Introdução À Aprendizagem de Máquina

Prof. Filipe Dwan Pereira
filipedwan@gmail.com
filipe.dwan@ufrr.br

O que é AM?

- Em uma definição mais geral:
  - “Um campo de estudo que dá aos computadores a habilidade de aprender sem ser explicitamente programado” – Arthur Samuel (1959)
- Em uma definição mais formal:
  - “Um programa de computador aprende a partir de experiências E relacionadas a uma classe de atividades T sendo medidas por uma métrica P, se o desempenho de uma tarefa em T, medida por P, melhora com a experiência E”. Tom M. Mitchell (1998)
- Em uma definição mais prática:
  - É a ciência (e arte) de programar computadores a fim de que eles possam **aprender através dos dados**

Exemplo – classificação ácero

ácero

Não é ácero

Não é ácero

ácero

ácero

Não é ácero

Exemplo do filtro de spam

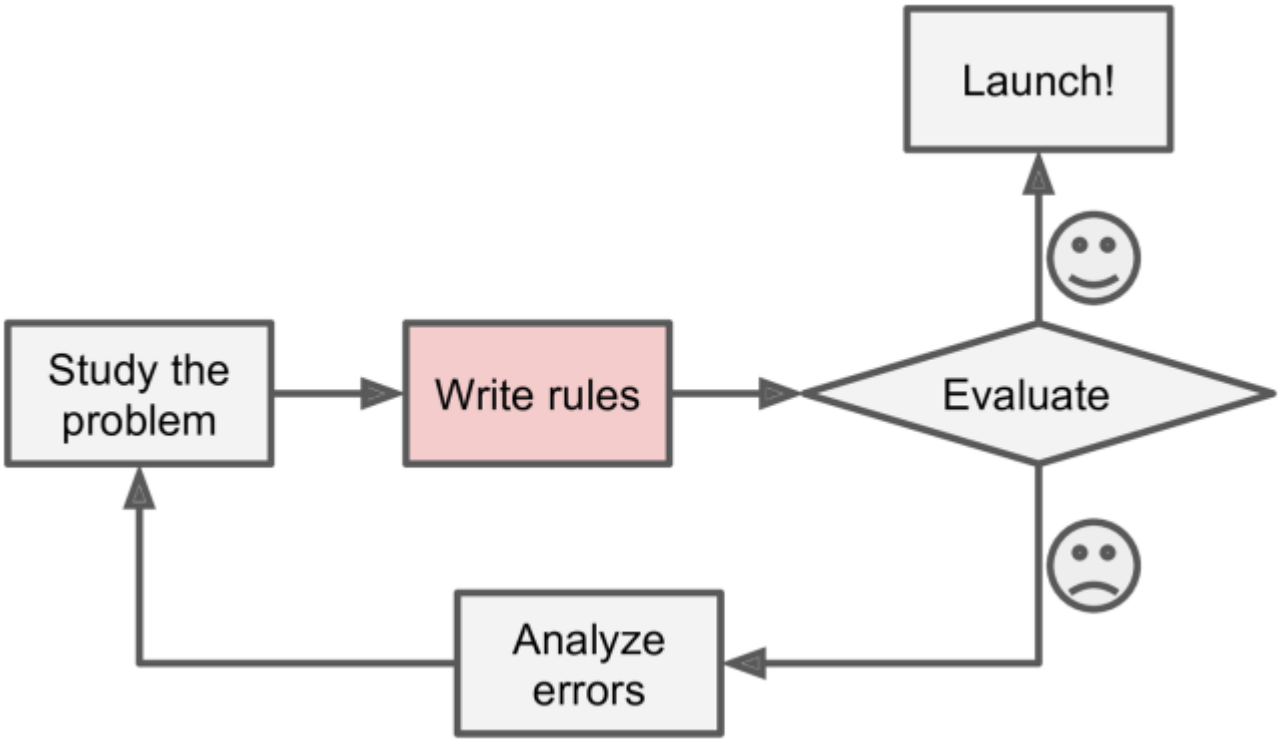
- A primeira aplicação de aprendizagem que fez sucesso foi o filtro de spam (feito nos anos 90):
  - Um filtro de spam é uma aplicação que emprega AM para rotular e-mails como spam e ham (não spam), dados exemplos de spam's e ham's rotolados por usuários;
  - Em AM supervisionada, a base de dados é normalmente dividida em conjunto de treinamento e conjunto de teste;
    - Os exemplos que o sistema de AM usa para aprender é chamado de **conjunto de treinamento**;
    - O complemento é usado para mensurar o desempenho do sistema de AM e é chamado de **conjunto de teste**;
  - Cada exemplo na base de dados é chamado de instância (amostra ou observação);
  - No caso de uma aplicação de filtro de spam:
    - a tarefa T é rotular novos e-mails como spam ou ham;
    - A experiência E é o conjunto de treinamento;
    - A métrica de desempenho P pode ser, por exemplo, a razão de instâncias da base de teste corretamente classificadas pela quantidade de instâncias do conjunto de teste;

Por que usar AM?

- Considere a forma que escreveríamos um filtro de spam usando programação tradicional:
  1. Analisar os padrões de um spam: palavras-chave, endereço de e-mail do remetente, corpo do e-mail e etc.
  2. Escrever um algoritmo baseado em todos os padrões que foram notados para caracterizar um spam;
  3. Testar o programa escrito e repetir os passos 1 e 2 até que o programa estava aceitável;

Por que usar AM?

- Filtro de spam usando programação tradicional



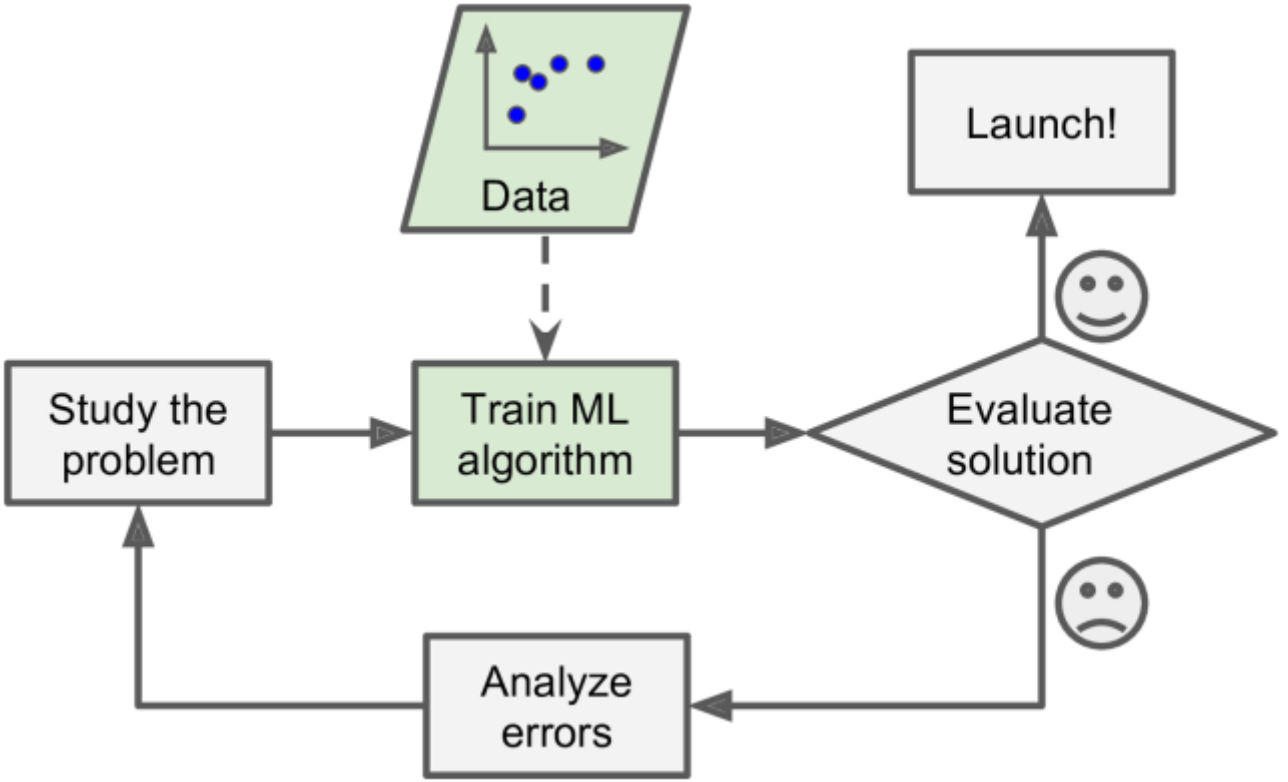
Perceba que nosso conjunto de regras tende a ficar cada vez maior!

Por que usar AM?

- Por outro lado, um filtro de spam que usa AM aprenderá automaticamente quais palavras e frases são bons preditores de spam;
- Isso é feito por exemplo através da detecção da frequência de certas palavras em e-mails spam e ham;
- Tal programa é muito menor, mais fácil de manter e provavelmente bem mais preciso (maior acurácia);

Por que usar AM?

- Filtro de spam usando AM

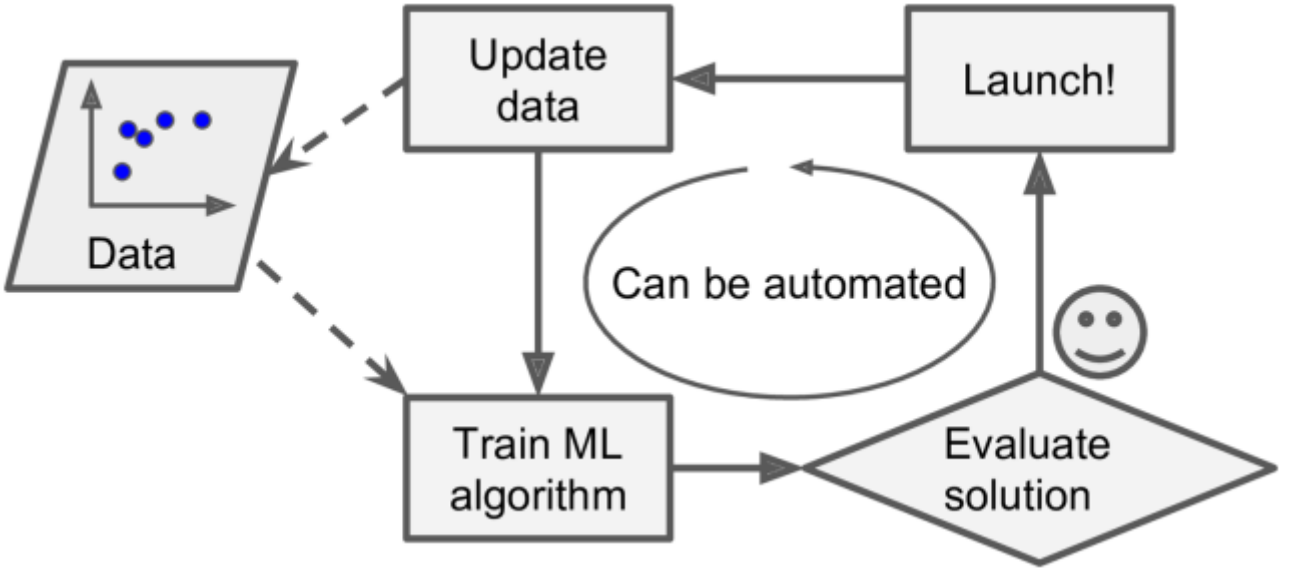


Por que usar AM?

- Note que se os *spammers* percebem que certas palavras estão sendo utilizadas para detectar um spam, eles podem trocá-las (e.g. 4U ao invés de “For U”).
  - Um filtro de spam que utiliza programação tradicional precisaria ser atualizado para rotular 4U manualmente.
  - Já um filtro de spam que utiliza AM perceberia automaticamente essa mudança;

Por que usar AM?

- Filtro de spam com AM pode perceber automaticamente mudanças em padrões de e-mails com spam ou ham:

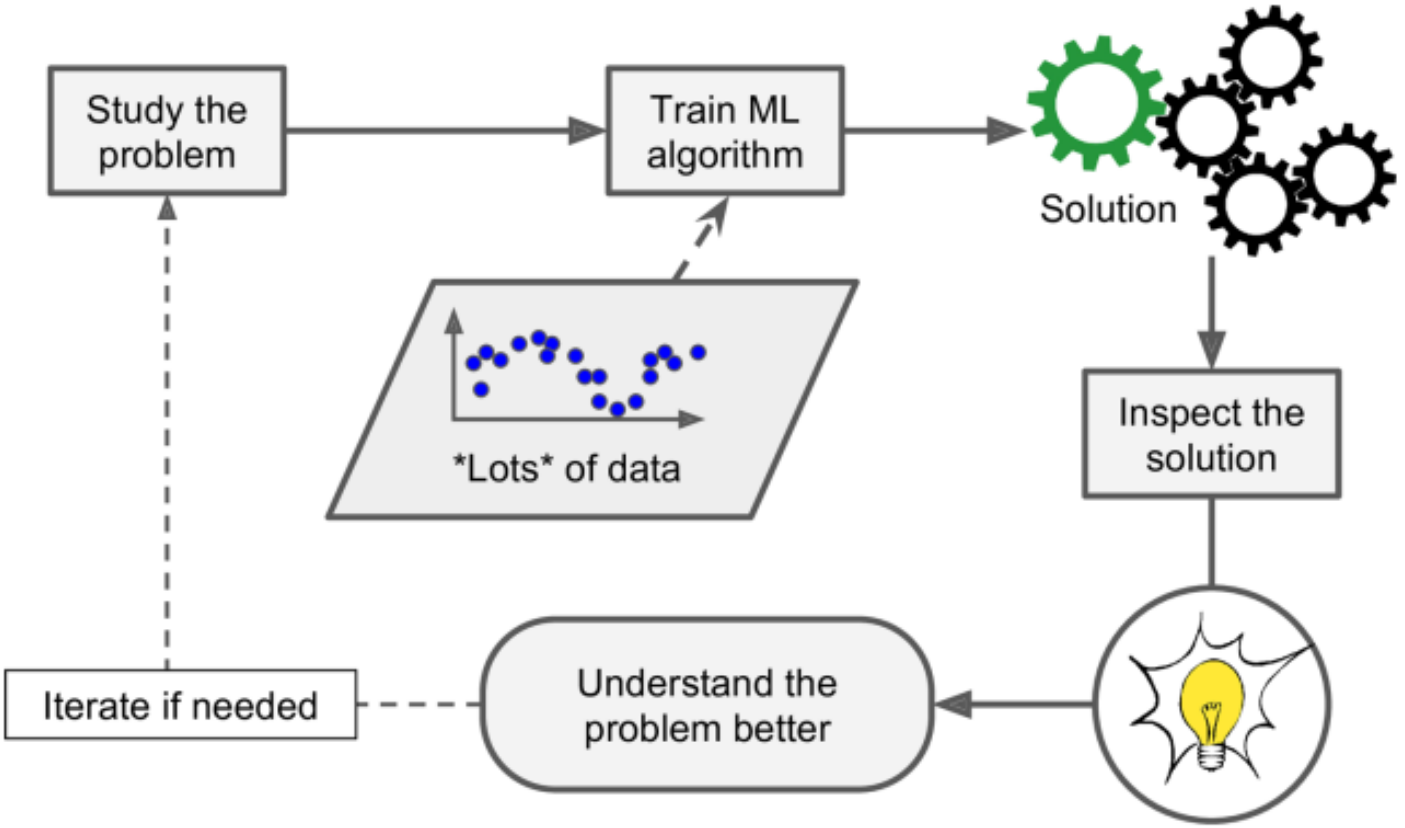


Por que usar AM?

- Existem problemas que são muito complexos e dinâmicos, tornando inviável um mapeamento de todos os padrões possíveis, exemplo:
  - Reconhecimento de voz: mapear todos os picos de som que representam palavras, frases e etc;
  - Sistemas de recomendação: mapear uma recomendação para cada preferência de usuário em um contexto dinâmico;
  - Reconhecimento de objetos em imagens: cada objeto pode aparecer com diferentes variações;

Por que usar AM?

- AM pode ser útil para descobrir padrões implícitos (data mining);

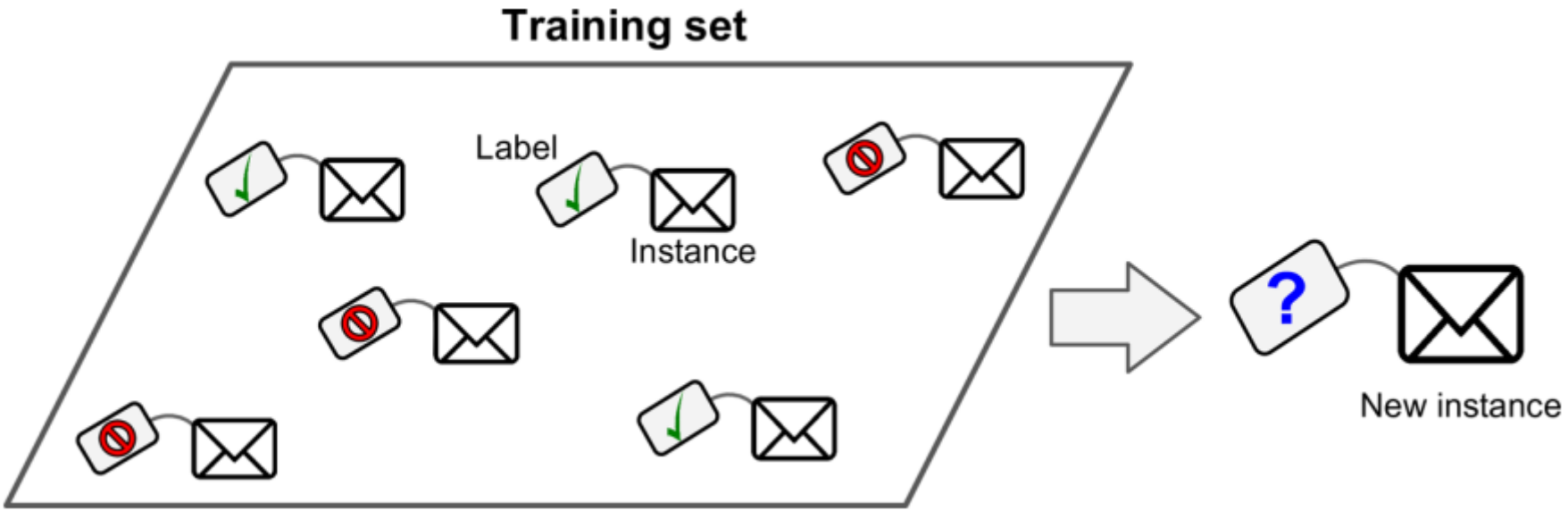


Tipos de Aprendizagem de máquina

- Existem muitos tipos de sistemas de AM.
- Iremos classificá-los em categorias baseadas em:
  - Se eles foram treinados ou não com supervisão humana (supervisionada, não-supervisionada, semi-supervisionada e reforço);
  - Se eles podem continuar aprendendo (online versus batch learning)
  - Se eles aprendem comparando novas amostras com amostras conhecidas ou se eles detectam padrões no conjunto de treinos para construir modelos preditivos (instance-based versus model-based)
- Note que tais categorias não são multuamente exclusivas.
  - Por exemplo, um filtro de spam moderno usa uma aprendizagem online, model-based, supervisionada;

AM Supervisionada

- Em AM supervisionada o conjunto de treino possui as respostas certas, o que chamamos de *labels* (**classes**, *targets* ou variáveis dependentes);



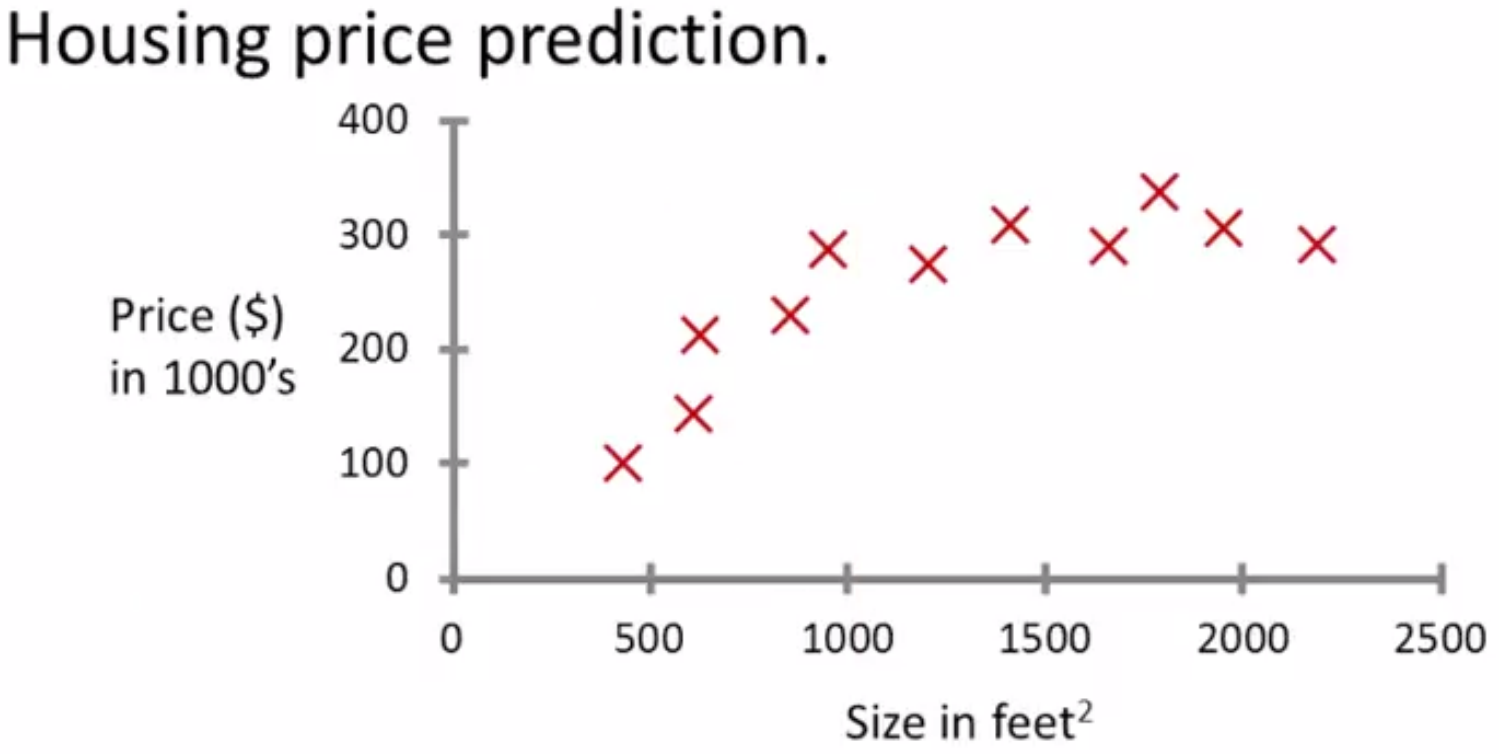
AM Supervisionada

- Formalmente, a tarefa de AM supervisionada é a seguinte:
  - Dado um conjunto de treinamento de *N* pares de exemplos de entrada e saída
    - $(x_1, y_1), (x_2, y_2), \ldots (x_N, y_N)$
  - Onde cada y_i foi gerado por uma função desconhecida $y = f(x)$,
  - Descobrir uma função *h* que se aproxime da função verdadeira *f*.
- A função h é uma hipótese;
- AM é uma busca através do espaço de hipóteses por uma que tenha bom desempenho mesmo em exemplos diferentes dos do conjunto de treino;
- Para mensurar a precisão da hipótese, usa-se um conjunto de teste de exemplos que são distintos do conjunto de treino;
- Dizemos que uma hipótese **generaliza** bem se infere com alta acurácia o valor de *y em* novos exemplos.

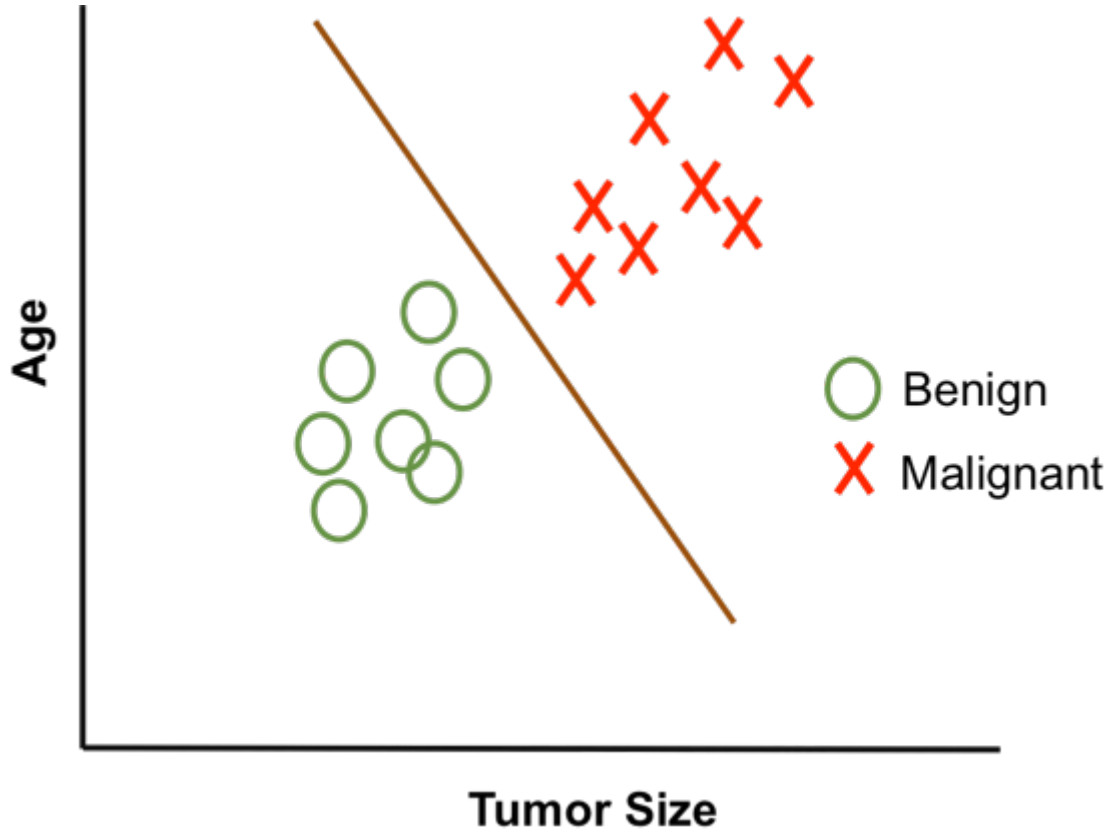
Tarefas em AM supervisionada

- Em AM supervisionada podemos ter tarefas de classificação ou regressão;
- Na classificação, o *target* (classe) pertence a um conjunto finito de valores ($C = \{c_1, \ldots c_n\}$), enquanto que na regressão é uma variável quantitativa;
  - Um filtro de spam é uma tarefa de classificação já que o *target* só pode ser *spam* ou *ham*;
  - Um sistema para inferir o preço de um carro é um exemplo de regressão, pois preço (target) é um valor quantitativo;
- Em classificação e regressão tenta-se inferir o target usando um conjunto de **atributos** (carcterísticas, **features**, variáveis indepentes, preditores)
  - Conjunto de atributos $z = [x_1,..., x_n]$.
  - Por exemplo, para inferir o preço do carro podemos usar como atributos:
    - Ano do carro;
    - Marca;
    - Cor;
    - Quantidade de donos anteriores;
    - Etc.

AM supervisionada: regressão

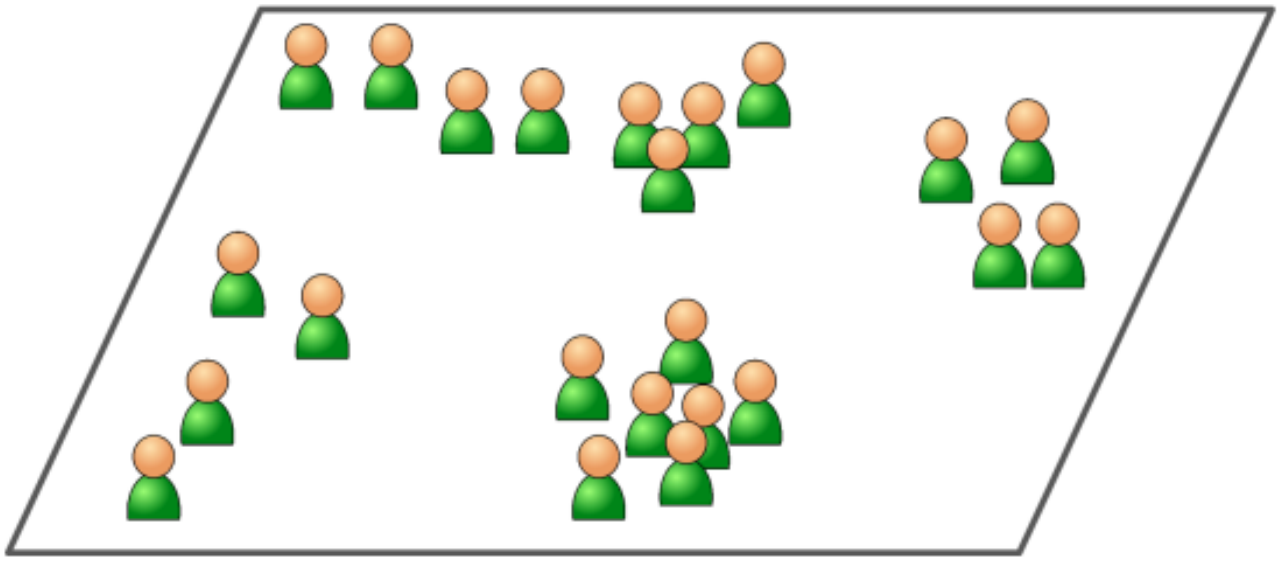


AM supervisionada: classificação

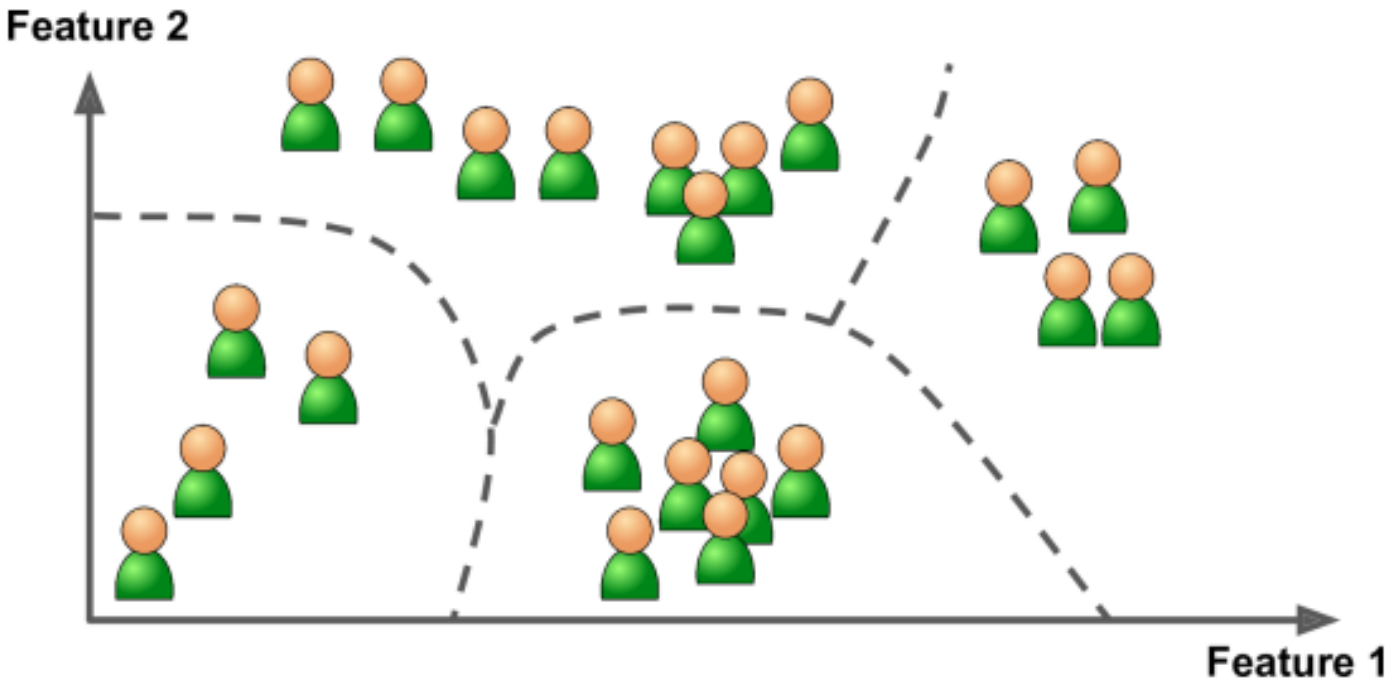


AM não supervisionada

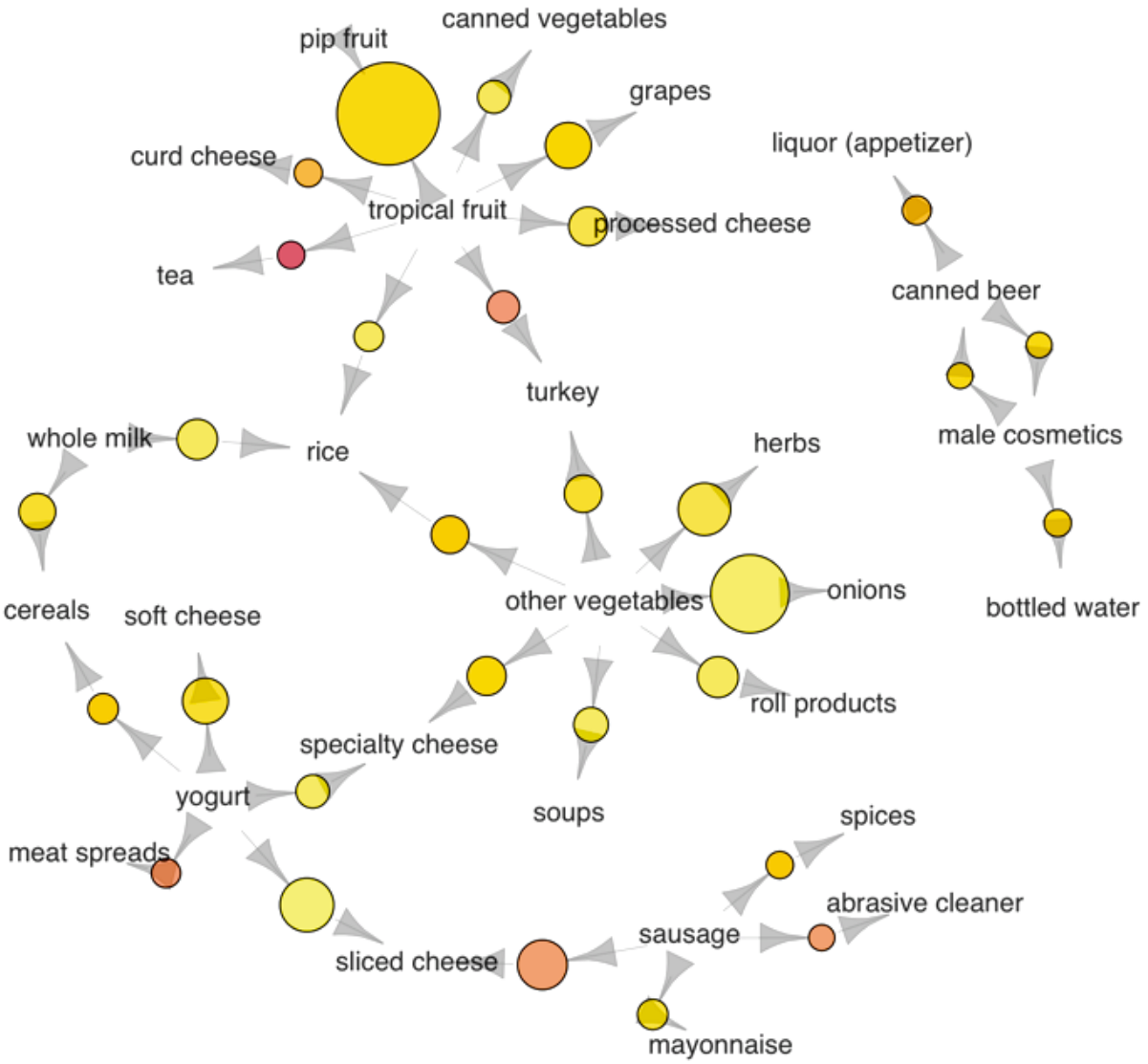
- Em AM não supervisionada as instâncias não possuem label/target.
  - Ex.: Agrupamento (clustering), regras de associação, redução de dimensionalidade, detecção de anomalias e etc.



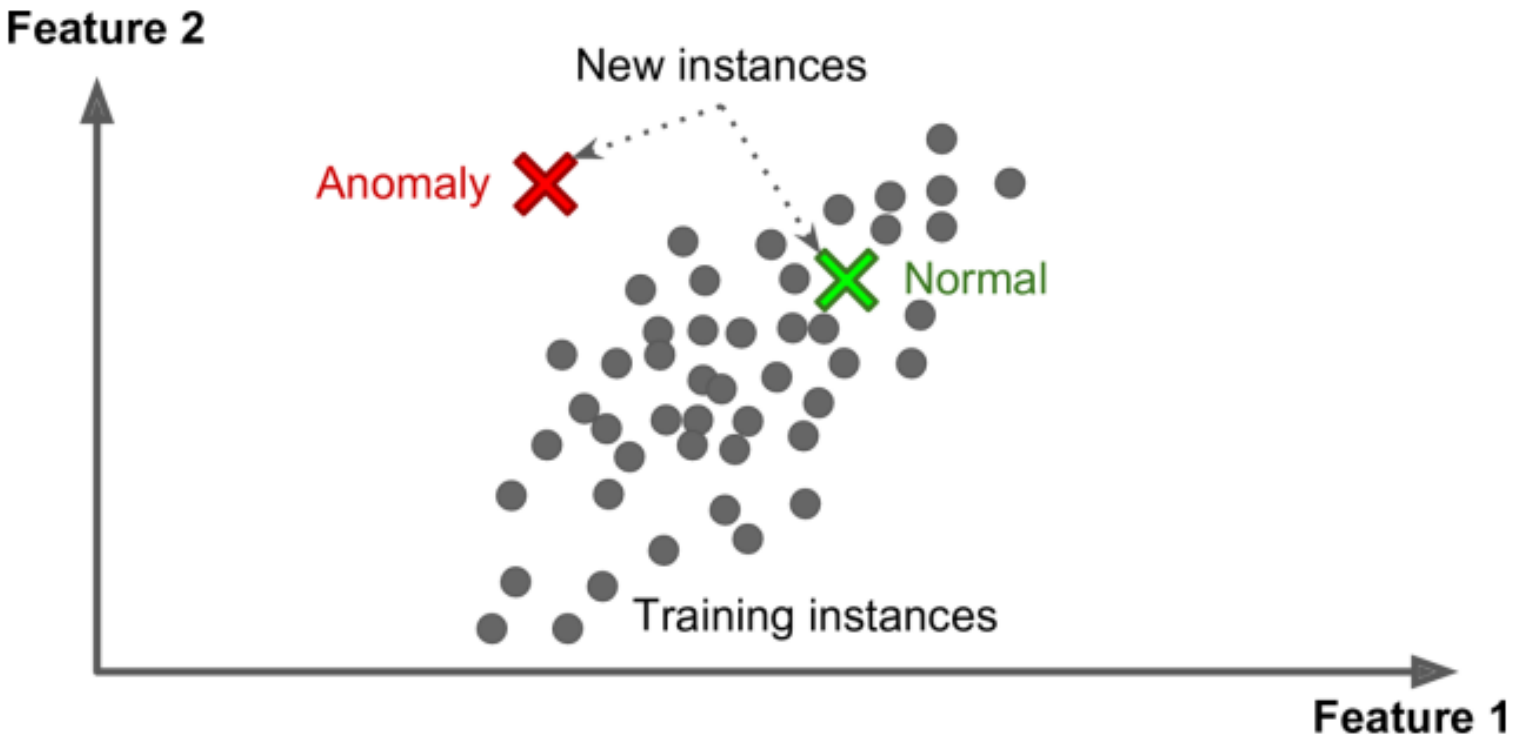
AM não supervisionado: clustering



AM não supervisionada: associação

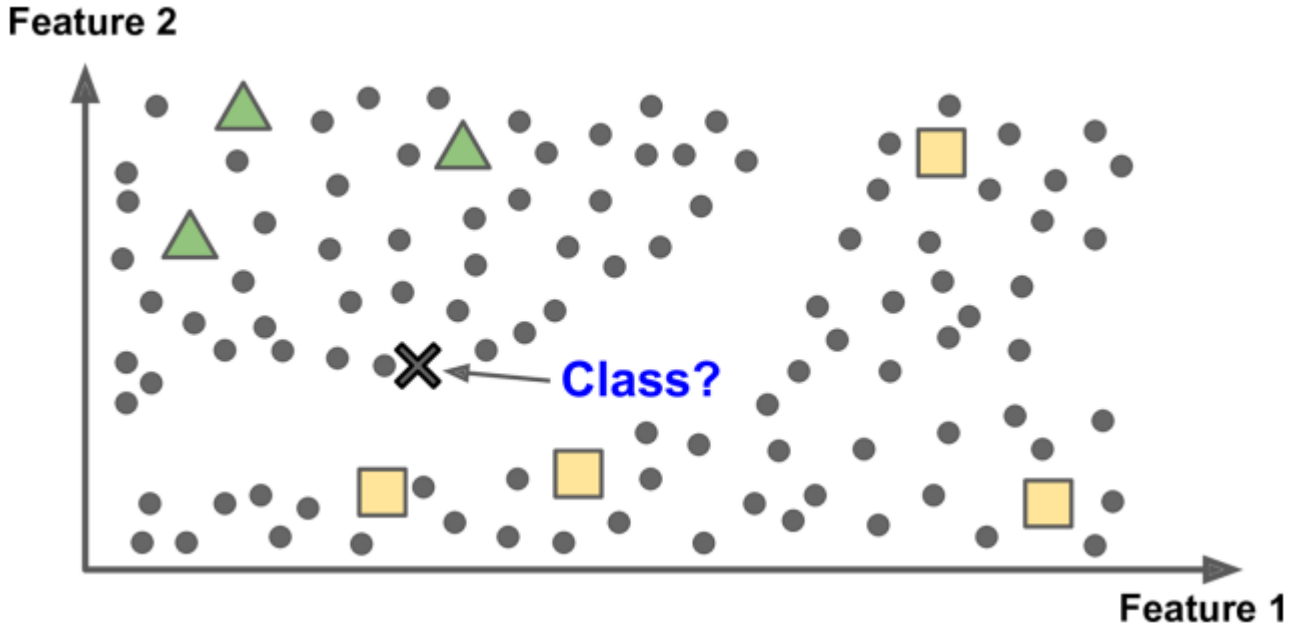


AM não supervisionada: detecção de anomalias



AM semi-supervisionada

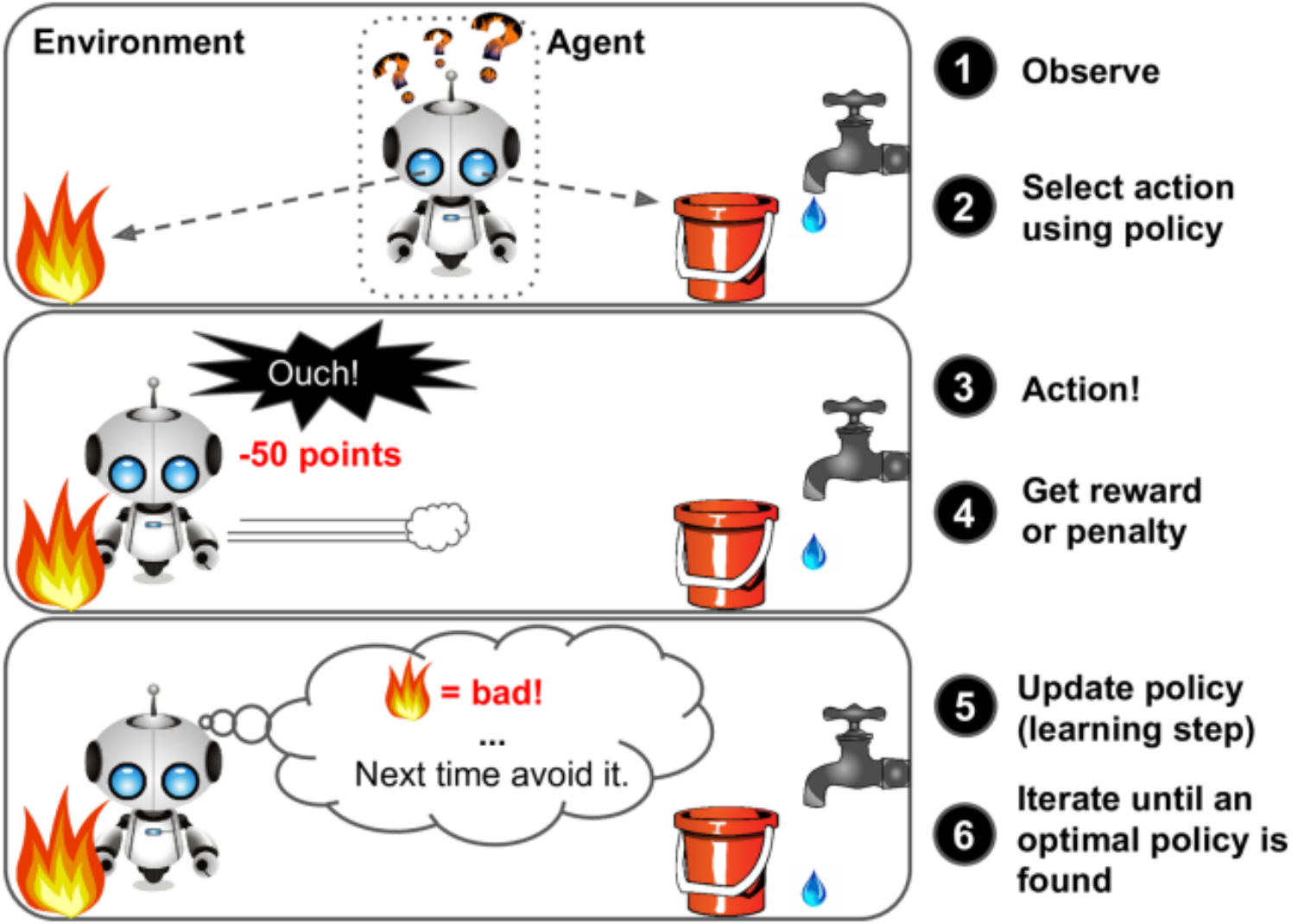
- Existem bases de dados que possuem labels apenas para algumas instâncias;
  - Nesse caso, estamos lidando com AM semi-supervisionada;
  - Ex.: google photos;



Aprendizagem por reforço

- O sistema de aprendizagem, chamado de agente, percebe o ambiente, seleciona ações e recebe uma recompensa/penalização;
- Ele aprende sozinho usando uma função (*policy*) para maximizar a recompensa;
- A função define qual ação deve ser selecionada dada a configuração do ambiente.

Aprendizagem por reforço

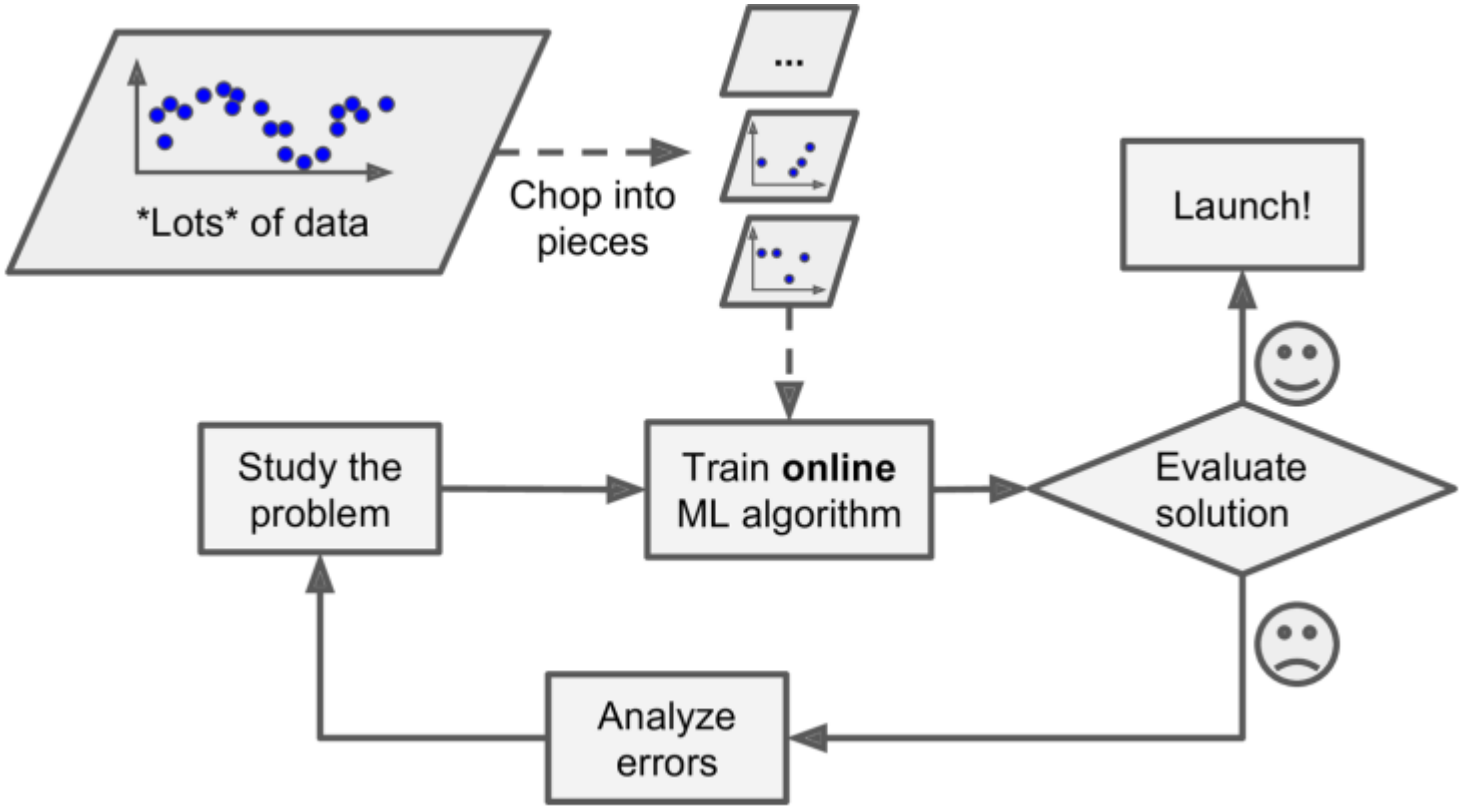


Aprendizagem em batch

- Nesse tipo de aprendizagem o sistema é incapaz de aprender de modo incremental
  - O sistema é treinado e então colocado em produção (sem aprender mais) – *offline learning*;
  - Caso deseje que o sistema aprenda sobre novos dados, deve-se treinar o sistema novamente (dados antigos + dados novos);
    - Após isso, interrompe-se o sistema antigo e coloca-se em produção o sistema novo.
  - Observe que esse processo de retrino pode ser executado periodicamente;

Aprendizagem online

- Sistema treinado de forma incremental;
- Algoritmos são alimentados por dados sequencialmente ou por mini-batches.

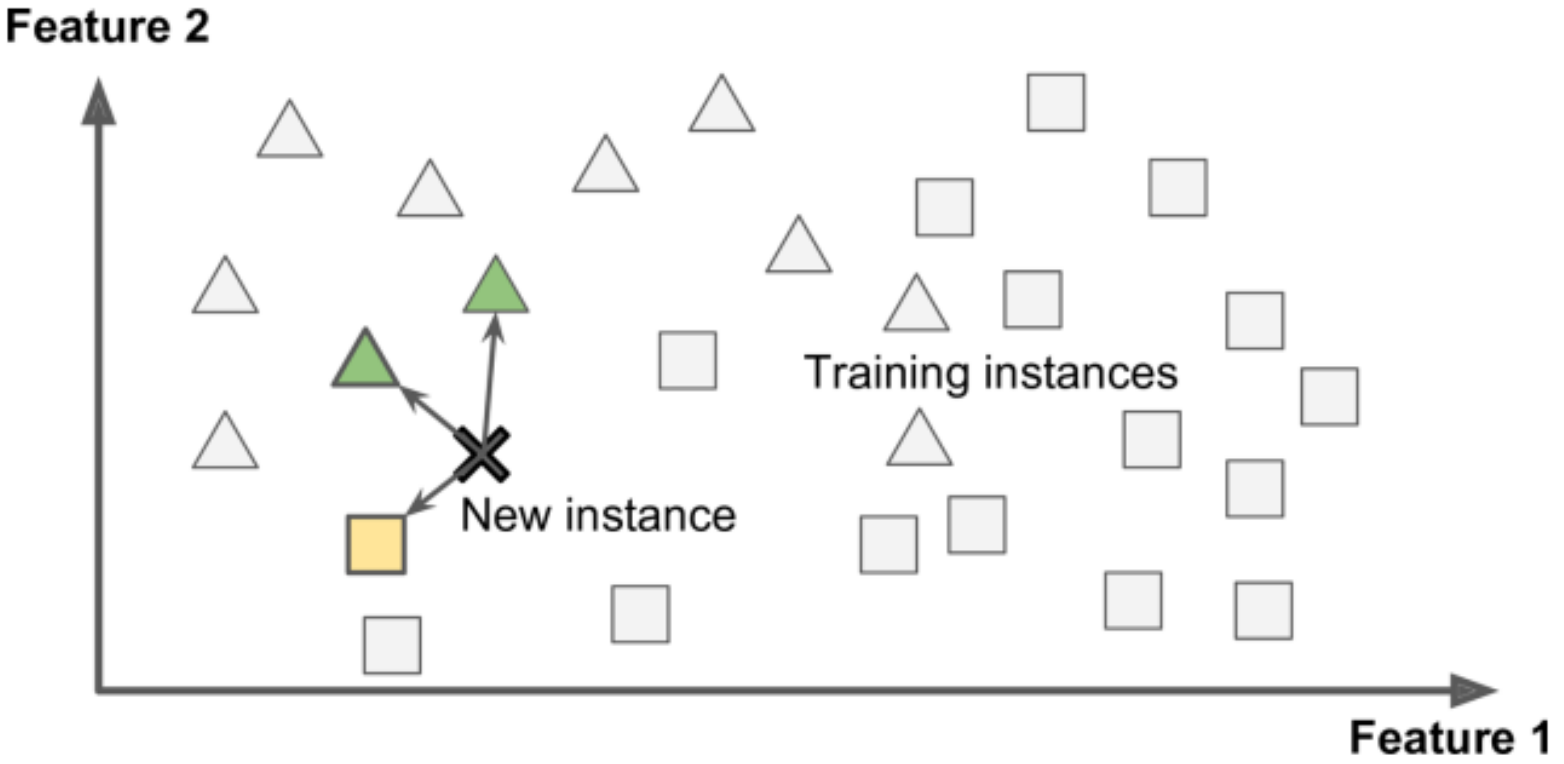


Instance-based versus model-based

- Uma outra forma de categorizar AM é em relação a como o sistema generaliza;
- Existem basicamente duas abordagens para o sistema realizar a generalização: instance-based versus model-based;

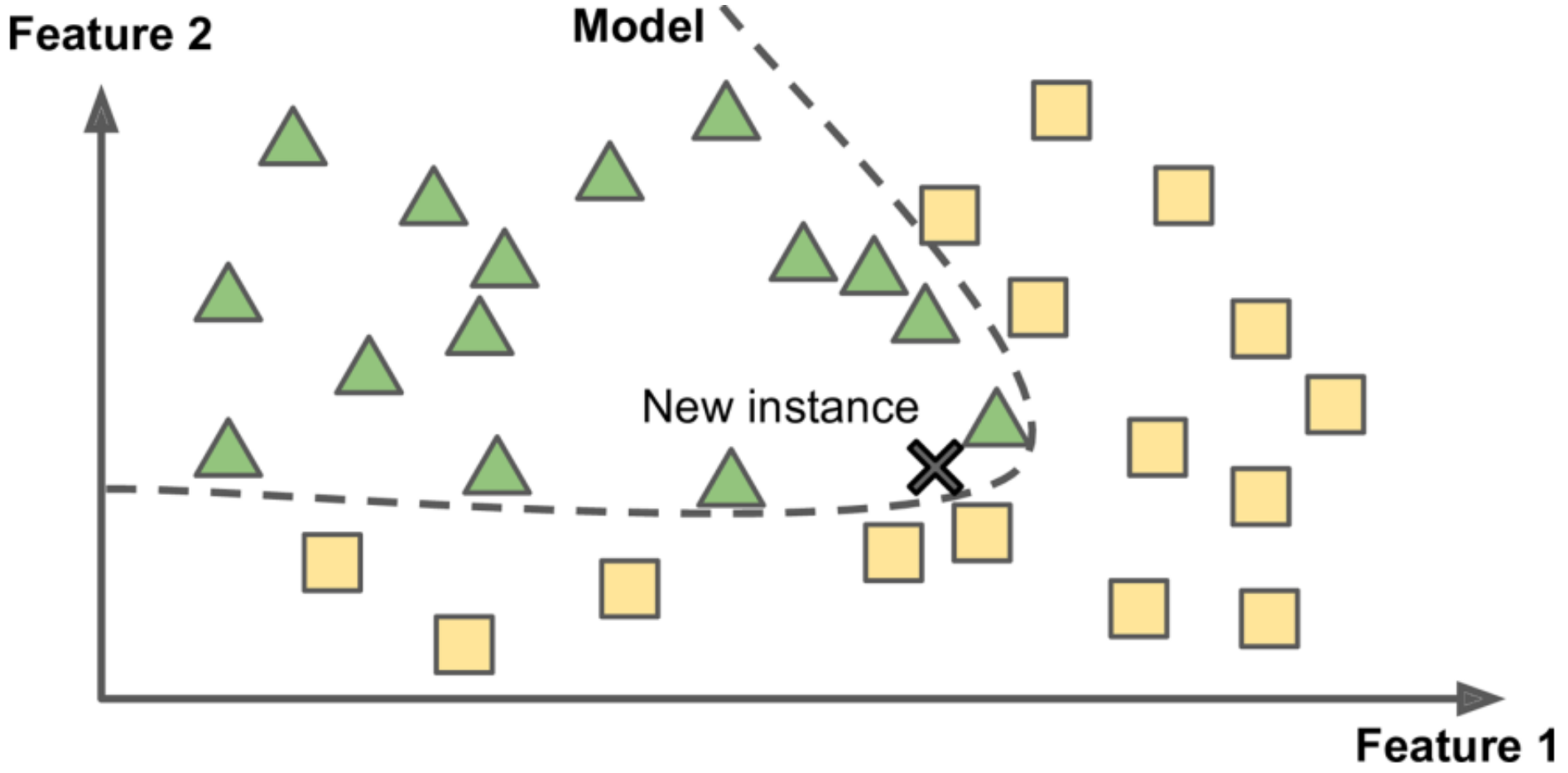
Instance-based

- Usa uma *medida de similaridade* para generalizar na base de teste com base em amostras conhecidas na base de treino;



Model-based

- Constroi um modelo das instâncias (baseado em padrões) para então realizar as predições;



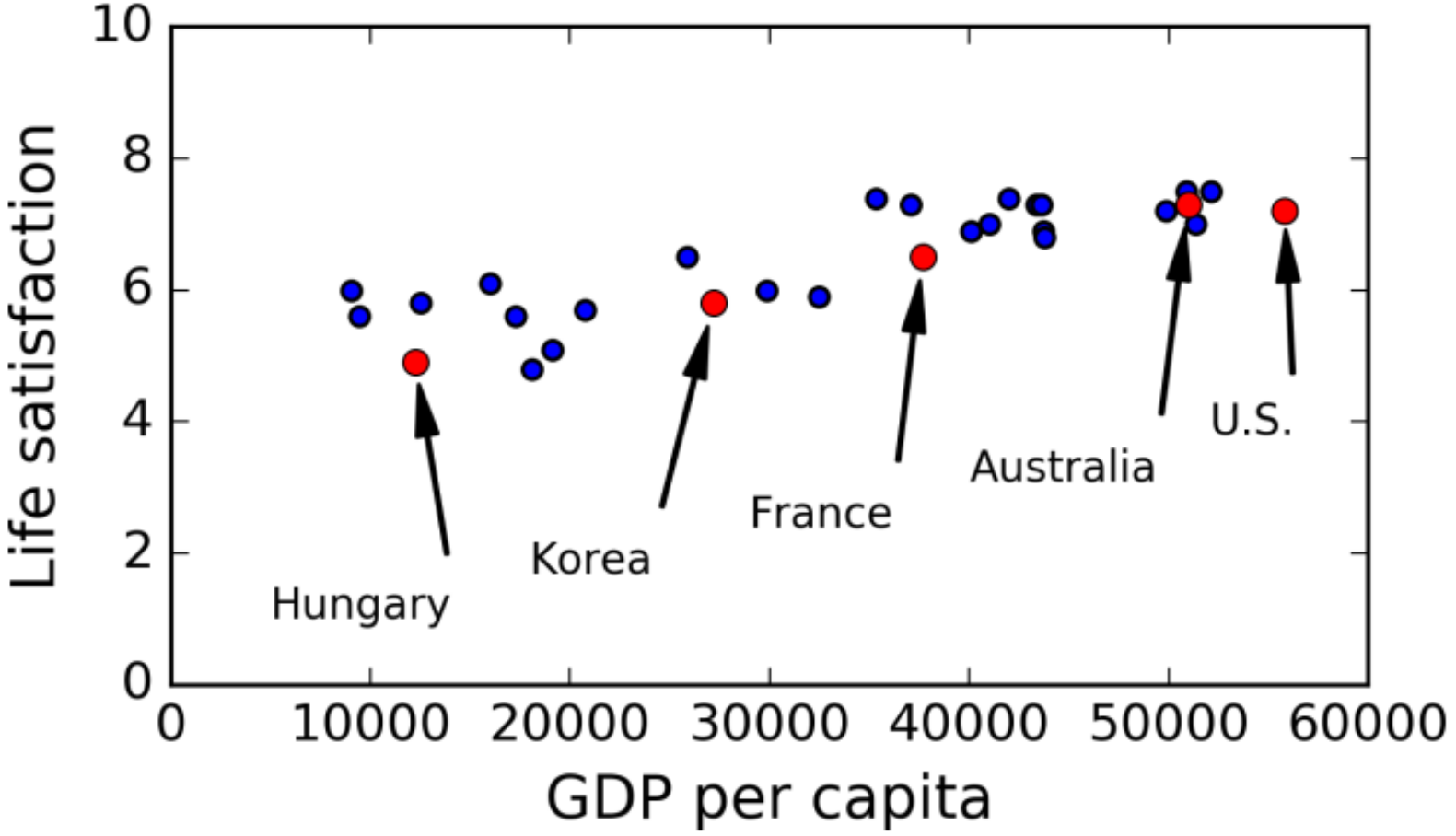
Exemplo de AM model-based

- Dinheiro deixa as pessoas mais felizes?
  - Veja uma amostra de um join das seguintes bases de dados:
    - https://goo.gl/0Eht9W
    - http://goo.gl/j1MSKe

Country	GDP per capita (USD)	Life satisfaction
Hungary	12,240	4.9
Korea	27,195	5.8
France	37,675	6.5
Australia	50,962	7.3
United States	55,805	7.2

Exemplo de AM model-based

- Plotando os dados de alguns países aleatórios:



Existe alguma tendência na figura?

Exemplo de AM model-based

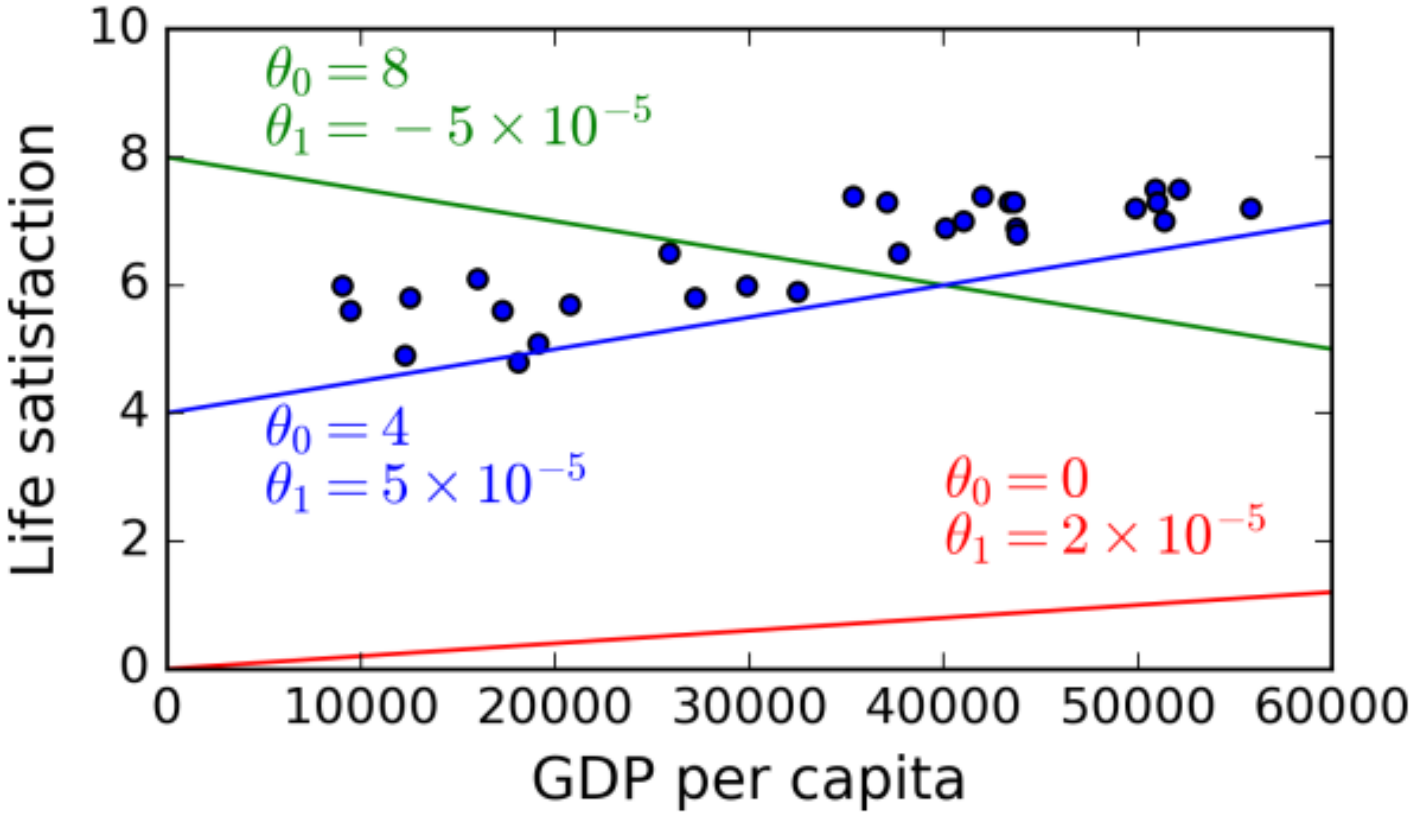
- Observa-se que à medida que o *GDP per capita* aumenta, o *life satisfaction* também aumenta;
- Em virtude disso, modelaremos o *life satistaction* com um função linear da *GDP per capita:*

$$life\_satisfaction = \theta_0 + \theta_1 \times GDP\_per\_capita$$

  - Nota: esse processo é chamado de model selection, onde *GDP per capita é um atributo e life satistaction o target;*

Exemplo de AM model-based

- Esse modelo possui dois parâmetros, θ_0 e θ_1.
- Mexendo nesses parâmetros, podemos construir qualquer funcão linear (modelos lineares):

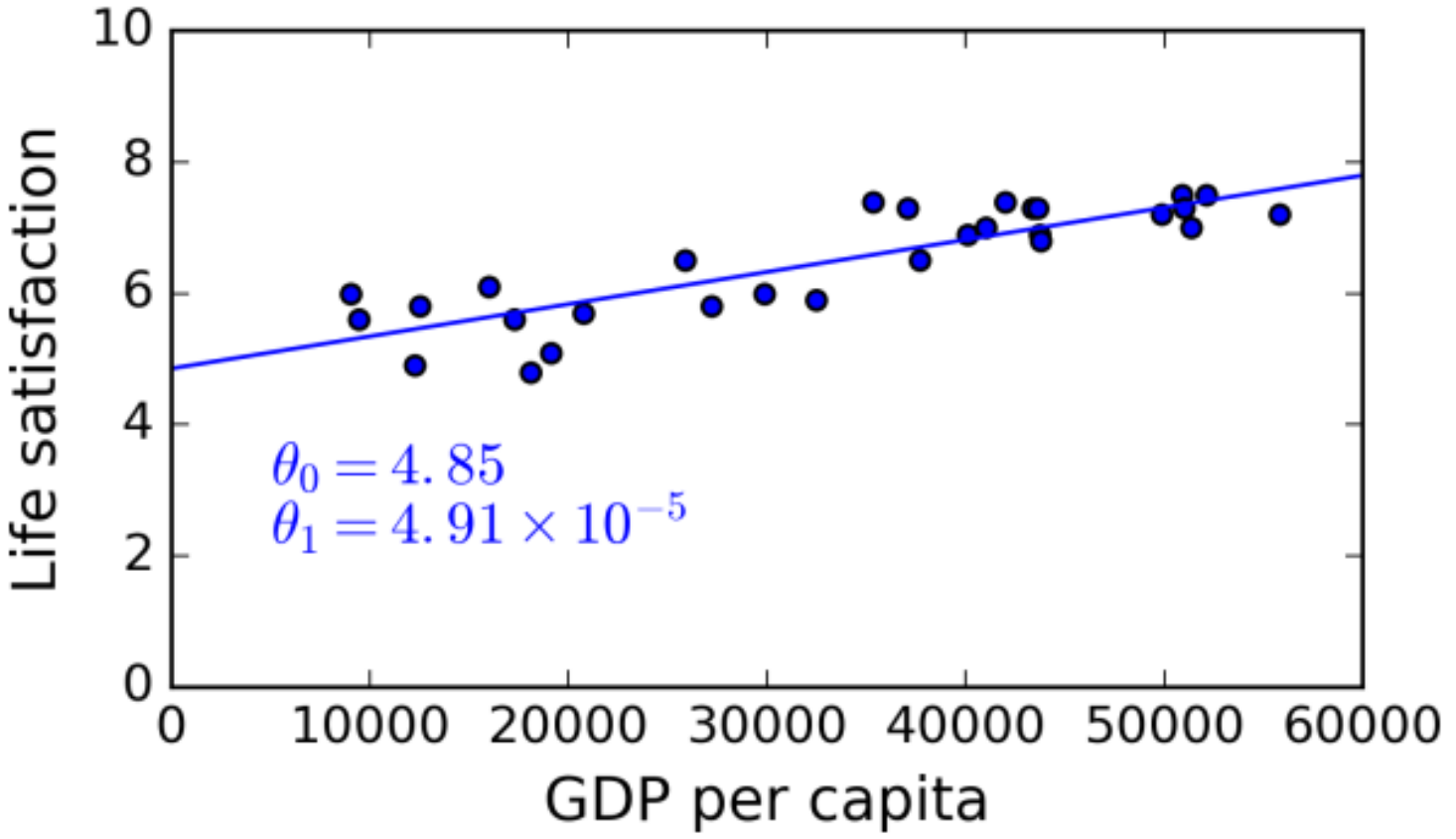


Exemplo de AM model-based

- Quais valores ótimos para os parâmetros θ_0 and θ_1?
  - Precisamos de uma métrica de desempenho (***utility function*** or ***fitness function***) que mensuara o quão bom o modelo é ou um ***cost function*** que mensura o quão ruim o modelo é;
  - Um cost function muito usada nesses casos é distância entre a predição do modelo linear e o volar real;
  - Assim, o objetivo é minimizar essas distâncias;

Exemplo de AM model-based

- O algoritmo chamado Regressão Linear faz exatamente isso e nesse caso os melhores parâmetros encontrados foram $\theta_0 = 4.85$ and $\theta_1 = 4.91 \times 10^{-5}$

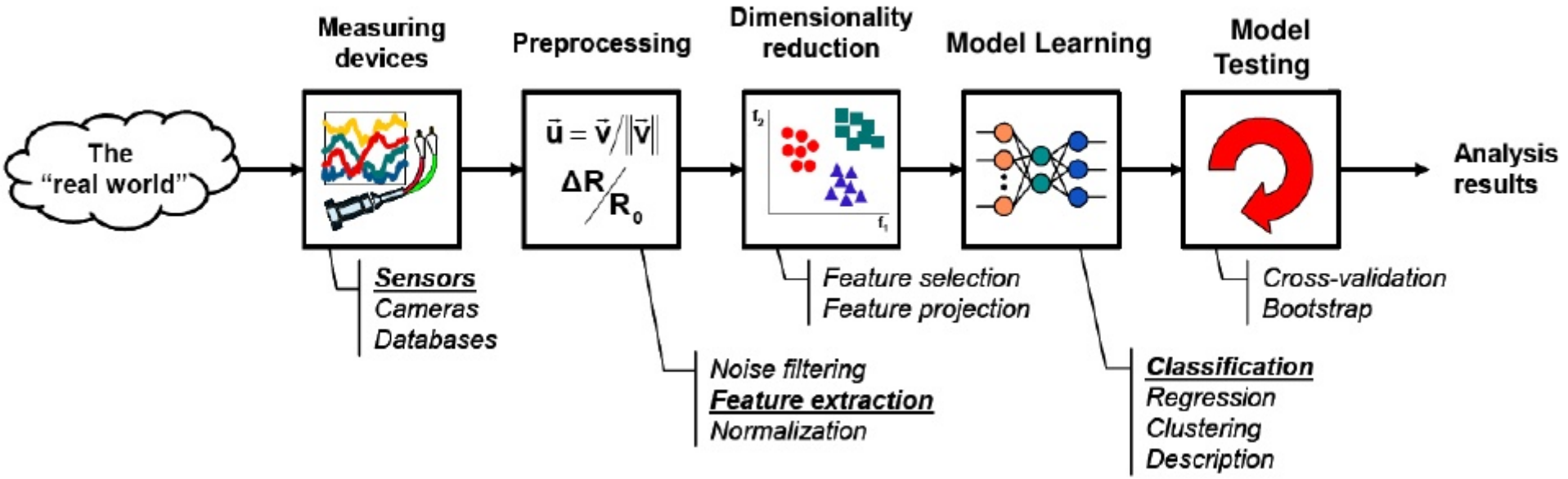


Exemplo de AM model-based

- Se o modelo não trouxe bons resultados, existem algumas opções:
  - Usar mais atributos (e.g. taxa de desemprego, qualidade de saúde, poluição, etc);
  - Coletar mais dados (com mais qualidade);
  - Tentar usar algoritmos de AM mais poderosos (e.g. Polynomial Regression)

Pipeline de AM

- A tarefa de AM está em aprender em um conjunto de treino e **generalizar** em um conjunto de teste;

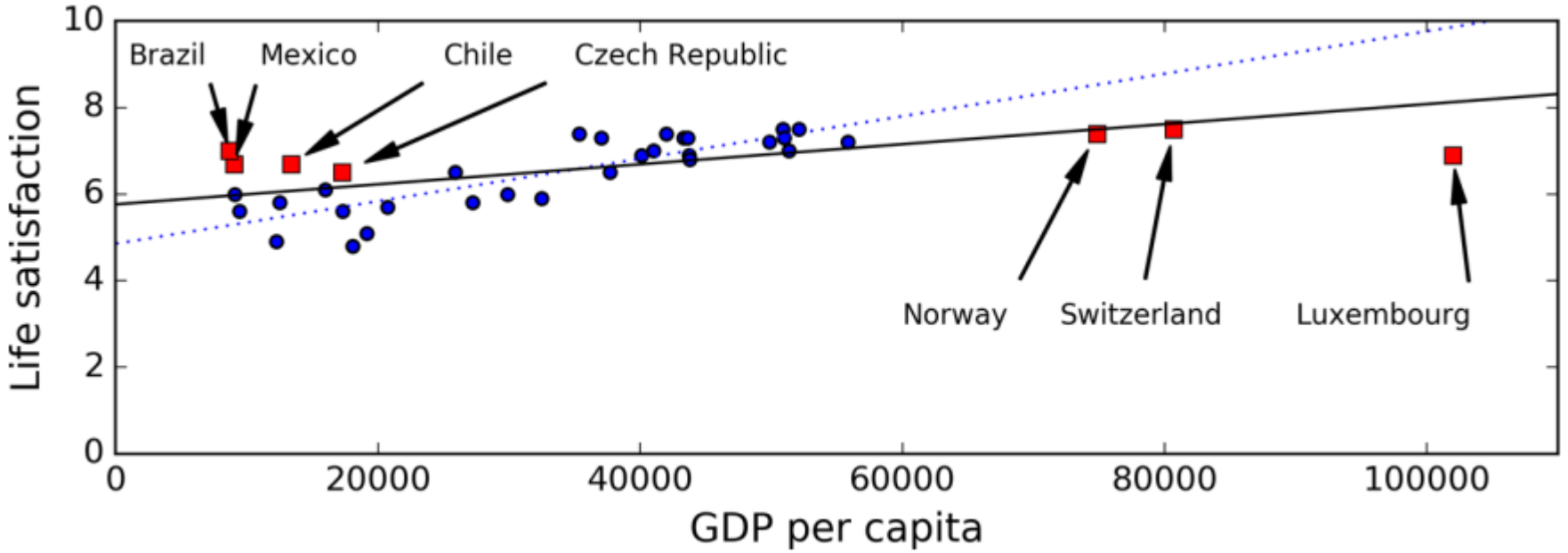


Principais desafios em AM

- Quantidade insuficiente de dados de treino:
  - Para que uma criança aprenda o que é uma maçã, basta você apontar para **uma** maçã e dizer “maçã” (possivelmente repetindo este procedimento algumas vezes).
    - Agora a criança é capaz de reconhecer maçãs em todos os tipos de cores e formas. Gênio!
  - Em AM não é bem assim;
    - Mesmo para problemas simples normalmente é necessário milhares de amostras;
    - Em problemas complexos como speach recognition ou image recognition é necessário milhões de amostras ou modelos pré-treinados;

Principais desafios em AM

- Dados não representativos
  - Para generalizar bem, é essencial que os dados sejam representativos
  - Por exemplo a base sobre renda per capita e nível de felicidade não é tão representativa quando inserimos os dados de todos os paízes:



Nota: no gráfico, a linha contínua representa o modelo treinado com estes dados e linha pontilhada representa os dados treinados com a subamostra utilizada anteriormente

Principais desafios em AM

- Dados não representativos
  - Sampling bias
    - Note que uma amostra pequena pode ter ruídos;
    - Mesmo uma amostra grande pode ter sido mal coletada.
      - Nas eleições presidenciais de 1936, onde Landon concorreu contra Roosevelt, a empresa Literary Digest conduziu um survey perguntando por telefone para 10 milhões de pessoa qual seria o seu voto.
      - 2,4 milhões de pessoas responderam, apontando para uma vitória de Landon com 57% dos votos;
      - Nas eleições, Roosevelt venceu com 62% dos votos;
        - Para coletar as respostas a empresa usou listas de telefones, dados de inscrições em revistas e etc. Tais fontes são normalmente vinculadas a pessoas ricas, as quais tendem a votar em republicanos;
        - Além disso, 75% não respondeu (nonresponse bias)

Principais desafios em AM

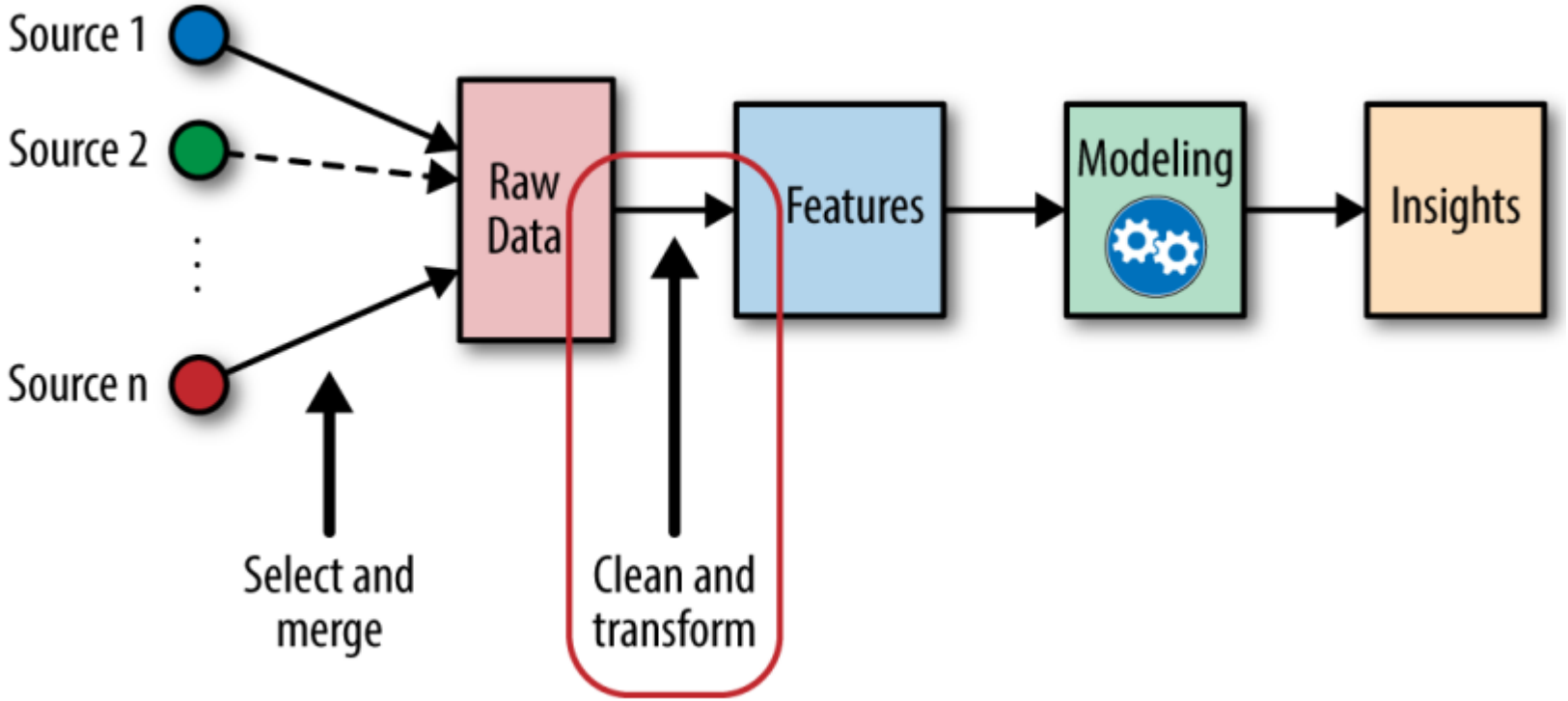
- Dados com baixa qualidade
  - Dados com erros, outliers, dados faltantes e etc.
  - Os cientistas dos dados normalmente investem boa parte do tempo preparando as bases de dados:
    - Remoção de outliers
    - Ajustar erros;
    - Interpolação
    - Etc.

Principais desafios em AM

- Atributos irrelevantes
  - *Garbage in, garbage out*;
  - O sucesso de um projeto de AM depende bastante do conjunto de atributos utilizado;
  - Lidando com Feature:
    - Feature Engeneering;
    - Feature selection (seleção de atributos): selecionar as features (atributos) mais relevantes;
    - Feature combination: combinar features existentes para produzir features com mais poder preditivo (e.g. redução de dimensionalidade);
    - Criar novas features através da obtenção de mais dados;

Principais desafios em AM

- Atributos irrelevantes
  - Feature engeneering (engenharia de atributos):

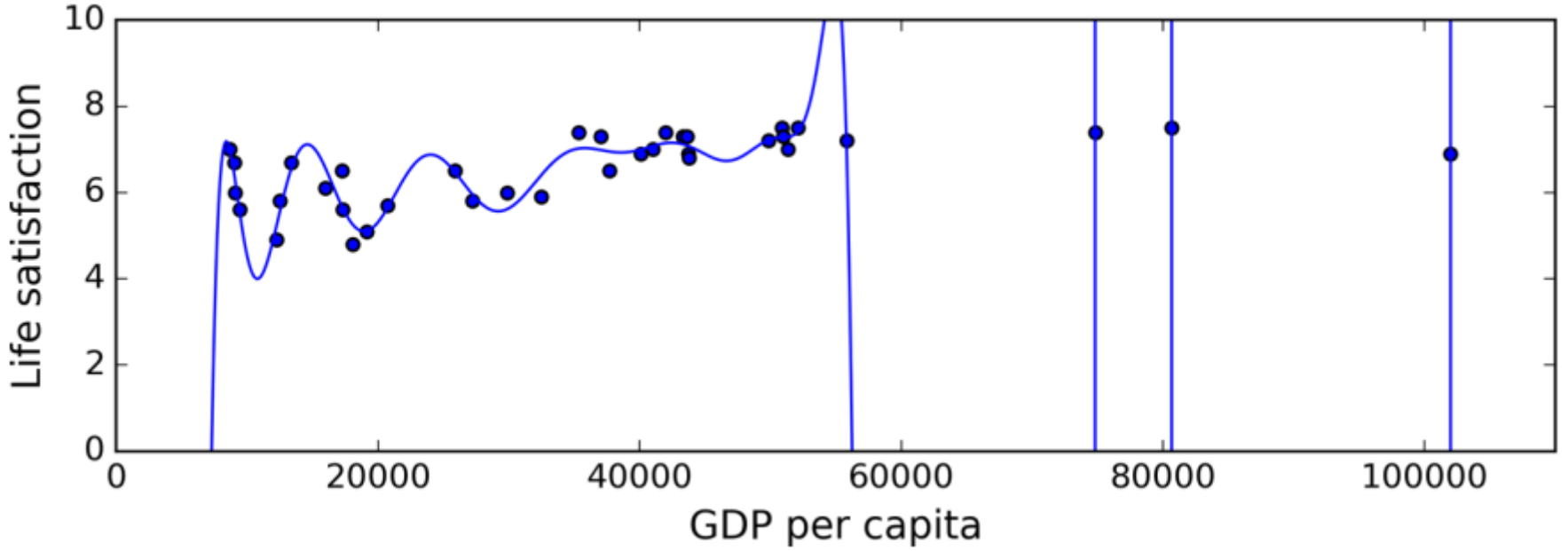


Principais desafios em AM

- Overfitting
  - Modelo é acurado na conjunto de treinamento, mas não generaliza bem;
  - Normalmente causado pela adaptação do modelo preditivo às peculiaridades do conjunto de treino;
  - Pode ocorrer quando o modelo preditivo é treinado com um conjunto muito grande de exemplos com pequena variação intra-classe;
  - Pode ocorrer em função de muitas iterações de treinamento;

Principais desafios em AM

- Exemplo de overfitting: regressão polinomial (alto grau)

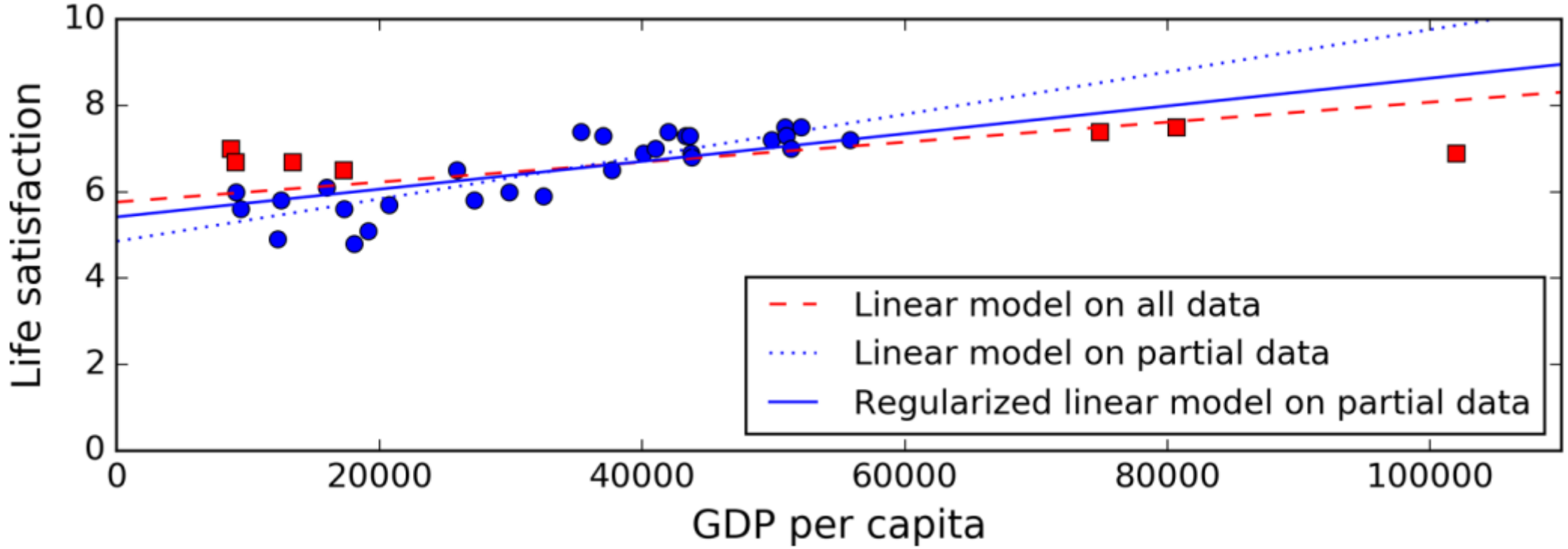


Principais desafios em AM

- Soluções para evitar overfitting
  - Aumentar a quantidade de dados de treinamento – ajuda até certo ponto;
  - Reduzir quantidade de ruídos
  - Ajustar hiperparâmetros do algoritmos de AM para tornar o modelo menos complexo
  - Regularização (restringir o modelo)
  - Usar **conjunto de validação**
    - Testar os diferentes algoritmos em validação e escolher a com menor erro em validação;
    - Encerrar o treinamento quando o erro em validação começar a aumentar – **critério de parada**;

Principais desafios em AM

- Exemplo de redução de overfitting por *regularization* e remoção de ruído

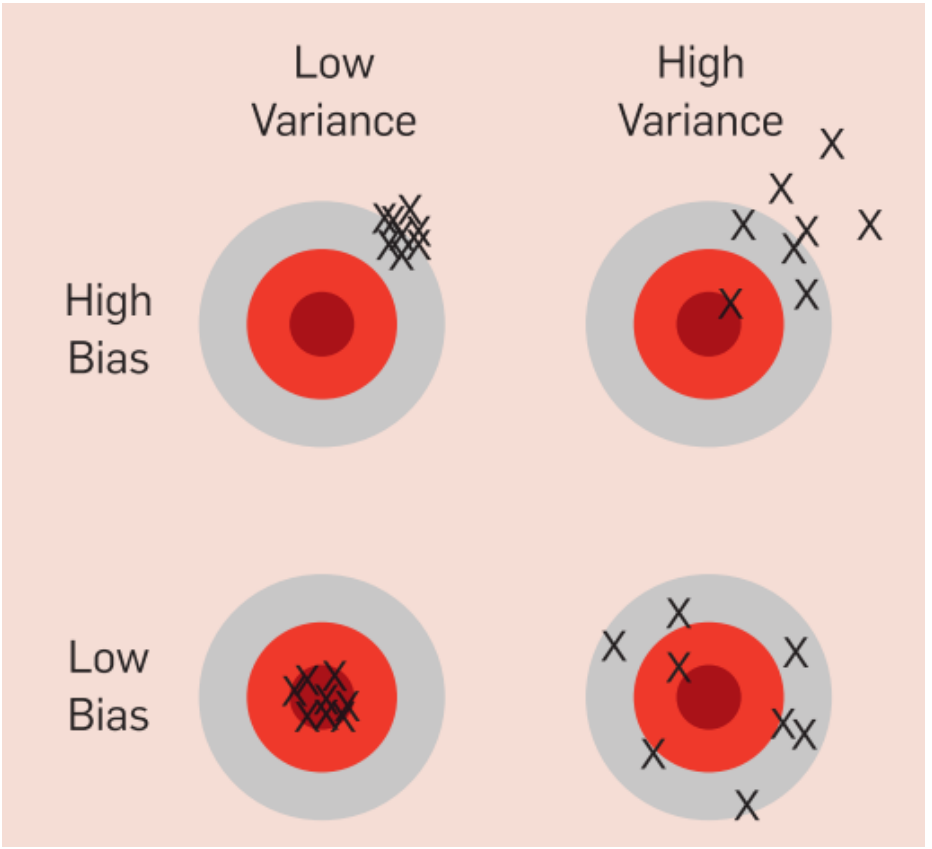


Principais desafios em AM

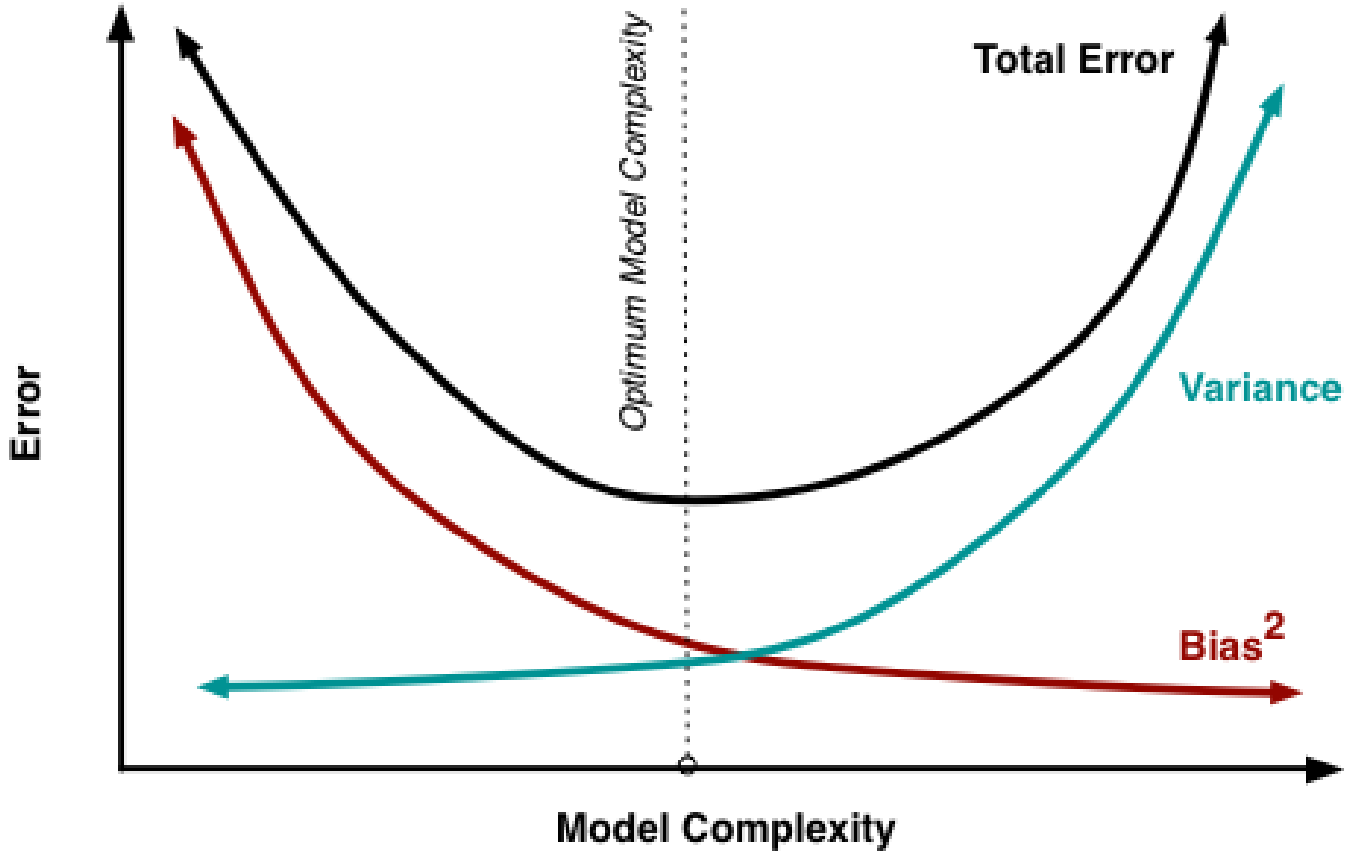
- Underfitting
  - Occorre quando o modelo é muito simples para a natureza dos dados;
- Soluções para evitar underfitting
  - Selecionar um modelo mais poderoso, com mais parâmetros;
  - Feature engeneering;
  - Reduzir os constraints no modelo (e.g. reduzir os valores dos parâmetros da regularização);

Dilema viés-variância

- O erro de predição pode ser decomposto em viés e variância (*Bias and Variance Tradeoff*).
- Viés é a tendência do modelo a aprender a mesma padrão errado;
- Variância é a tendência em aprender comportamento aleatórios que não de acordo com o sinal nos dados;



Dilema viés-variância



Exercícios

Nota: As terminologias mostradas nesta aula serão utilizadas com frequência no decorrer desta disciplina. Assim, é importantíssimo que o aluno responda o seguinte exercício antes da próxima aula.

1. Como você definiria aprendizagem de máquina?

2. Você poderia mencionar quatro tipos de problemas onde aprendizagem de máquina seria aplicável?

3. O que é um conjunto de treinamento com labels?

4. Quais são as duas tarefas mais comuns em aprendizagem de máquina supervisionada?

5. Cite duas tarefas comuns em aprendizagem de máquina não supervisionada?

6. Que tipo de aprendizagem de máquina seria usada para permitir que um robô andasse em vários terrenos desconhecidos?

Exercícios

7. Qual tipo de algoritmo você usaria para segmenter consumidores em múltiplos grupos?

8. O problema de detecção de spam é supervisionado ou não supervisionado?

9. O que é um método online em aprendizagem de máquina?

10. Que tipo de algoritmo de aprendizagem de máquina que usa uma métrica de similaridade para realizar a predição?

11. Mencione 4 dos principais desafios em AM.

12. Se o modelo preditivo tem boa perfoamance no conjunto de treino mas generaliza mal em novos exemplos, o que está acontecendo? Você pode enumerar 3 soluçoes potenciais?

13. O que é um conjunto de teste e porque é importante usá-lo?

14. O que é um conjunto de validação e porque é importante usá-lo?

15. Qual a relação do dilema viés-variância com overfitting e underfitting?

Referências

- Géron, Aurélien. Hands-on machine learning with Scikit-Learn and TensorFlow: concepts, tools, and techniques to build intelligent systems. " O'Reilly Media, Inc.", 2017.
- Müller, Andreas C., and Sarah Guido. Introduction to machine learning with Python: a guide for data scientists. " O'Reilly Media, Inc.", 2016.
- Zheng A., Casari A. Feature Engineering for Machine Learning: principles and techniques for data scientists. " O'Reilly Media, Inc.", 2018.
- VanderPlas, Jake. Python data science handbook: essential tools for working with data. " O'Reilly Media, Inc.", 2016.
- Domingos, Pedro. "A few useful things to know about machine learning." Communications of the ACM 55.10 (2012): 78-87.
- Russell, Stuart J., and Peter Norvig. Artificial intelligence: a modern approach. Malaysia; Pearson Education Limited,, 2016.