Machine Learning

Formação Metodológica do MAPE

Felipe Lamarca

Instituto de Estudos Sociais e Políticos (IESP-UERJ)

2025-09-19

Um curso de ML, em média

Se eu tivesse um semestre, provavelmente falaríamos de:

- k-NN
- Regressão linear
- Regressão logística
- Processos gaussianos
- Técnicas de seleção de modelos
- Técnicas de redução de dimensionalidade
- Redes neurais
- ...

Hoje

Não falaremos dos vários algoritmos e modelos utilizados em machine learning. Antes, e talvez mais importante, vamos falar de *machine learning* como um **paradigma** próprio.

Uma anedota: machine learning $\times$ modelagem estatística

No 5º período da graduação em ciência de dados, cursei machine learning e modelagem estatística. Uma aula era seguida da outra, às segundas e quartas.

Quais são as diferenças entre uma coisa e outra? Qual é mais legal?

Uma anedota: machine learning $\times$ modelagem estatística

No 5º período da graduação em ciência de dados, cursei machine learning e modelagem estatística. Uma aula era seguida da outra, às segundas e quartas.

Qual é a diferença entre um e outro? Qual é mais legal?

A saber

Esse foi o motivo pelo qual eu e minha namorada brigamos pela primeira vez!

Na aula de machine learning...

Figure 1: Diego Parente, professor de Machine Learning

O tema da aula era regressão linear. Em particular, Diego terminou a aula mostrando, sob certas condições, o estimador por máxima verossimilhança e o estimador por mínimos quadrados coincidem no caso normal. De fato, sendo $\mathcal{L}$ a verossimilhança:

$$\mathcal{L}(\beta) = (2\pi\sigma^2)^{-n/2} \exp\left(-\frac{1}{2\sigma^2}(y - X\beta)^T(y - X\beta)\right),$$

queremos β que minimiza a log-verossimilhança negativa $l(\beta)$:

$$-l(\beta) = \frac{n}{2}\log(2\pi) + \frac{n}{2}\log\sigma^2 + \frac{1}{2\sigma^2}(y - X\beta)^T(y - X\beta)$$

Então...

$$\min_{\beta} -l(\beta) = \min_{\beta} ||y - X\beta||_2^2$$

Uau! ✋ 😮 ✋

Na aula de modelagem estatística...

Figure 2: Luiz Max, professor de modelagem estatística

O tema da aula era regressão linear. Em particular, Max terminou a aula mostrando que, sob certas condições, o estimador por máxima verossimilhança e o estimador por mínimos quadrados coincidem no caso normal. De fato, sendo $\mathcal{L}$ a verossimilhança,

$$\mathcal{L}(\beta) = (2\pi\sigma^2)^{-n/2} \exp\left(-\frac{1}{2\sigma^2}(y - X\beta)^T(y - X\beta)\right)$$

queremos β que minimiza a log-verossimilhança negativa $l(\beta)$:

$$-l(\beta) = \frac{n}{2}\log(2\pi) + \frac{n}{2}\log\sigma^2 + \frac{1}{2\sigma^2}(y - X\beta)^T(y - X\beta).$$

Então...

$$\min_{\beta} -l(\beta) = \min_{\beta} ||y - X\beta||_2^2$$

Ué?! 😕

E tem diferença, então?

Mas e aí? Tem diferença, então?

E tem diferença, então?

Tem! Mas a diferença não tá na matemática. Trata-se sobretudo de uma diferença de objetivos.

Eu destacaria, em particular: enquanto a modelagem estatística quer reconstruir o processo gerador dos dados (Lego I e II), machine learning quer acertar as previsões sobre a variável de interesse. Isso tem implicações:

- Foco em previsão ao invés de explicação
- Seleção automática de variáveis
- Menor preocupação com interpretabilidade
- Alta dimensionalidade, conjuntos enormes de dados

O que é Machine Learning?

> *Machine learning, a subset of AI, uses algorithms to analyze data, identify patterns, and make predictions. It learns from data on its own, improving over time.* – *Microsoft*

Em geral, a ideia do aprendizado de máquina é encontrar a função f que melhor relaciona as variáveis preditoras à variável resposta, ou *target*.

Regressão é machine learning

Suponha que nosso objetivo é estimar a **probabilidade** de um candidato ser eleito em 2026. Para isso, temos à nossa disposição um banco de dados históricos (2012-2020) que inclui o sexo biológico dos candidatos, seus gastos de campanha, uma variável que indica se o candidato é incumbente ou desafiante e o resultado das urnas.

Regressão é machine learning

Sendo X a matriz de covariáveis (*preditores*) e Y a variável resposta (*target*), podemos estimar os coeficientes de um modelo de regressão logística:

$$\mathbb{P}(Y_i = 1) = \text{logit}^{-1}(X_i\beta)$$

Estimados os coeficientes, podemos estimar a probabilidade de um candidato qualquer ser eleito. Machine learning é isso, mas de maneira um pouco mais sofisticada.

Tipos de problema

1. Aprendizado `supervisionado`: quando o *target* é conhecido
   - queremos estimar a probabilidade de um candidato ser eleito
2. Aprendizado `não-supervisionado`: quando o *target* é desconhecido
   - queremos agrupar os eleitores em grupos; fazemos análise temática de *corpus* textuais
3. Aprendizado `por reforço`: quando o agente aprende por tentativa e erro, recebendo recompensas ou punições
   - humanos avaliam respostas do modelo para guiar comportamento (RLHF)

Treino, teste e validação

Em geral, dividimos o nosso banco de dados 2 ou 3 subconjuntos:

- O conjunto de **treino** é usado para ensinar o modelo a reconhecer padrões (i.e., *treinar* o modelo)
- O conjunto de teste contém dados nunca vistos pelo modelo e serve para medir sua capacidade de generalização (i.e., *testar* o modelo)
- O conjunto de validação ajuda a ajustar os hiperparâmetros, evitando overfitting.

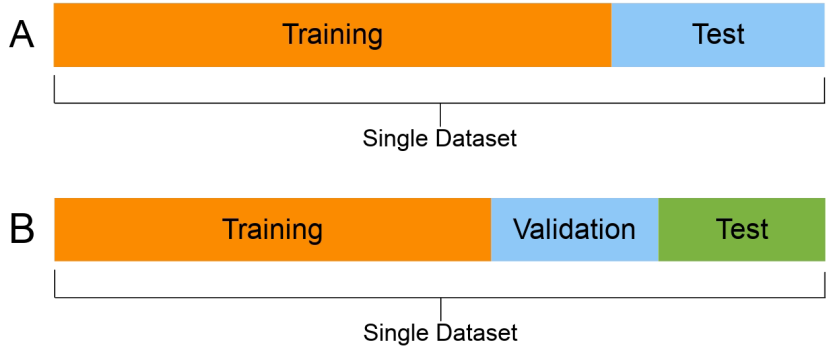


Figure 3: Treino, teste e validação

Overfitting e underfitting

- **Underfitting**
  - Modelo muito simples, não consegue capturar os padrões dos dados.
  - Erro alto no treino **e** no teste.
- **Overfitting**
  - Modelo muito complexo, "decora" o conjunto de treino.
  - Erro baixo no treino, mas alto no teste.
- **Objetivo:** encontrar o **equilíbrio** $\rightarrow$ boa performance no treino **e** no teste.

Overfitting e underfitting

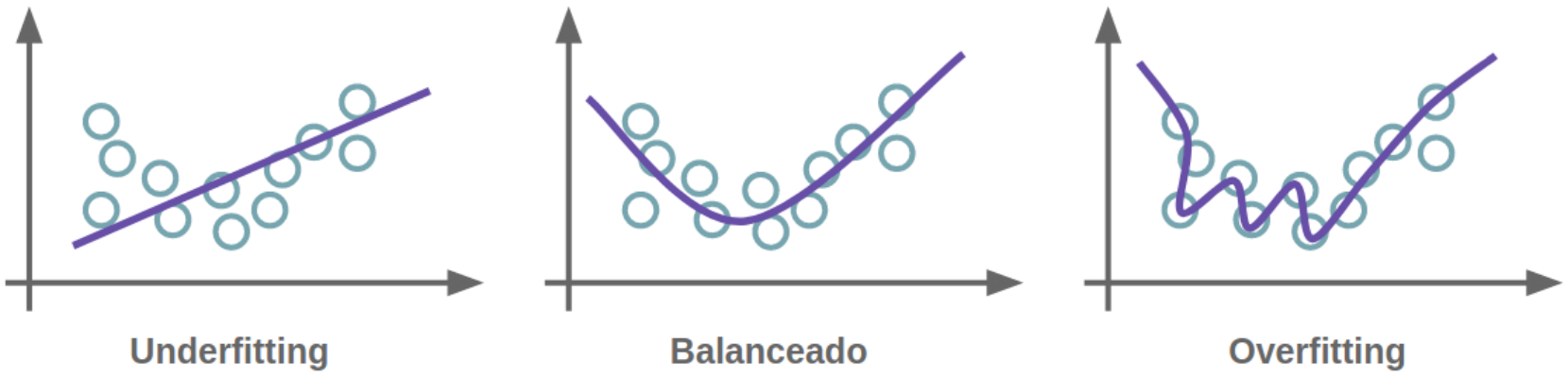


Figure 4: Overfitting e underfitting

Avaliação de modelos – Regressão

- **Erro Médio Absoluto (MAE):** média dos erros em valor absoluto

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

- **Erro Quadrático Médio (MSE):** penaliza mais os erros grandes

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

- **Raiz do Erro Quadrático Médio (RMSE):** raiz do MSE

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

Avaliação de modelos – Classificação

As métricas de classificação se baseiam, principalmente, na matriz de confusão:

	Predicted Positive	Predicted Negative
Actual Positive	True Positive (TP)	False Negative (FN)
Actual Negative	False Positive (FP)	True Negative (TN)

Avaliação de modelos – Classificação

- **Acurácia**: proporção de acertos totais

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

- **Precisão**: dos preditos como positivos, quantos realmente são positivos?

$$Precision = \frac{TP}{TP + FP}$$

- **Recall (Sensibilidade)**: dos positivos reais, quantos foram identificados?

$$Recall = \frac{TP}{TP + FN}$$

- **F1-Score**: equilíbrio entre precisão e recall

$$F1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall}$$

É possível treinar modelos para maximizar uma ou outra métrica, dependendo do target.

Outros modelos – k-NN

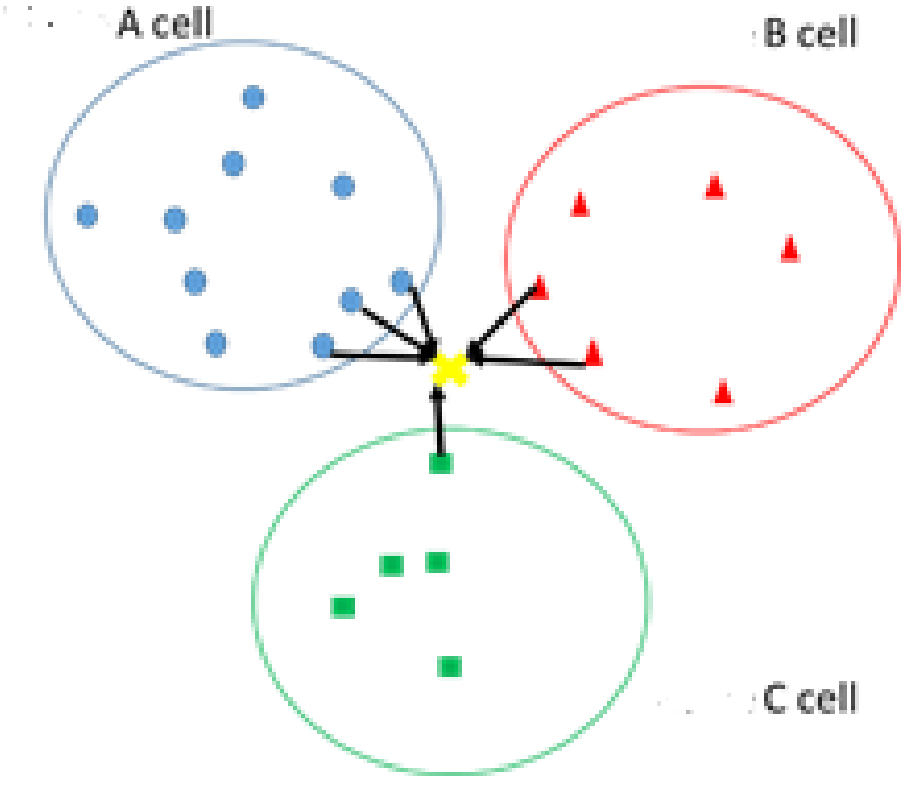
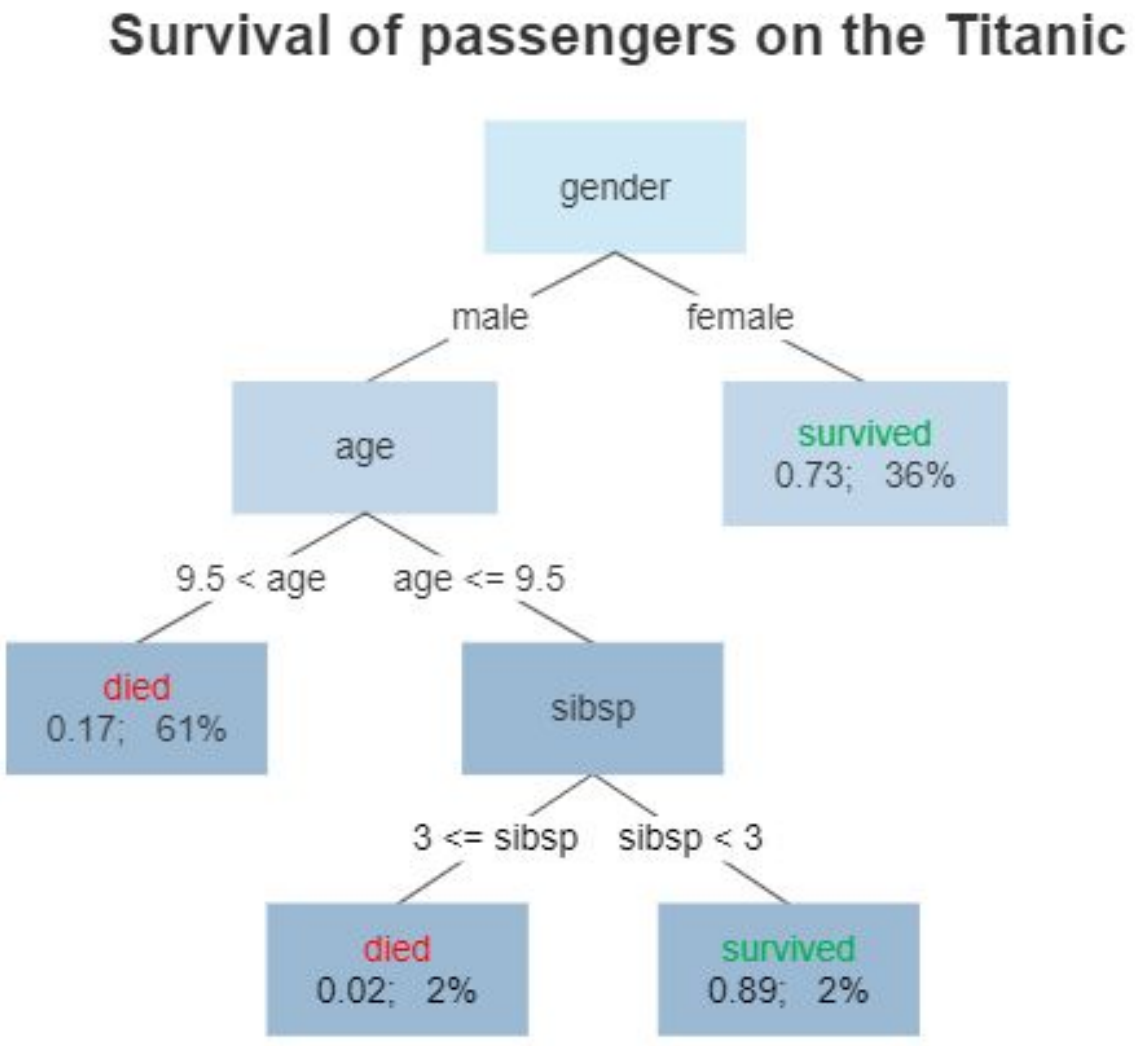


Figure 5: k-nearest neighbors

Outros modelos – Árvores de decisão



Outros modelos – Redes neurais

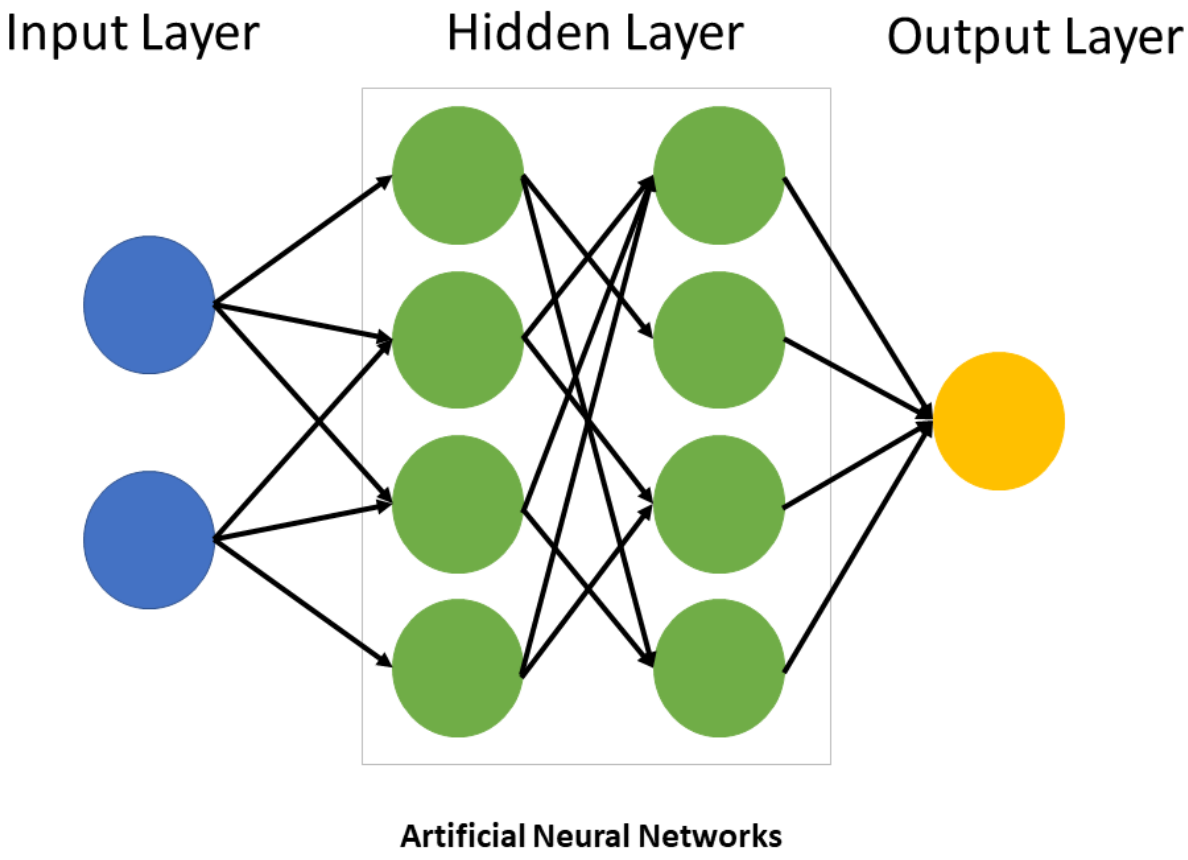


Figure 7: Neural networks

Tópicos especiais (que não veremos)

- Regularização (L1, L2, Dropout) para evitar *overfitting*
- k-fold cross-validation
- Técnicas de redução de dimensionalidade (PCA, UMAP, t-SNE)

Hands-on

Código!