Insper

MACHINE LEARNING · 2026.2

Caderno de atividades

As 27 aulas, de F1 a E8

Humberto Sandmann

Adaptado do caderno de Machine Learning da disciplina Inteligência Artificial,
de Fabio Roberto de Miranda, Diego Pavan Soler e Humberto Sandmann.

Nome: _____

Dupla do projeto: _____

Versão 2026.2 · outubro de 2026

Como usar este caderno

Cada aula de Machine Learning tem uma página no site, com texto, código, simuladores e um lab. Este caderno é o seu lado da aula: uma atividade por aula, feita no papel, para treinar o que a página ensina com números pequenos o bastante para fazer à mão. Os cenários vêm das aulas sempre que possível (o Hexapawn, Galton e as alturas, os distritos da Califórnia, o Titanic, os pinguins, os vinhos, as meias-luas, os 20 pacientes das métricas, os e-mails de spam, os dígitos e o banco); os casos curtos criados para o caderno trazem no próprio enunciado tudo o que é preciso para resolvê-los.

Cada atividade tem quatro partes

1. **A pergunta da aula**, a mesma da página. Ao terminar a atividade, você deve saber respondê-la.
2. **O que você precisa saber**: um resumo curto, para consultar enquanto resolve. Ele não substitui a página da aula (o QR code no topo leva até ela).
3. **Os exercícios**, em ordem de dificuldade. Faça na ordem: cada um prepara o seguinte.
4. **As respostas comentadas**, no fim do caderno. Tente antes de olhar; quando errar, leia o comentário inteiro, porque é nele que está o raciocínio.

Os três níveis

● AQUECIMENTO	reconhecer e aplicar uma ideia da aula diretamente: uma definição, uma conta curta, uma classificação.
●● PRÁTICA	usar a ideia num caso parecido com o da aula, com contas e uma decisão justificada.
●●● DESAFIO	combinar várias ideias num caso aberto, como se faz no projeto e nas avaliações: formular, diagnosticar, escrever para alguém.

As cores

dados e divisão · modelo · avaliação	as etapas do fluxo que a aula trabalha: preparar e dividir os dados, treinar o modelo, avaliar. As mesmas cores da folha, do mapa e das cartas.
O que você precisa saber	o resumo da aula; os termos-chave saem nesta cor.
Confira no simulador	um QR code que abre o simulador da aula. Não pede respostas novas: serve para ver, mexendo, o que você acabou de calcular.
resposta × Cuidado	nas respostas comentadas: em verde, a resposta e o raciocínio; em vermelho, o erro mais comum.

O fio destas aulas. As 27 atividades seguem a ordem do curso, em sete temas: **F**, os fundamentos (o que é aprender com dados); **D**, os dados (olhar, preparar e encadear tudo num pipeline sem vazamento); **U**, o não supervisionado e o texto (reduzir dimensões, agrupar, transformar texto em vetores, achar tópicos); **R** e **C**, a regressão e a classificação (os modelos, as métricas, o threshold e a validação honesta); **M**, os modelos que vencem em dados tabulares (escolher, árvores, florestas e boosting); e **E**, as extensões (redes neurais, explicabilidade, AutoML, produção e a fronteira). Cada tema usa os anteriores.

Convenções: vírgula decimal (0,75); nomes de métricas e métodos como nas páginas das aulas; “à mão” quer dizer sem computador (uma calculadora simples ajuda).

Sumário

F1. Introdução e história 4

F2. O panorama de ML 8

D1. Análise exploratória de dados 12

D2. Pré-processamento de dados 17

D3. Pipelines 21

U1. Redução de dimensionalidade 26

U2. Agrupamento 32

R1. Regressão linear 38

R2. Gradiente descendente e regularização 44

C1. Regressão logística 50

C2. k-vizinhos mais próximos (k-NN) 56

C3. Métricas de classificação 63

C4. ROC, PR e dados desbalanceados 68

C5. Máquinas de vetores de suporte (SVM) 74

C6. Validação cruzada e vazamento de dados 79

M1. Árvores de decisão 84

M2. Random forest 89

M3. Gradient boosting 93

M4. Seleção de modelos 99

E1. Agrupamento hierárquico 105

E2. Representação de texto 111

E3. Modelagem de tópicos e BERTopic 117

E4. Redes neurais 124

E5. Explicabilidade 130

E6. AutoML 135

E7. MLOps 139

E8. A fronteira **144**

Respostas comentadas **149**

ATIVIDADE DA AULA

F1

Introdução e história

insper.github.io/ML/pt/2026.2/classes/introduction/

respostas comentadas na página 149

A pergunta da aula. Um filtro de spam feito de regras escritas à mão começa a falhar assim que os spammers mudam de truque. Quando vale a pena trocar as regras por um programa que aprende com exemplos e o que quer dizer, exatamente, que ele "aprendeu"?

ETAPAS DO FLUXO NESTA AULA dados e divisão | modelo | avaliação

O que você precisa saber

- **Machine learning (ML)**: em vez de escrever as regras, escrevemos um programa que as **infere a partir de exemplos**. O fluxo se inverte: regras + dados → respostas vira dados + respostas → **modelo**, uma aproximação das regras usada em casos novos.
- **Mitchell (1997)**: um programa aprende com a experiência E numa classe de tarefas T, com a medida de desempenho P, se o desempenho em T, medido por P, **melhora com** E. Guardar mais dados não basta: P tem de subir.
- Aprender compensa quando as regras são **desconhecidas**, **mudam**, são **numerosas demais** ou precisam de **personalização**. Uma regra curta, fixa e conhecida se programa.
- **Por que agora**: as ideias são antigas (mínimos quadrados, 1805), mas só nas últimas duas décadas se juntaram **dados**, **poder computacional** (as unidades de processamento gráfico, GPUs, e a nuvem) e **software aberto** (scikit-learn, XGBoost, PyTorch).
- Três lições da história: os **clássicos nunca foram embora**; os **ciclos de hype** são reais (dois invernos da inteligência artificial, a IA) e o antídoto é a avaliação honesta; os métodos modernos são **composições** de ideias clássicas.
- A **máquina de caixas de fósforos** (Gardner, 1962) aprende o Hexapawn só com feedback: sorteia uma conta da caixa da posição e, quando perde, tira a conta da jogada perdedora. Aprender é **ajustar parâmetros** (as contas) com base em feedback: aprendizagem por reforço.

1 Três lições da trajetória

• AQUECIMENTO

Escreva ao lado de cada fato a lição da história que ele ilustra: **(1)** os clássicos nunca foram embora; **(2)** os ciclos de hype são reais; **(3)** os métodos modernos são composições de ideias clássicas.

(a) 1969: Minsky e Papert provam que um perceptron sozinho não resolve o XOR. _____

(b) Num problema de regressão, a equipe começa pelos mínimos quadrados (1805). _____

(c) O BERTopic (2022) encadeia embeddings, redução de dimensão, agrupamento e TF-IDF (frequência do termo × inverso da frequência nos documentos). _____

(d) A regressão logística, de 1958, continua sendo um cavalo de batalha em produção. _____

(e) O gradient boosting é gradiente descendente funcional sobre árvores de decisão. _____

(f) 1973: o Relatório Lighthill conclui que a IA não entregou o que prometeu. _____

(g) Que antídoto a aula propõe para a lição 2? .....

2 Programar ou aprender? • AQUECIMENTO

Para cada tarefa, diga se você escreveria a regra à mão (P) ou aprenderia com exemplos (A). Quando for A, diga qual dos quatro motivos pesa mais: regras desconhecidas, que mudam, numerosas demais ou personalização.

Tarefa	P ou A?	se A, o motivo
(a) Conferir os dois dígitos verificadores de um CPF (a fórmula é pública).		
(b) Ler a placa de um carro numa foto tirada por uma câmera de rua, de dia ou de noite.		
(c) Barrar compras fraudulentas com cartão; os fraudadores mudam de tática quando uma regra os pega.		
(d) Sugerir a próxima música para cada ouvinte de um aplicativo.		
(e) Converter uma temperatura de graus Celsius para Fahrenheit.		
(f) Dizer se uma radiografia de tórax mostra pneumonia.		

3 Tarefa, experiência e desempenho • AQUECIMENTO

Descreva cada sistema no vocabulário de Mitchell.

Sistema	tarefa T	experiência E	desempenho P
(a) A máquina de caixas de fósforos que joga Hexapawn			
(b) Um leitor de dígitos manuscritos nos cheques de um banco			
(c) Um sistema que prevê, em dias, o tempo de internação de cada paciente			

(d) Um programa de e-mail guarda todas as mensagens que recebe há um ano, mas continua barrando 70% do spam, como no primeiro dia. Pela definição de Mitchell, ele aprendeu? Por quê?

. . . . . .

. . . . . .

4 Uma regra escolhida pelos dados

●● PRÁTICA

Um filtro de spam vai ser uma regra de uma linha, mas quem escolhe a regra são os dados. Há três candidatas: **R1** = "spam se contém *grátis*"; **R2** = "spam se contém *prêmio*"; **R3** = "spam se contém *grátis* ou *prêmio*". Os e-mails 1 a 8 já foram rotulados por uma pessoa; A a D chegaram depois.

	rotulados								**novos**			
e-mail	1	2	3	4	5	6	7	8	A	B	C	D
contém *grátis*	✓	✓			✓			✓		✓		
contém *prêmio*	✓		✓				✓		✓		✓	
é spam?	sim	sim	sim	não	não	não	sim	não	sim	sim	não	não

(a) Quantos dos 8 e-mails rotulados cada regra acerta?

(b) Qual regra os dados escolhem? No fluxo invertido (dados + respostas → modelo), o que entrou e o que saiu?

(c) Quantos dos 4 e-mails novos a regra escolhida acerta? O desempenho é o mesmo dos rotulados?

(d) Os spammers passam a escrever *pr3mio*. Qual dos quatro motivos para aprender isso ilustra? O que fazer?

5 A caixa de fósforos, à mão

●● PRÁTICA

Numa posição do Hexapawn a máquina tem três jogadas, cada uma com contas de uma cor na caixa: verde (sempre leva à vitória), amarela e vermelha (sempre levam à derrota). A caixa começa com uma conta de cada cor; a máquina sorteia uma conta ao acaso.

(a) Na primeira visita a essa posição, qual é a chance de a máquina jogar a verde?

(b) *Só punição.* Nas duas primeiras visitas ela sorteia a amarela e depois a vermelha e perde as duas partidas. O que sobra na caixa? Qual é a chance de jogar a verde na terceira visita?

(c) *Punição e recompensa* (cada vitória acrescenta uma cópia da conta jogada). Recomeçando com uma conta de cada cor, as três primeiras visitas sorteiam verde (vence), vermelha (perde) e verde (vence). Complete a tabela.

(d) Nos termos de Mitchell, quem são T, E e P? Quais são os "parâmetros" que a máquina ajusta?

	verdes	**amarelas**	**vermelhas**	**chance de jogar a verde**
no início	1	1	1	1/3
depois da visita 1				
depois da visita 2				
depois da visita 3				

CONFIRA NO SIMULADOR

A máquina de caixas de fósforos

Jogue uma partida contra a máquina (você é o azul e começa). Depois aperte *treinar N partidas* e veja as contas sumirem das caixas enquanto a taxa de vitórias sobe: a partir de certo ponto a máquina não pode mais ser vencida.

6 A proposta do fornecedor

DESAFIO

Um hospital recebe esta proposta: "nosso sistema sinaliza as radiografias de tórax com pneumonia e acerta 99% das vezes". A direção pede a sua opinião por escrito.

(a) Descreva o sistema com T, E e P.

(b) Por que esse é um caso para aprender com dados e não para regras escritas à mão?

(c) Quais das três forças (dados, poder computacional, software) tornaram esse sistema viável só nas últimas décadas? Justifique cada uma.

(d) Que duas perguntas você faria ao fornecedor antes de acreditar nos 99%? Relacione com a lição dos invernos da IA.

ATIVIDADE DA AULA

F2

O panorama de ML

insper.github.io/ML/pt/2026.2/classes/ml-landscape/

respostas comentadas na página 149

A pergunta da aula. Um modelo acerta 99,8% das vezes nos dados em que foi treinado e só 62% em dados novos. Antes de escolher qualquer algoritmo, que mapa ajuda a entender esse resultado: que tipo de problema é, em que passo do projeto estamos e o que faz um modelo funcionar em dados que nunca viu?

ETAPAS DO FLUXO NESTA AULA dados e divisão | modelo | avaliação

O que você precisa saber

- Os paradigmas de machine learning (ML) se separam pelo **feedback**. **Supervisionado**: entradas x e respostas y; aprende f com $f(x) \approx y$ em dados novos (**regressão** se y é contínuo, **classificação** se é categórico). **Não supervisionado**: só x; procura estrutura. **Por reforço**: um agente recebe recompensas. No meio-termo: semissupervisionado, **autossupervisionado** (rótulos tirados dos próprios dados) e supervisão fraca.
- O projeto é um **ciclo**: (1) formular, (2) obter e explorar os dados, (3) prepará-los, (4) treinar, (5) avaliar, (6) implantar e monitorar. Duas setas voltam: *iterar* (de 5 para 3) e *os dados mudam*, o drift (de 6 para 2). O grosso do trabalho fica em 2 e 3.
- **Underfitting**: simples demais, erra muito no treino e no teste. **Overfitting**: decora o ruído, erra pouco no treino e muito no teste. Quem avisa é o erro de teste.
- **Regra de ouro**: nunca avalie um modelo nos dados em que ele foi treinado; o teste simula o futuro e é usado uma vez. **Sem almoço grátis** (Wolpert, 1996): nenhum algoritmo é melhor em todos os problemas; compare famílias e valide.
- Antes de um modelo decidir sobre pessoas: **justiça**, **privacidade**, **transparência** e **ciclos de retroalimentação**. "O modelo disse isso" não justifica decisão nenhuma.

1 Em que passo do ciclo? • AQUECIMENTO

Escreva o passo do ciclo (1 a 6) de cada frase de uma equipe que prevê o preço de venda de imóveis. Nas duas últimas, diga também para que passo o ciclo volta.

(a) "O sucesso será errar, em média, menos de 10% do preço de venda." _____

(b) "Um quinto das áreas está faltando e quase todas nos anúncios antigos." _____

(c) "Preenchemos as áreas faltantes com a mediana das linhas de treino." _____

(d) "Ajustamos uma regressão linear e uma árvore de decisão." _____

(e) "Medimos o erro dos dois em imóveis que nenhum deles viu." _____

(f) "A árvore erra nos imóveis antigos: vamos criar a coluna idade do prédio." _____

(g) "Um ano depois de entrar no ar, o erro dobrou sem ninguém mexer no código." _____

2 Que feedback?

• AQUECIMENTO

Para cada situação, diga o paradigma (supervisionado, não supervisionado, por reforço ou um do meio-termo) e, se for supervisionado, se é regressão ou classificação.

Situação	paradigma	regressão ou classificação?
(a) Prever o consumo de energia de um prédio amanhã, em kWh, com três anos de consumo diário registrado.		
(b) Separar 50 mil notícias em grupos de assuntos parecidos, sem uma lista de assuntos.		
(c) Um robô aprende a andar ganhando pontos quando avança sem cair.		
(d) Dizer se um tumor é benigno ou maligno a partir de exames já laudados por médicos.		
(e) Prever quantos dias um paciente vai ficar internado.		
(f) Resumir 300 medidas dos sensores de uma máquina em 3 números.		
(g) Pré-treinar um modelo de linguagem escondendo uma palavra de cada frase e prevendo-a, sem rótulo humano nenhum.		

3 A nota que não vale

•• PRÁTICA

Um "modelo" de spam guarda os 10 e-mails de treino com os rótulos. Para um e-mail que já viu, devolve o rótulo guardado; para qualquer outro, responde "não spam". O teste tem 10 e-mails novos, nenhum igual aos de treino, e 3 deles são spam.

(a) Qual é a acurácia (fração de acertos) dele no treino? E no teste?

(b) Compare com uma regra que responde sempre "não spam". O que o modelo aprendeu?

(c) Que número vai no relatório? Que regra da aula a escolha obedece?

(d) Outra equipe testou 20 variações de um modelo e ficou com a de maior nota no teste. Essa nota ainda mede o desempenho em dados novos? Por quê?

4 O diagnóstico pelos dois erros

●● PRÁTICA

Num conjunto de dados sintético, polinômios de graus diferentes foram ajustados a 15 pontos de treino e medidos em 2.000 pontos novos. O erro é o erro quadrático médio (MSE, do inglês *mean squared error*): a média de (previsto − real)^2. O ruído somado aos dados tem variância $\sigma^2 = 9$.

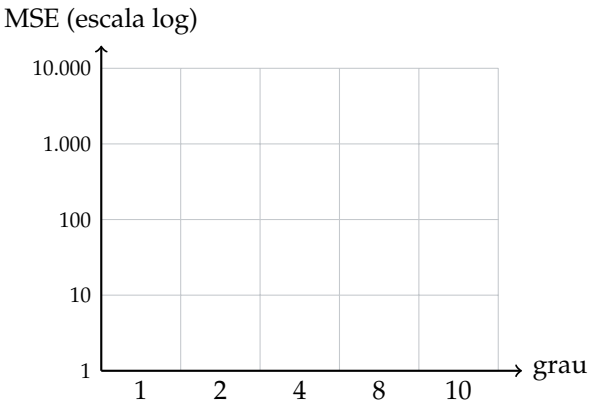
grau	1	2	4	8	10
MSE de treino	29,9	27,1	7,2	4,3	3,6
MSE de teste	38,1	31,4	10,4	61,2	6.064,5
diagnóstico					

(a) Complete o diagnóstico: underfitting, bom ajuste ou overfitting.

(b) Esboce as duas curvas no gráfico (o eixo do erro está em escala log) e marque a linha do ruído.

(c) Nenhum grau tem MSE de teste abaixo de 9, mas o grau 10 tem MSE de treino 3,6. Por que o primeiro limite existe e por que o 3,6 é um mau sinal?

(d) Com 100 pontos de treino em vez de 15, qual erro de teste você espera que caia mais? Por quê?



CONFIRA NO SIMULADOR

Generalização: underfitting e overfitting

Mova o grau de 1 a 15 e acompanhe os dois erros. Depois aperte *mais dados* e suba o grau de novo: qual das duas curvas muda mais?

5 Antes de decidir sobre pessoas

●● PRÁTICA

Para cada caso, diga qual das quatro perguntas da aula (justiça, privacidade, transparência, ciclo de retroalimentação) ele levanta e o que você verificaria.

(a) Um modelo de crédito treinado com 20 anos de decisões de gerentes aprova 70% dos pedidos num bairro e 35% noutro, de renda parecida.

(b) Uma clínica publica "sem nomes" uma tabela de pacientes com CEP, data de nascimento e sexo.

(c) Uma candidata recusada por uma triagem automática pergunta o motivo; a empresa responde que "o modelo decidiu".

(d) Uma plataforma recomenda os vídeos que cada pessoa mais clica e é retreinada toda semana com os cliques que as próprias recomendações geraram.

6 O mapa de um projeto

●●● DESAFIO

Uma rede de farmácias quer saber, toda segunda-feira, quantas caixas de cada remédio cada loja vai vender na semana, para abastecer as lojas. Há quatro anos de vendas semanais.

(a) Qual é o paradigma e o tipo de problema? O que é uma linha da tabela?

(b) Que conjunto de teste "simula o futuro" aqui? Por que sortear semanas ao acaso não serve?

(c) Um modelo muito flexível erra, em média, 1 caixa nas semanas de treino e 9 nas de teste; um simples erra 6 nas duas. Diagnostique os dois e diga qual levar para a loja.

(d) Por que vale a pena experimentar mais de uma família de modelos antes de decidir?

(e) Seis meses depois de implantado, o erro sobe. Que seta do ciclo explica isso e o que fazer?

ATIVIDADE DA AULA

D1

Análise exploratória de dados

insper.github.io/ML/pt/2026.2/classes/eda/

respostas comentadas na página 150

A pergunta da aula. Uma planilha com os 20.640 distritos da Califórnia no censo de 1990 chega antes de qualquer modelo; o alvo é o valor mediano das casas. O que é preciso descobrir sobre esses dados, olhando só a parte de treino, antes de confiar neles?

ETAPAS DO FLUXO NESTA AULA **dados e divisão**

O que você precisa saber

- O produto da análise exploratória de dados (EDA, do inglês *exploratory data analysis*) é uma **lista escrita** de fatos, problemas e hipóteses. **Divida antes de explorar**: tudo o que é estimado e alimenta uma decisão vem só do treino. Primeiro, **o que é uma linha** (aqui, um distrito, não uma casa).
- A **média** é puxada pela cauda; a **mediana** só conta posições. Média bem acima da mediana: cauda longa à direita. Dispersão: desvio padrão s (divide por $n-1$) e o intervalo interquartil, $\text{IQR} = Q_3 - Q_1$.
- **Outliers**: a regra do IQR sinaliza o que fica fora de $[Q_1 - 1{,}5\,\text{IQR};\ Q_3 + 1{,}5\,\text{IQR}]$; a de $|z| > 3$, com $z = (x - \bar{x})/s$, sofre **mascaramento**. As regras só apontam candidatos: **erro, raro mas real ou outra coisa**?
- **Sempre faça gráficos** (Anscombe): poucos bins escondem **tetos**, picos na borda de valores truncados (500.001 vale "pelo menos 500 mil").
- **Pearson** mede o quanto os pontos seguem uma reta; **Spearman** é o Pearson dos postos e mede se y sobe quando x sobe. $r \approx 0$ não quer dizer "sem relação". Uma tendência no total pode se inverter dentro dos grupos (**Simpson**): condicione na terceira variável.
- **Faltantes**: MCAR (*missing completely at random*: ao acaso), MAR (*at random*: depende de outra coluna), MNAR (*not at random*: do próprio valor). Compare as linhas com e sem o valor; guarde um indicador.

1 Centro, dispersão e cercas

• AQUECIMENTO

Cômodos por domicílio é `total_rooms / households`, em que `households` conta só os domicílios *ocupados*. Nove distritos têm, arredondando, 3, 4, 4, 5, 6, 7, 8, 9, 44.

(a) Calcule a média e a mediana. Tire o 44 e calcule de novo. Qual das duas se mexeu?

(b) Com a convenção do pandas (o quantil p fica na posição $p\,(n-1)$ da lista ordenada, contando do 0), ache Q_1, Q_3 e o IQR.

(c) Calcule as cercas da regra do IQR. O 44 é sinalizado?

2 O que a outra regra deixa passar

●● PRÁTICA

Continue com os nove distritos do exercício anterior (média 10).

(a) Calcule o desvio padrão s (divida por $n-1=8$) e o escore z do 44. A regra $|z|>3$ o sinaliza?

(b) Numa amostra de n valores, nenhum $|z|$ passa de $(n-1)/\sqrt{n}$. Quanto dá para $n=9$? O que isso diz sobre a regra $|z|>3$ aqui?

(c) Escreva uma hipótese para cada explicação do 44: um **erro** de registro, um caso **raro mas real** e **outra coisa** (dica: releia o que `households` conta). O que você faria com o distrito em cada caso?

3 Um ponto fora da ordem

●● PRÁTICA

Cinco pontos: $x=1,2,3,4,5$ e $y=1,2,3,4,0$. Os quatro primeiros estão numa reta perfeita.

x	y	$x-\bar{x}$	$y-\bar{y}$	produto	$(x-\bar{x})^2$	$(y-\bar{y})^2$	posto x	posto y
1	1							
2	2							
3	3							
4	4							
5	0							
			soma					

(a) Complete a tabela e calcule $r=\dfrac{\sum(x-\bar{x})(y-\bar{y})}{\sqrt{\sum(x-\bar{x})^2}\,\sqrt{\sum(y-\bar{y})^2}}$.

(b) Calcule o Spearman (o Pearson dos postos). Na aula, com $y=1,2,3,4,20$, deu Pearson 0,8 e Spearman 1. Por que o Spearman resiste ao 20, mas não ao 0?

(c) No California Housing, a latitude tem Pearson $-0{,}146$ e Spearman $-0{,}169$ com o valor das casas. Para o norte, grau a grau, a mediana do valor é (mil US$) 158,5; 194,9; 199,8; 84,5; 90,3; 236,5; 139,1; 96,2; 80,4. A latitude é inútil?

CONFIRA NO SIMULADOR

Pearson e Spearman

Carregue *os 5 pontos da tabela* (o exemplo da aula, com $y = 20$ no último) e arraste o último ponto para baixo, até $y = 0$, o caso do exercício 3: os dois coeficientes caem juntos. Depois leve-o de volta para bem alto: qual dos dois para de se mexer?

4 A barra mais alta

●● PRÁTICA

O alvo (valor mediano das casas) foi desenhado duas vezes. Com 10 bins, a última barra tem 1.243 distritos, 30% da barra mais alta. Com 50 bins, ela vira a barra mais alta do histograma: 1.030 distritos contra 47 da vizinha. O valor mais frequente da coluna é exatamente 500.001, em 965 distritos; a renda média deles é 7,83 (3,87 na tabela inteira) e só 27 são do interior.

(a) O que explica a barra? Por que ela quase some com 10 bins?

(b) Quanto vale, de verdade, um distrito registrado como 500.001?

(c) Esses 965 distritos são um sorteio ao acaso da tabela? Que consequência isso tem para um modelo treinado com eles?

(d) Dê duas atitudes defensáveis e uma indefensável.

5 Por que falta?

●● PRÁTICA

Na parte de treino (16.512 distritos), 163 não têm `total_bedrooms`. A comparação das linhas com e sem o valor:

	renda mediana	valor médio (US$)	idade mediana das casas
tem o valor	3,86	206.191	28,58
falta o valor	3,79	198.106	29,91

A média de 163 distritos sorteados ao acaso oscila, tipicamente, cerca de US$ 8.991 (o erro padrão).

(a) A diferença de valor médio entre os dois grupos é maior ou menor que essa oscilação? O que se conclui sobre o mecanismo? É uma prova?

(b) O que fazer com os faltantes? De quais linhas sai o número usado para preencher?

(c) Se os distritos sem o valor fossem quase todos do interior (mais baratos), qual seria o mecanismo mais provável e o que mudaria?

6 Melhor nas duas, pior no total

● ● ● DESAFIO

Duas linhas de produção fazem peças pequenas e grandes. As grandes são mais difíceis.

	peças pequenas (defeitos / peças)	**peças grandes** (defeitos / peças)	**total**
linha A	9 / 90	63 / 210	
linha B	36 / 240	21 / 60	

(a) Calcule a taxa de defeito de cada linha em cada tamanho e no total. Qual linha é melhor em cada tamanho? E no total?

(b) Explique a inversão. Qual é a terceira variável e por que ela é um confundidor?

(c) No California Housing, a fração de cômodos que são quartos tem $r = -0{,}256$ com o valor das casas na tabela inteira, mas r entre $+0{,}24$ e $+0{,}42$ dentro de cada uma das quatro faixas de renda mais baixas. O que você escreveria na lista de achados sobre essa coluna?

ATIVIDADE DA AULA

D2

Pré-processamento de dados

insper.github.io/ML/pt/2026.2/classes/preprocessing/

respostas comentadas na página 151

A pergunta da aula. Um classificador de k vizinhos mais próximos (k-NN) que olha as colunas cruas do Titanic acerta 67,6% dos passageiros de teste; com as mesmas colunas padronizadas, 82,1%. Os dados e o modelo são os mesmos: o que mudou? E o que mais uma tabela crua precisa antes de chegar a um modelo?

ETAPAS DO FLUXO NESTA AULA dados e divisão

O que você precisa saber

- **A escala decide a distância**: a fatia de cada coluna na distância quadrática média é a fatia dela na variância total. Escalonadores: padrão $(x-\mu)/\sigma$; min-max $(x-x_{\min})/(x_{\max}-x_{\min})$; robusto $(x-\text{mediana})/\text{IQR}$; max-abs $x/\max|x|$ (μ é a média, σ o desvio padrão com divisão por n, IQR $= Q_3 - Q_1$ o intervalo interquartil). Árvores não se importam.
- **Ajuste no treino, transforme o resto**: toda estatística (média, mediana, λ, lista de categorias) é aprendida só nas linhas de treino e aplicada, congelada, ao teste e à produção.
- Escalonar não muda a **forma**: a assimetria continua a mesma. Para uma cauda longa à direita, $\log(1+x)$ ou Yeo-Johnson (o Box-Cox exige valores positivos). Por coluna, a ordem é **faltantes** → **forma** → **escala**.
- Categorias **nominais** viram one-hot (`handle_unknown='ignore'; drop='first'` só para modelo linear com intercepto e sem penalidade); **ordinais** viram inteiros na ordem que você declara. Alta cardinalidade: extraia estrutura, agrupe as raras; target encoding só com cross-fitting.
- **Imputar**: primeiro pergunte por que falta. Preencher com a média multiplica a variância pela fração observada; `KNNImputer` e `IterativeImputer` usam as outras colunas; `add_indicator=True` guarda o fato de que faltava.

1 Qual é o vizinho mais próximo?

• AQUECIMENTO

Três passageiros, descritos por idade e tarifa: D (30 anos, £10), E (60 anos, £12) e F (32 anos, £70). Nos passageiros com idade conhecida, os desvios padrão são $\sigma_{\text{idade}} = 14{,}52$ anos e $\sigma_{\text{tarifa}} = £52{,}88$.

(a) Calcule as distâncias euclidianas $d(D,E)$ e $d(D,F)$ nas unidades brutas. Quem é o vizinho mais próximo de D?

(b) Divida cada diferença pelo desvio padrão da coluna e calcule de novo. E agora?

(c) Que fração da distância quadrática média entre dois passageiros vem da tarifa, nas unidades brutas? Use $\sigma^2_{\text{tarifa}}/(\sigma^2_{\text{idade}} + \sigma^2_{\text{tarifa}})$. O que isso diz de um k-NN nas colunas cruas?

CONFIRA NO SIMULADOR

Escalonar muda quem são seus vizinhos

Deslize a unidade da renda de milhares para centavos e veja os cinco vizinhos do ponto vermelho virarem outros cinco, sem que ninguém tenha mudado. Depois aperte *padronizar*.

2 Quatro escalonadores à mão

• AQUECIMENTO

Uma coluna de renda (em centenas de reais) tem, nas linhas de treino, os valores 10, 20, 30, 40 e 200.

(a) Calcule a média, o desvio padrão (divida por $n = 5$), o mínimo, o máximo, a mediana, Q_1, Q_3 (são o 2º e o 4º valores) e o maior valor absoluto.

(b) Complete a tabela.

(c) Em qual escalonador as quatro rendas comuns ficam mais espalhadas? Por quê?

(d) No teste aparece uma renda de 400. Quanto ela vale no min-max ajustado no treino? É um erro?

valor	padrão	min-max	robusto	max-abs
10				
40				
200				

3 Forma, não escala

•• PRÁTICA

Seis tarifas, em libras: 0, 7, 15, 31, 63 e 511.

(a) Calcule a média e a mediana. O que a diferença entre elas diz sobre a cauda?

(b) Depois de passar pelo `StandardScaler`, a cauda continua lá? Por quê?

(c) Aplique $\log_2(1 + x)$ (a base 2 deixa as contas redondas; trocar a base só multiplica tudo por uma constante) e calcule de novo a média e a mediana. O que aconteceu com a cauda?

(d) Por que $\log(1 + x)$ e não $\log x$? Por que o Box-Cox recusaria esta coluna e o Yeo-Johnson não?

(e) Num pipeline, o log vem antes ou depois do escalonador? Por quê?

4 Categorias viram colunas

●● PRÁTICA

Um `OneHotEncoder(handle_unknown='ignore')` foi ajustado com os portos de embarque S, C, S e Q das linhas de treino.

(a) Escreva a saída para os passageiros com porto S, C, Q e B (Belfast, que nunca apareceu no treino), com as colunas na ordem C, Q, S.

(b) Com códigos inteiros (C = 0, Q = 1, S = 2), quanto valem as distâncias C–Q e C–S? E com one-hot? Que afirmação os inteiros fazem sem ninguém ter medido?

(c) Num modelo linear com intercepto, por que se usa `drop='first'`? Que porto vira a referência?

(d) Uma coluna de risco tem os níveis baixo, médio e alto. Que códigos o `OrdinalEncoder()` padrão dá a cada um (ele ordena alfabeticamente)? O que está errado e como corrigir?

5 O preço de preencher

●● PRÁTICA

(a) Uma coluna tem os valores observados 3, 5, 7 e 9 e mais 4 faltantes. Calcule a variância (divida por n) dos 4 observados e a dos 8 valores depois de preencher os buracos com a média. Qual é a razão entre as duas?

(b) No Titanic, 714 das 891 idades são observadas, com desvio padrão 14,53 anos (divisão por $n-1$). Que desvio você prevê depois de imputar pela média? (O fator agora é $\sqrt{713/890}$.)

(c) Um `KNNImputer` com $k = 2$ preenche a idade do último passageiro abaixo, usando a tarifa para medir a distância. Que idade ele dá? E a imputação pela média?

(d) Na lista do Titanic, 29,4% dos passageiros sem idade sobreviveram, contra 40,6% dos outros. Por que isso pede `add_indicator=True`?

idade	40	20	50	24	?
tarifa (£)	30	8	26	10	9

6 Monte o pré-processamento

● ● ● DESAFIO

A lista de passageiros do Titanic (891 linhas) tem estas colunas. O alvo é `Survived`.

coluna	o que se sabe	o que fazer
`PassengerId`	891 valores distintos, um número de linha	
`Pclass`	1, 2 ou 3 (classe da passagem)	
`Sex`	2 valores	
`Age`	177 faltantes (19,9%), mais na 3ª classe	
`SibSp`, `Parch`	contagens de familiares (de 0 a 8 e de 0 a 6)	
`Fare`	de £0 a £512, 15 zeros, assimetria 4,79	
`Embarked`	3 portos, 2 faltantes	
`Ticket`	681 valores distintos	
`Cabin`	687 faltantes (77,1%)	

(a) Complete a tabela (sai, imputar, transformar a forma, escalonar, codificar como?).

(b) O que muda na sua tabela se o modelo for uma árvore de decisão em vez de um k-NN?

(c) Um colega troca cada bilhete pela taxa de sobrevivência dos passageiros com o mesmo bilhete, calculada nas 891 linhas. Por que a acurácia dele vai parecer ótima? Na aula, essa codificação "explicou" 77,5% de rótulos sorteados no cara ou coroa. Como fazer direito?

ATIVIDADE DA AULA

D3
Pipelines

insper.github.io/ML/pt/2026.2/classes/pipelines/

respostas comentadas na página 152

A pergunta da aula. O mesmo modelo de k vizinhos mais próximos (k-NN), treinado do mesmo jeito, erra US$ 43 mil por distrito quando o código de previsão está certo e US$ 116 mil quando os distritos chegam um de cada vez a um scaler reajustado. Como garantir que o que foi aprendido no treino é exatamente o que se aplica a toda linha nova?

ETAPAS DO FLUXO NESTA AULA dados e divisão avaliação

O que você precisa saber

- O fluxo manual tem **três bugs clássicos**: (1) `fit_transform` onde devia ser `transform`, invisível no teste inteiro e grave em lotes pequenos; (2) um passo esquecido na hora de prever; (3) outra ordem de colunas, silenciosa num array.
- Um `Pipeline` é uma lista de passos que se comporta como um estimador. O `fit` chama `fit_transform` em cada transformador, na saída do anterior, só nas linhas de treino; o `predict` só chama `transform` e termina no `predict` do modelo.
- O `ColumnTransformer` manda cada grupo de colunas para o seu ramo, uma tripla `(nome, transformador, colunas)`, escolhe as colunas **pelo nome** e descarta as não mencionadas (`remainder='drop'`). As saídas ganham prefixos (`num__`, `cat__`).
- **Passos seus**: sem estado, um `FunctionTransformer`; com estado, uma classe em que o `__init__` só guarda hiperparâmetros e o `fit` aprende em atributos terminados em `_` e devolve `self`.
- O pipeline ajustado se **inspeciona** (`named_steps`, `statistics_`, `mean_`) e se **salva** num arquivo só (`joblib`). Cada parâmetro tem um caminho `passo__parametro`.
- O tamanho de um **vazamento** depende do que o passo aprende: um scaler ajustado em tudo mexe pouco; um passo que usa o alvo pode fabricar acerto. Com tudo dentro do pipeline, não há nada a medir.

1 À mão: o que o pipeline aprende

AQUECIMENTO

Um pipeline com `SimpleImputer(strategy='median')` e `StandardScaler()` é ajustado numa coluna com as linhas de treino 4, NaN, 8, 12 e 16. Depois chegam duas linhas novas: NaN e 22.

(a) Que mediana o imputador aprende? Escreva a coluna imputada e calcule a média e o desvio padrão (divida por $n = 5$) que o scaler aprende.

(b) Que valores chegam ao modelo para as duas linhas novas?

(c) E se alguém chamar `fit_transform` nas duas linhas novas? E numa linha só, a de 22?

2 Que bug é este?

• AQUECIMENTO

Um k-NN foi treinado com os distritos da Califórnia (imputador, scaler e modelo, ajustados à mão no treino). O correto erra US$ 43.023 por distrito; prever sempre a média do treino erra US$ 92.894. Associe cada sintoma ao bug (1, 2 ou 3) e diga se ele é silencioso (S) ou dá erro (E).

Sintoma	bug	S ou E
(a) No teste inteiro, o MAE (erro absoluto médio) sai US$ 42.814, até um pouco menor que o correto.		
(b) Com os distritos enviados um de cada vez, todos recebem a mesma previsão; MAE de US$ 116.049.		
(c) `ValueError: Input X contains NaN.`		
(d) MAE de US$ 105.978, sem aviso nenhum na tela.		
(e) Um `UserWarning` sobre nomes de colunas e MAE de US$ 101.517.		
(f) `ValueError: The feature names should match those that were passed during fit.`		

(g) Quais desses sintomas são piores que prever sempre a média? Qual é o mais perigoso? Por quê?

3 Quanto vazou?

•• PRÁTICA

Para dar ao k-NN um "CEP", as coordenadas viraram zonas de cerca de 1 km: 12.590 zonas, 8.237 delas com um só distrito. Um target encoder troca cada zona pelo valor médio das casas da zona. Erro absoluto médio no teste:

versão	MAE (US$)
encoder ajustado nas 20.640 linhas, antes da divisão	14.567
encoder como passo do pipeline (só vê o treino)	42.552
sem a zona	60.383

(a) Por que a primeira versão parece tão boa? Pense num distrito de teste que está sozinho na sua zona.

(b) A zona ajuda de verdade? Quanto?

(c) Em 20 divisões diferentes, ajustar o *scaler* em todas as linhas mudou o MAE em só US$ 4, em média. Por que a diferença entre os dois passos é tão grande?

(d) Que regra sai daqui?

4 O traço do fit

●● PRÁTICA

Entre os passos de um pipeline foi posto um "espião" que não muda nada e imprime o formato dos dados e, para a coluna `total_bedrooms`, quantos valores faltam e a média naquele ponto.

```
traced = make_pipeline(Spy('raw'), SimpleImputer(strategy='median'),
                       Spy('imputed'), StandardScaler(), Spy('scaled'),
                       KNeighborsRegressor(n_neighbors=10))
traced.fit(X_train, y_train)
  raw      fit       (16512, 8)  missing 163 · mean  538.92
  raw      transform (16512, 8)  missing 163 · mean  538.92
  imputed  fit       (16512, 8)  missing   0 · mean  537.92
  imputed  transform (16512, 8)  missing   0 · mean  537.92
  scaled   fit       (16512, 8)  missing   0 · mean    0.00
  scaled   transform (16512, 8)  missing   0 · mean    0.00
traced.predict(X_test)
  raw      transform (4128, 8)   missing  44 · mean  533.66
  imputed  transform (4128, 8)   missing   0 · mean  532.63
  scaled   transform (4128, 8)   missing   0 · mean   -0.01
```

(a) Durante o `fit`, quantas linhas cada passo vê? Por que cada passo aparece com `fit` *e* `transform`?

(b) Imputar baixou a média de 538,92 para 537,92. A mediana do treino é maior ou menor que a média? O que isso diz da coluna?

(c) Por que, no `predict`, a média padronizada é $-0{,}01$ e não exatamente 0? O que uma média 0,00 ali indicaria?

CONFIRA NO SIMULADOR

Fit em quais linhas?

Com *fit nas linhas de treino*, veja o z médio das linhas de teste sair perto de 0, mas não exatamente 0, como no traço do exercício 4. Troque para *fit em todas as linhas*: numa divisão aleatória quase nada muda; com *teste = período posterior*, a média e o desvio vazados já contêm os preços novos.

5 Para onde vai cada coluna? ●● PRÁTICA

A tabela crua do Titanic tem 11 colunas: `PassengerId`, `Pclass`, `Name`, `Sex`, `Age`, `SibSp`, `Parch`, `Ticket`, `Fare`, `Cabin` e `Embarked`.

```
preprocess = ColumnTransformer([
  ('num',  make_pipeline(SimpleImputer(strategy='median'),
                         StandardScaler()),
           ['Age', 'SibSp', 'Parch']),
  ('fare', make_pipeline(SimpleImputer(strategy='median'),
                         FunctionTransformer(np.log1p),
                         StandardScaler()),
           ['Fare']),
  ('cat',  make_pipeline(SimpleImputer(strategy='most_frequent'),
                         OneHotEncoder(handle_unknown='ignore')),
           ['Pclass', 'Sex', 'Embarked'])])
model = Pipeline([('prep', preprocess),
                  ('knn', KNeighborsClassifier())])
```

Depois do `fit`, o ramo `'num'` guarda, para Age, SibSp e Parch: medianas 28,5, 0 e 0; médias 29,556, 0,493 e 0,390; desvios 13,016, 1,060 e 0,838.

(a) Que colunas nunca chegam ao modelo? Quantas colunas chegam? (Há 3 classes, 2 sexos e 3 portos.)

(b) Que valores chegam para um passageiro novo com idade faltante, 2 irmãos e nenhum pai ou filho? E se ele embarcou num porto que não existia no treino?

(c) Escreva o caminho do parâmetro que troca a estratégia do imputador do ramo `'cat'` e o do número de vizinhos.

(d) O que mudaria com `remainder='passthrough'`?

6 Um passo seu

● ● ● DESAFIO

(a) Para cada passo proposto, diga se ele precisa aprender algo no treino (classe com `fit`) ou não (`FunctionTransformer`) e se ele usa o alvo: (1) $\log(1+x)$; (2) cômodos por domicílio, `total_rooms / households`; (3) cortar cada coluna nos percentis 1 e 99; (4) trocar por `'Rara'` as categorias com menos de 10 linhas; (5) trocar cada categoria pela média do alvo.

(b) Escreva a classe `AgrupaRaras` do passo (4), seguindo as convenções do scikit-learn (hiperparâmetro `min_count=10`).

(c) Por que as categorias frequentes têm de ser aprendidas no `fit` e não recalculadas no `transform`?

(d) Depois de `joblib.dump(model, 'modelo.joblib')`, o que o arquivo contém? O que garante que uma linha crua recebida em produção seja tratada como as de treino?

ATIVIDADE DA AULA

U1
Redução de dimensionalidade

insper.github.io/ML/pt/2026.2/classes/dimensionality-reduction/

respostas comentadas na página 152

A pergunta da aula. O conjunto *digits* tem 1.797 imagens de $8 \times 8 = 64$ pixels. Quantos números por imagem bastam para guardar quase tudo o que elas têm? E o que um mapa 2D bonito dessas imagens pode (e não pode) afirmar?

ETAPAS DO FLUXO NESTA AULA dados e divisão modelo

O que você precisa saber

- **Maldição da dimensionalidade:** colunas que não carregam estrutura somam ruído a toda distância e o vizinho mais próximo deixa de significar algo. O que importa é quantas direções de fato variam (a **dimensão intrínseca**), não o número de colunas.
- **PCA** (análise de componentes principais)**:** centralize (X_c são os n pontos menos a média), calcule a covariância $C = X_c^\top X_c/(n-1)$ e ache seus autovetores. O autovalor λ_k *é* a variância ao longo do componente k; a soma dos autovalores é o traço, a variância total; a **variância explicada** é $\lambda_k / \sum_j \lambda_j$.
- **Reconstruir** com k componentes custa um erro igual à soma dos autovalores descartados. Escolha k por um limiar acumulado (90%, 95%), pelo cotovelo ou pelo desempenho do modelo que vem depois.
- O PCA persegue variância, que tem unidade: **padronize** colunas em unidades diferentes. Ele aprende com os dados, então fica num **pipeline**, ajustado só no treino.
- **t-SNE** (do inglês *t-distributed stochastic neighbor embedding*) e **UMAP** (*uniform manifold approximation and projection*) preservam vizinhanças locais. Tamanho dos grupos, distância entre grupos e eixos do mapa não têm significado; o t-SNE acha "grupos" até em ruído e não tem `transform`; o UMAP tem.
- A **trustworthiness** pergunta quantos vizinhos no mapa eram vizinhos de verdade. Nota alta diz que o mapa é fiel, não que há grupos.

1 Para que reduzir?

• AQUECIMENTO

Para cada situação, escolha PCA, t-SNE ou UMAP e justifique em poucas palavras.

Situação	método	por quê
(a) Comprimir 300 leituras de sensores antes de um modelo que roda num celular.		
(b) Olhar 3.000 avaliações de clientes (384 números cada) para ter ideias de segmentos.		
(c) Pôr, toda semana, os produtos novos num mapa que já foi publicado.		
(d) Comprimir embeddings de documentos antes de um agrupamento por densidade.		
(e) Tirar o ruído de imagens de 12×12 pixels.		

2 Quantas direções importam?

• AQUECIMENTO

Um PCA nas 6 colunas padronizadas de um conjunto (a matriz de correlação, cujo traço é 6) deu estes autovalores:

componente	1	2	3	4	5	6
autovalor λ	3,0	1,5	0,9	0,3	0,2	0,1
variância explicada						
acumulada						

(a) Complete a tabela. Quantos componentes guardam pelo menos 90%? E 95%?

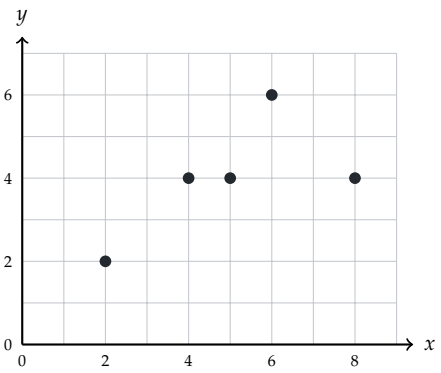
(b) Com $k = 2$, quanto vale o erro de reconstrução (por linha, na mesma escala dos autovalores)? Que fração da variância total ele é?

(c) Um colega diz: "meu conjunto tem 5.000 colunas, então o k-NN (k vizinhos mais próximos) nem vai funcionar". Que número você pediria a ele antes de concordar? Como esta tabela responderia?

3 O PCA de cinco pontos

●● PRÁTICA

Os pontos são $(2,2)$, $(4,4)$, $(5,4)$, $(6,6)$ e $(8,4)$.



(a) Centralize os pontos e escreva a matriz de covariância C (divisor $n-1=4$).

(b) Ache os dois autovalores pela equação $\lambda^2 - (\text{traço})\,\lambda + \det = 0$. Que fração da variância a PC1 explica?

(c) Ache a direção da PC1, w_1, de comprimento 1, e desenhe-a no gráfico, passando pela média.

(d) Calcule o escore de $(6,6)$ na PC1, a sua reconstrução só com a PC1 e o erro quadrático dessa reconstrução.

(e) Sem calcular os outros quatro pontos: quanto vale a soma dos erros quadráticos dos cinco?

CONFIRA NO SIMULADOR

Gire a reta: variância capturada e descartada

Gire a reta e veja as duas barras, variância projetada e erro de reconstrução, somarem sempre a variância total, como $\lambda_1 + \lambda_2$ no exercício 3. Aperte *encontrar PC1*: a reta para onde a primeira barra é máxima e a segunda, mínima.

4 A unidade e a ordem

●● PRÁTICA

Os 178 vinhos têm 13 medidas químicas. A prolina tem desvio padrão de 314,9; o magnésio, 14,3; a tonalidade fica em torno de 1. Dois PCAs:

	colunas brutas	padronizadas
variância explicada da PC1 e da PC2	99,8% e 0,2%	36,2% e 19,2%
maior peso da PC1	prolina, 1,000	flavonoides, 0,423
k-NN sobre as 2 primeiras PCs (acurácia de teste)	0,722	0,926

(a) O que os 99,8% medem? É uma descoberta sobre vinho?

(b) Nas padronizadas, os 13 autovalores somam cerca de 13. Por quê?

(c) Um colega rodou o código abaixo e relatou a acurácia. O que está errado? Reescreva com um pipeline.

```
Z = PCA(n_components=2).fit_transform(StandardScaler().fit_transform(X))
Z_tr, Z_te, y_tr, y_te = train_test_split(Z, y, test_size=0.3, random_state=0)
print(KNeighborsClassifier().fit(Z_tr, y_tr).score(Z_te, y_te))
```

5 O que o mapa autoriza

●● PRÁTICA

Uma analista rodou o t-SNE nos clientes de uma loja e trouxe seis frases. Para cada uma, diga se o mapa sustenta a frase e, se não, por quê. Use os números medidos na aula: um grupo 11,6 vezes mais largo que outro saiu do mesmo tamanho no mapa; um grupo 10 vezes mais distante saiu só 1,3 vez mais distante; ruído puro, com silhueta 0,09, virou um mapa com silhueta 0,42.

(a) "O grupo dos clientes premium é o maior do mapa, então é o mais variado."

(b) "Os grupos A e B estão colados, então os clientes deles são parecidos."

(c) "O mapa tem seis manchas, então a loja tem seis segmentos de clientes."

(d) "Dois clientes muito parecidos nos dados costumam sair perto no mapa."

(e) "Vou usar as duas coordenadas do mapa como features do modelo de churn."

(f) "Todo mês vou pôr os clientes novos neste mesmo mapa."

CONFIRA NO SIMULADOR

t-SNE rodando de verdade

Escolha *ruído uniforme* e rode: veja aparecerem "grupos" que não existem. Depois mude só a perplexidade nos três grupos iguais.

6 O que se perde num ponto

● ● ● DESAFIO

Um PCA em 3 colunas tem média $(5, 5, 2)$, autovalores 8, 3 e 1 e componentes $w_1 = (1, 1, 0)/\sqrt{2}$, $w_2 = (0, 0, 1)$ e $w_3 = (1, -1, 0)/\sqrt{2}$. Chega um ponto novo, $x = (7, 3, 4)$.

(a) Que fração da variância total cada componente explica?

(b) Calcule os três escores de x (produtos escalares de x − média com cada w_k). Confira que a soma dos quadrados dos escores é $\|x - \text{média}\|^2$.

(c) Reconstrua x com $k = 1$ e com $k = 2$ componentes. Qual é o erro quadrático de cada reconstrução?

(d) A PC3 explica só $1/12$ da variância do conjunto. Que fração do afastamento de x em relação à média está nela? O que isso diz sobre julgar um ponto pela variância explicada do conjunto?

ATIVIDADE DA AULA

U2
Agrupamento

insper.github.io/ML/pt/2026.2/classes/clustering/

respostas comentadas na página 153

A pergunta da aula. 342 pinguins de Palmer, quatro medidas de cada um e nenhuma espécie anotada. Quantos grupos há? E como saber se eles existem nos dados ou se o algoritmo só fez o que sempre faz, devolver uma partição?

ETAPAS DO FLUXO NESTA AULA modelo avaliação

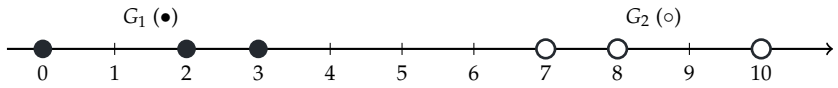
O que você precisa saber

- Agrupar recebe só X e **sempre** devolve uma partição. Índices **internos** (inércia, silhueta, Calinski–Harabasz, Davies–Bouldin) usam só X; o **ARI** (índice de Rand ajustado) compara com rótulos conhecidos e serve só para conferir, nunca para escolher.
- **k-means** minimiza a **inércia** $J = \sum_j \sum_{x_i \in C_j} \|x_i - \mu_j\|^2$. O algoritmo de Lloyd alterna *atribuir* (cada ponto ao centro mais próximo) e *atualizar* (cada centro na média do grupo); J nunca sobe, mas pode parar num **mínimo local**: use `n_init` e k-means++.
- A inércia sempre cai com mais grupos: procure o **cotovelo**. A **silhueta** de um ponto é $s = (b - a)/\max(a, b)$, com a a distância média ao próprio grupo e b ao grupo vizinho mais próximo.
- Ruído puro dá silhueta perto de 0,40: compare sempre com os mesmos dados de **colunas embaralhadas**. Padronize por um argumento sobre as unidades; a silhueta não avisa quando a escala está errada.
- O k-means supõe grupos convexos e de espalhamento parecido. O **DBSCAN** (*density-based spatial clustering of applications with noise*) separa, com `eps` e `min_samples`, pontos **centrais**, **de borda** e **ruído** (-1); um `eps` global é uma densidade só; o **HDBSCAN** (hierárquico) aceita várias.

1 A silhueta de um ponto

• AQUECIMENTO

Seis pontos sobre uma reta, em dois grupos: $G_1 = \{0, 2, 3\}$ e $G_2 = \{7, 8, 10\}$.



(a) Calcule a, b e a silhueta s dos pontos 0 e 3.

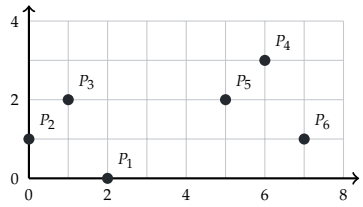
(b) O ponto 3 se desloca para a posição 5 e continua em G_1. Recalcule a, b e s para ele. O que o sinal de s diz?

(c) Por que a silhueta usa o $\max(a, b)$ no denominador?

2 Lloyd à mão

• AQUECIMENTO

Seis pontos e $k = 2$. Os centros iniciais caíram sobre P_1 e P_2: $c_1 = (2, 0)$ e $c_2 = (0, 1)$.



ponto		iteração 0 d^2 a c_1	iteração 0 d^2 a c_2	iteração 0 grupo	iteração 1 d^2 a c'_1	iteração 1 d^2 a c'_2	iteração 1 grupo
P_1	(2, 0)						
P_2	(0, 1)						
P_3	(1, 2)						
P_4	(6, 3)						
P_5	(5, 2)						
P_6	(7, 1)						
inércia							

(a) Preencha a iteração 0 e calcule a inércia. Algum ponto foi parar no grupo "errado"?

(b) Atualize os centros (c'_1 e c'_2) e preencha a iteração 1. Qual ponto troca de grupo?

(c) Atualize de novo. O algoritmo convergiu? Qual é a inércia final?

CONFIRA NO SIMULADOR

Lloyd, um passo de cada vez

Aperte *passo* várias vezes: atribuir e atualizar se alternam e o histórico da inércia nunca sobe. Troque a inicialização e veja se o resultado final muda.

3 Que k os pinguins pedem?

●● PRÁTICA

O k-means (`n_init=10`) nas quatro medidas padronizadas dos 342 pinguins deu:

k	2	3	4	5	6	7	8
inércia	565,7	379,4	300,4	232,6	204,3	187,1	171,0
queda da inércia	—						
silhueta	0,532	0,447	0,400	0,378	0,372	0,335	0,296
Calinski–Harabasz (maior é melhor)	482,2	441,7	400,4	411,2	382,7	352,3	334,0
Davies–Bouldin (menor é melhor)	0,714	0,944	0,944	0,973	0,951	1,048	1,151

(a) Complete as quedas. Onde está o cotovelo? Que k a silhueta, o Calinski–Harabasz e o Davies–Bouldin escolhem?

(b) Há três espécies. O ARI com a espécie (que um projeto real não teria) é máximo em $k = 3$ (0,793; com $k = 2$ é 0,655). Os índices internos erraram? O que eles mediram? (Duas das espécies quase só diferem no comprimento do bico.)

(c) Em 300 pontos sorteados ao acaso num quadrado, sem grupo nenhum, o k-means com $k = 4$ chega a silhueta 0,40. Que conta você faria antes de relatar a silhueta dos pinguins?

CONFIRA NO SIMULADOR

As duas curvas em função de k

Varra o k nos conjuntos de bolhas, tamanhos desiguais e ruído: veja quando o cotovelo e a silhueta concordam e o número que a silhueta dá ao ruído.

4 Quem é o vizinho do pinguim P?

●● PRÁTICA

Três pinguins (comprimento e profundidade do bico, nadadeira, todos em mm, e massa em g):

	bico, compr.	**bico, prof.**	**nadadeira**	**massa**
P	39	18,0	190	3.700
Q	47	15,0	215	3.800
R	40	18,5	192	4.200
desvio padrão da coluna	5,5	2,0	14	800

(a) Nas unidades originais, P fica mais perto de Q ou de R? Que coluna decide?

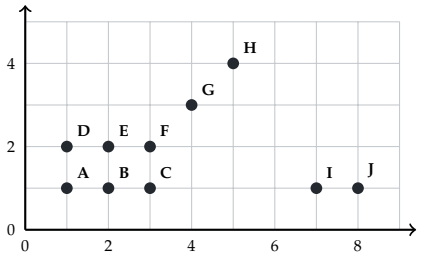
(b) Divida cada diferença pelo desvio padrão da coluna e responda de novo.

(c) Nos 342 pinguins, o k-means com $k = 3$ deu silhueta 0,580 nas unidades originais e 0,447 nas padronizadas; o ARI com a espécie foi 0,331 e 0,793. Por que a silhueta preferiu os dados brutos? Como se decide a escala sem rótulos?

5 DBSCAN à mão

●● PRÁTICA

Dez pontos, `eps` $= 1{,}5$ e `min_samples` $= 4$ (o próprio ponto conta, como no scikit-learn).



ponto	a até 1,5 (ele incluído)	central?	alcança central?	tipo
A, C, D				
B, E				
F				
G				
H				
I, J				

(a) Complete a tabela (distâncias úteis: 1, $\sqrt{2} \approx 1{,}41$, $\sqrt{5} \approx 2{,}24$). Quantos grupos o DBSCAN devolve? Quem recebe -1?

(b) H está a 1,41 de G, que pertence a um grupo. Por que H não entra nele?

(c) Refaça com `min_samples` $= 2$: quantos grupos e quantos pontos de ruído?

6 Que algoritmo e o que relatar

● ● ● DESAFIO

Cinco conjuntos 2D de 500 pontos com grupos conhecidos. O k-means recebeu o k certo; o HDBSCAN (`min_cluster_size=15`) não recebeu k nenhum.

	bolhas	**luas**	**círculos**	**esticados**	**densidades**
k-means: silhueta	0,823	0,489	0,356	0,501	0,605
k-means: ARI	1,000	0,253	−0,002	0,555	0,436
HDBSCAN: grupos e pontos de ruído	3 e 0	2 e 0	2 e 0	3 e 49	3 e 45
HDBSCAN: ARI	1,000	1,000	1,000	0,852	0,856

(a) Ordene as partições do k-means pela silhueta e pelo ARI. As ordens concordam? Por que *densidades* tem a segunda maior silhueta?

(b) Por que nenhum k e nenhum reinício fazem o k-means acertar os dois anéis concêntricos?

(c) Em *densidades*, todos os pontos que o HDBSCAN atribuiu a um grupo estão no grupo certo. De onde vem o ARI de 0,856?

(d) Nos 178 vinhos padronizados, $k = 3$ deu silhueta 0,285 (com $k = 2$, 0,268) e os mesmos vinhos com as colunas embaralhadas deram 0,068. Escreva, em duas frases, o resultado como uma *escolha* com o seu critério e a sua evidência, não como uma descoberta.

ATIVIDADE DA AULA

R1

Regressão linear

insper.github.io/ML/pt/2026.2/classes/linear-regression/

respostas comentadas na página 154

A pergunta da aula. Galton mediu as alturas de 928 filhos adultos e de seus pais: pais 3 polegadas acima da média tiveram filhos previstos só cerca de 2 polegadas acima. Os filhos estão sendo "puxados" para a média? E como se escolhe a reta, se mede o que ela errou e se usa esse erro para melhorar?

ETAPAS DO FLUXO NESTA AULA modelo avaliação

O que você precisa saber

- O modelo: $\hat{y} = w_0 + w_1 x_1 + \cdots + w_d x_d$. O peso w_j é a variação de $\hat{y}$ por unidade de x_j, **com os outros atributos fixos**; tem unidade, então não mede importância.
- Os **mínimos quadrados ordinários** (OLS, de *ordinary least squares*) escolhem os pesos que minimizam a soma dos resíduos ao quadrado, SSE $= \sum_i e_i^2$, com $e_i = y_i - \hat{y}_i$. Com um atributo: $\hat{w}_1 = \dfrac{\sum_i (x_i - \bar{x})(y_i - \bar{y})}{\sum_i (x_i - \bar{x})^2} = r_{xy} \dfrac{s_y}{s_x}$ e $\hat{w}_0 = \bar{y} - \hat{w}_1 \bar{x}$ ($\bar{x}, \bar{y}$ médias; s desvios padrão; r_{xy} correlação).
- Em matriz (X com uma coluna de 1s): não se inverte X, que nem é quadrada; resolvem-se as **equações normais** $X^\top X\, w = X^\top y$. Elas equivalem a $\sum_i e_i = 0$ e $\sum_i e_i x_i = 0$.
- Métricas sobre os resíduos: MAE (*mean absolute error*) $= \frac{1}{n}\sum |e_i|$; MSE (*mean squared error*) $= \frac{1}{n}\sum e_i^2$; RMSE (*root mean squared error*) $= \sqrt{\text{MSE}} \geq$ MAE. O $R^2 = 1 - \text{SSE}/\text{SST}$, com SST $= \sum_i (y_i - \bar{y})^2$, compara com prever sempre $\bar{y}$ e pode ser negativo fora do treino.
- O **gráfico de resíduos** (e contra $\hat{y}$) mostra o que o R^2 esconde (curva, funil, ondas, pontos extremos). Um **outlier** tem resíduo grande; um ponto de **alta alavancagem** fica longe de $\bar{x}$ e pode puxar a reta até ficar com resíduo pequeno.
- Com a perda média $J = \frac{1}{n}\sum e_i^2$: $\partial J/\partial w_0 = -\frac{2}{n}\sum e_i$ e $\partial J/\partial w_1 = -\frac{2}{n}\sum e_i x_i$, uma **soma ponderada pelos resíduos**. É a base do gradiente descendente (R2).
- **Regressão à média**: com $r < 1$, o caso extremo na primeira medida tende a ficar mais perto da média na segunda, nos dois sentidos. É aritmética (sinal mais sorte), não uma força.

1 Lendo um modelo de preços

• AQUECIMENTO

Um modelo de preços de apartamentos (preço em R$, área em m², idade em anos) foi ajustado:

$$\widehat{\text{preço}} = 50.000 + 3.200 \cdot \text{área} - 1.500 \cdot \text{idade}$$

(a) Escreva, numa frase completa, o que o 3.200 quer dizer.

(b) Quanto o modelo prevê para um apartamento de 70 m² com 10 anos?

(c) Dois apartamentos de mesma área, um 5 anos mais velho que o outro: qual a diferença prevista?

(d) Se a área fosse medida em km², o que aconteceria com o 3.200? Dá para dizer que a área importa mais que a idade porque $3.200 > 1.500$?

(e) O que o 50.000 significaria, ao pé da letra? Faz sentido?

2 Mínimos quadrados à mão

• AQUECIMENTO

Quatro pontos: $x = (2, 4, 6, 8)$ e $y = (3, 7, 6, 12)$. Complete a tabela e responda.

x_i	y_i	$x_i - \bar{x}$	$y_i - \bar{y}$	produto	$(x_i - \bar{x})^2$	$\hat{y}_i$	e_i
2	3						
4	7						
6	6						
8	12						
			soma:				

(a) Calcule $\bar{x}$, $\bar{y}$, $\hat{w}_1$ e $\hat{w}_0$.

(b) Calcule $\hat{y}_i$ e os resíduos e_i. Quanto valem $\sum e_i$ e $\sum e_i x_i$? Por que esses valores não são coincidência?

3 Quanto a reta erra

●● PRÁTICA

(a) Com os resíduos do exercício anterior, calcule o MAE, o MSE, o RMSE e o R^2.

(b) Dois modelos de tempo de entrega erraram, em minutos, em cinco pedidos: o modelo A errou 3, 3, 3, 3, 3; o modelo B errou 0, 0, 0, 0, 15. Calcule o MAE e o RMSE de cada um. Qual você prefere se um atraso grande irrita muito mais que vários pequenos?

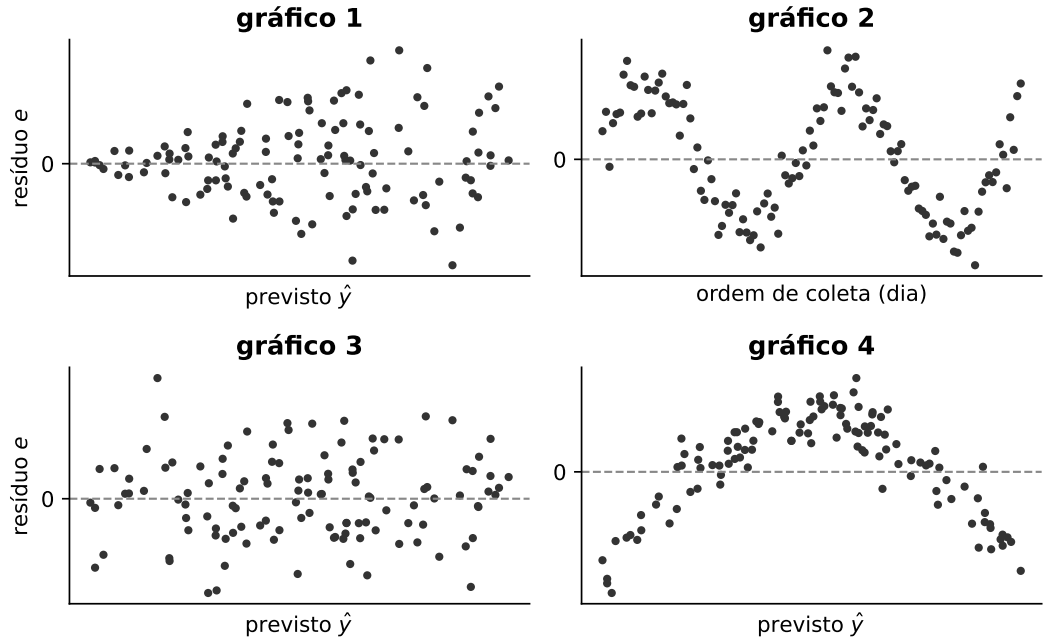
(c) No teste, um modelo tem SSE $= 50$ e SST $= 40$. Quanto vale o R^2 e o que ele diz?

(d) Um colega acrescentou uma coluna de números sorteados ao acaso e o R^2 *de treino* subiu de 0,80 para 0,81. Ele conclui que a coluna ajuda. O que você responde?

4 O que os resíduos contam

●● PRÁTICA

Os quatro gráficos mostram os resíduos de quatro regressões diferentes. Nos gráficos 1, 3 e 4 o eixo horizontal é o valor previsto; no gráfico 2 é a ordem em que os dados foram coletados.



(a) Para cada gráfico, diga o que ele revela e o que você faria.

(b) Num quinto conjunto, um único ponto tem x muito acima de todos os outros e resíduo pequeno. Ele pode estar estragando a reta? Por que a regra "descarte os pontos de resíduo grande" não o encontraria?

CONFIRA NO SIMULADOR

Quatro conjuntos, quatro diagnósticos

Diagnostique cada caso antes de ler o veredicto e compare o R^2 de cada um com o desenho dos resíduos. O quarto caso é o ponto de alta alavancagem.

5 Três equações, duas incógnitas

●● PRÁTICA

Três pontos, $(x, y) = (0,1)$, $(1,3)$ e $(2,2)$, e o modelo $\hat{y} = w_0 + w_1 x$, ou seja, $\hat{y} = Xw$ com

$$X = \begin{pmatrix} 1 & 0 \\ 1 & 1 \\ 1 & 2 \end{pmatrix}, \qquad y = \begin{pmatrix} 1 \\ 3 \\ 2 \end{pmatrix}, \qquad w = \begin{pmatrix} w_0 \\ w_1 \end{pmatrix}.$$

(a) Por que não dá para calcular $w = X^{-1}y$? Existe algum w com $Xw = y$ exatamente?

(b) Calcule $X^\top X$ e $X^\top y$ e resolva as equações normais.

(c) Calcule os resíduos e confira que $X^\top e = 0$. Quanto vale $\|e\|^2$? Compare com a reta $w = (1,1)$.

(d) Quanto vale o gradiente $\left(\partial J/\partial w_0,\ \partial J/\partial w_1\right)$ da perda média nesse ponto? Por quê?

CONFIRA NO SIMULADOR

A reta persegue o mínimo

Arraste um ponto do meio e depois um da ponta, para longe dos demais: qual dos dois gira mais a reta?

6 O reforço funcionou? ••• DESAFIO

Numa escola, as provas 1 e 2 têm a mesma média (6,0), o mesmo desvio padrão (1,5) e correlação $r = 0{,}6$ entre as notas de um mesmo aluno. Os alunos com as piores notas na prova 1 (média 3,4) tiveram aulas de reforço e chegaram a 4,9 na prova 2. Os melhores da prova 1 (média 8,6), sem reforço, caíram para 7,6. A diretora conclui: "o reforço deu 1,5 ponto e os bons alunos se acomodaram".

(a) Calcule a inclinação da reta de mínimos quadrados que prevê a prova 2 pela prova 1.

(b) Que média essa reta prevê na prova 2 para cada grupo, sem reforço algum? Compare com o observado.

(c) O que sobra das duas conclusões da diretora?

(d) Regredindo a prova 1 na prova 2, a inclinação também é 0,6: os melhores da prova 2 tinham sido menos extremos na prova 1. Por que isso mostra que não há uma "força" puxando as notas para a média?

(e) Como você mediria o efeito do reforço sem depender dessa conta?

ATIVIDADE DA AULA

R2

Gradiente descendente e regularização

insper.github.io/ML/pt/2026.2/classes/gradient-descent-regularization/

respostas comentadas na página 155

A pergunta da aula. Três polinômios foram ajustados aos mesmos 100 pontos. O de grau 30 tem o menor erro de treino (RMSE, a raiz do erro quadrático médio, de 2,46) e erra, em média, 10,43 na validação. Como treinar um modelo quando não há fórmula fechada? E como frear um modelo flexível antes que ele decore o ruído?

ETAPAS DO FLUXO NESTA AULA modelo avaliação

O que você precisa saber

- **Gradiente descendente**: $w \leftarrow w - \eta \nabla J(w)$, com η a **taxa de aprendizado** (o tamanho do passo). O gradiente aponta para onde a perda mais cresce; o sinal de menos faz descer.
- Na regressão linear com perda média $J = \frac{1}{n}\sum_i e_i^2$ e $e_i = y_i - \hat{y}_i$: $\partial J/\partial w_j = -\frac{2}{n}\sum_i e_i\, x_{ij}$ (com $x_{i0} = 1$ para o intercepto); em matriz, $\nabla J = -\frac{2}{n} X^\top e$. Onde $\nabla J = 0$ estão as equações normais da R1.
- η pequeno: lento; grande: passa do mínimo, oscila ou **diverge**. Sem centrar ou padronizar, a paisagem vira um vale estreito e o caminho ziguezagueia; a forma fechada não muda. Batch usa todas as linhas por passo; o **SGD** (*stochastic gradient descent*, gradiente descendente estocástico), uma; o mini-batch, um lote.
- Atributos polinomiais $(x, x^2, \ldots, x^p)$ deixam o modelo linear nos pesos e curvo em x. **Underfitting**: treino e validação altos e parecidos. **Overfitting**: treino baixo, validação alta e instável entre as dobras.
- **Ridge** (L2): $J = \|y - Xw\|^2 + \alpha \sum_j w_j^2$; encolhe todos os pesos, nenhum vira zero; divide o peso entre atributos correlacionados. **Lasso** (L1): $+\alpha \sum_j |w_j|$; zera os fracos (seleção de atributos). Com um atributo e $\sum_i x_i^2 = 1$: $w^{\text{ridge}} = \hat{w}/(1+\alpha)$ e $w^{\text{lasso}} = \text{sinal}(\hat{w})\max(|\hat{w}| - \alpha/2,\ 0)$, sendo $\hat{w}$ o peso dos mínimos quadrados ordinários (OLS).
- O α é escolhido na validação. **Padronize antes de regularizar** e não penalize o intercepto.

1 Um passo, à mão

• AQUECIMENTO

Quatro pontos: $x = (0, 1, 2, 3)$ e $y = (2, 1, 4, 5)$. O modelo é $\hat{y} = w_0 + w_1 x$, a perda é a média $J = \frac{1}{4}\sum_i e_i^2$, o ponto de partida é $w_0 = w_1 = 0$ e $\eta = 0{,}05$.

(a) Calcule J, $\partial J/\partial w_0$ e $\partial J/\partial w_1$ no ponto de partida.

(b) Dê um passo. Quanto valem w_0, w_1 e J depois dele?

(c) Calcule o gradiente no novo ponto. Os sinais mudaram? O que isso diz sobre o próximo passo?

(d) Depois de muitos passos, os pesos param em $(1{,}2;\ 1{,}2)$. Confira com as fórmulas da R1. Quanto vale o gradiente lá?

2 O tamanho do passo

• AQUECIMENTO

Na perda $J(w) = w^2$, a derivada é $2w$. Partindo de $w = 4$:

(a) Mostre que um passo leva w a $(1 - 2\eta)\, w$.

(b) Complete a tabela com os três primeiros passos.

η	0,1	0,5	0,75	1	1,25
fator $1 - 2\eta$					
passo 1					
passo 2					
passo 3					

Para cada η, descreva o que acontece (converge devagar, de uma vez, oscilando, fica preso, diverge). A partir de que η a descida diverge?

CONFIRA NO SIMULADOR

A taxa de aprendizado

Avance passo a passo com $\eta = 0{,}5$, depois 0,9, 1 e 1,05: os mesmos quatro comportamentos da tabela.

3 Um vale estreito

●● PRÁTICA

Três pontos: $x = (10, 11, 12)$ e $y = (1, 2, 3)$, com o modelo e a perda média do exercício 1.

(a) Em $w_0 = w_1 = 0$, calcule $\partial J/\partial w_0$ e $\partial J/\partial w_1$. Quantas vezes a segunda é maior que a primeira? Por que um η bom para w_0 é ruim para w_1?

(b) Centre o atributo, $x' = x - \bar{x}$, e recalcule $\partial J/\partial w_1$ no mesmo ponto.

(c) Com x', a reta de mínimos quadrados é $\hat{y} = 2 + 1 \cdot x'$. Escreva-a em função de x original. A reta mudou? O que o centramento mudou, então?

(d) O SGD usaria um único ponto por passo. O que muda no gradiente e no caminho?

4 Underfitting ou overfitting? ●● PRÁTICA

Cem pontos de $y = 0{,}5x^2 + x + 2$ mais um ruído de desvio padrão 3. Três regressões polinomiais, com o RMSE no treino e numa validação cruzada de cinco dobras:

grau	RMSE de treino	RMSE de validação (média)	desvio entre as dobras
1	3,22	3,26	0,19
2	2,64	2,75	0,33
30	2,46	10,43	13,46

As cinco dobras do grau 30 deram 3,14; 3,02; 5,51; 37,29 e 3,20.

(a) Classifique cada grau (underfitting, adequado, overfitting) e diga qual você escolhe.

(b) Confira a média das cinco dobras do grau 30. Qual número da tabela denuncia o overfitting antes da média? Por quê?

(c) Por que nenhum modelo vai chegar a um RMSE de validação muito abaixo de 3?

(d) Um polinômio de grau 10 ajustado a 10 pontos tem RMSE de treino praticamente zero. Por quê? É bom sinal?

5 Ridge divide, lasso subtrai

●● PRÁTICA

Um atributo padronizado de modo que $\sum_i x_i^2 = 1$. Use as fórmulas do resumo com $\alpha = 0{,}6$.

$\hat{w}$ (OLS)	1,6	1,0	−0,5	0,2
ridge				
lasso				

(a) Complete a tabela.

(b) Que pesos o lasso zera com $\alpha = 0{,}6$? A partir de que α ele zera os quatro? E o ridge, zera algum?

(c) Na forma de orçamento, o lasso restringe $\sum |w_j| \leq t$ (um losango) e o ridge, $\sum w_j^2 \leq t^2$ (um círculo). Por que só o losango costuma dar pesos exatamente zero?

CONFIRA NO SIMULADOR

Por que o L1 zera e o L2 não

Arraste o ótimo do OLS e veja onde a elipse toca o losango e onde toca o círculo.

6 Dois termômetros

● ● ● DESAFIO

Uma sorveteria prevê as vendas do dia com a temperatura medida por dois termômetros vizinhos (correlação 0,99 entre as leituras). Refazendo a regressão linear em três meses diferentes, os coeficientes dos termômetros saíram $(2{,}1;\ 0{,}1)$, $(-0{,}4;\ 2{,}6)$ e $(1{,}3;\ 0{,}9)$; as previsões quase não mudaram.

(a) Explique o fenômeno. Por que a soma dos dois coeficientes é estável e cada um, não?

(b) O que o ridge faria com esses dois coeficientes? E o lasso?

(c) Um ridge foi medido para vários α (RMSE, em sorvetes por dia):

α	0,01	0,1	1	10	100	1000
treino	10,1	10,2	10,6	12,0	18,5	30,2
validação	14,8	13,9	12,3	12,9	19,0	30,5

Que α você escolhe? Em que ponta há overfitting e em qual há underfitting?

(d) Sem padronizar, um termômetro em °C e outro em °F. O mesmo efeito vale 2,2 sorvetes por °C ou cerca de 1,22 por °F. Que coluna a penalidade $\alpha \sum w_j^2$ favorece? Por que isso é um problema?

ATIVIDADE DA AULA

C1

Regressão logística

insper.github.io/ML/pt/2026.2/classes/logistic-regression/

respostas comentadas na página 156

A pergunta da aula. Com 23 estudantes, uma reta ajustada aos rótulos 0 (reprovado) e 1 (aprovado) prevê $-0{,}23$ para quem estudou uma hora e passa a errar o dobro quando entram cinco aprovados que estudaram muito. Como transformar horas de estudo numa probabilidade de aprovação que faça sentido e numa decisão?

ETAPAS DO FLUXO NESTA AULA modelo avaliação

O que você precisa saber

- Classificar tem dois passos: o modelo dá um **escore** ou uma **probabilidade**; um **limiar** (*threshold*) a transforma em rótulo.
- A regressão logística calcula o escore linear $z = w^\top x + b$ e o passa pela **sigmoide**: $\hat{p} = \sigma(z) = \dfrac{1}{1+e^{-z}}$, sempre entre 0 e 1, com $\sigma(0) = 0{,}5$. A **fronteira de decisão** $z = 0$ é uma reta (um hiperplano); as curvas de mesmo $\hat{p}$ são paralelas a ela.
- A **chance** (*odds*) é $p/(1-p)$; a do modelo é e^z: o escore é o log da chance. Uma unidade a mais de x_j, com os outros fixos, **multiplica a chance por** e^{w_j}; a variação em probabilidade depende do ponto de partida.
- A perda é a **entropia cruzada** (*log loss*): $J = -\frac{1}{n}\sum_i \left[y_i \ln \hat{p}_i + (1-y_i)\ln(1-\hat{p}_i)\right]$, a média de $-\ln(\text{probabilidade dada à classe verdadeira})$. Prever sempre 0,5 custa $\ln 2 \approx 0{,}693$. É convexa e seu gradiente é $\frac{1}{n}X^\top(\hat{p}-y)$.
- Fronteiras curvas pedem atributos construídos (x_1^2, x_2^2, polinômios). Com k classes, o **softmax** $\hat{p}_c = e^{z_c}/\sum_j e^{z_j}$ dá probabilidades que somam 1.
- `predict` é `predict_proba` com limiar 0,5, uma **escolha**, não parte do modelo. Nesta aula o modelo é o puro, `LogisticRegression(C=np.inf)`: o scikit-learn, por padrão, acrescenta uma penalidade L2.

1 Uma reta para aprovar? • AQUECIMENTO

Uma regressão linear foi ajustada aos rótulos 0 (reprovado) e 1 (aprovado) de 23 estudantes, com as horas de estudo como atributo. Ela prevê de $-0{,}23$ a $1{,}18$ e cruza 0,5 em 6,67 horas. Depois entram cinco estudantes que estudaram de 25 a 45 horas e foram aprovados: a reta passa a cruzar 0,5 em 8,66 horas e os erros vão de 2 para 4. Uma regressão logística, nos dois casos, cruza 0,5 em 6,75 horas e erra 2.

(a) Por que $-0{,}23$ e $1{,}18$ não servem como probabilidades?

(b) Os cinco novos estudantes estão rotulados corretamente e são casos fáceis. Por que pioraram a reta?

(c) Um estudante que estudou 8 horas e passou: o que a reta com os cinco novos prevê para ele?

(d) Por que a curva logística quase não se mexeu?

2 Escores viram probabilidades • AQUECIMENTO

Um modelo de aprovação tem $z = 1{,}5\,x - 9$, com x em horas de estudo. Use $e^{1{,}5} \approx 4{,}48$ e $e^{3} \approx 20{,}1$.

horas x	4	5	6	7	8
escore z					
$\hat{p} = 1/(1+e^{-z})$					

(a) Complete a tabela. Em quantas horas fica a fronteira? Que rótulo o `predict` dá a quem estudou 5 horas? E a quem estudou 7?

(b) Um modelo de três classes dá os escores $(1;\ 0;\ -1)$. Calcule as probabilidades do softmax (use $e \approx 2{,}718$ e $e^{-1} \approx 0{,}368$). O que muda se somarmos 5 aos três escores?

3 Chances

●● PRÁTICA

(a) Converta em chance: $p = 0{,}8$; $p = 0{,}5$; $p = 0{,}2$. Que probabilidade corresponde à chance 3 ("3 para 1")?

(b) No modelo do exercício anterior, calcule a chance de aprovação com 5, 6 e 7 horas. Por quanto uma hora a mais multiplica a chance? Quantos pontos de probabilidade ela soma de 5 para 6 horas? E de 7 para 8?

(c) Num modelo do Titanic, a categoria de referência de `Sex` é `female` e o coeficiente de `Sex_male` é $-2{,}685$ ($e^{-2{,}685} \approx 0{,}068$). Complete: "com classe, idade, família, tarifa e porto fixos, a chance de sobreviver de um homem é _____ vez a de uma mulher; a de uma mulher é _____ vezes a de um homem".

(d) Para um passageiro, o modelo dá $\hat{p} = 0{,}093$ como homem e $\hat{p} = 0{,}601$ como mulher. Calcule a razão das probabilidades e a razão das chances. Qual das duas é a mesma para todo passageiro?

4 A perda que pune a certeza errada

●● PRÁTICA

Quatro previsões de um modelo binário. Use a tabela de logaritmos naturais.

v	0,01	0,1	0,2	0,3	0,4	0,5	0,6	0,7	0,8	0,9
$\ln v$	−4,605	−2,303	−1,609	−1,204	−0,916	−0,693	−0,511	−0,357	−0,223	−0,105

exemplo	y	$\hat{p}$ (classe 1)	probabilidade dada à verdade	$-\ln(\cdot)$
1	1	0,8		
2	0	0,1		
3	1	0,3		
4	0	0,6		

(a) Complete a tabela e calcule a entropia cruzada média. O modelo é melhor que dizer 0,5 para todo mundo?

(b) Que exemplo pesa mais na perda e por quê?

(c) Um exemplo da classe 1 recebeu $\hat{p} = 0{,}01$. Quanto ele custa no erro quadrático $(1-\hat{p})^2$ e na entropia cruzada? A força com que o treino corrige o escore dele é $\hat{p} - y$ na entropia cruzada e $2(\hat{p}-y)\,\hat{p}\,(1-\hat{p})$ no erro quadrático. Calcule as duas e conclua.

CONFIRA NO SIMULADOR

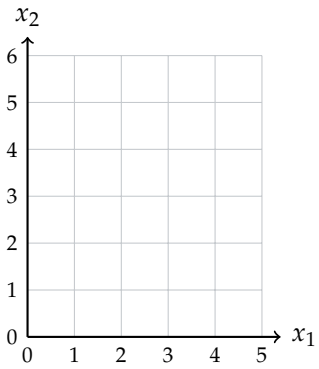
Duas perdas, lado a lado

Mexa em w e em b: o que cada um faz com a curva? No painel da direita, compare a entropia cruzada com o erro quadrático.

5 A fronteira é uma reta

●● PRÁTICA

Com dois atributos padronizados, uma regressão logística ajustou $z = 2x_1 + x_2 - 4$.



(a) Desenhe a fronteira $\hat{p} = 0{,}5$ e marque o lado que o `predict` chama de classe 1.

(b) Calcule $\hat{p}$ nos pontos $(3, 1)$ e $(1, 1)$ (use $e^3 \approx 20{,}1$ e $e \approx 2{,}718$).

(c) Onde fica a curva de $\hat{p} = 0{,}9$? Desenhe-a ($\ln 9 \approx 2{,}197$).

(d) A classe 1 forma um disco de raio 0,5 no centro; a classe 0, um anel de raio 1 em volta. Por que nenhuma reta separa as duas? Com os atributos x_1^2 e x_2^2, o modelo ajustou $z = -16{,}83\,x_1^2 - 16{,}00\,x_2^2 + 9{,}70$: que forma tem a fronteira? O modelo deixou de ser linear?

6 Onde pôr o limiar

● ● ● DESAFIO

Um modelo de triagem deu estas probabilidades de doença a oito pacientes (1 = doente):

paciente	A	B	C	D	E	F	G	H
$\hat{p}$	0,92	0,75	0,64	0,48	0,41	0,33	0,22	0,08
doente?	1	1	0	1	0	1	0	0

(a) Com os limiares 0,3; 0,5 e 0,7 (sinaliza quem tem $\hat{p} \geq$ limiar), conte os doentes perdidos, os alarmes falsos e a acurácia.

(b) Deixar passar um doente custa 5 vezes mais que um alarme falso. Qual limiar tem o menor custo total?

(c) Por que a acurácia não serve para essa escolha?

(d) Para mudar o limiar é preciso treinar o modelo de novo? Escreva, em até três linhas, a recomendação para o hospital.

ATIVIDADE DA AULA

C2

k-vizinhos mais próximos (k-NN)

insper.github.io/ML/pt/2026.2/classes/knn/

respostas comentadas na página 156

A pergunta da aula. Um serviço de streaming sabe quantas horas de ação e de documentário cada assinante assiste por semana e se ele gostou de uma série nova. Para um assinante novo, ouvir 1, 3 ou 5 assinantes parecidos dá respostas diferentes. Quantos vizinhos ouvir, como medir "parecido" e como saber se funciona?

ETAPAS DO FLUXO NESTA AULA dados e divisão | modelo | avaliação

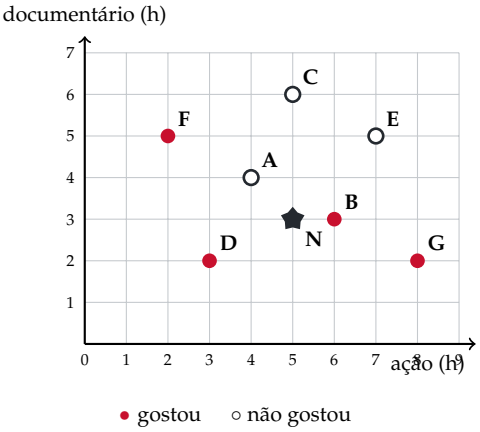
O que você precisa saber

- O **k-NN** (*k-nearest neighbors*, k vizinhos mais próximos) mede a distância do ponto novo a cada linha de treino, pega as k mais próximas e as deixa votar. O `predict_proba` é a fração dos k vizinhos em cada classe.
- O `fit` só **guarda as linhas** (aprendiz preguiçoso, baseado em instâncias): o trabalho fica para a previsão, que custa uma distância por linha guardada. A regressão logística (C1) guarda $d + 1$ números e prevê quase de graça.
- A distância soma diferenças **nas unidades de cada atributo**: sem escalonar, a coluna de números maiores decide sozinha. O escalonador entra num **pipeline**, ajustado só no treino.
- O k controla a complexidade ao contrário: k pequeno segue cada ponto (overfitting); k grande vira a classe majoritária (underfitting). Com $k = 1$, a acurácia de treino é 100% **por construção**. Escolha k na **validação** e meça o teste uma vez. Com duas classes, use k ímpar.
- Outras réguas: Manhattan $\sum_j |a_j - b_j|$, euclidiana $\sqrt{\sum_j (a_j - b_j)^2}$, Chebyshev $\max_j |a_j - b_j|$; cosseno para texto. Com `weights='distance'`, cada vizinho vota com peso $1/d$.
- **Maldição da dimensionalidade**: com muitas colunas, as distâncias ficam parecidas e colunas irrelevantes afogam as úteis. O k-NN brilha com muitas linhas e poucas colunas, todas com significado.

1 Quem vota

• AQUECIMENTO

Sete assinantes (horas de ação e de documentário por semana) e um assinante novo, $\mathrm{N} = (5, 3)$.



assinante	(ação, doc.)	gostou?	d^2 até N
A	(4, 4)	não	
B	(6, 3)	sim	
C	(5, 6)	não	
D	(3, 2)	sim	
E	(7, 5)	não	
F	(2, 5)	sim	
G	(8, 2)	sim	

(a) Complete a coluna da distância ao quadrado (a ordem dos vizinhos é a mesma da distância).

(b) Dê a previsão e o `predict_proba` de "gostou" com $k = 1, 3, 5$ e 7.

(c) Com $k = 7$, o que o modelo responde para *qualquer* assinante novo? Por quê?

2 A unidade escolhe o vizinho

• AQUECIMENTO

Uma agência de viagens quer prever se um cliente novo, X, com renda de R$ 4.000 por mês e 30 anos, compra um pacote de mochilão. Dois clientes antigos: P (R$ 4.300; 58 anos; não comprou) e Q (R$ 5.200; 31 anos; comprou).

(a) Sem escalonar, calcule a distância euclidiana de X a P e a Q. Qual é o 1-NN e o que ele prevê?

(b) Divida a renda pelo desvio padrão dela (R$ 2.000) e a idade pelo dela (14 anos) e refaça. E agora?

(c) Sem escalonar, mas com a renda anotada em milhares de reais (4,0; 4,3; 5,2), quem fica mais perto? O que isso diz sobre o k-NN sem escalonador?

(d) Onde as médias e os desvios do escalonador devem ser calculados?

3 O código da colega

●● PRÁTICA

Uma colega rodou o código abaixo e escreveu: “escolho $k = 1$, que acerta 100%”. Um colega respondeu: “não, escolha $k = 5$, que tem o melhor teste, e reporte 0,900”.

```
X, y = make_moons(n_samples=400, noise=0.3, random_state=1)
Xs = StandardScaler().fit(X).transform(X)
X_tr, X_te, y_tr, y_te = train_test_split(Xs, y, test_size=0.25, random_state=0)
for k in [1, 5, 15, 45, 135]:
    knn = KNeighborsClassifier(n_neighbors=k).fit(X_tr, y_tr)
    print(f"k={k:3d}  treino {knn.score(X_tr, y_tr):.3f}  teste {knn.score(X_te, y_te):.3f}")
```

```
k=  1  treino 1.000  teste 0.840
k=  5  treino 0.920  teste 0.900
k= 15  treino 0.910  teste 0.890
k= 45  treino 0.893  teste 0.890
k=135  treino 0.837  teste 0.860
```

(a) O que está errado no argumento da colega?

(b) O que está errado no argumento do colega? Com 100 linhas de teste, quanto vale uma linha?

(c) Há um terceiro problema no código, antes do laço. Qual?

(d) Reescreva a receita em quatro passos curtos.

4 Escolher o k

●● PRÁTICA

Mil pontos de duas luas com ruído, divididos em 600 de treino, 200 de validação e 200 de teste. Acurácia do k-NN:

k	1	3	5	15	73	151	451
treino	1,000	0,913	0,900	0,888	0,878	0,865	0,772
validação	0,850	0,865	0,880	0,895	0,910	0,875	0,765

(a) Que k você escolhe? Onde há overfitting e onde há underfitting?

(b) Na validação, quantas linhas separam $k = 15$ de $k = 73$? Essa diferença é convincente?

(c) Com o k escolhido, o modelo foi reajustado em treino + validação e acertou 0,900 no teste, menos que os 0,910 da validação. Por que isso era esperado? Pode-se agora voltar e trocar o k?

CONFIRA NO SIMULADOR

O voto muda com k

Sem mover o ponto de consulta (a estrela), percorra o k de 1 a 25 e veja a resposta mudar; depois arraste-o pela zona em que as classes se misturam e compare k pequeno com k grande.

5 Outras réguas

●● PRÁTICA

(a) Calcule as distâncias de Manhattan, euclidiana e de Chebyshev entre $a = (2, 1, 0)$ e $b = (5, 5, 12)$.

(b) Uma consulta em $(0, 0)$ e dois pontos de treino, $U = (3, 3)$ e $V = (5, 0)$. Qual é o mais próximo em cada uma das três distâncias?

(c) Volte ao exercício 1 e refaça o voto com `weights='distance'` (peso $1/d$ por vizinho) para $k = 3$ e $k = 5$. Alguma resposta muda?

6 Quando o k-NN quebra

● ● ● DESAFIO

Três medições:

- 1.000 pontos ao acaso num cubo de d dimensões; para uma consulta típica, (mais distante − mais próximo)/mais próximo vale 71,21 com $d = 2$; 2,69 com $d = 10$; 0,44 com $d = 100$; 0,12 com $d = 1.000$.
- Um conjunto de vinhos com 13 atributos e colunas de ruído puro acrescentadas, todas padronizadas: com 0, 10, 100 e 1.000 colunas de ruído, o 5-NN acerta 0,963; 0,963; 0,796; 0,556 no teste e uma regressão logística, 1,000; 0,944; 0,907; 0,685.
- Duas classes em círculos concêntricos (dois atributos): regressão logística 0,493; 15-NN 0,993.

(a) O que a primeira medição diz sobre a ideia de "vizinho mais próximo" em dimensão alta?

(b) Por que o k-NN cai mais rápido que a regressão logística quando entram colunas de ruído? Padronizar resolve?

(c) Que modelo você usaria (k-NN ou regressão logística) em cada caso: (i) dois atributos, 50 mil linhas, fronteira curva; (ii) 300 colunas, poucas relevantes, previsões em milissegundos; (iii) explicar a um cliente por que *o pedido dele* foi recusado?

ATIVIDADE DA AULA

C3

Métricas de classificação

insper.github.io/ML/pt/2026.2/classes/classification-metrics/

respostas comentadas na página 157

A pergunta da aula. Em 1.000 pacientes, só 10 estão doentes. Responder "saudável" para todo mundo acerta 99,0%; um modelo que encontra 5 dos 10 doentes, com 10 alarmes falsos, acerta só 98,5%. A acurácia escolheu o modelo que não encontra ninguém. Quais erros cada modelo comete e com que número medir cada um?

ETAPAS DO FLUXO NESTA AULA avaliação

O que você precisa saber

- O modelo dá um **escore** a cada caso; nós o transformamos em rótulo com um **limiar** t (o `.predict()` usa 0,5). Cada caso cai num de quatro grupos: TP (verdadeiro positivo: doente e marcado), FN (falso negativo: doente e chamado de saudável, um escape), FP (falso positivo: saudável e marcado, um alarme falso) e TN (verdadeiro negativo).
- **Matriz de confusão** no layout padrão: real nas linhas, previsto nas colunas, positivo primeiro, $\begin{bmatrix} TP & FN \\ FP & TN \end{bmatrix}$. No scikit-learn, `confusion_matrix(y_true, y_pred, labels=[1, 0])`; sem `labels`, os rótulos saem em ordem crescente e o canto superior esquerdo vira o TN.
- **Acurácia** $= (TP + TN)/n$. Com classes desbalanceadas, a resposta constante já acerta muito (o **paradoxo da acurácia**): reporte sempre a baseline (`DummyClassifier`) ao lado.
- Cada métrica lê **uma linha ou uma coluna**. Linha dos doentes: **revocação** (recall, sensibilidade ou TPR, taxa de verdadeiros positivos) $= TP/(TP + FN)$. Linha dos saudáveis: **especificidade** $= TN/(TN + FP)$ e FPR (taxa de falsos positivos) $= FP/(FP + TN) = 1-$ especificidade. Coluna dos marcados: **precisão** $= TP/(TP + FP)$. Coluna dos liberados: VPN (valor preditivo negativo) $= TN/(TN + FN)$.
- Subir o limiar marca menos casos: a revocação só pode cair; a precisão tende a subir, mas nem sempre. O limiar é uma decisão, tomada na **validação**.
- **F1** $= \dfrac{2 \cdot \text{precisão} \cdot \text{revocação}}{\text{precisão} + \text{revocação}} = \dfrac{2\,TP}{2\,TP + FP + FN}$: a média harmônica, puxada para o menor dos dois; nunca olha o TN. O F_β faz um escape pesar β^2 alarmes falsos.
- Revocação e especificidade são proporções dentro de uma classe e não mudam com a **prevalência** (a fração de positivos); precisão, VPN e acurácia misturam as duas linhas e mudam.

1 Uma pergunta por métrica

• AQUECIMENTO

Associe cada pergunta à métrica que a responde: acurácia, precisão, revocação, especificidade ou VPN.

(a) Rastreamento de câncer: “dos doentes, quantos o exame encontra?” _____

(b) Pronto-socorro que manda para casa quem não está infartando: “dos que libero, quantos estão bem?” _____

(c) Equipe antifraude com 50 ligações por dia: “das ligações, quantas são fraude de verdade?” _____

(d) Rastreamento em massa: “dos saudáveis, quantos *não* recebem uma carta de alarme?” _____

(e) “Em quantos casos, doentes ou não, o modelo acertou?” _____

2 Quatro grupos e uma matriz

• AQUECIMENTO

Vinte pacientes encaminhados a um especialista, em ordem de escore do modelo (1 = doente, segundo exames posteriores):

id	escore	doente	grupo	id	escore	doente	grupo
P01	0,95	1		P11	0,40	0	
P02	0,90	1		P12	0,35	1	
P03	0,85	1		P13	0,30	0	
P04	0,80	1		P14	0,25	0	
P05	0,75	0		P15	0,20	1	
P06	0,70	1		P16	0,15	0	
P07	0,65	1		P17	0,10	0	
P08	0,60	0		P18	0,08	0	
P09	0,55	1		P19	0,05	0	
P10	0,45	1		P20	0,02	0	

(a) Com limiar $t = 0{,}5$ (marcar quem tem escore $\geq 0{,}5$), escreva o grupo de cada paciente (TP, FN, FP ou TN) e complete a matriz no layout padrão.

(b) Um colega rodou `confusion_matrix(y_true, y_pred)`, sem `labels`, e recebeu `[[8 2] [3 7]]`. Ele escreveu “TP = 8”. O que está em cada canto dessa matriz? Como ele obtém o layout padrão?

	previsto: doente	**previsto: saudável**
real: doente		
real: saudável		

3 Uma triagem de fraude

●● PRÁTICA

Uma triagem de fraude olhou 200 transações, das quais 20 eram fraude. O modelo marcou 25 transações; 15 delas eram fraude.

(a) Monte a matriz de confusão (layout padrão).

(b) Calcule acurácia, precisão, revocação, especificidade, FPR, VPN e F1. Para cada uma, diga se ela lê uma linha, uma coluna ou a matriz inteira.

(c) Quanto acerta um modelo que nunca marca nada? Compare com a acurácia do modelo: o que ela diz sobre ele?

CONFIRA NO SIMULADOR

Uma matriz, todas as métricas

Aperte *o modelo preguiçoso* e compare a acurácia com a barra da baseline sempre-maioria. Depois suba a prevalência sem mexer na sensibilidade nem na especificidade e acompanhe a precisão.

4 Média comum ou média harmônica

●● PRÁTICA

Três maneiras de cortar os mesmos 20 pacientes: marcar só o P01; marcar com limiar 0,5; marcar os 20.

(a) Para cada corte, calcule a precisão, a revocação, a média aritmética das duas e o F1.

(b) Qual corte cada média põe em primeiro lugar? Por que se usa a média harmônica?

(c) Um modelo tem precisão 1,0 e revocação 0,02. Quanto vale o F1, aproximadamente?

5 Subir e descer a linha

●● PRÁTICA

Com os 20 pacientes do exercício 2, complete a tabela.

limiar	marcados	TP	FP	FN	TN	precisão	revocação	F1
0,3								
0,5								
0,7								

(a) Quando o limiar sobe de 0,3 para 0,7, para onde vão a precisão e a revocação? O modelo mudou?

(b) Calcule a precisão em $t = 0{,}60$, 0,65 e 0,70. Por que a revocação só pode cair quando o limiar sobe, mas a precisão nem sempre sobe?

(c) Numa triagem em que um escape custa muito mais que um exame a mais, qual dos três limiares da tabela você levaria e por quê?

6 O mesmo modelo, outra população

● ● ● DESAFIO

O modelo dos 20 pacientes tem revocação 0,70 e especificidade 0,80 no limiar 0,5. Ele vai ser usado para rastrear a população geral, em que a doença é mais rara. Conte 10.000 pessoas em cada prevalência:

prevalência	doentes	TP	FP	precisão	VPN	acurácia	acurácia balanceada
50%							
10%							
1%							

Dica: a acurácia balanceada é a média da revocação e da especificidade.

(a) Complete a tabela. Com 1% de prevalência, quantos alarmes falsos há para cada doente encontrado?

(b) Quais colunas mudaram e quais não? Explique pela matriz: que linhas cada métrica usa?

(c) O hospital mediu precisão 0,778 no consultório do especialista e quer anunciá-la para o rastreamento. O que você responde? O que deve ser reportado junto com uma precisão?

(d) No limiar 0,5, o $F_2 = \dfrac{5\,TP}{5\,TP + 4\,FN + FP}$ dos 20 pacientes vale quanto? Fica acima ou abaixo do F1 do exercício 5? Por quê?

ATIVIDADE DA AULA

C4

ROC, PR e dados desbalanceados

insper.github.io/ML/pt/2026.2/classes/roc-imbalanced/

respostas comentadas na página 158

A pergunta da aula. Com limiar 0,5, os 20 pacientes da aula de métricas dão TP = 7 (verdadeiros positivos), FN = 3 (falsos negativos), FP = 2 (falsos positivos) e TN = 8 (verdadeiros negativos). Mas o modelo deu uma probabilidade a cada paciente e o 0,5 foi escolha nossa. Como avaliar o modelo em todos os limiares de uma vez? E como escolher o limiar que vai para produção, sobretudo quando a classe de interesse é 2% das linhas?

ETAPAS DO FLUXO NESTA AULA avaliação

O que você precisa saber

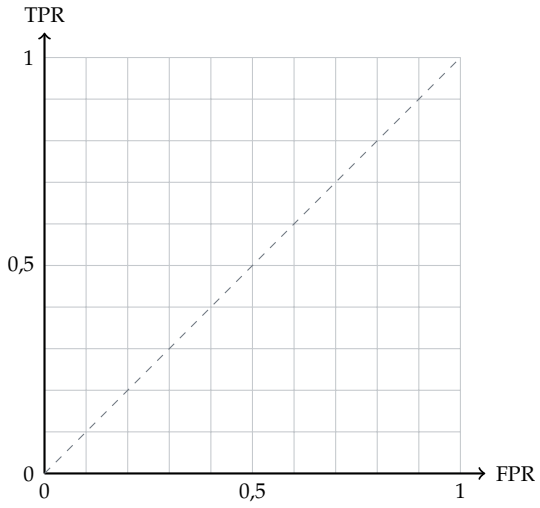
- Um classificador que dá escores é uma família de classificadores, um por limiar t. Em cada um: TPR (taxa de verdadeiros positivos, o recall) $= TP/(TP+FN)$, a fração dos doentes pegos; FPR (taxa de falsos positivos) $= FP/(FP+TN)$, a fração dos saudáveis alarmados.
- A **curva ROC** (*receiver operating characteristic*) liga os pontos (FPR, TPR) de todos os limiares. Baixar o limiar além de um doente sobe $1/P$ (P = número de positivos); além de um saudável, anda $1/N$ para a direita (N = negativos). A diagonal é um modelo que sorteia escores.
- **AUC** (área sob a curva ROC) = fração dos pares (positivo, negativo) em que o positivo tem o escore maior; empate vale meio. 0,5 é o acaso, 1 é perfeito, abaixo de 0,5 a ordem está invertida. **Só a ordem importa**: a AUC não diz nada sobre calibração nem sobre um limiar. Passe escores, nunca rótulos.
- A **curva PR** (precisão–recall) põe a precisão contra o recall. O modelo sem habilidade é uma horizontal na **prevalência** π (a fração de positivos). A **AP** (*average precision*) $= \sum_k (R_k - R_{k-1})\, P_k$ é a média da precisão na posição de cada positivo; leia-a contra π.
- Em N pessoas há πN doentes e $(1-\pi)N$ saudáveis, então $TP = \text{TPR}\cdot\pi N$, $FP = \text{FPR}\cdot(1-\pi)N$ e precisão $= \dfrac{\pi\,\text{TPR}}{\pi\,\text{TPR} + (1-\pi)\,\text{FPR}}$. A ROC não muda com a prevalência; a PR muda.
- O limiar é escolhido na **validação**, por um critério declarado: meta de recall, F1 máximo, **J de Youden** $=$ TPR $-$ FPR, custo mínimo. Com probabilidades calibradas, sinalizar custa $(1-p)\,c_{FP}$ e não sinalizar $p\,c_{FN}$: sinalize se $p > t^* = c_{FP}/(c_{FP}+c_{FN})$.
- Pesos de classe (`class_weight='balanced'`: $w_c = n/(k\,n_c)$, com n linhas, k classes e n_c linhas da classe c) e **reamostragem** (subamostrar, sobreamostrar ou criar pontos sintéticos da minoria com o SMOTE, *synthetic minority over-sampling technique*) basicamente movem o ponto de operação. Reamostre só o treino, dentro de um pipeline do `imblearn`.

1 Um ponto por limiar

• AQUECIMENTO

Os 20 pacientes, em ordem de escore (1 = doente):

	P01	P02	P03	P04	P05	P06	P07	P08	P09	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20
escore	0,95	0,90	0,85	0,80	0,75	0,70	0,65	0,60	0,55	0,45	0,40	0,35	0,30	0,25	0,20	0,15	0,10	0,08	0,05	0,02
doente	1	1	1	1	0	1	1	0	1	1	0	1	0	0	1	0	0	0	0	0



(a) Comece em (0, 0), com o limiar acima de todos os escores, e baixe-o paciente a paciente até marcar todos. Desenhe a curva ROC na grade.

(b) Marque o ponto do limiar 0,5. Quais são as coordenadas dele?

(c) Como seria a curva de um modelo perfeito? E a de um que sorteia escores?

(d) Quantos pontos distintos a curva tem? Um colega rodou `roc_curve` e recebeu só 12. Que pontos sumiram e por que a curva não muda?

2 Contar pares

• AQUECIMENTO

Com os 20 pacientes do exercício 1, cada par (doente, saudável) está na ordem certa quando o doente tem o escore maior.

saudável	P05	P08	P11	P13	P14	P16	P17	P18	P19	P20	**total**
doentes acima dele											

(a) Complete a tabela. Quantos pares há ao todo e quanto vale a AUC?

(b) Compare cada número da tabela com a altura da sua curva ROC em cada passo para a direita. Por que a área sob a escada dá o mesmo valor?

(c) Quanto vale a AUC dos escores ao quadrado? E de $100 \cdot \text{escore} - 7$? E de $1 - \text{escore}$?

(d) Um colega passou os rótulos de `predict` (limiar 0,5) ao `roc_auc_score`. A "curva" tem um só ponto, ligado aos cantos (0, 0) e (1, 1) por retas. Calcule a área. Que métrica da aula de métricas ela é?

3 A curva que mede os marcados

•• PRÁTICA

Ainda com os 20 pacientes.

(a) Em que posição da fila (1 = maior escore) está cada um dos 10 doentes? Calcule a precisão de quem é marcado até cada um deles.

(b) Calcule a AP. Qual seria a AP de um modelo que sorteia escores, nestes dados?

(c) Um colega calculou a área sob a curva PR ligando os pontos por retas (`auc(recall, precision)`) e obteve 0,8608. Qual dos dois números reportar e por quê?

doente nº	1	2	3	4	5	6	7	8	9	10
posição										
precisão										

4 Quando a doença fica rara

●● PRÁTICA

Um modelo opera num limiar com TPR 0,80 e FPR 0,12. Ele é usado em três populações, com prevalência π de 50%, 20% e 1%.

(a) Calcule a precisão nas três. Com $\pi = 1\%$, quantos alarmes falsos há para cada doente encontrado?

(b) O ponto da curva ROC muda entre as três populações? E a curva PR? Explique pelos denominadores da FPR e da precisão.

(c) Dois relatórios dizem "AP 0,25". Um foi medido com 50% de positivos e o outro com 1%. Qual é bom? Quantas vezes melhor que o acaso?

(d) Que curva você usaria para uma triagem em que só 2% das linhas são positivas e cada caso marcado é revisto por um radiologista?

5 Quatro critérios, nem sempre o mesmo limiar

●● PRÁTICA

Trate os 20 pacientes como o conjunto de validação. Deixar passar um doente custa 10 vezes mais que um encaminhamento desnecessário: custo $= 10 \cdot FN + FP$.

t	TP	FP	FN	precisão	recall	F1	J de Youden	custo
0,50								
0,45								
0,35								
0,20								
0,10								

(a) Complete a tabela. Que limiar escolhe cada critério: F1 máximo; J máximo; recall $\geq 0{,}9$ com a melhor precisão; custo mínimo?

(b) Se os escores fossem probabilidades calibradas, qual seria o limiar t^*? Por que o mínimo da tabela não cai exatamente nele?

(c) Por que os critérios discordam, se o modelo é o mesmo? Quem decide qual usar?

CONFIRA NO SIMULADOR

O limiar é uma escolha

Deslize o limiar de 0,5 para baixo e veja o ponto ROC andar para cima e para a direita, enquanto a precisão cai. Leve o limiar até 0,98 e compare a acurácia com o recall.

6 Um 5-NN quase perfeito

● ● ● DESAFIO

Uma triagem de fraude simulada tem 12.000 transações de treino, 233 delas fraude (cerca de 2%).

(a) Calcule os pesos w_0 e w_1 de `class_weight='balanced'`. Cada fraude pesa quantas transações legítimas?

(b) No mesmo teste, o modelo *sem* pesos com limiar 0,0194 (a prevalência do treino) marca 685 transações, recall 0,833, precisão 0,095; o modelo *com* pesos no limiar 0,5 marca 673, recall 0,821, precisão 0,095. A ROC-AUC é 0,930 e 0,929. O que os pesos mudaram e o que não mudaram? Que alternativa mais barata existe?

(c) Um colega mediu assim:

```
X_r, y_r = RandomOverSampler().fit_resample(X, y)   # repete cada fraude ~50x
Xa, Xb, ya, yb = train_test_split(X_r, y_r, test_size=0.2, stratify=y_r)
pred = KNeighborsClassifier(5).fit(Xa, ya).predict(Xb)
```

Ele reporta recall 1,00 e precisão 0,96. Feito do jeito certo, o mesmo 5-NN (k vizinhos mais próximos, k-NN, com $k = 5$) tira recall 0,42 e precisão 0,20. Aponte as duas coisas que inflaram o número dele e reescreva o procedimento em palavras.

ATIVIDADE DA AULA

C5

Máquinas de vetores de suporte (SVM)

insper.github.io/ML/pt/2026.2/classes/svm/

respostas comentadas na página 159

A pergunta da aula. Seis pontos, três de cada classe, e infinitas retas que separam todos eles: todas com 100% de acurácia no treino, então a acurácia não escolhe. Qual reta a máquina de vetores de suporte (SVM, de *support vector machine*) escolhe, quais pontos decidem a resposta e como ela traça fronteiras curvas sem deixar de ser linear?

ETAPAS DO FLUXO NESTA AULA modelo avaliação

O que você precisa saber

- Rótulos $y \in \{-1, +1\}$; escore $f(x) = w^\top x + b$; a previsão é o sinal de f. A **margem com sinal** $m = y\,f(x)$ é positiva do lado certo e grande longe da fronteira $f(x) = 0$.
- Distância de x até a fronteira: $|w^\top x + b| / \|w\|$. A SVM fixa $|f| = 1$ nos pontos mais próximos; as bordas da **rua** são $f = \pm 1$ e ela tem largura $2/\|w\|$. Maximizar a rua é $\min \frac{1}{2}\|w\|^2$ com $y_i f(x_i) \geq 1$: um problema convexo, com um único ótimo.
- **Vetores de suporte**: os pontos sobre a borda (ou dentro da rua). $w = \sum_i \alpha_i y_i x_i$, com $\alpha_i = 0$ para todo ponto com $m > 1$: apagar um deles não muda nada.
- **Margem suave**: a folga mais barata do ponto i é a **perda hinge** $\max(0,\ 1 - m_i)$; minimiza-se $\frac{1}{2}\|w\|^2 + C \sum_i \max(0,\ 1 - m_i)$. A regressão logística tem a mesma forma, com a log loss $\ln(1 + e^{-m})$, que nunca é zero.
- O C é o preço de uma violação: C pequeno dá rua larga e muitos vetores de suporte (regularização forte); C grande, rua estreita e poucos. Escolha-o na validação cruzada (CV).
- **Kernel**: $K(u, v) = \langle \phi(u), \phi(v) \rangle$ calcula o produto interno num espaço de atributos maior sem montar ϕ. RBF (função de base radial): $K = e^{-\gamma \|u - v\|^2}$, uma semelhança; γ grande deixa cada vetor de suporte com alcance curto. C e γ são buscados juntos; escalonar os atributos é obrigatório.

1 A que distância da fronteira

• AQUECIMENTO

Uma SVM linear tem $w = (3, 4)$ e $b = -5$.

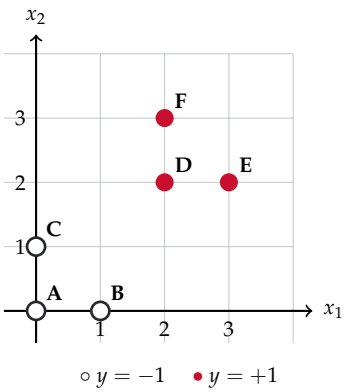
ponto	x	y	$f(x)$	$m = y\,f(x)$	distância	hinge
A	$(3, 1)$	$+1$				
B	$(0, 0)$	$+1$				
C	$(1; 0{,}5)$	-1				
D	$(1, 1)$	$+1$				
E	$(0, 1)$	-1				

(a) Complete a tabela. Quanto mede a rua?

(b) Quais pontos estão do lado errado? Quais estão sobre uma borda da rua? Quais a hinge ignora?

2 Seis pontos, à mão

•• PRÁTICA



Margem rígida (nenhuma violação), três pontos de cada classe.

(a) Por simetria, a fronteira é perpendicular à diagonal: $w = (a, a)$. Quais pontos de cada classe ficam mais perto dela? Onde passa a fronteira?

(b) Imponha $f = +1$ no positivo mais próximo e $f = -1$ num negativo mais próximo e resolva para a e b.

(c) Calcule a largura da rua e desenhe a fronteira e as duas bordas.

(d) Calcule m dos seis pontos. Quais são os vetores de suporte?

3 Apagar um ponto

●● PRÁTICA

Com a solução do exercício 2:

(a) O solver devolve $\alpha_i y_i = -\frac{2}{9}$ para B, $-\frac{2}{9}$ para C e $\frac{4}{9}$ para D e zero para os demais. Confira que $w = \sum_i \alpha_i y_i x_i$.

(b) Apague o ponto E e reajuste. O que muda? Por quê?

(c) Apague o ponto D e refaça a conta do exercício 2. Qual é a nova fronteira e a nova largura da rua?

(d) Numa regressão logística, apagar o ponto E mudaria a fronteira? Por quê?

4 Três perdas, quatro pontos

●● PRÁTICA

Com $w = (\frac{2}{3}, \frac{2}{3})$ e $b = -\frac{5}{3}$, quatro pontos novos, todos da classe $+1$. A coluna da log loss, $\ln(1 + e^{-m})$, já está calculada.

ponto	$m = y\,f(x)$	hinge	log loss	perda 0-1
(3, 3)			0,093	
(2, 2)			0,313	
(1,5; 1,5)			0,540	
(1; 0,5)			1,081	

(a) Complete a tabela. Qual ponto está certo mas é cobrado pela hinge? Por quê?

(b) Que perda ainda "escuta" o ponto (3, 3)? O que isso tem a ver com os vetores de suporte?

5 O botão C

●● PRÁTICA

SVM linear num conjunto de 263 retornos de radar (34 atributos padronizados), CV 5-fold:

C	0,01	0,1	1	10	100
largura da rua	2,979	1,219	0,525	0,256	0,135
vetores de suporte	137	96	70	58	50
acurácia de treino	0,886	0,913	0,947	0,954	0,962
acurácia na CV	0,863	0,871	0,886	0,871	0,852

(a) Por que a rua encolhe e os vetores de suporte diminuem quando o C sobe?

(b) Que C você escolhe? Onde há underfitting e onde há overfitting? Use o vão entre treino e CV.

(c) Por que não escolher pelo treino?

CONFIRA NO SIMULADOR

Margem suave: o que o C compra de fato

Arraste um ponto sem círculo para longe da rua e veja a fronteira ficar parada; depois arraste um vetor de suporte. Mova o C e conte os círculos.

6 Uma reta em outro espaço

● ● ● DESAFIO

(a) Cinco pontos numa reta, $x = -2, -1, 0, 1, 2$, com $y = +1$ nos dois das pontas e -1 nos três do meio. Qual a melhor acurácia de um único limiar em x? Leve cada ponto a (x, x^2): ache a fronteira de margem máxima nesse plano (por simetria, ela é horizontal) e diga a que limiares ela corresponde em x.

(b) Para $\phi(v) = (v_1^2,\ \sqrt{2}\,v_1 v_2,\ v_2^2)$, $u = (2, 1)$ e $v = (1, 1)$: calcule $\phi(u) \cdot \phi(v)$ e $(u \cdot v)^2$. Por que isso importa quando há 34 atributos e grau 3?

(c) Kernel RBF com $\gamma = 0{,}5$: quanto vale K entre dois pontos a distância 1? E a distância 3? E com $\gamma = 2$, a distância 1? O que um γ grande faz com a fronteira?

(d) No radar, com $\gamma = 1$, a acurácia de treino é 1,000 e a da CV fica em 0,665, logo acima dos 0,643 de responder sempre a classe maior. O que aconteceu?

Dica: $e^{-0{,}5} \approx 0{,}607$; $e^{-2} \approx 0{,}135$; $e^{-4{,}5} \approx 0{,}011$.

ATIVIDADE DA AULA

C6

Validação cruzada e vazamento de dados

insper.github.io/ML/pt/2026.2/classes/validation/

respostas comentadas na página 159

A pergunta da aula. Mesmo modelo, mesmos dados, só a semente da divisão 80/20 mudou: um aluno reporta 100% de acurácia e outro 93%. Qual dos dois números vale? Como medir uma receita sem depender da sorte de uma divisão e sem deixar informação de fora do treino vazar para dentro dele?

ETAPAS DO FLUXO NESTA AULA dados e divisão avaliação

O que você precisa saber

- Três conjuntos: **treino** (ajusta os parâmetros), **validação** (escolhe modelos e hiperparâmetros; desgasta-se com o uso) e **teste** (usado uma vez, no fim). O escore do escolhido, no conjunto em que foi escolhido, é otimista: a **maldição do vencedor**.
- **Validação cruzada** (CV, de *cross-validation*) k-fold: divida o treino em k partes; cada parte valida uma vez enquanto as outras $k-1$ treinam. São k ajustes; reporte **média $\pm$ desvio**. A CV avalia uma **receita** (algoritmo, hiperparâmetros e pré-processamento): os k modelos são descartados e a receita é reajustada em todo o treino.
- O divisor diz quais linhas podem cair juntas na validação: `StratifiedKFold` (classificação, mantém a fração de cada classe), `GroupKFold` (várias linhas por paciente ou cliente), `TimeSeriesSplit` (só valida no futuro), `KFold(shuffle=True)` (regressão sem ordem).
- Todo passo que aprende (scaler, imputador, seleção de atributos, *target encoding*, SMOTE, a sobreamostragem sintética da minoria) vai **dentro do `Pipeline`**, para ser reajustado em cada fold. Se o passo olha o y, ajustá-lo antes da CV pode inflar muito o escore.
- **Vazamento** é informação de fora do fold de treino entrando no treino. Cinco padrões: pré-processamento antes da divisão; atributo registrado depois do momento da previsão; linhas duplicadas; várias linhas da mesma entidade dos dois lados (grupo); dados com tempo embaralhados. Bom demais para ser verdade? Suspeite de vazamento primeiro.
- Comparar com justiça: os mesmos folds para todos (um único objeto `cv`), cada candidato ajustado, diferenças **fold a fold**. Se a média das diferenças é menor que o desvio delas, é **empate** e fica o mais simples. A **CV aninhada** avalia o procedimento "buscar e escolher" em folds que a busca não viu.

1 Quantas vezes, com quantas linhas

• AQUECIMENTO

Um conjunto de treino tem 455 linhas e é avaliado com `StratifiedKFold(n_splits=5)`.

(a) Em cada fold, quantas linhas treinam e quantas validam? Quantos ajustes são feitos? Quantas vezes cada linha é validada?

(b) Terminada a CV, o que se faz com os cinco modelos ajustados? Que modelo vai para produção?

(c) Um `GridSearchCV` testa 13 valores de `C` com esse mesmo `cv` e, no fim, reajusta o melhor (`refit=True`). Quantos ajustes ao todo? E uma `RepeatedStratifiedKFold(n_splits=5, n_repeats=10)` de um modelo só?

2 Que divisor?

• AQUECIMENTO

Escreva o divisor de validação cruzada adequado a cada caso (ou "nenhum resolve").

(a) Classificar transações como fraude, 2% de positivos, sem ordem no tempo. _____

(b) Diagnóstico a partir de exames; cada paciente tem 10 exames na tabela. _____

(c) Prever o preço de amanhã a partir dos preços diários passados. _____

(d) Prever o preço de casas, linhas sem ordem nem grupos. _____

(e) Prever readmissão hospitalar usando o número de consultas de acompanhamento após a alta. _____

E se o arquivo tiver 100 linhas, 8 positivas, ordenadas por classe (as 8 positivas primeiro), e alguém usar `KFold(5)` sem embaralhar? Quantas positivas caem em cada fold de validação?

3 Cinco folds, cinco notas

●● PRÁTICA

Recall de uma regressão logística em cada fold de validação de uma CV 5-fold; cada fold tem 34 tumores malignos. O recall médio de treino foi 0,971.

fold	0	1	2	3	4
recall	0,882	0,971	1,000	0,941	0,971
tumores perdidos					

(a) Complete a última linha. Quanto de recall vale um tumor?

(b) Calcule a média e o desvio padrão (divida por 5). Escreva o resultado como se reporta.

(c) O modelo é pior no fold 0? De onde vem a diferença entre os folds?

(d) Compare treino e validação: há sinal de overfitting? E de underfitting?

CONFIRA NO SIMULADOR

A sorte da divisão

Com $n = 80$, aperte *Sortear 100* e compare a faixa do hold-out com a da média da CV. Depois aperte *Novo conjunto* e veja o que a CV não corrige.

4 O mesmo ruído, dois números

●● PRÁTICA

Cem linhas, 5.000 colunas de puro ruído e um alvo no cara ou coroa: não há nada a aprender.

```
# versão A
X_sel = SelectKBest(f_classif, k=20).fit_transform(X, y)
cross_val_score(LogisticRegression(), X_sel, y, cv=cv).mean()      # 0.89
# versão B
pipe = make_pipeline(SelectKBest(f_classif, k=20), LogisticRegression())
cross_val_score(pipe, X, y, cv=cv).mean()                          # 0.53
```

(a) Em cada versão, o `SelectKBest` aprende com quais linhas? Qual número é honesto?

(b) Como a versão A tira 89% em dados sem sinal nenhum?

(c) Trocar o `SelectKBest` por um `StandardScaler` ajustado antes da CV, num conjunto de 569 linhas, não mudou nada mensurável (97,9% nos dois casos). Isso prova que escalonar fora do pipeline é seguro? Qual a diferença entre os dois passos?

CONFIRA NO SIMULADOR

Vazamento, medido em puro ruído

Aumente o número de atributos de ruído e compare a barra reportada com a honesta: só uma delas sai do acaso.

5 Um 1-NN que acerta no cara ou coroa

●● PRÁTICA

Trezentas linhas de ruído com rótulos sorteados. Cada linha ganha uma gêmea quase idêntica (a mesma linha mais um ruído minúsculo), com o mesmo rótulo: 600 linhas. Um 1-NN (k vizinhos mais próximos, k-NN, com $k = 1$) é avaliado com `KFold(5, shuffle=True)`.

(a) Para uma linha de validação, com que probabilidade (aproximada) a gêmea dela está no treino? O que o 1-NN prevê nesse caso? E no outro?

(b) Calcule a acurácia esperada. Quanto ela seria sem as gêmeas?

(c) Que divisor dá o número honesto? Em que situações reais aparecem gêmeas assim?

6 Empate ou vitória?

●●● DESAFIO

Precisão média (AP, *average precision*) de quatro candidatos nos mesmos cinco folds de uma triagem com 2,3% de positivos:

candidato	fold 1	2	3	4	5	média ± desvio
logística	0,670	0,643	0,567	0,556	0,672	0,622 ± 0,050
logística com `class_weight='balanced'`	0,652	0,603	0,527	0,533	0,611	0,585 ± 0,048
k-NN ajustado ($k = 25$)	0,715	0,771	0,734	0,609	0,729	0,712 ± 0,054
logística de grau 2 ajustada	0,738	0,761	0,678	0,631	0,747	0,711 ± 0,049

(a) Olhando só a última coluna, as duas primeiras linhas parecem empatadas? Agora calcule as diferenças fold a fold (logística menos balanceada), a média e o desvio delas. Empate ou vitória?

(b) Faça o mesmo para k-NN menos grau 2. Qual você põe em produção e por quê? (A logística de grau 2 são 28 números; o k-NN guarda as 8.946 linhas de treino.)

(c) O 0,712 do k-NN é o melhor de 35 configurações avaliadas nesses mesmos folds. Uma CV aninhada do procedimento inteiro deu 0,702. Por que ficou abaixo?

ATIVIDADE DA AULA

M1

Árvores de decisão

insper.github.io/ML/pt/2026.2/classes/decision-trees/

respostas comentadas na página 160

A pergunta da aula. Uma operadora quer saber quais clientes vão cancelar o plano no próximo trimestre. Com a conta mensal e o tempo de casa de cada cliente, que pergunta uma árvore faz primeiro, como ela responde a um cliente novo e por que ela não deve continuar perguntando até acertar todos os clientes de treino?

ETAPAS DO FLUXO NESTA AULA modelo avaliação

O que você precisa saber

- Uma árvore faz perguntas **atributo $\leq$ limiar**, da **raiz** até uma **folha**. A folha guarda as contagens de treino que caíram nela: `predict` devolve a classe majoritária; `predict_proba`, as proporções.
- **Impureza** de um nó com frações p_c de cada classe: **Gini** $G = 1 - \sum_c p_c^2$; **entropia** $H = -\sum_c p_c \log_2 p_c$. As duas valem 0 num nó puro; com duas classes em 50/50, $G = 0{,}5$ e $H = 1$ bit.
- Um corte que manda n_L dos n exemplos para a esquerda é pontuado pela **queda de impureza** $\Delta = I(\text{pai}) - \frac{n_L}{n} I(L) - \frac{n_R}{n} I(R)$. Os limiares testados são os **pontos médios** entre valores vizinhos; vence a maior queda. O crescimento é **guloso e recursivo**.
- Sem regra de parada, a árvore **decora**: folhas puras e 100% de acurácia de treino, até com rótulos aleatórios. **Pré-poda**: `max_depth`, `min_samples_leaf`, `max_leaf_nodes`. **Pós-poda**: `ccp_alpha`.
- Numa **árvore de regressão**, a folha prevê a média dos alvos e a impureza é o erro quadrático médio em torno dela. A previsão é uma escada: **a árvore não extrapola**.

1 Quão misturado está o nó?

• AQUECIMENTO

Complete a tabela. A taxa de erro é a fração de exemplos que não são da classe majoritária.

contagens no nó	Gini	entropia (bits)	taxa de erro
8 da classe A, 0 da B			
4 A, 4 B			
6 A, 2 B			
2 A, 1 B, 1 C (três classes)			

Dica: $\log_2 0{,}75 \approx -0{,}415$; $\log_2 0{,}5 = -1$; $\log_2 0{,}25 = -2$.

Por que o Gini do último nó passa de 0,5, o "máximo" citado no resumo?

2 A primeira pergunta

●● PRÁTICA

Dez clientes de uma operadora: `tenure` é o tempo de casa (meses), `bill` a conta mensal (R$) e `churn` vale 1 para quem cancelou.

cliente	A	B	C	D	E	F	G	H	I	J
`tenure`	30	3	12	5	24	2	8	36	4	6
`bill`	40	50	60	70	80	90	100	110	120	130
`churn`	0	0	0	1	0	1	1	0	1	1

(a) Calcule o Gini da raiz.

(b) Complete a tabela para três cortes candidatos. Qual deles a árvore escolhe e qual é a queda de Gini?

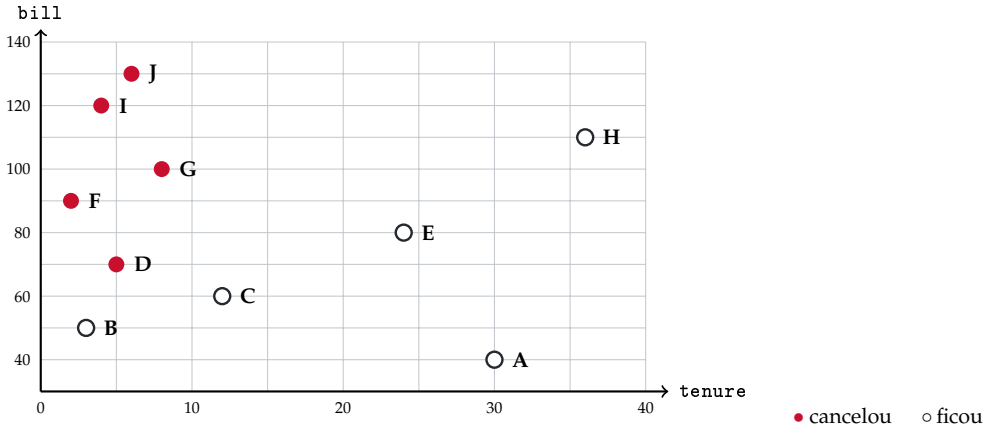
corte	**esq. (fica/cancela)**	**dir. (fica/cancela)**	**Gini esq.**	**Gini dir.**	**Gini ponderado**
`bill` ≤ 65					
`bill` ≤ 85					
`tenure` ≤ 10					

(c) Por que o CART (*classification and regression trees*, o algoritmo das árvores do scikit-learn) testa `bill` ≤ 65 e não ≤ 60 ou ≤ 70?

3 Crescer, prever e parar

●● PRÁTICA

No ramo `tenure` ≤ 10 do exercício anterior (clientes B, D, F, G, I, J), o melhor corte é `bill` ≤ 60, que deixa os dois filhos puros. O outro ramo já é puro.



(a) Desenhe no gráfico os dois cortes da árvore. Quantas folhas ela tem e qual é a acurácia de treino?

(b) Para os clientes novos P (7 meses, R$ 55), Q (20 meses, R$ 125) e R (9 meses, R$ 95), dê a previsão e a probabilidade de cancelar.

(c) Com `max_depth=1`, o que muda nas respostas de (a) e (b)? Por que o segundo corte merece desconfiança?

CONFIRA NO SIMULADOR

Árvore crescendo

Aumente a profundidade um nível por vez: as regiões viram caixinhas em volta de pontos isolados enquanto a impureza de treino continua caindo.

4 Uma chave para pinguins

●● PRÁTICA

Uma árvore de profundidade 2 treinada em 256 pinguins de três espécies. Os pesos de cada folha são as contagens de treino de (Adelie, Chinstrap, Gentoo):

```
|--- flipper_length_mm <= 206.50
|   |--- culmen_length_mm <= 43.35
|   |   |--- weights: [107.00, 4.00, 0.00] class: Adelie
|   |--- culmen_length_mm >  43.35
|   |   |--- weights: [4.00, 42.00, 1.00] class: Chinstrap
|--- flipper_length_mm >  206.50
|   |--- culmen_depth_mm <= 17.55
|   |   |--- weights: [0.00, 0.00, 91.00] class: Gentoo
|   |--- culmen_depth_mm >  17.55
|   |   |--- weights: [2.00, 5.00, 0.00] class: Chinstrap
```

(a) Escreva as quatro regras em português (nadadeira, comprimento e profundidade do bico, em mm).

(b) Classifique o pinguim X (nadadeira 210, bico de 50 de comprimento e 18,0 de profundidade) e o Y (nadadeira 190, bico de 40 de comprimento e 18,5 de profundidade). Com que probabilidade?

(c) Quantos pinguins de treino a chave erra? Em qual folha você confiaria menos e por quê?

5 Podar o elo mais fraco

DESAFIO

Volte à árvore completa do exercício 3. Seja $R(T)$ a impureza total das folhas de uma árvore T: a soma, sobre as folhas, de (fração das linhas de treino na folha) $\times$ (Gini da folha). A poda por custo-complexidade minimiza $R(T) + \alpha|T|$, em que $|T|$ é o número de folhas. Colapsar a subárvore T_t de um nó interno t numa folha passa a compensar a partir de

$$\alpha_{\text{ef}}(t) = \frac{R(t) - R(T_t)}{|T_t| - 1},$$

em que $R(t)$ é a impureza de t como folha (fração das linhas $\times$ Gini de t).

(a) Calcule α_{ef} do nó `bill` ≤ 60 (o que fica no ramo `tenure` ≤ 10) e o da raiz.

(b) Qual nó é podado primeiro e a partir de que α? Depois dessa poda, a partir de que α a raiz vira folha?

(c) Quantas folhas tem a árvore podada com $\alpha = 0{,}2$? E com $\alpha = 0{,}4$? Quem escolhe o α num projeto?

6 Aluguel em escada

DESAFIO

Seis apartamentos: área (m^2) e aluguel mensal (milhares de R$).

área	30	40	50	70	80	100
aluguel	1	2	3	6	7	8

A impureza de um nó é o erro quadrático médio (MSE, do inglês *mean squared error*) dos aluguéis em torno da média do nó.

(a) Calcule a média e o MSE da raiz.

(b) Calcule o MSE ponderado dos cortes área ≤ 45, área ≤ 60 e área ≤ 75. Qual vence? Que fração do MSE da raiz ele remove?

(c) Com esse único corte, quanto a árvore prevê para apartamentos de 65, 100 e 200 m^2? Uma árvore mais funda acertaria o de 200 m^2? Por que isso pesa num alvo que cresce com o tempo, como preços sob inflação?

ATIVIDADE DA AULA

M2
Random forest

insper.github.io/ML/pt/2026.2/classes/random-forest/

respostas comentadas na página 161

A pergunta da aula. No filtro de spam, as 200 árvores de uma floresta acertam em média 0,900 dos e-mails de validação e a melhor delas, 0,935; o voto das 200 acerta 0,961. Como uma média de árvores instáveis acerta mais que qualquer uma delas e o que impede essa média de melhorar para sempre?

ETAPAS DO FLUXO NESTA AULA modelo avaliação

O que você precisa saber

- A média de n estimativas, cada uma com variância σ^2 e correlação ρ entre pares, tem **variância** $\rho\sigma^2 + (1-\rho)\sigma^2/n$. O segundo termo some com n grande; o **piso** $\rho\sigma^2$ fica.
- **Bootstrap**: sortear n linhas *com reposição* das n que você tem. Cada linha fica de fora com probabilidade $(1-1/n)^n \approx 1/e \approx 0{,}37$.
- **Bagging**: árvores profundas (viés baixo, variância alta), cada uma no seu bootstrap; a previsão é a média ou o voto. A média corta variância, **não viés**.
- **Random forest** = bagging + sorteio de atributos, **ao mesmo tempo**: cada árvore tem o seu bootstrap das linhas e, em cada divisão, só `max_features` atributos sorteados são candidatos (padrão $\sqrt{d}$ na classificação, com d = número de atributos). O bagging sozinho não sorteia atributos. Cada árvore fica pior, mas as árvores ficam **menos correlacionadas** e a floresta, melhor.
- **Out-of-bag (OOB)**: cada linha de treino é votada só pelas árvores que não a sortearam; é uma validação de graça. Mais árvores **nunca pioram** o erro, só custam tempo.

1 A variância de uma média

• AQUECIMENTO

Num certo ponto, cada árvore de um bagging tem variância de previsão $\sigma^2 = 0{,}09$ e duas árvores quaisquer têm correlação $\rho = 0{,}5$.

(a) Calcule a variância da média de $n = 1$, 10 e 100 árvores. Qual é o limite com infinitas árvores?

(b) Sortear os atributos de cada divisão deixa cada árvore pior ($\sigma^2 = 0{,}12$), mas menos parecida com as outras ($\rho = 0{,}2$). Recalcule para $n = 100$ e para o limite. Valeu a troca? Em qual termo da fórmula o sorteio age?

2 Um bootstrap à mão

• AQUECIMENTO

Um conjunto de treino tem 8 linhas, numeradas de 1 a 8. Os 8 sorteios com reposição de uma amostra bootstrap deram: 3, 1, 3, 6, 8, 1, 2, 3.

(a) Quantas vezes cada linha foi sorteada? Quais linhas ficaram fora da sacola dessa árvore?

(b) Compare a fração deixada de fora com $(1 - 1/8)^8$. Com 1.000 linhas, quantas linhas distintas uma amostra bootstrap contém, mais ou menos?

3 O voto de quem não viu

•• PRÁTICA

Uma floresta de 5 árvores classifica e-mails como spam (S) ou legítimo (L). Na tabela, "·" quer dizer que a árvore treinou naquele e-mail (ele estava no seu bootstrap); uma letra é o voto da árvore num e-mail que ela *não* viu.

e-mail (classe real)	T1	T2	T3	T4	T5	previsão OOB	certa?
e1 (S)	S	·	S	L	·		
e2 (L)	·	L	·	L	S		
e3 (S)	L	·	·	L	S		
e4 (L)	·	·	L	·	·		
e5 (S)	S	S	·	·	S		
e6 (L)	·	S	S	·	L		

(a) Complete as duas colunas e calcule o erro OOB da floresta.

(b) Por que essa estimativa é honesta, se todas as árvores foram treinadas no mesmo conjunto de treino?

(c) Em qual e-mail a previsão OOB merece menos confiança? Por que o erro OOB de uma floresta de 5 árvores engana?

4 Árvores diferentes ou mais árvores?

●● PRÁTICA

Florestas de 200 árvores nos dígitos manuscritos (64 pixels por imagem), variando `max_features`; "correlação" é a correlação média entre os erros de duas árvores.

`max_features`	64 (bagging)	20	8	3	1
correlação entre árvores	0,304	0,233	0,198	0,160	0,133
erro de uma árvore	0,195	0,215	0,249	0,309	0,375
erro da floresta	0,053	0,027	0,023	0,025	0,030

(a) De 64 para 8, cada árvore piora e a floresta melhora. Explique com a fórmula do resumo.

(b) Por que a floresta volta a piorar com `max_features` $= 1$?

(c) Noutro conjunto de dados, o erro OOB de uma floresta de árvores completas foi 0,0755, 0,0500, 0,0467 e 0,0457 com 10, 50, 200 e 400 árvores; com árvores de profundidade 3, 0,1226, 0,0965, 0,0867 e 0,0886. Com 4.000 árvores rasas, a floresta rasa alcançaria a completa? Por quê?

CONFIRA NO SIMULADOR

A floresta votando

Acrescente árvores: o painel das árvores individuais não melhora, enquanto a curva de erro do voto cai e depois achata.

5 O júri

●●● DESAFIO

Cada jurado acerta, sozinho e de forma independente dos outros, com probabilidade $p = 0{,}7$. O júri decide por maioria.

(a) Calcule a chance de a maioria acertar com 3 jurados e com 5 jurados.

Dica: com 3: os três acertam ou exatamente dois acertam; o 2.º caso tem 3 arranjos, um para cada jurado que erra.

(b) Agora, dos 5 jurados, 3 votam sempre como manda o mesmo jornal (que acerta 70% das vezes) e só 2 decidem sozinhos. Qual é a chance de a maioria acertar?

(c) Que parte de uma random forest faz o papel do "jornal"? Que dois mecanismos dela impedem que as árvores o sigam todas?

6 Quem a floresta escuta?

●●● DESAFIO

Uma floresta de 300 árvores prevê maus pagadores de empréstimos. Foram acrescentadas duas colunas de puro ruído: `random_id`, um número de série com 1.000 valores distintos, e `coin`, um cara-ou-coroa. A importância por impureza (MDI, do inglês *mean decrease in impurity*) vem do treino; a por permutação é a queda de ROC-AUC (área sob a curva ROC) ao embaralhar a coluna em 300 empréstimos separados.

coluna	valores distintos	MDI (posição)	permutação (posição)
`credit_amount`	921	0,129 (1.ª)	0,007 (8.ª)
`random_id`	1.000	0,099 (2.ª)	$0{,}002 \pm 0{,}007$ (14.ª)
`duration`	33	0,087 (3.ª)	0,056 (2.ª)
`checking_status`	4	0,086 (5.ª)	0,144 (1.ª)
`coin`	2	0,020 (17.ª)	$-0{,}002$ (22.ª)

(a) Por que um número de série sem sinal nenhum fica em 2.º lugar pela MDI? Em qual das duas colunas de importância você confia?

(b) Uma cópia exata de `checking_status` é acrescentada à tabela. A importância por permutação de cada cópia cai de 0,144 para cerca de 0,05. Por quê? Que erro isso pode induzir?

ATIVIDADE DA AULA

M3

Gradient boosting

insper.github.io/ML/pt/2026.2/classes/gradient-boosting/

respostas comentadas na página 161

A pergunta da aula. Numa senoide com ruído, uma floresta de 200 tocos (árvores de um corte só) quase não melhora um toco: erro quadrático médio de teste 0,572 contra 0,608. Os mesmos 200 tocos somados pelo boosting chegam a 0,280, perto do piso do ruído, 0,25. O que muda quando cada árvore pequena é ajustada ao que as anteriores erraram e como impedir que essa soma exagere?

ETAPAS DO FLUXO NESTA AULA modelo avaliação

O que você precisa saber

- **A média corta variância; a soma corta viés.** A floresta tira a média de árvores fundas e instáveis; o boosting soma árvores rasas e estáveis, uma depois da outra.
- O boosting **não sorteia linhas**: toda árvore vê todos os pontos; o que muda de uma árvore para a outra é o **alvo** (o resíduo), não a amostra. O AdaBoost muda os **pesos** dos pontos errados; XGBoost, LightGBM e `HistGradientBoosting` são o mesmo gradient boosting com histogramas, passo de Newton e penalidades.
- O laço, na perda quadrática: comece com a média, F_0; calcule os **resíduos** $r_i = y_i - F_{m-1}(x_i)$; ajuste uma árvore pequena h_m aos resíduos (não a y); some-a encolhida pela **taxa de aprendizado** η: $F_m = F_{m-1} + \eta\, h_m$.
- Na perda $L = \frac{1}{2}(y - F)^2$, o resíduo $y - F = -\partial L/\partial F$ é o **gradiente negativo**: o boosting é um gradiente descendente sobre as previsões.
- **Mais árvores podem piorar o modelo** (numa floresta, nunca). Um η menor precisa de mais árvores (metade de η, cerca do dobro), chega mais longe e sobe mais devagar depois do mínimo.
- A **parada antecipada** observa a perda numa fatia separada do treino e para quando ela deixa de melhorar: o número de árvores vira uma saída. Na classificação, o boosting soma **log-odds** e $p = 1/(1 + e^{-F})$.

1 Média ou soma?

• AQUECIMENTO

Cinco modelos treinados nos mesmos 200 pontos de uma senoide com ruído de variância 0,25, avaliados em 5.000 pontos novos pelo erro quadrático médio (MSE, do inglês *mean squared error*):

modelo	MSE de treino	MSE de teste
um toco	0,552	0,608
floresta de 200 tocos	0,511	0,572
boosting de 200 tocos ($\eta = 0{,}1$)	0,215	0,280
uma árvore profunda	0,000	0,509
floresta de 200 árvores profundas	0,054	0,369

(a) Por que a floresta de tocos quase não melhora o toco, enquanto a floresta de árvores profundas melhora muito a árvore profunda? Use as palavras viés e variância.

(b) Por que nenhum dos cinco pode ter MSE de teste abaixo de 0,25?

2 Boosting à mão

●● PRÁTICA

Cinco pontos, perda quadrática, tocos e $\eta = 0{,}5$:

x	1	2	3	4	5
y	1	2	3	6	13

(a) Calcule F_0 e os resíduos da rodada 1.

(b) Complete a tabela: cada folha do toco prevê o resíduo médio dos seus pontos e SSE é a soma dos quadrados dos desvios em torno dessas médias. Qual corte vence? Quanto vale F_1 em cada ponto?

corte	**média dos resíduos à esq.**	**média dos resíduos à dir.**	**SSE**
$x < 1{,}5$			
$x < 2{,}5$			
$x < 3{,}5$			
$x < 4{,}5$			

(c) Faça a rodada 2 testando só $x < 3{,}5$ e $x < 4{,}5$. Quanto vale F_2? Calcule o MSE de treino de F_0, F_1 e F_2.

(d) Com $\eta = 1$, quanto valeria F_1? O que esse passo inteiro faz com o último ponto e por que isso preocupa?

CONFIRA NO SIMULADOR

O resíduo encolhendo

Ponha $\nu = 1$ com 6 tocos e depois $\nu = 0{,}15$ com 40 (ν é a taxa de aprendizado): ajustes parecidos, um em passos inteiros, outro em passos cautelosos.

3 Freio e parada

●● PRÁTICA

Num banco que oferece um depósito por telefone, um booster foi treinado sem parada antecipada, com número fixo de árvores. A tabela dá a precisão média (AP, do inglês *average precision*, a área sob a curva precisão–recall) num fold de validação; 11,7% dos clientes assinam.

árvores	5	35	200	1.000
$\eta = 0{,}5$	0,418	0,382	0,338	0,336
$\eta = 0{,}05$	0,417	0,433	0,435	0,408

(a) Para cada η, qual número de árvores é o melhor? Mais árvores sempre ajudam? E numa random forest?

(b) Por que o pico de $\eta = 0{,}05$ é mais alto e vem mais tarde?

(c) A parada antecipada escolheria o número de árvores sozinha. Com que dados ela decide e por que não com o conjunto de teste?

4 Para quem ligar primeiro?

●● PRÁTICA

O banco quer um modelo que ordene, *antes* de cada campanha, os clientes pela chance de assinar. Com todas as colunas, o booster tem ROC-AUC (área sob a curva ROC) de 0,934 e AP de 0,624; sem a coluna `duration`, 0,797 e 0,447. A coluna `duration` é a duração da última ligação, em segundos.

(a) O modelo de 0,934 pode ser usado? Por quê?

(b) Para cada coluna, diga se o valor existe no momento da previsão: idade; saldo em conta; resultado da campanha anterior; duração da ligação; número de ligações feitas na campanha anterior.

(c) Uma validação cruzada mais cuidadosa teria detectado o problema de (a)? Por quê?

5 Uma rodada de classificação

DESAFIO

Cinco pontos com rótulo 0 ou 1, um toco e $\eta = 0{,}5$:

x	1	2	3	4	5
y	0	0	1	1	1

O boosting soma log-odds F e lê $p = 1/(1+e^{-F})$. Ele começa no log-odds da taxa base, $F_0 = \ln \frac{p_0}{1-p_0}$, e os pseudo-resíduos são $y - p$. Cada folha recebe um passo de Newton: $w = \sum(y_i - p_i) / \sum p_i(1-p_i)$, somas sobre os pontos da folha.

(a) Calcule p_0, F_0 e os pseudo-resíduos. Qual corte o toco escolhe?

(b) Calcule os valores das duas folhas, F_1 e as probabilidades p_1.

(c) O XGBoost usa $w = \sum(y_i - p_i) / \left(\sum p_i(1-p_i) + \lambda\right)$. Recalcule as folhas com $\lambda = 1$. O que a penalidade faz?

Dica: $\ln 1{,}5 \approx 0{,}405$; $e^{0{,}845} \approx 2{,}33$; $e^{-1{,}239} \approx 0{,}290$.

6 Floresta ou boosting? ••• DESAFIO

Preços de casas na Califórnia, mesmos 5 folds. RMSE (raiz do erro quadrático médio, do inglês *root mean squared error*) em milhares de US$:

fold	1	2	3	4	5	**média**
random forest, padrões	49,7	48,6	48,7	49,2	49,3	49,1
boosting, padrões	47,4	47,3	47,4	47,8	47,9	47,5

Além disso, 16 configurações aleatórias de cada família tiveram R^2 de validação entre 0,762 e 0,826 (floresta) e entre 0,650 e 0,844 (boosting). Num núcleo de CPU, um ajuste levou 11,5 s (floresta) e 0,5 s (boosting).

(a) Calcule a diferença fold a fold (floresta − boosting), a média e o desvio-padrão. A vantagem é real?

(b) O que a faixa das 16 configurações diz sobre o risco de ajustar mal cada família?

(c) Recomende uma das duas para (i) um baseline forte numa tarde, sem tempo para ajuste; (ii) a última fração de erro num conjunto grande, com tempo para busca e parada antecipada.

ATIVIDADE DA AULA

M4

Seleção de modelos

insper.github.io/ML/pt/2026.2/classes/model-selection/

respostas comentadas na página 162

A pergunta da aula. No lab de preços de casas, a regressão linear erra US$ 66.369 nos distritos de treino e US$ 66.460 nos de validação; a árvore sem limites erra 0 e US$ 67.266. Alguém oferece 50.000 distritos a mais. Para qual dos dois modelos isso ajuda e o que fazer com o outro?

ETAPAS DO FLUXO NESTA AULA modelo avaliação

O que você precisa saber

- **Erro de teste esperado = viés^2 + variância + ruído σ^2.** Viés: quanto a previsão *média* (sobre muitos conjuntos de treino) erra a verdade. Variância: quanto as previsões se espalham em torno dessa média. Ruído: o piso que nenhum modelo baixa.
- **Underfitting (viés alto):** treino ≈ validação, os dois altos. **Overfitting (variância alta):** treino muito melhor que a validação; a distância entre os dois é o sintoma.
- **Curva de validação:** dados fixos, complexidade variando; o fundo do U é a escolha. **Curva de aprendizado:** modelo fixo, linhas variando. Num modelo de capacidade fixa, **mais dados curam a variância, não o viés**. A **busca em grade** faz a mesma conta da curva de validação, mas escolhe em vez de desenhar; só a decomposição viés–variância separa os dois termos, e ela exige conhecer a verdade.
- **Todo modelo tem um botão de complexidade.** Giram ao contrário: `alpha`, `n_neighbors` e `min_samples_leaf` maiores deixam o modelo *mais simples*.
- **Regressão à média:** a nota do vencedor de uma busca é em parte sorte e cai quando medida de novo. Nenhuma família vence em todo problema: **não existe almoço grátis**. Três sentidos de *viés*: o do viés–variância (erro sistemático), o **indutivo** (a aposta de cada família) e o **de seleção** (a nota otimista do vencedor).

1 Quatro modelos, duas notas cada

● AQUECIMENTO

Raiz do erro quadrático médio (RMSE, do inglês *root mean squared error*) de quatro modelos de preço de casas, em US$, na média de 5 folds:

modelo	treino	validação	distância (validação − treino)
baseline (prevê a média)	114.784	114.793	
regressão linear	66.369	66.460	
k-vizinhos (k-NN), $k = 5$	48.898	59.967	
árvore sem limites	0	67.266	

(a) Complete a última coluna. Qual modelo tem viés alto? Qual tem variância alta?

(b) Ordene os três últimos pela validação. O que essa ordem, sozinha, não diz?

2 Cinco conjuntos de treino, um ponto

• AQUECIMENTO

Num problema inventado, conhecemos a verdade num ponto x_0: $f(x_0) = 2$. O ruído tem desvio-padrão 0,2. Dois modelos foram treinados em cinco conjuntos de treino independentes; eis o que cada versão prevê em x_0:

conjunto de treino	1	2	3	4	5	**média**
modelo A	1,2	1,4	1,3	1,5	1,1	
modelo B	2,6	1,2	2,3	1,5	2,4	

(a) Para cada modelo, calcule o viés^2 (média − verdade, ao quadrado) e a variância (média dos quadrados dos desvios em torno da própria média).

(b) Some o ruído σ^2 e compare os erros esperados. Qual erro nenhum dos dois consegue baixar?

(c) Qual dos dois está em underfitting? Qual mudaria mais de resposta se você recebesse outro conjunto de treino?

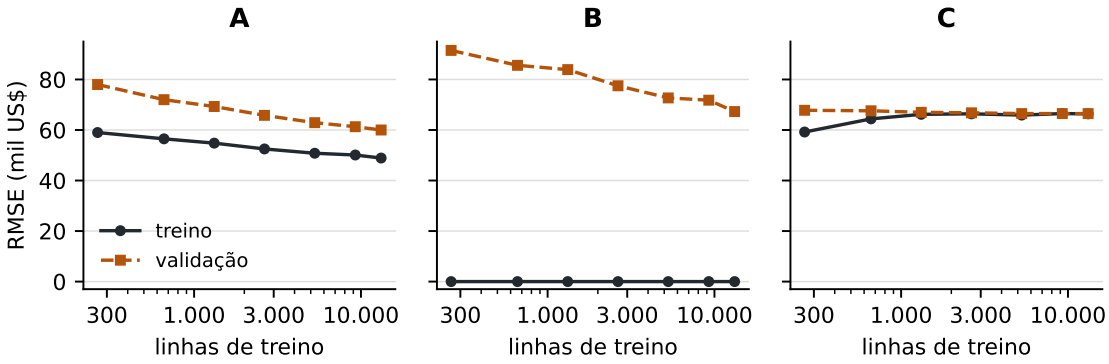
CONFIRA NO SIMULADOR

Viés e variância

Deslize do grau 0 ao 12: veja o erro de treino só descer, o de teste desenhar um U e os dois se separarem do lado da variância.

3 Mais dados valem a pena? ●● PRÁTICA

As três curvas de aprendizado abaixo (RMSE em milhares de US$, nos mesmos folds) são de uma regressão linear, de um k-NN com $k = 5$ e de uma árvore sem limites, não necessariamente nessa ordem.



(a) Qual painel é de qual modelo? Diga o que denuncia cada um.

(b) Para quais dos três 50.000 linhas a mais ajudariam? Para qual seriam dinheiro jogado fora?

(c) Para cada modelo, diga o que tentar em vez de (ou além de) mais dados.

4 Para que lado girar o botão

●● PRÁTICA

Cada linha é um modelo já medido. A coluna "melhor" é a melhor validação que a mesma família conseguiu com outro valor do mesmo botão (RMSE em US$; na última linha, o erro quadrático médio, MSE, do inglês *mean squared error*, num seno com ruído).

modelo e botão	treino	valid.	melhor	diagnóstico	o botão…
Ridge grau 3, `alpha` $= 0{,}01$	51.931	66.098	57.473		
Ridge grau 3, `alpha` $= 10^6$	106.790	106.832	57.473		
k-NN, `n_neighbors` $= 1$	0	73.873	59.182		
árvore, `min_samples_leaf` $= 100$	61.413	63.505	57.796		
polinômio, `degree` $= 1$ (MSE)	0,267	0,319	0,114		

Complete as duas colunas da direita: viés alto ou variância alta e se o botão deve aumentar ou diminuir. Quais linhas giram "ao contrário" da intuição de que número maior é modelo mais complexo?

5 O primeiro da turma

● ● ● DESAFIO

Uma busca em grade testou 20 configurações nos mesmos folds. As notas de validação cruzada (CV, do inglês *cross-validation*) têm média 0,80. Cada nota carrega um ruído dos folds de desvio-padrão 0,02; as acurácias *verdadeiras* das configurações diferem pouco entre si, com desvio-padrão 0,01. A vencedora tirou 0,84.

(a) A confiabilidade é $r = \sigma^2_{\text{verdade}} / (\sigma^2_{\text{verdade}} + \sigma^2_{\text{ruído}})$. Um grupo que ficou D acima da média é esperado $r \cdot D$ acima dela numa medida nova. Que nota você espera da vencedora em dados novos?

(b) E se o ruído dos folds tivesse desvio-padrão 0,005?

(c) Que número vai para o relatório e onde ele deve ser medido?

(d) Um colega viu a máquina de vetores de suporte (SVM, do inglês *support vector machine*) vencer nos dados de radar (0,946 contra 0,897 da melhor árvore) e decidiu usar SVM em todo projeto. Outro colega prefere sempre um ensemble de árvores, que nos mesmos dados empatou com ela (0,943 e 0,940). Responda aos dois com esta tabela (acurácia de CV da melhor configuração de cada família):

dados	logística	k-NN	SVM	árvore	floresta	boosting
fronteira linear, 40 dim., 200 linhas	0,820	0,710	0,805	0,725	0,735	0,720
duas luas (curva em 2 dim.)	0,853	0,900	0,895	0,887	0,893	0,895
limiares em 2 de 20 colunas	0,640	0,636	0,686	0,938	0,948	0,944

ATIVIDADE DA AULA

E1

Agrupamento hierárquico

insper.github.io/ML/pt/2026.2/classes/hierarchical-clustering/

respostas comentadas na página 163

A pergunta da aula. 101 animais descritos por 16 perguntas simples (tem pelos? põe ovos? quantas pernas?). Uma árvore construída só com essas respostas redescobre a classificação dos zoólogos? E em que altura ela deve ser cortada?

ETAPAS DO FLUXO NESTA AULA modelo avaliação

O que você precisa saber

- **Aglomerativo:** comece com um grupo por ponto, funda os dois grupos mais próximos, registre a **altura da fusão** e repita até sobrar um. São $n-1$ fusões, sem sorteio.
- A **ligação** define a distância entre grupos: **simples** (o par mais próximo; segue cadeias e sofre com pontes), **completa** (o par mais distante), **média** (a média de todos os pares) e **Ward**, o aumento da soma de quadrados, $\Delta = \frac{|P||Q|}{|P|+|Q|}\|\mu_P - \mu_Q\|^2$, desenhado na altura $\sqrt{2\Delta}$.
- No **dendrograma**, o eixo vertical é distância. A altura em que dois pontos se juntam é a **distância cofenética**; folhas vizinhas no desenho podem ser muito diferentes.
- **Cortar** numa altura dá os grupos; cortes da mesma árvore são **aninhados**. Um k que vale numa faixa larga de alturas (um **salto** grande entre fusões) é sustentado pelos dados.
- A **correlação cofenética** mede quão bem as alturas preservam as distâncias originais, não quão bons são os grupos.
- Qualquer distância serve (exceto em Ward, que exige a euclidiana): **correlação** agrupa perfis pelo formato, **cosseno** é o padrão para texto. A memória cresce com n^2.

1 Distância entre grupos

• AQUECIMENTO

Cinco pontos sobre uma reta: A $= 0$, B $= 2$, C $= 5$, D $= 11$, E $= 18$.

(a) Complete a matriz de distâncias (metade de cima basta).

(b) A primeira fusão junta A e B, na altura 2, qualquer que seja a ligação. Por quê? .....

.....

(c) Complete a distância do grupo $\{A, B\}$ aos outros pontos em cada ligação.

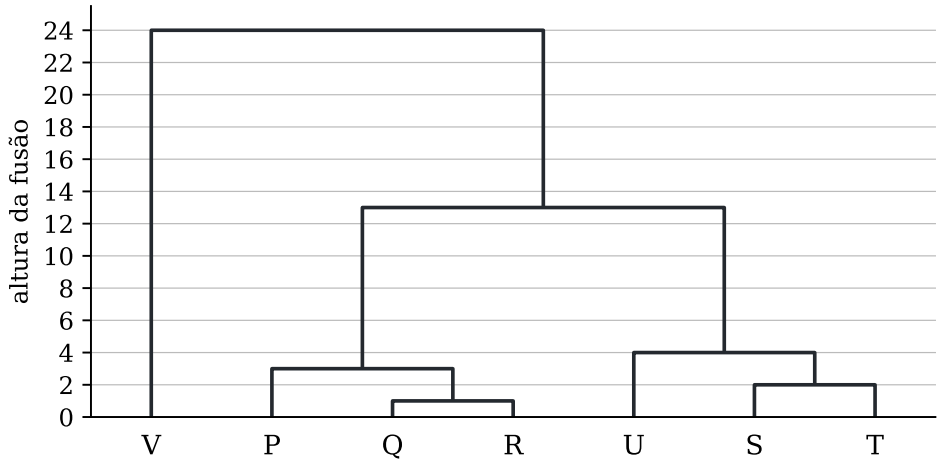
	A	B	C	D	E
A	0				
B		0			
C			0		
D				0	
E					0

de $\{A, B\}$ até	simples	completa	média
C			
D			
E			

2 Ler o dendrograma

• AQUECIMENTO

Dendrograma (ligação completa) de sete clientes, P a V:



(a) Em que altura P e U se juntam? E Q e R? Que nome tem essa altura?

(b) No desenho, V fica ao lado de P; R fica ao lado de U. Esses pares são parecidos? Quantos desenhos diferentes representam esta mesma árvore?

(c) Corte a árvore na altura 8. Quantos grupos saem e quais são?

(d) Para $k = 2, 3$ e 4, em que faixa de alturas o corte dá esse k? Que valores de k a árvore sustenta?

3 Uma matriz, três árvores

●● PRÁTICA

Continue com os pontos do exercício 1.

(a) Termine as árvores das ligações simples, completa e média: para cada fusão, quem se funde e em que altura.

(b) Corte cada árvore em dois grupos. Quais ligações concordam?

(c) Na árvore simples, as alturas são 2, 3, 6 e 7. Na completa, 2, 5, 7 e 18. Qual das duas mostra uma divisão em dois grupos sustentada pelos dados? Por quê?

(d) Ward: na segunda fusão, calcule o custo Δ de juntar $\{A, B\}$ com C, C com D e D com E. Qual fusão acontece e em que altura ($\sqrt{2\Delta}$)?

	simples		**completa**		**média**	
fusão	funde	altura	funde	altura	funde	altura
1	A + B	2	A + B	2	A + B	2
2						
3						
4						

CONFIRA NO SIMULADOR

Construa e corte uma árvore

Aperte *fundir o próximo* nas três bolhas e veja cada fusão virar uma barra; depois troque a ligação nas luas e na cadeia e procure o maior salto antes de cortar.

4 Onde cortar

●● PRÁTICA

400 pontos em quatro bolhas, que vêm em dois pares. As fusões mais altas da árvore de Ward, do topo para baixo, estão nas alturas 159,2; 32,3; 30,3; 9,7; 9,2; 8,9; 8,7.

k	2	3	4	5	6	7
faixa de alturas	32,3 a 159,2					
salto	126,9					
silhueta do corte	0,787	0,664	0,545	0,473	0,393	0,349

(a) Complete a tabela. Que dois valores de k os saltos destacam? O que isso diz sobre os dados?

(b) A silhueta concorda com os saltos? O que ela não enxerga?

(c) Por que dá para relatar $k = 2$ e $k = 4$ ao mesmo tempo sem contradição? Duas rodadas do k-means com $k = 2$ e $k = 4$ dariam a mesma garantia?

5 Quão fiel é a árvore

●● PRÁTICA

Quatro pontos sobre uma reta: A = 0, B = 1, C = 4, D = 6. A ligação simples funde AB em 1, CD em 2 e os dois pares em 3.

(a) Complete a tabela com as distâncias originais d e as cofenéticas c dos seis pares.

(b) Calcule a correlação cofenética, $r = \dfrac{\sum(d-\bar{d})(c-\bar{c})}{\sqrt{\sum(d-\bar{d})^2\sum(c-\bar{c})^2}}$.

(c) Em quatro bolhas, as quatro ligações tiveram correlação cofenética entre 0,944 e 0,953, mas o ARI (índice de Rand ajustado) do corte em 4 grupos foi de 0,496 (simples) a 0,954 (completa). O que a correlação cofenética mede, então? Quando ela é o critério certo?

par	AB	AC	AD	BC	BD	CD
original d						
cofenética c						

6 Que distância?

DESAFIO

Gasto mensal de quatro clientes em três meses:

	mês 1	mês 2	mês 3
A	1	2	3
B	5	10	15
C	3	2	1
D	15	10	5

(a) Calcule as distâncias euclidianas A–B, A–C e B–D. Cortada em dois grupos, que partição uma ligação média daria com essa distância?

(b) Sem conta longa, qual é a correlação de Pearson r entre A e B? E entre A e C? Com a distância de correlação $1 - r$, que dois grupos saem?

(c) Qual das duas partições serve a uma equipe que quer saber "quem está gastando cada vez mais"? E a quem quer separar clientes grandes de pequenos?

(d) Por que o scikit-learn recusa `linkage='ward'` com a distância de correlação ou a do cosseno?

(e) Quanta memória ocupam as distâncias entre todos os pares de 50.000 clientes, a 8 bytes por número?

ATIVIDADE DA AULA

E2

Representação de texto

insper.github.io/ML/pt/2026.2/classes/text-representation/

respostas comentadas na página 163

A pergunta da aula. Na busca da central de ajuda, "erro 0x80004005" acha a resposta certa na hora, enquanto "perdi a fatura" não acha nada, embora exista uma pergunta sobre a segunda via do boleto. Como um texto vira um vetor? E quando contar palavras deixa de bastar?

ETAPAS DO FLUXO NESTA AULA dados e divisão modelo

O que você precisa saber

- **Tokenização:** o analisador padrão do scikit-learn põe tudo em minúsculas e guarda só sequências de duas ou mais letras ou dígitos (some "o", "a", "é"). **Stop words**, **n-gramas**, **stemming** (corta sufixos: "venc") e **lematização** (forma de dicionário: "vencer") mudam as colunas.
- **Bag-of-words:** uma linha por documento, uma coluna por palavra do vocabulário, contagens na célula. O `fit` aprende o vocabulário; palavra nova no `transform` é descartada. Quase tudo é zero.
- **TF-IDF** (frequência do termo × frequência inversa nos documentos), no scikit-learn: tf × idf, com $\text{idf}(t) = \ln \frac{1+n}{1+\text{df}(t)} + 1$, em que n é o número de documentos e $\text{df}(t)$ quantos contêm t; depois cada linha é dividida pelo seu comprimento. Palavra presente em todo documento pesa menos.
- **Cosseno** $= \frac{u \cdot v}{\|u\|\|v\|}$ compara direções e ignora o tamanho do texto; com linhas de comprimento 1, é o produto escalar.
- **Esparso** não vê sinônimos (cosseno 0), ordem nem sentido, mas acerta tokens raros e exatos. **Word embeddings** (hipótese distribucional: palavras com as mesmas vizinhas ficam perto) dão um vetor por palavra; **sentence embeddings** dão um vetor por frase, calculado com o contexto.
- Embeddings vencem com **poucos rótulos**, paráfrases e outras línguas; os cossenos deles ocupam uma escala comprimida (vale a ordem, não o valor). Na dúvida, a **busca híbrida** combina os dois.

1 De uma frase a tokens

• AQUECIMENTO

A frase é: "O time não venceu: o time empatou de novo."

(a) Que tokens o analisador padrão do scikit-learn produz? Quantas colunas tem o vocabulário?

(b) Com a lista de stop words {o, a, os, as, do, da, de, no, na, e, um, uma}, quantas colunas sobram? A negação sobrevive?

(c) Só com bigramas (`ngram_range=(2, 2)`), quantas colunas? Qual bigrama junta palavras que não eram vizinhas no texto?

(d) Uma ferramenta leva "venceu" a "venc"; outra, a "vencer". Qual faz stemming e qual lematiza?

2 O saco de palavras

• AQUECIMENTO

Quatro avaliações curtas: d1 “O filme é bom.”; d2 “O filme é muito bom, muito bom.”; d3 “O livro é ruim.”; d4 “O filme do livro é ruim.”

	bom	do	filme	livro	muito	ruim
d1						
d2						
d3						
d4						

(a) Complete a matriz de contagens (o vocabulário já está nas colunas). Por que “é” e “o” não têm coluna?

(b) Calcule o cosseno entre d1 e d2, entre d1 e d4 e entre d1 e d3.

(c) “O filme é bom. O filme é bom.” tem que cosseno com d1? E que distância euclidiana? Qual das duas medidas serve para texto?

(d) Que vetor recebe “O filme é ótimo.”? O que aconteceu com “ótimo”?

3 Pesar pela raridade

●● PRÁTICA

Continue com as quatro avaliações ($n = 4$). Use $\ln 2{,}5 = 0{,}9163$, $\ln(5/3) = 0{,}5108$ e $\ln 1{,}25 = 0{,}2231$.

(a) Calcule o idf de cada palavra. Qual pesa menos? Por quê?

(b) Calcule a linha TF-IDF de d1 (tf $\times$ idf, depois divida pelo comprimento).

(c) A linha TF-IDF de d4 é: do 0,614; filme 0,392; livro 0,484; ruim 0,484. Calcule o cosseno TF-IDF entre d1 e d4 e compare com o das contagens. Por que caiu?

(d) A consulta "filme ruim" vira (filme 0,629; ruim 0,777). A linha de d3 é (livro 0,707; ruim 0,707). Ordene d1, d3 e d4 pelo cosseno com a consulta.

CONFIRA NO SIMULADOR

TF-IDF e busca ao vivo

Digite as quatro avaliações e a consulta "filme ruim": confira o idf, a linha de d1 e a ordenação que você calculou.

4 Esparso ou denso? ●● PRÁTICA

Na aula, os mesmos pares de frases foram comparados pelo cosseno do TF-IDF e pelo de um modelo multilíngue de sentence embeddings:

frase A	frase B	TF-IDF	embedding
o carro é rápido	o automóvel é veloz	0,00	0,97
a manga da camisa rasgou	o punho da blusa descosturou	0,21	0,87
a manga da camisa rasgou	comi uma manga madura	0,21	0,35
o cachorro mordeu o homem	o homem mordeu o cachorro	1,00	0,98
a taxa de juros subiu	o time venceu o jogo	0,00	0,14

Na busca por "erro 0x80070005" entre duas entradas que só diferem no código, o TF-IDF deu 0,55 à certa e 0,18 à outra; o embedding, 0,80 e 0,77. Com 5 posts rotulados por classe, um classificador acertou 55,7% sobre TF-IDF e 81,9% sobre embeddings; com todos os 2.097 posts, 90,4% e 90,1%.

(a) Que limite do TF-IDF cada uma das quatro primeiras linhas mostra? Algum deles o embedding também tem?

(b) Por que frases sem relação dão 0,14 em vez de 0? O que isso diz sobre usar um limiar fixo de cosseno?

(c) Escolha esparso ou denso e justifique: (1) achar o chamado que cita a nota fiscal 000.482.117; (2) agrupar 30.000 comentários curtos de um aplicativo; (3) classificar e-mails em 4 filas com 15 exemplos rotulados; (4) buscar artigos em inglês com perguntas em português.

5 Conhecer uma palavra pelas vizinhas

●● PRÁTICA

Num corpus de onze frases (animais que comem e bebem; veículos que andam com combustível), contou-se, com janela de duas palavras, quem aparece perto de quem: $N = 126$ pares no total. A linha de "carro" é {o: 2, usa: 1, gasolina: 1, corre: 1, na: 1}, que soma 6; como contexto, "o" aparece 22 vezes e "gasolina", 4.

(a) Calcule a informação mútua pontual, $\mathrm{PMI}(w, c) = \log_2 \dfrac{n(w,c)\,N}{n(w)\,n(c)}$, em que $n(w,c)$ é a contagem do par, $n(w)$ o total da linha de w e $n(c)$ o total de c como contexto, para (carro, o) e (carro, gasolina). Qual contexto diz mais sobre "carro"? (Use $\log_2 1{,}91 \approx 0{,}93$; $\log_2 2{,}63 \approx 1{,}40$; $\log_2 5{,}25 \approx 2{,}39$; $\log_2 10{,}5 \approx 3{,}39$.)

(b) Com contagens brutas, gato e carro tiveram cosseno 0,50; depois da PPMI (a PMI positiva: os valores negativos trocados por 0), 0,05; depois da PPMI e de uma SVD (decomposição em valores singulares) para 2 números, gato e cachorro tiveram 0,99. Explique cada passo em uma frase.

(c) Nos vetores GloVe, "hot"/"cold" (quente/frio) têm cosseno 0,80, o mesmo de "doctor"/"physician" (médico). Por que antônimos ficam tão perto? Para que tarefa isso é uma armadilha?

(d) "bank" tem um único vetor GloVe, perto de "money" (0,64) e longe de "river" (0,41). Que limite isso mostra e que representação o resolve?

6 Uma busca de brinquedo

DESAFIO

Uma central de ajuda tem três perguntas. Imagine que um modelo de sentence embeddings as levasse a vetores de comprimento 1 com só 3 números (vetores de brinquedo; os reais têm 384):

	pergunta	vetor
F1	Como trocar a senha da conta?	(0,8; 0,6; 0)
F2	Qual é o prazo de entrega?	(0; 0,6; 0,8)
F3	Como cancelar uma compra?	(0,6; 0; 0,8)

As consultas são “esqueci minha senha”, com vetor (0,6; 0,8; 0), e “desisti do pedido”, com vetor (0,48; 0,36; 0,8). No TF-IDF ajustado nas três perguntas, “esqueci minha senha” tem cosseno 0,467 com F1 e 0 com as outras.

(a) Calcule o cosseno de cada consulta com F1, F2 e F3 (os vetores já têm comprimento 1). Quem vem em primeiro?

(b) Qual é o cosseno TF-IDF de “desisti do pedido” com cada pergunta? Por quê?

(c) Combine as notas da primeira consulta numa busca híbrida, $0{,}5 \cdot \cos_{\text{TF-IDF}} + 0{,}5 \cdot \cos_{\text{embedding}}$. A ordem muda? Para que tipo de consulta a parte TF-IDF faria diferença de verdade?

(d) Na segunda consulta, a primeira e a segunda notas estão perto. O que o sistema deveria fazer quando a melhor nota é baixa ou mal separada?

(e) Se cada pergunta fosse representada pela média dos vetores das suas palavras, “o cachorro mordeu o homem” e “o homem mordeu o cachorro” teriam que cosseno? Por quê?

CONFIRA NO SIMULADOR

As quatro representações lado a lado

Escolha o par de paráfrases e depois o da ordem trocada: veja as faixas esparsas quase vazias, as densas cheias e quais cossenos mudam.

ATIVIDADE DA AULA

E3

Modelagem de tópicos e BERTopic

insper.github.io/ML/pt/2026.2/classes/topic-modeling-bertopic/

respostas comentadas na página 164

A pergunta da aula. 2.155 posts de quatro fóruns, sem rótulo nenhum. Do que eles tratam? A LDA (alocação latente de Dirichlet), a NMF (fatoração de matrizes não negativas) e o BERTopic devolvem três respostas diferentes: em qual acreditar? E quantos tópicos existem, afinal?

ETAPAS DO FLUXO NESTA AULA modelo avaliação

O que você precisa saber

- Duas ideias de tópico: uma **distribuição sobre palavras**, com cada documento uma mistura de tópicos (LDA, NMF); ou um **grupo de documentos**, nomeado depois pelas suas palavras (BERTopic).
- **LDA:** para cada palavra, sorteia-se um tópico da mistura θ_d do documento e depois a palavra da distribuição φ_k do tópico, então $p(w \mid d) = \sum_k \theta_{d,k}\, \varphi_{k,w}$. O α da Dirichlet controla quão misturado é um documento típico. Precisa de muitas palavras e depende da semente.
- **NMF:** $X \approx WH$ com $W, H \geq 0$; cada linha de H é um tópico, cada linha de W diz quanto de cada tópico um documento tem. Sem pesos negativos, os tópicos são partes que se somam (a SVD, decomposição em valores singulares, dá "esportes menos espaço").
- O número de tópicos é uma escolha de granularidade: a perplexidade não acompanha o que as pessoas entendem; a **coerência** (média da NPMI, a informação mútua pontual normalizada, entre as palavras do topo) e a **diversidade** (fração de palavras distintas no topo) ajudam, mas é preciso ler os tópicos.
- **BERTopic:** sentence embeddings → **UMAP** (*uniform manifold approximation and projection*; reduz a 5 dimensões) → **HDBSCAN** (agrupamento hierárquico por densidade; outliers no tópico -1) → **c-TF-IDF** (o TF-IDF, frequência do termo vezes inverso da frequência nos documentos, calculado por classe) $= \frac{n_{t,c}}{\sum_{t'} n_{t',c}} \times \ln(1 + A/f_t)$, com f_t a contagem de t em todos os grupos e A a média de palavras por grupo.
- **Armadilhas:** stop words que dão nome aos tópicos; outliers demais (reatribuí-los é uma troca); um `min_cluster_size` que muda o número de tópicos em degrau; sementes que mudam até o número de tópicos.

1 A história da LDA

• AQUECIMENTO

Dois tópicos sobre seis palavras e um documento que é 75% culinária e 25% espaço:

	pão	**forno**	**massa**	**foguete**	**órbita**	**lua**
φ culinária	0,50	0,30	0,10	0,05	0	0,05
φ espaço	0	0,05	0,05	0,40	0,30	0,20
$p(w \mid d)$						

(a) Complete a última linha. Confira que ela soma 1.

(b) Num documento de 20 palavras escrito por essa história, quantas, em média, vêm do tópico espaço? Por que a ordem das palavras não importa para a LDA?

(c) Avaliações curtas costumam falar de um assunto só. Você usaria $\alpha = 0{,}1$ ou $\alpha = 10$? (Na aula, a maior fatia de um documento ficou, em média, com 0,89 e 0,42.)

(d) Nas onze frases escritas à mão da aula (53 palavras), a LDA separou os três temas em só 1 de 20 sementes. Por quê?

CONFIRA NO SIMULADOR

A história da LDA ao vivo

Arraste o ponto branco (a mistura θ) pelo triângulo e escreva documentos: as palavras mudam de cor com o tópico. Depois ponha α em 0,1 e em 10 e compare a nuvem de misturas.

2 Partes que se somam

●● PRÁTICA

Uma NMF com dois tópicos sobre as palavras (gol, time, jogo, foguete, lua) deu $H = \begin{pmatrix} 2 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 2 & 1 \end{pmatrix}$ e, para quatro documentos, as linhas de W: d1 $(1;\ 0)$, d2 $(0;\ 1)$, d3 $(0{,}5;\ 0{,}5)$, d4 $(0{,}2;\ 0{,}6)$.

(a) Que tema é cada tópico? Descreva cada documento em palavras.

(b) Calcule a reconstrução WH das linhas de d3 e d4.

(c) Uma componente de uma SVD truncada nos mesmos dados tem pesos (gol 0,5; time 0,4; foguete $-0{,}5$; lua $-0{,}4$). Como ela se lê? Por que a NMF não pode produzir algo assim?

(d) Na aula, nos posts, a NMF concordou mais com os fóruns que a LDA (ARI, o índice de Rand ajustado, de 0,767 contra 0,456), mas no lab, com outros cinco fóruns, perdeu (0,333 contra 0,603), porque gastou um tópico inteiro com a assinatura repetida de um usuário. O que isso diz sobre "qual modelo é melhor"?

3 Os tópicos são bons? ●● PRÁTICA

Num corpus de 2.000 documentos, “foguete” aparece em 200, “lançamento” em 100 e os dois juntos em 80.

(a) Calcule a NPMI, $\ln\big[P(a,b)/\big(P(a)\,P(b)\big)\big] \;/\; \big(-\ln P(a,b)\big)$, em que P é a fração dos documentos que contêm as palavras. (Use $\ln 8 = 2{,}079$ e $\ln 0{,}04 = -3{,}219$.)

(b) Quanto vale a NPMI se os dois aparecem juntos em 10 documentos? E em nenhum?

(c) Três tópicos com as quatro palavras do topo: {deus, jesus, igreja, gente}, {time, jogo, hóquei, gente}, {espaço, nasa, lançamento, gente}. Calcule a diversidade (palavras distintas ÷ total). O que a palavra repetida revela?

(d) A perplexidade da LDA em posts separados subiu sem parar de $k = 2$ a $k = 20$; a coerência teve pico em $k = 4$, com uma variação entre sementes quase do tamanho da curva. Como você escolheria k?

4 Dar nome a um grupo

●● PRÁTICA

O HDBSCAN formou três grupos de documentos. Juntando cada grupo num pseudodocumento, as contagens são:

	esportes	espaço	culinária
palavras no grupo	20	10	15
"time"	4	0	0
"lua"	0	2	0
"forno"	0	0	3
"que"	6	3	5

(a) Calcule A (a média de palavras por grupo, truncada para inteiro) e o c-TF-IDF de "time", "lua" e "forno" no seu grupo. (Use $\ln 4{,}75 = 1{,}558$; $\ln 8{,}5 = 2{,}140$; $\ln 6 = 1{,}792$; $\ln(29/14) = 0{,}728$.)

(b) Calcule o c-TF-IDF de "que" nos três grupos. Ela ganha de alguma palavra do assunto? O que isso explica sobre nomes como `3_the_of_and_to`?

(c) Trocar a lista de stop words e chamar `update_topics` muda quais documentos estão em cada tópico? Por quê?

5 O BERTopic passo a passo

●● PRÁTICA

(a) Ponha os quatro passos do BERTopic em ordem e diga o que cada um recebe e devolve: HDBSCAN; c-TF-IDF; sentence embeddings; UMAP.

(b) Nas primeiras 20 palavras de cada post, a NMF sobre TF-IDF chegou a ARI 0,115 com os fóruns; um k-means sobre embeddings, a 0,582. "I want my money back" e "please process a refund" têm cosseno 0 no TF-IDF e 0,60 nos embeddings. Explique a ligação entre os dois resultados.

(c) Por que reduzir para 5 dimensões? Por que não 2 nem 384?

(d) Com os mesmos embeddings, `min_cluster_size` 5, 10, 15 e 25 deram 96, 24, 5 e 3 tópicos (28%, 20%, 0% e 0% de outliers); somar ao embedding um ruído de um milionésimo levou uma rodada de 24 para 3 tópicos. O que você responderia a quem pergunta "quantos tópicos existem"? Que configuração você evitaria?

CONFIRA NO SIMULADOR

O degrau do `min_cluster_size`

Mova `min_cluster_size` um valor por vez e veja o número de tópicos ficar parado e depois despencar de uma vez, com a fração de outliers saltando junto.

6 Que modelo recomendar

DESAFIO

Os mesmos 2.155 posts, 24 tópicos para os três modelos, semente 42:

	outliers	coerência	diversidade	homogeneidade	ARI semente 42 × 43
LDA	0%	0,265	0,72	0,526	0,164
NMF	0%	0,391	0,92	0,646	0,736
BERTopic	20%	0,251	0,88	0,927	0,324

A homogeneidade (1 quando cada tópico só tem posts de um fórum) do BERTopic foi medida sem os outliers; depois de reatribuí-los, caiu para 0,822 ou 0,882, conforme a estratégia. Com a semente 43, o BERTopic achou 16 tópicos em vez de 24.

(a) Que modelo "vence" em cada coluna? Por que a comparação de homogeneidade não é justa como está?

(b) Por que a NMF tira a melhor coerência, embora o BERTopic tenha os tópicos mais puros?

(c) Uma equipe quer os tópicos de 30.000 avaliações curtas (10 a 20 palavras), em português, de um aplicativo de música. Recomende um modelo e três cuidados de configuração. O que ela deve relatar junto com a lista de tópicos?

ATIVIDADE DA AULA

E4

Redes neurais

insper.github.io/ML/pt/2026.2/classes/neural-networks/

respostas comentadas na página 165

A pergunta da aula. Num XOR (ou exclusivo) ruidoso, a regressão logística acerta 0,485, o nível do acaso; uma rede com uma camada oculta de 8 unidades ReLU acerta 0,983 nos mesmos folds. O que a camada oculta calcula que um neurônio sozinho não calcula? E como cada peso aprende com o erro?

ETAPAS DO FLUXO NESTA AULA modelo avaliação

O que você precisa saber

- Um **neurônio** calcula a pré-ativação $z = w^\top x + b$ (pesos w, viés b) e a ativação $a = \varphi(z)$. Com a sigmoide $\sigma(z) = 1/(1+e^{-z})$, é a regressão logística.
- O **perceptron** usa um degrau e rótulos $y \in \{-1, +1\}$. Se $y_i z_i \leq 0$ (lado errado ou sobre a reta), $w \leftarrow w + \eta\, y_i x_i$ e $b \leftarrow b + \eta\, y_i$, com taxa de aprendizado η. Para quando uma **época** (uma passada pelos dados) não tem erro; se nenhuma reta separa as classes, como no XOR, nunca para.
- Uma **camada oculta** calcula atributos novos, nos quais o problema fica linearmente separável; a saída é um classificador linear sobre eles. Sem ativação não linear as camadas **colapsam**: $W_2(W_1 x + b_1) + b_2$ é uma camada só.
- **Passagem direta**: um produto de matrizes e uma função elemento a elemento por camada. Uma camada com n_{ent} entradas e n_{sai} unidades tem $(n_{\text{ent}} + 1)\, n_{\text{sai}}$ parâmetros.
- **Retropropagação**: a regra da cadeia, da perda L para trás. O **delta** de uma unidade é $\delta = \partial L / \partial z$; o gradiente de um peso é o delta da unidade que ele alimenta vezes o valor que entrou por ele. Com saída sigmoide e entropia cruzada, $\delta_y = \hat{y} - y$. A ReLU (unidade linear retificada), $\max(0, z)$, tem derivada 1 se $z > 0$ e 0 se $z < 0$; a da sigmoide nunca passa de 0,25 (o **gradiente que desaparece**).
- Na prática: `StandardScaler` no pipeline; largura e `alpha` (o **weight decay**, a penalidade L2 do Ridge) escolhidos por validação cruzada; a perda não é convexa, então a semente importa. Em tabelas, gradient boosting primeiro.

1 O perceptron à mão

• AQUECIMENTO

Quatro pontos de treino, visitados na ordem A, B, C, D, a partir de $w = (0,0)$, $b = 0$ e $\eta = 1$:

ponto	x_1	x_2	y	$z = w^\top x + b$	$y\,z$	erro?	w depois	b depois
A	2	1	$+1$					
B	1	2	-1					
C	0	1	-1					
D	3	2	$+1$					

(a) Complete a tabela da época 1. Lembre que z é calculado com os pesos de *antes* da visita.

(b) Faça a época 2 com os pesos finais. Ela tem algum erro? O que o perceptron faz depois dela?

(c) Escreva a fronteira final como uma equação e diga de que lado dela fica a classe $+1$.

CONFIRA NO SIMULADOR

O simulador do perceptron

Rode o exemplo de seis pontos com **Passo** e confira cada $y\,z \le 0$; depois troque para o XOR e veja os erros por época nunca chegarem a zero.

2 Quantas camadas, de verdade?

• AQUECIMENTO

(a) Uma rede $2 \to 2 \to 1$ *sem* ativação na camada oculta calcula $h = W_1 x + b_1$ e $u = w_2^\top h + b_2$, com $W_1 = \begin{pmatrix} 1 & 2 \\ 3 & -1 \end{pmatrix}$, $b_1 = (1,0)$, $w_2 = (2,-1)$ e $b_2 = 0{,}5$. Escreva u como uma única camada, $u = w^\top x + b$: quanto valem w e b? Confira com a entrada $x = (1,1)$ pelos dois caminhos.

(b) Conte os parâmetros (pesos e vieses) de uma rede com 20 entradas e 1 saída quando a parte oculta é (i) uma camada de 16 unidades; (ii) duas camadas de 8 unidades. Qual tem mais?

(c) Uma rede com 2 entradas, 50 unidades ocultas sem ativação e uma saída sigmoide: que fronteiras de decisão ela consegue traçar?

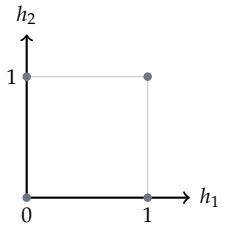
3 Uma rede montada à mão

●● PRÁTICA

Uma rede com degraus, $\text{degrau}(z) = 1$ se $z > 0$ e 0 caso contrário, todos os pesos escolhidos à mão:

$$h_1 = \text{degrau}(x_1 - x_2 - 0{,}5), \qquad h_2 = \text{degrau}(-x_1 + x_2 - 0{,}5), \qquad \text{saída} = \text{degrau}(h_1 + h_2 - 0{,}5).$$

x_1	x_2	h_1	h_2	**saída**
0	0			
0	1			
1	0			
1	1			



o espaço oculto: marque as 4 entradas

(a) Complete a tabela. Que função das duas entradas a rede calcula? Descreva em palavras o que cada unidade oculta detecta.

(b) No quadro à direita, marque onde cada uma das quatro entradas cai no espaço (h_1, h_2), indicando a classe. Por que agora uma reta basta? Qual é a reta que a saída traça?

(c) Para treinar essa rede por gradiente descendente, o degrau precisa ser trocado por uma sigmoide ou uma ReLU. Por quê?

4 Largura e penalidade

●● PRÁTICA

Uma camada oculta treinada em 300 pontos de duas meias-luas ruidosas. Acurácia média nos folds de treino e na validação cruzada (CV); a regressão logística marca 0,830 na CV.

unidades (`alpha` = 0,0001)	1	3	10	100
treino	0,842	0,897	0,938	0,997
CV	0,843	0,873	0,893	0,857

100 unidades, `alpha` =	0,0001	0,01	0,1	1	10
treino	0,997	0,993	0,939	0,914	0,837
CV	0,857	0,880	0,903	0,917	0,830

(a) Por que 1 unidade oculta não faz melhor que a regressão logística?

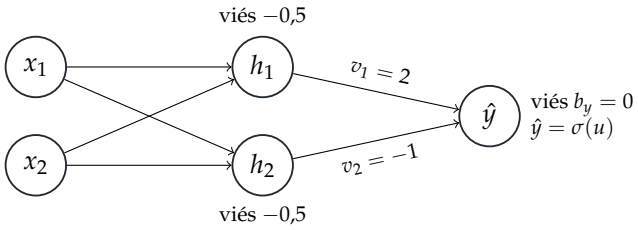
(b) Onde há overfitting? Que número da tabela mostra isso?

(c) Que configuração você escolhe? Por que `alpha` = 10 volta à marca da regressão logística?

5 Um passo de treino

●● PRÁTICA

Uma rede 2-2-1 com unidades ocultas ReLU e saída sigmoide, treinada com a entropia cruzada $L = -\ln \hat{y}$ (o alvo é $y = 1$). Um exemplo, $x = (1, 2)$:



Os pesos que entram em h_1 são $w_{11} = 0{,}5$ (de x_1) e $w_{12} = 0{,}25$ (de x_2), com viés $b_{h_1} = -0{,}5$; em h_2, $w_{21} = -1$ e $w_{22} = 0{,}5$. Use $\sigma(1) = 0{,}731$ e $\ln 0{,}731 = -0{,}313$.

(a) Passagem direta: calcule z_1, z_2, h_1, h_2, $u = v_1 h_1 + v_2 h_2 + b_y$, $\hat{y}$ e L.

(b) Retropropagação: calcule $\delta_y = \hat{y} - y$, os gradientes $\partial L/\partial v_1$ e $\partial L/\partial v_2$, os deltas ocultos $\delta_{h_j} = \delta_y \, v_j \, \varphi'(z_j)$ (com φ' a derivada da ReLU) e os gradientes $\partial L/\partial w_{11}$, $\partial L/\partial w_{12}$ e $\partial L/\partial b_{h_1}$.

(c) Por que nenhum peso que entra em h_2 muda com este exemplo?

(d) Dê um passo com $\eta = 0{,}1$ ($p \leftarrow p - \eta \, \partial L/\partial p$) em v_1, w_{11}, w_{12}, b_{h_1} e b_y. Com os pesos novos, $\hat{y}$ passa a 0,843: a perda caiu?

CONFIRA NO SIMULADOR

Retropropagação, passo a passo

Na rede 2-2-1 sigmoide da aula, avance do passo 1 ao 7 e compare o delta da saída com os deltas ocultos: veja o gradiente encolher ao voltar para a primeira camada.

6 Três notas de uma colega

● ● ● DESAFIO

Uma colega treinou redes neurais e mandou três notas. Para cada uma, diga o que está acontecendo e o que você faria.

(i) "Num conjunto de e-mails (57 atributos numéricos, com desvios padrão de 0,076 a 606), o MLP (perceptron multicamadas) direto nos dados deu acurácia de CV 0,927; com um `StandardScaler` no pipeline, 0,948. Os dados são os mesmos, então deve ser sorte."

(ii) "Num problema pequeno, de duas entradas, a mesma rede de 2 unidades ocultas, nos mesmos dados, com `random_state` de 0 a 9, deu acurácias de treino 0,76; 0,66; 0,74; 0,94; 0,94; 0,94; 0,94; 0,94; 0,74; 0,66. Acho que o otimizador está com defeito."

(iii) "Nos e-mails, comparei com gradient boosting nos mesmos 5 folds. Boosting − MLP, fold a fold: 0,003; 0,002; 0,021; 0,016; −0,002. Conclusão: redes neurais não servem para tabelas."

Em (iii), calcule a média e o desvio padrão (divida por 5) das diferenças e escreva a conclusão honesta numa frase. Diga também o que o boosting não precisou e a rede precisou.

ATIVIDADE DA AULA

E5
Explicabilidade

insper.github.io/ML/pt/2026.2/classes/explainability/

respostas comentadas na página 166

A pergunta da aula. Uma random forest de crédito marcou um solicitante com risco previsto de 0,710 de o empréstimo dar errado. Em que atributos o modelo se apoia? Por que ele deu esse número a essa pessoa? E o que essa explicação não autoriza a concluir?

ETAPAS DO FLUXO NESTA AULA modelo avaliação

O que você precisa saber

- Explicações **globais** (como o modelo se comporta no geral) e **locais** (por que esta previsão). Toda explicação descreve **o modelo**, não o mundo: não é causal.
- **Importância por permutação**: embaralhe uma coluna em dados **separados** e meça a queda média do escore em alguns embaralhamentos (quedas perto de zero, até negativas, são ruído). A **MDI** (*mean decrease in impurity*, o `feature_importances_` das árvores) é medida no treino e premia colunas com muitos valores distintos, até ruído.
- Atributos correlacionados **dividem o crédito** (embaralhe o grupo junto) e geram **linhas impossíveis** quando um deles é mexido sozinho.
- **PDP** (gráfico de dependência parcial): $\mathrm{PD}_j(v)$ é a média, sobre as linhas, da previsão com o atributo j trocado por v. As curvas **ICE** (de expectativa condicional individual) fazem a mesma conta linha a linha; curvas em leque indicam interações.
- **SHAP** (*SHapley Additive exPlanations*): previsão $= \phi_0 + \phi_1 + \cdots + \phi_d$, com o **valor base** ϕ_0 (a previsão média num conjunto de fundo) e uma parcela ϕ_j por atributo. O valor de uma coalizão S de atributos, $v(S)$, é a média das previsões no fundo com os atributos de S trocados pelos da linha. Com dois atributos, $\phi_1 = \frac{1}{2}[v(\{1\}) - v(\varnothing)] + \frac{1}{2}[v(\{1,2\}) - v(\{2\})]$.
- **LIME** (*Local Interpretable Model-agnostic Explanations*): as inclinações de um modelo linear ajustado em volta da linha; responde "o que mudaria a previsão?" e depende da largura da vizinhança.

1 Embaralhar e medir

• AQUECIMENTO

(a) Um classificador de crédito tem ROC-AUC (a área sob a curva ROC) de 0,80 nos dados de teste. Cinco embaralhamentos de cada coluna deram:

coluna embaralhada	ROC-AUC					queda média
`conta_corrente`	0,66	0,68	0,65	0,67	0,64	
`finalidade`	0,80	0,79	0,81	0,80	0,79	

Complete a tabela. De qual coluna o modelo depende? Como uma nota embaralhada pode sair *acima* da original (0,81)?

(b) Uma random forest de preços de casas recebeu uma coluna de puro ruído contínuo. Ela tem MDI 0,018 (uma coluna de cara ou coroa tem 0,002); embaralhá-la custa 0,016 de R^2 nos dados de treino e $-0{,}000$ nos de teste. Explique os três números. Qual deles vai para o relatório?

2 A dependência parcial à mão

• AQUECIMENTO

Um modelo prevê o valor das casas de um distrito, $\hat{f} = 100 + 30\,\text{renda} + 15\,\text{renda} \cdot \text{antigo}$ (em US$ 1.000), a partir da renda mediana (em US$ 10.000) e de `antigo` (1 se as casas têm mais de 30 anos).

distrito	renda	antigo	$\hat{f}$ com renda $= 2$	$\hat{f}$ com renda $= 8$	ganho (ICE)
D1	3	0			
D2	5	1			
D3	2	1			
D4	6	0			
média			PD(2) =	PD(8) =	

(a) Complete a tabela. Por que a renda de cada distrito (a segunda coluna) não entra na conta?

(b) As curvas ICE são paralelas ou se abrem em leque? O que o ganho médio do PDP esconde?

(c) Num modelo em que latitude e longitude têm correlação $-0{,}92$, que cuidado pede o PDP só da latitude?

3 Duas surpresas da permutação

●● PRÁTICA

(a) Numa random forest de preços de casas, embaralhar a renda (nos dados de teste) custa 0,754 de R^2. A equipe acrescenta uma cópia exata da renda e retreina: agora embaralhar a renda custa 0,196 e embaralhar a cópia custa 0,196; embaralhar as duas juntas custa 0,748. Por que cada cópia parece valer tão pouco? Por que as duas separadas somam menos que as duas juntas? Que número vai para o relatório?

(b) Num modelo de gradient boosting com R^2 de teste 0,825, embaralhar a latitude "custa" 1,330 de R^2. Para quanto foi o R^2? Como um modelo pode ficar pior do que prever a média para todo mundo? (Latitude e longitude têm correlação $-0{,}92$; 45% das linhas embaralhadas ficaram a mais de 25 km de qualquer distrito real.) O que fazer?

4 Shapley à mão

●● PRÁTICA

Um modelo de risco usa dois atributos sim/não, `atraso` (teve atraso de pagamento no último ano) e `jovem` (tem menos de 25 anos):

$$\hat{f} = 0{,}10 + 0{,}20\,\text{atraso} + 0{,}10\,\text{jovem} + 0{,}20\,\text{atraso}\cdot\text{jovem}.$$

O fundo tem quatro solicitantes: $(\text{atraso}, \text{jovem}) = (0,0)$, $(1,0)$, $(0,1)$ e $(0,0)$. Explique o solicitante $x = (1,1)$.

(a) Calcule $v(\varnothing)$, $v(\{\text{atraso}\})$, $v(\{\text{jovem}\})$ e $v(\{\text{atraso}, \text{jovem}\})$. (Para $v(\{\text{atraso}\})$, ponha atraso $= 1$ nos quatro do fundo, mantenha o `jovem` de cada um e tire a média.)

(b) Calcule ϕ_{atraso} e ϕ_{jovem}. Confira que $\phi_0 + \phi_{\text{atraso}} + \phi_{\text{jovem}}$ dá a previsão de x.

(c) No próprio modelo, se x não fosse jovem, quanto cairia a previsão? Por que isso não é ϕ_{jovem}? E o banco pode concluir que, se a pessoa fosse mais velha, o risco real dela seria menor?

CONFIRA NO SIMULADOR

Valores de Shapley

No modelo de brinquedo da aula, siga as seis ordens de chegada; depois ponha a interação em 0 e em 80 e veja como ela é dividida ao meio entre os dois atributos que a criam.

5 Duas perguntas sobre a mesma casa

●● PRÁTICA

Para um distrito de San Francisco, o modelo de preços (em US$ 1.000) dá valor base 207,4 e previsão 429,7. A parcela SHAP da renda é $+2{,}6$. O LIME, ajustado em volta do mesmo distrito, dá à renda uma inclinação de $+42{,}1$ por desvio padrão (US$ 19.000 de renda) e à ocupação (pessoas por domicílio) $-49{,}9$ por desvio padrão. Com outras larguras de vizinhança, a inclinação da renda fica entre 38,2 e 54,8.

(a) O SHAP e o LIME se contradizem sobre a renda? Qual dos dois responde "por que esta casa é cara?" e qual responde "o que mudaria o preço dela?"

(b) Um vereador lê "ocupação $-49{,}9$" e propõe incentivar mais gente por casa para baratear a moradia. O que está errado?

(c) Que informação precisa acompanhar uma explicação LIME num relatório?

6 Antes de entregar o modelo

● ● ● DESAFIO

Uma random forest de crédito (ROC-AUC de validação cruzada 0,785) está quase pronta. A equipe juntou três achados.

(i) Para o solicitante 11, o valor base é 0,305 e a previsão, 0,710. As seis maiores parcelas SHAP: duração de 48 meses $+0{,}114$; conta corrente negativa $+0{,}114$; poupança abaixo de 100 marcos $+0{,}045$; `personal_status` "mulher divorciada, separada ou casada" $+0{,}033$; idade 24 anos $+0{,}021$; histórico de crédito $+0{,}020$. O modelo tem 20 atributos.

(ii) `personal_status` tem importância por permutação de $-0{,}001$; retreinar sem ele muda a ROC-AUC de validação cruzada em $-0{,}004 \pm 0{,}005$. Mesmo assim, sua parcela SHAP passa de 0,02 (em valor absoluto) para metade dos solicitantes de teste.

(iii) Uma coluna nova, `cartas_de_cobranca`, levou a ROC-AUC de validação cruzada de 0,785 para 0,927 e é a primeira em permutação (0,209) e em SHAP médio absoluto (0,154).

(a) Quanto somam as parcelas dos outros 14 atributos do solicitante 11? Quais motivos o banco pode escrever na carta de recusa e quais não deveria usar?

(b) `personal_status` fica ou sai? Tirar a coluna resolve tudo?

(c) Que pergunta fazer sobre `cartas_de_cobranca` antes de comemorar?

(d) Escreva uma limitação para a documentação do modelo, na forma "o modelo _____ porque _____; quem usa deve _____".

ATIVIDADE DA AULA

E6
AutoML

insper.github.io/ML/pt/2026.2/classes/automl/

respostas comentadas na página 167

A pergunta da aula. Com cinco hiperparâmetros e cinco valores cada, uma grade custa 3.125 validações cruzadas. Como gastar um orçamento de poucas dezenas de avaliações? E como saber se o melhor escore que a busca relata é honesto?

ETAPAS DO FLUXO NESTA AULA modelo avaliação

O que você precisa saber

- Ajustar é maximizar $f(\lambda)$, o escore de validação cruzada (CV) de uma configuração λ de hiperparâmetros: uma função **cara, ruidosa e sem gradiente**. Um método de busca é uma regra para escolher o próximo λ a avaliar.
- **Escala log** para tudo o que age de forma multiplicativa (taxa de aprendizado, `C`, `alpha`, número de folhas ou de vizinhos): cada década recebe a mesma fração dos sorteios.
- **Grade ou aleatória**: uma grade com k valores por eixo em d eixos custa k^d avaliações e testa cada hiperparâmetro em só k valores; B sorteios aleatórios testam B valores distintos de cada um, qualquer que seja d.
- **Otimização bayesiana**: um modelo substituto prevê, em cada λ, uma média μ e um desvio padrão σ. A próxima avaliação maximiza a **melhoria esperada** (EI, *expected improvement*), $\mathrm{EI} = (\mu - f^+)\,\Phi(z) + \sigma\,\varphi(z)$, com f^+ o melhor escore até agora, $z = (\mu - f^+)/\sigma$ e Φ e φ a acumulada e a densidade da normal padrão. O **TPE** (*tree-structured Parzen estimator*, o padrão do Optuna) sorteia onde as tentativas boas são densas e as ruins, raras.
- **Halving sucessivo**: n configurações com orçamento r; fica a melhor fração $1/\eta$, com η vezes mais orçamento, até sobrar uma. Supõe que a ordem com orçamento pequeno prevê a ordem final.
- A busca otimiza também o **vazamento** que você deu; e, quanto mais longa, mais **otimista** fica o melhor escore (a maldição do vencedor). Relate o teste trancado ou uma **CV aninhada** da busca inteira.

1 Onde caem os sorteios

• AQUECIMENTO

(a) A força de regularização `C` é sorteada entre 0,01 e 100. Que fração dos sorteios fica abaixo de 1 se o sorteio é uniforme? E se é uniforme na escala log?

(b) Marque os hiperparâmetros que pedem escala log: ( ) taxa de aprendizado entre 0,01 e 0,5; ( ) `alpha` entre 10^{-4} e 10; ( ) número de folhas entre 4 e 128; ( ) fração de atributos por divisão entre 0,5 e 1.

2 Grade ou sorteio?

• AQUECIMENTO

Um escore depende de três hiperparâmetros em $[0,1]$, mas só o primeiro, x_1, importa: o escore é alto perto de um valor ótimo c desconhecido, que pode estar em qualquer lugar. O orçamento é de 64 avaliações.

(a) Uma grade $4 \times 4 \times 4$ usa os valores $1/8$, $3/8$, $5/8$ e $7/8$ em cada eixo. Quantos valores distintos de x_1 ela testa? E 64 sorteios aleatórios?

(b) Com a grade, qual é a maior distância possível de c ao valor testado mais próximo? E a distância média? (Com 64 sorteios, a média fica perto de $1/(2 \cdot 65) \approx 0{,}008$.)

(c) Uma grade de 5 valores para cada um de 5 hiperparâmetros custa quantas avaliações? Quantos valores distintos ela dá a cada hiperparâmetro?

CONFIRA NO SIMULADOR

Grade, sorteio e halving

Com orçamento 9, conte os valores distintos de x_1 que a grade e o sorteio testam; aumente o orçamento e aperte *novo problema* algumas vezes.

3 Onde avaliar agora?

•• PRÁTICA

O melhor escore até agora é $f^+ = 0{,}80$. O substituto prevê, em três candidatos:

candidato	μ	σ	z	$\Phi(z)$	$\varphi(z)$	$(\mu - f^+)\Phi(z)$	$\sigma\varphi(z)$	EI
A	0,81	0,01						
B	0,77	0,06						
C	0,79	0,02						

Da tabela da normal: $\Phi(1) = 0{,}841$, $\varphi(1) = 0{,}242$, $\Phi(-0{,}5) = 0{,}309$, $\varphi(-0{,}5) = 0{,}352$.

(a) Complete a tabela. Qual candidato é avaliado a seguir? Em qual deles quem trabalha é a média e em qual é a incerteza?

(b) B e C têm o mesmo z. Por que a EI de B é três vezes a de C?

(c) O substituto supõe que o escore é suave numa escala de 0,15; o pico verdadeiro tem largura 0,06. O que pode dar errado?

4 Um torneio

●● PRÁTICA

(a) Halving sucessivo com 81 configurações, $\eta = 3$ e 5 árvores por configuração na primeira rodada. Monte as rodadas (configurações $\times$ árvores) até sobrar uma. Quanto custa o torneio, em árvores? E treinar as 81 com o orçamento completo?

(b) Num torneio de 27 configurações de gradient boosting com o *número de árvores* como orçamento, todas as taxas de aprendizado pequenas (0,01 a 0,03) saíram na primeira rodada; o vencedor (taxa 0,062) ficou em 13.º de 27 quando todas foram treinadas por completo. Com o número de *linhas* como orçamento, o vencedor ficou em 5.º. Por que a primeira unidade de orçamento enganou o torneio?

5 Dois escores que a busca relatou

●● PRÁTICA

(a) Num conjunto de módulos de software (prever quais terão defeito), a mesma busca aleatória relatou ROC-AUC (área sob a curva ROC) de CV 0,792 sem a coluna `bug_reports` e 0,916 com ela. A coluna conta os relatos de bug abertos contra o módulo depois do lançamento; o modelo é usado antes. No teste, com a coluna preenchida, o modelo com ela marca 0,962; com `bug_reports` $= 0$, como será usado, 0,782; o modelo sem ela marca 0,804. Qual modelo vai para produção? Que método de busca teria pego o problema?

(b) Com 300 linhas de desenvolvimento, de 10 para 150 tentativas, o escore de CV da configuração escolhida foi de 0,772 para 0,788 e o escore dela num teste grande, de 0,765 para 0,769. Que fração do ganho relatado é otimismo? E por que, com 1.687 linhas, as duas curvas subiram juntas?

6 O relatório da busca

● ● ● DESAFIO

Quatro buscas com o mesmo orçamento ajustaram um gradient boosting (ROC-AUC, CV de 3 folds). Os valores padrão marcam $0{,}9468 \pm 0{,}0042$.

busca	melhor CV	vencedor
grade $3 \times 3 \times 3$	0,9498	taxa 0,3 (de 0,03; 0,1; 0,3), 128 folhas (de 8; 32; 128), folha mínima 5 (de 5; 20; 80)
aleatória, 27 sorteios	0,9499	
TPE, 27 tentativas	0,9510	
halving, mesmo orçamento em linhas	0,9528	

O vencedor do halving, escolhido pela regra fixada antes (o melhor escore de CV), marca 0,9455 no teste; os valores padrão, só como referência, marcam 0,9374 no mesmo teste. Uma CV aninhada da busca aleatória inteira dá $0{,}9472 \pm 0{,}0067$.

(a) O que diz o vencedor da grade estar onde está?

(b) Algum método venceu os outros? Compare as diferenças com o desvio padrão entre folds.

(c) O vencedor caiu 0,007 da CV para o teste. Isso prova a maldição do vencedor? Quanto o ajuste vale de verdade, no teste?

(d) A busca aleatória relatou 0,9499 para si mesma. Quanto disso era otimismo?

(e) Escreva a frase do relatório: que número você divulga para "o modelo ajustado" e por quê?

ATIVIDADE DA AULA

E7
MLOps

insper.github.io/ML/pt/2026.2/classes/mlops/

respostas comentadas na página 167

A pergunta da aula. Um banco põe no ar, em maio de 2009, um modelo que ordena os clientes antes de cada ligação de telemarketing; a fração de ligações que terminam em contratação, 5,1% em 2008, chega a 51,6% em 2010. O que precisa ir no arquivo do modelo para que ele seja servido, verificado e reconstruído? E como perceber a mudança antes de os rótulos chegarem?

ETAPAS DO FLUXO NESTA AULA dados e divisão | modelo | avaliação

O que você precisa saber

- O artefato é o **pipeline inteiro** (pré-processamento e modelo) com seus metadados: versões das bibliotecas, impressão digital (hash) dos dados, schema e métricas de validação. Teste de **ida e volta**: salvar, carregar e conferir predições idênticas. Um pickle é um programa: só carregue arquivos de fontes confiáveis.
- O **schema** guarda as faixas e as categorias do treino. **Rejeite** o impossível (campo ausente ou inesperado, categoria desconhecida, tipo errado, valor fora dos limites do negócio) e **avise** o que é só novo (fora da faixa de treino). Sem schema, o pipeline responde a lixo com uma probabilidade normal.
- **Reprodutibilidade**: semente fixa, hash dos dados, versões fixadas, um registro de experimentos que nunca é editado.
- Três mudanças: **covariate shift** ($P(x)$, as entradas, muda), **concept drift** ($P(y \mid x)$ muda; invisível sem rótulos) e **prior shift** ($P(y)$, o balanço das classes, muda; a precisão e as probabilidades sofrem).
- **PSI** (índice de estabilidade populacional) $= \sum_i (a_i - e_i) \ln(a_i / e_i)$, com e_i a fração da referência e a_i a fração atual na faixa i: abaixo de 0,1, estável; de 0,1 a 0,25, moderada; acima de 0,25, significativa. **KS** (Kolmogorov–Smirnov): D é o maior salto entre as duas acumuladas; com muitas linhas, o p-valor acusa diferenças irrelevantes, então alerte por magnitude.
- Monitore em duas velocidades: entradas e escores na hora; métricas com rótulos depois. **Drift não é dano**. Troca segura: portão de avaliação, shadow, canary e rollback.

1 O PSI à mão

• AQUECIMENTO

Um atributo foi cortado nos quartis dos dados de treino, então cada faixa tem $e_i = 0{,}25$. Use $\ln 1{,}2 = 0{,}182$, $\ln 0{,}8 = -0{,}223$, $\ln 0{,}2 = -1{,}609$, $\ln 0{,}6 = -0{,}511$ e $\ln 2 = 0{,}693$.

	faixa 1	faixa 2	faixa 3	faixa 4	PSI	leitura
semana A: a_i	0,30	0,30	0,20	0,20		
termos						
semana B: a_i	0,05	0,15	0,30	0,50		
termos						

Complete a tabela. Por que nenhum termo pode ser negativo?

CONFIRA NO SIMULADOR

Drift e PSI

Desloque a média da produção até o PSI cruzar 0,25; volte-a para 0 e mude só a dispersão; por fim, ligue o concept drift e veja o PSI ficar em 0 enquanto a acurácia cai.

2 O maior salto

• AQUECIMENTO

(a) Referência: 2, 4, 6, 8. Dados atuais: 5, 7, 9, 11. Calcule as acumuladas empíricas $F_{\text{ref}}(x)$ e $F_{\text{cur}}(x)$ em cada valor que aparece e a estatística D.

(b) Num lote de um milhão de linhas, a média de um atributo se moveu 0,02 desvio padrão: o p-valor do KS é $1{,}6 \times 10^{-41}$ e o PSI, 0,0005. Você dispara o alerta? Por quê?

3 Quatro lotes, quatro diagnósticos

●● PRÁTICA

Um modelo foi treinado com um atributo $x \sim N(0,1)$. Em produção, quatro lotes de 5.000 linhas, já com os rótulos:

lote	PSI(x)	KS p	acurácia	precisão	taxa real	prevista média	mudança
I	0,152	0,000	0,794	0,491	0,200	0,339	
II	0,004	0,695	0,816	0,769	0,462	0,465	
III	0,003	0,340	0,660	0,329	0,167	0,459	
IV	3,505	0,000	0,728	0,729	0,720	0,780	

(a) Escreva na última coluna: sem mudança, covariate shift, concept drift ou prior shift. Justifique cada um com dois números.

(b) Qual das mudanças nenhum monitor de entrada consegue ver? O que é preciso para vê-la?

4 Rejeitar, avisar ou servir?

●● PRÁTICA

O schema do modelo: `idade` com faixa de treino de 19 a 95 e limites do negócio de 18 a 100; `pdays` (dias desde o último contato) com faixa de treino de -1 a 350; `profissao` com as categorias vistas no treino (`admin.`, `technician`, `services` e outras nove). Para cada requisição, escreva **servir**, **avisar** (servir e registrar) ou **rejeitar**, com o motivo.

	requisição (o resto está certo)	decisão e motivo
R1	`idade` = 17	
R2	`idade` = 97	
R3	`pdays` = 420	
R4	`profissao` = `'astronauta'`	
R5	`idade` = `'41'` (texto)	
R6	traz um campo a mais, `duracao_da_ligacao` = 240	

Sem o schema, com `OneHotEncoder(handle_unknown='ignore')` no pipeline, o que acontece com R1 e R4? Por que R6 merece mais atenção que um simples erro de formato?

5 Números que quase não se mexem

●● PRÁTICA

(a) Um aplicativo novo passa a mandar a duração do empréstimo em anos, não em meses. A ROC-AUC (área sob a curva ROC) de teste iria de 0,783 para 0,751; a fração de solicitantes marcados como arriscados, de 20,0% para 2,8%; 17,2% das decisões mudam. Um painel que acompanha só a ROC-AUC enxergaria o problema? Que verificação o pega na hora?

(b) Duas random forests que diferem só na semente marcam ROC-AUC 0,763 e 0,766 no teste, mas 18 de 250 solicitantes recebem decisões diferentes. Que fração é essa? Por que fixar a semente?

(c) Uma chamada ao pipeline custa cerca de 1,7 ms, quer leve uma linha, quer mil. Quanto tempo leva pontuar 10.000 clientes uma linha por chamada? Que padrão de serving você prefere quando a decisão pode esperar até amanhã?

6 Um ano de monitoramento

● ● ● DESAFIO

O modelo do banco (ROC-AUC de validação 0,619) foi monitorado por trimestre. As regras de alerta, escritas antes: avisos de schema $> 5\%$; PSI do pior atributo $> 0{,}25$; PSI do escore $> 0{,}25$; $|\text{prevista média} - \text{taxa real}| > 0{,}05$; ROC-AUC $<$ validação $-0{,}05$.

trimestre	avisos	pior atributo (PSI)	PSI do escore	taxa real	prevista média	ROC-AUC
2009Q2	14,7%	`contact` (3,247)	0,071	0,124	0,081	0,671
2009Q3	5,3%	`contact` (2,212)	1,360	0,367	0,150	0,584
2010Q2	12,0%	`poutcome` (2,145)	0,891	0,551	0,167	0,698
2010Q4	12,2%	`poutcome` (2,295)	1,098	0,427	0,174	0,749

(a) Que alertas disparam em cada trimestre?

(b) Em 2009Q2, os avisos vêm quase todos de `pdays` acima de 350 (os dados de treino começaram um ano antes) e o PSI de `contact` vem de o canal não ter sido registrado nos dois primeiros meses do treino. O modelo quebrou?

(c) O que quebrou a partir de 2009Q3: a ordenação dos clientes ou as probabilidades? Por que a regra da ROC-AUC nunca dispara? Que mudança é essa?

(d) Retreinar a cada trimestre com os últimos 6 meses levou o desvio médio de calibração de $-0{,}272$ para $-0{,}037$. Que aposta essa política faz? Que passos vêm antes de o modelo retreinado substituir o atual?

ATIVIDADE DA AULA

E8

A fronteira

insper.github.io/ML/pt/2026.2/classes/frontier/

respostas comentadas na página 168

A pergunta da aula. "Classifique este tíquete de suporte como cobrança, técnico ou outro": em 2018, um projeto de rotulagem e um classificador treinado; hoje, muitas vezes um único prompt. Quando um modelo de fundação substitui o `fit()`, quando a caixa de ferramentas das aulas anteriores continua vencendo e onde ela está dentro da pilha nova?

ETAPAS DO FLUXO NESTA AULA dados e divisão | modelo | avaliação

O que você precisa saber

- O **mapa do campo** agrupa as técnicas pelo trabalho que fazem; as da fronteira se apoiam nas do curso.
- **Transfer learning**: reusar um modelo pré-treinado, por **extração de atributos** (o modelo fica congelado; seus embeddings alimentam uma cabeça clássica) ou por **fine-tuning** (o treino continua com os seus rótulos). A **LoRA** (adaptação de baixo posto) treina só uma correção BA de cada matriz $n \times n$, com B de $n \times r$ e A de $r \times n$: $2nr$ números em vez de n^2.
- **Modelos de fundação**: pré-treino **autossupervisionado** (os alvos saem do próprio texto, como o próximo token) em escala enorme. Um **LLM** (grande modelo de linguagem) se adapta por zero-shot, few-shot (exemplos no prompt; nenhum peso muda), fine-tuning ou **RAG** (geração aumentada por recuperação: os trechos mais próximos da pergunta, por embeddings, vão para o prompt).
- Dentro da pilha nova: saída softmax, gradiente descendente com weight decay e entropia cruzada; o RAG é um k-NN (k-vizinhos mais próximos) sobre embeddings; o modelo de recompensa do **RLHF** (aprendizado por reforço com feedback humano) é uma regressão logística sobre pares, $P(\text{A preferida}) = \sigma(r_A - r_B)$; um benchmark contaminado é vazamento.
- **Onde o clássico vence**: tabelas (gradient boosting), latência e custo, decisões reguladas, problemas pequenos e bem estruturados, determinismo. **Roteamento**: linguagem e percepção para os modelos de fundação; predição estruturada para a pilha clássica. Em todo caso, um conjunto de teste rotulado.
- O **fluxo de ML**: formular o problema, obter e explorar os dados, prepará-los, treinar, avaliar com honestidade, implantar e monitorar. A avaliação manda de volta à preparação; o drift manda de volta aos dados.

1 Peças conhecidas

• AQUECIMENTO

Para cada componente da pilha moderna, escreva a peça do curso que ele usa e o código da aula.

componente	peça do curso	aula
a camada de saída de um LLM, que dá uma probabilidade a cada token do vocabulário		
o treino de um LLM: passos em mini-lotes com uma penalidade L2 nos pesos		
a busca dos trechos que entram no prompt de um RAG		
o modelo de recompensa do RLHF, treinado em pares de respostas comparadas		
um benchmark cujas perguntas apareceram nos dados de treino do LLM		
o BERTopic: embeddings, UMAP, HDBSCAN e um TF-IDF (frequência do termo ponderada pelo inverso da frequência nos documentos) por grupo		
medir as respostas de um LLM contra 200 rótulos feitos por pessoas		

2 Duas contas da fronteira

• AQUECIMENTO

(a) A LoRA adapta uma matriz de pesos 2048×2048 com posto $r = 16$. Quantos números são treinados? Que fração dos pesos da matriz isso é?

(b) Um modelo de recompensa dá $r_A = 1{,}5$ à resposta A e $r_B = 0{,}5$ à resposta B. Qual é a probabilidade de uma pessoa preferir A? Se a pessoa preferiu A, qual é a perda de entropia cruzada desse par? E se preferiu B? (Use $\sigma(1) = 0{,}731$, $\ln 0{,}731 = -0{,}313$ e $\ln 0{,}269 = -1{,}313$.)

CONFIRA NO SIMULADOR

O mapa do campo

Clique em RAG e siga as ligações até o k-NN e os embeddings; depois clique em alinhamento (RLHF) e ache a regressão logística.

3 Clássico ou fundação?

●● PRÁTICA

Para cada situação, escreva **clássico**, **fundação** ou **os dois**, com o motivo.

	situação	escolha e motivo
S1	prever churn numa tabela de 200.000 clientes com 40 colunas numéricas e categóricas	
S2	achar a intenção de mensagens livres de clientes, com 50 exemplos rotulados	
S3	uma decisão de crédito cujos motivos precisam ser mostrados a um auditor	
S4	pontuar 50 milhões de transações toda noite, com orçamento curto	
S5	classificar as linhas de uma tabela pequena, de 800 linhas	
S6	transcrever ligações de atendimento e prever, junto com os dados do cliente, quem vai cancelar	

4 Avaliar um prompt

●● PRÁTICA

Um prompt few-shot marca tíquetes de suporte como “cobrança”. A equipe o testou em 200 tíquetes rotulados por pessoas, 60 deles de cobrança: o prompt marcou 55 tíquetes, dos quais 45 eram de cobrança.

(a) Monte a matriz de confusão (real nas linhas, previsto nas colunas, cobrança primeiro) e calcule a acurácia, a precisão e o recall. Quanto acerta quem responde sempre “não é cobrança”?

(b) Os 5 exemplos do prompt foram copiados de tíquetes desses mesmos 200. Que problema isso cria? Como corrigir?

(c) O classificador antigo (TF-IDF e regressão logística), nos mesmos 200 tíquetes, teve precisão 0,80 e recall 0,70. Dá para dizer que o prompt é melhor? Quantos tíquetes de cobrança separam os dois recalls?

	previsto: cobrança	**previsto: outro**
real: cobrança		
real: outro		

5 Revisar um plano

● ● ● DESAFIO

Uma rede de academias quer prever quais alunos vão cancelar a matrícula no mês seguinte, para oferecer um desconto antes. A tabela tem 12.000 alunos; 8% cancelaram. O plano de um estagiário está abaixo. Para cada passo, escreva a etapa do fluxo (1 formular, 2 obter e explorar, 3 preparar, 4 treinar, 5 avaliar, 6 implantar e monitorar), se está certo ou errado, a aula que trata dele (código) e, se estiver errado, a correção.

	passo do plano	etapa	aula	certo? correção
P1	O alvo é "cancelou no mês seguinte"; o modelo roda no dia 1 de cada mês.			
P2	Fiz a análise exploratória (EDA) na tabela inteira e escolhi os atributos pelas correlações com o alvo.			
P3	Imputei a mediana e padronizei a tabela inteira; depois dividi em treino e teste.			
P4	A coluna `motivo_do_cancelamento` ajuda muito; ficou.			
P5	Usei k-NN com $k = 1$, sem escala: acurácia de treino 1,00.			
P6	No teste, a acurácia foi 0,92: ótimo resultado.			
P7	Testei 300 configurações e fiquei com a de melhor nota no teste.			
P8	O SHAP (valores de Shapley) mostra que ir pouco à academia aumenta o risco; vamos obrigar os alunos a vir.			
P9	Salvei só o modelo (sem o pré-processamento) e vou olhar a acurácia uma vez por ano.			

6 Do problema ao monitor

● ● ● DESAFIO

Uma prefeitura quer priorizar a fiscalização de restaurantes: prever quais vão ser reprovados na próxima inspeção. Há 3.000 inspeções passadas, de 2019 a 2025 (12% de reprovações), com atributos do restaurante (idade, tamanho, bairro, reprovações anteriores) e o texto livre das reclamações de clientes. A fiscalização visita 100 restaurantes por mês. Escreva uma decisão concreta para cada etapa do fluxo, citando a aula que a fundamenta:

(a) formular: o que é previsto, para quem e em que momento; qual erro custa mais;

(b) preparar: o que fazer com o texto das reclamações e com as colunas da tabela;

(c) treinar: que modelo começa e qual é o baseline;

(d) avaliar: que divisão (as inspeções têm data) e que métrica, comparada com que número;

(e) explicar e implantar: o que o fiscal precisa ver e o que monitorar depois que o modelo estiver no ar.

Respostas comentadas

Os números foram conferidos em Python. Quando a resposta é aberta, o comentário diz o que uma boa resposta precisa ter. **Como ler:** **em verde e negrito**, a resposta; × **Cuidado**, em vermelho, o erro mais comum.

F1 Introdução e história

1. (a) 2; (b) 1; (c) 3; (d) 1; (e) 3; (f) 2. (g) A **avaliação honesta**: medir o modelo em dados que ele nunca viu, assunto de C6 · Validação cruzada e vazamento de dados.

2. (a) P; (b) A, regras numerosas demais; (c) A, regras que mudam; (d) A, personalização; (e) P; (f) A, regras desconhecidas. Em (b) há maneiras demais de uma placa aparecer (ângulo, luz, sujeira), como os jeitos de escrever um "7". Em (f), como no tumor da aula, ninguém sabe escrever a regra exata a partir da imagem bruta.

× **Cuidado:** (a) e (e) não pedem ML: uma fórmula conhecida, curta e fixa se programa e acerta sempre.

3. (a) T**: jogar Hexapawn;** E**: as partidas jogadas;** P**: a taxa de vitórias.** (b) T: dizer que dígito está na imagem; E: imagens de dígitos já rotuladas; P: a fração de dígitos lidos corretamente. (c) T: prever os dias de internação; E: internações passadas com o tempo real; P: quantos dias o sistema erra, em média. (d) **Não aprendeu**: E cresceu, mas P não melhorou.

× **Cuidado:** guardar dados não é aprender; a definição exige que o desempenho medido suba.

4. (a) R1 acerta 1, 2, 4 e 6: **4 de 8**; R2 erra só o 2: **7 de 8**; R3 erra o 5 e o 8: **6 de 8**. (b) **R2.** Entraram os e-mails (dados) e os rótulos (respostas); saiu a regra R2, o modelo. Ninguém a escreveu: ela foi escolhida pelo acerto nos exemplos. (c) R2 acerta A e D e erra B (spam sem *prêmio*) e C (*prêmio* sem spam): **2 de 4**, bem menos que 7/8. O número medido nos exemplos usados para escolher a regra é otimista; F2 · O panorama de ML mostra por quê. (d) Regras que **mudam**: rotular e-mails novos e escolher a regra de novo (retreinar).

× **Cuidado:** o acerto nos e-mails usados para escolher a regra não diz quanto ela acerta em e-mails novos.

5. (a) $1/3$. (b) Sobra só a verde: **chance 1**. (c) Visita 1: 2, 1, 1, chance $2/4 = 1/2$; visita 2: 2, 1, 0, chance $2/3$; visita 3: **3, 1, 0, chance** $3/4$. (d) T: jogar Hexapawn; E: as partidas; P: a taxa de vitórias. Os parâmetros são **as contas de cada caixa**: o feedback (vitória ou derrota) as ajusta e P sobe com E. Nada ali "entende" o jogo.

6. Uma boa resposta. (a) T: sinalizar radiografias com pneumonia; E: radiografias já laudadas por médicos; P: a fração de radiografias classificadas corretamente (ou uma métrica mais cuidadosa, como em C3 · Métricas de classificação). (b) **Regras desconhecidas**: ninguém sabe escrever, pixel a pixel, o que distingue uma pneumonia. (c) As três: **dados** (exames digitalizados e laudados em grande quantidade), **poder computacional** (GPUs para treinar redes com milhões de imagens) e **software** (bibliotecas abertas). (d) "Os 99% foram medidos em radiografias que o sistema nunca viu, de outro hospital?" e "quantas das radiografias tinham pneumonia?": se só 1% tem, um sistema que nunca sinaliza nada também acerta 99%. Os invernos vieram de promessas maiores que os resultados; o antídoto é medir com honestidade.

× **Cuidado:** um acerto alto sem saber onde foi medido e qual é a proporção de doentes não diz se o sistema é útil.

F2 O panorama de ML

1. (a) 1; (b) 2; (c) 3; (d) 4; (e) 5; (f) 5, volta a 3 (iterar); (g) 6, volta a 2 (os dados mudam, drift). Em (c), a mediana vem só das linhas de treino porque preparar é aprender com os dados.

2. (a) supervisionado, regressão; (b) não supervisionado (agrupamento ou tópicos); (c) por reforço; (d) supervisionado, classificação; (e) supervisionado, regressão; (f) não supervisionado (redução de dimensionalidade); (g) autossupervisionado. Em (g) o "rótulo" (a palavra escondida) sai do próprio texto: ninguém rotulou nada à mão.

× **Cuidado:** em (b) e (f) não há y: sem resposta conhecida não existe regressão nem classificação.

3. (a) **Treino 100%; teste** $7/10 = 70\%$. (b) A regra "sempre não spam" também tira 70% no teste: **o modelo não aprendeu nada**, só decorou (overfitting extremo). (c) O do teste, 70%: a **regra de ouro**, nunca avaliar nos dados de treino. (d) Não: o teste foi usado para escolher e a nota da variação vencedora é otimista. O teste deve ser usado **uma única vez**; para escolher entre variações, use dados de validação (C6).

× **Cuidado:** 100% no treino diz só que o modelo guardou as respostas que já tinha visto.

4. (a) **Graus 1 e 2: underfitting** (os dois erros altos e próximos). **Grau 4: bom ajuste** (o menor teste, 10,4, perto do ruído). **Graus 8 e 10: overfitting** (o treino cai, o teste dispara). (b) O treino desce sempre; o teste faz um U: desce até o grau 4, encosta na linha do 9 sem cruzá-la e depois sobe muito (por isso a

escala log). (c) O ruído não depende de x: nem a curva verdadeira o prevê, então o MSE esperado em dados novos nunca fica abaixo de $\sigma^2 = 9$. No treino, todo ajuste absorve um pouco do ruído (o grau 4 já marca 7,2); quanto mais abaixo de 9, mais ruído o modelo "previu". O grau 10, com 11 coeficientes para 15 pontos, chega a 3,6: decorou o ruído (o teste de 6.064,5 confirma). É a cara do overfitting. (d) **Os dos graus altos (8 e 10)**: com 100 pontos o polinômio tem pouco espaço para decorar e o teste cai para perto de 9. O grau 1 quase não muda: mais dados não curam o underfitting.

× **Cuidado:** erro de treino baixo não é sinal de modelo bom; só o erro em dados novos diz se ele generaliza.

5. (a) **Justiça**: as decisões históricas podem carregar viés; compare as taxas e os erros entre os bairros com perfis parecidos. (b) **Privacidade**: CEP, nascimento e sexo juntos podem reidentificar pessoas; verifique o consentimento e se a combinação é única. (c) **Transparência**: a decisão tem de poder ser explicada a quem foi afetado; "o modelo decidiu" não é justificativa (E5 · Explicabilidade). (d) **Ciclo de retroalimentação**: o modelo molda os dados com que é retreinado; verifique se a variedade do que é mostrado está encolhendo.

6. Uma boa resposta. (a) **Supervisionado, regressão** (o alvo é um número de caixas); uma linha é uma loja × remédio × semana. (b) **As semanas mais recentes**, depois de todas as de treino: o modelo vai prever o futuro e um sorteio poria semanas futuras no treino. (c) O flexível está em **overfitting** (1 contra 9); o simples erra igual nos dois, sem sinal de decorar. **Leve o simples**: erra 6 no teste contra 9. (d) Sem almoço grátis: nenhuma família ganha sempre; só comparando várias, na validação, se descobre a melhor para estes dados. (e) A seta **os dados mudam** (drift): preços, hábitos e remédios mudam; monitorar o erro e retreinar com as semanas novas.

× **Cuidado:** escolher pelo erro de treino levaria o modelo flexível, justamente o pior em semanas novas.

D1 Análise exploratória de dados

1. (a) **Média $90/9 = 10$; mediana 6.** Sem o 44: média $46/8 = 5{,}75$ e mediana $(5+6)/2 = 5{,}5$. A média caiu quase pela metade; a mediana mal se mexeu. (b) Posições $0{,}25 \cdot 8 = 2$ e $0{,}75 \cdot 8 = 6$: $Q_1 = 4$, $Q_3 = 8$, **IQR $= 4$**. (c) Cercas $4 - 6 = -2$ e $8 + 6 = 14$: **o 44 é sinalizado**.

× **Cuidado:** a média 10 não descreve distrito nenhum: oito dos nove ficam abaixo dela.

2. (a) Desvios ao quadrado: 49, 36, 36, 25, 16, 9, 4, 1 e 1.156, soma 1.332; $s = \sqrt{1.332/8} = \sqrt{166{,}5} \approx 12{,}90$ e $z = (44 - 10)/12{,}90 \approx 2{,}63$: **não é sinalizado**. O 44 inflou a média e o desvio que o julgam (mascaramento). (b) $8/\sqrt{9} \approx 2{,}67$: com 9 valores, $|z| > 3$ **não sinaliza nada, sejam quais forem os dados**. Por isso a regra do IQR é a usual na EDA. (c) *Erro*: `households` digitado errado (1 em vez de 10); corrigir na fonte ou marcar como faltante. *Raro*: poucas casas enormes; manter e preferir mediana, quantis ou log. *Outra coisa*: **casas de veraneio vazias**; como `households` conta só domicílios ocupados, sobram cômodos sem morador; sinalizar o distrito ou tratá-lo à parte.

3. (a) $\bar{x} = 3$, $\bar{y} = 2$; produtos 2, 0, 0, 2 e -4, soma 0; as duas somas de quadrados valem 10: $r = 0/10 = 0$. (b) Postos de y: 2, 3, 4, 5, 1, com média 3; os desvios dos postos são os mesmos da tabela: **Spearman $= 0$**. O 20 continua sendo o maior y, então os postos não mudam e o Spearman fica 1; o 0 **inverte a ordem** (o último x ganha o menor y) e derruba os dois. (c) **Não**: o valor sobe, cai, salta e cai de novo. Uma relação que não é monótona não cabe num coeficiente; a localização importa muito, só não em linha reta.

× **Cuidado:** $r \approx 0$ quer dizer "nenhuma tendência linear", nunca "nenhuma relação".

4. (a) **Um teto**: todo valor acima de 500 mil foi registrado como 500.001. Com 10 bins de 48.500 dólares, os 965 se misturam aos vizinhos numa barra larga. (b) **Pelo menos 500 mil**: o valor verdadeiro se perdeu. (c) Não: são distritos ricos e do litoral. Um modelo treinado com eles aprende que nada passa de 500 mil e **subestima os distritos caros**. (d) Defensáveis: manter e documentar a limitação; remover e declarar que o modelo só vale abaixo de 500 mil (ou usar um modelo para dados censurados). Indefensável: não perceber e tratar 500.001 como um preço real.

5. (a) A diferença é $198.106 - 206.191 = -8.085$, **menor que a oscilação de 8.991**; renda e idade também são parecidas. **Não há evidência contra MCAR**, o que não prova MCAR (o MNAR não se detecta pelos dados). (b) Imputar com a **mediana do treino** e guardar uma coluna indicadora de falta. A mediana sai só das 16.512 linhas de treino. (c) A falta

dependeria de uma coluna observada (a localização): **MAR**. A mediana geral daria aos distritos do interior valores de distritos caros; impute por grupo ou a partir das outras colunas e mantenha o indicador.

× **Cuidado:** calcular a mediana de imputação no arquivo inteiro usa as linhas de teste: só os fatos de como o arquivo foi feito podem vir dele todo.

6. (a) A: pequenas $9/90 = 10\%$, grandes $63/210 = 30\%$, **total** $72/300 = 24\%$. B: 15%, 35%, **total** $57/300 = 19\%$. **A é melhor nos dois tamanhos e pior no total.** (b) O **tamanho da peça** é o confundidor: A faz sobretudo peças grandes (210 de 300), que têm mais defeito; B, sobretudo pequenas. O total compara um lote difícil com um fácil. (c) Algo como: "a correlação negativa no total é sobretudo renda disfarçada (a fração de quartos cai com a renda); dentro de cada faixa de renda ela é positiva. É uma **hipótese a testar**, não um achado: não escrever que mais quartos baixam o preço".

× **Cuidado:** uma correlação no agregado não sobrevive sozinha: condicione nas terceiras variáveis óbvias (renda, localização) antes de concluir.

D2 Pré-processamento de dados

1. (a) $d(D,E) = \sqrt{30^2 + 2^2} \approx 30{,}07$ e $d(D,F) = \sqrt{2^2 + 60^2} \approx 60{,}03$: **E, o senhor de 60 anos**. (b) $d(D,E) = \sqrt{(30/14{,}52)^2 + (2/52{,}88)^2} \approx 2{,}07$ e $d(D,F) = \sqrt{(2/14{,}52)^2 + (60/52{,}88)^2} \approx 1{,}14$: **agora F**. (c) $52{,}88^2/(14{,}52^2 + 52{,}88^2) = 2.796{,}3/3.007{,}1 \approx$ **93,0%**: nas colunas cruas, um k-NN é praticamente um k-NN na tarifa.

× **Cuidado:** nenhuma das duas respostas é "a verdadeira"; a bruta é um acidente das unidades, a padronizada é uma escolha que dá a cada coluna o mesmo peso.

2. (a) $\mu = 60$; $\sigma = \sqrt{(2.500 + 1.600 + 900 + 400 + 19.600)/5} = \sqrt{5.000} \approx 70{,}71$; mín. 10, máx. 200, mediana 30, $Q_1 = 20$, $Q_3 = 40$ (IQR 20), máx. absoluto 200. (b) 10: $-0{,}71$; 0; -1; 0,05. 40: $-0{,}28$; 0,16; 0,5; 0,2. 200: 1,98; 1; 8,5; 1. (c) **No robusto**: de 10 a 40 as rendas comuns ocupam 1,5, contra 0,42 no padrão e cerca de 0,15 no min-max e no max-abs. Mediana e IQR vêm do meio dos dados, então o 200 não define a escala; ele só fica longe. (d) $(400 - 10)/190 \approx$ **2,05**. Não é erro: o mínimo e o máximo são os do treino e não se recalculam no teste.

3. (a) Média $627/6 = 104{,}5$, mediana $(15 + 31)/2 = 23$: a média muito acima da mediana indica **cauda longa à direita** (o 511). (b) **Continua**: o escalonador desloca e estica o eixo, mas não muda a forma; a assimetria fica igual. (c) Os valores viram 0, 3, 4, 5, 6 e 9: média $27/6 = 4{,}5$ e mediana 4,5. **A cauda sumiu**: o log comprime os valores grandes muito mais que os pequenos. (d) Há tarifas 0 e $\log 0 = -\infty$; o Box-Cox exige valores estritamente positivos, o Yeo-Johnson aplica a mesma família a $x + 1$. (e) **Antes**: escalonar primeiro cria valores negativos (o log falha) e o log desfaria a escala.

4. (a) S → (0, 0, 1); C → (1, 0, 0); Q → (0, 1, 0); **B → (0, 0, 0)**. (b) Inteiros: C–Q = 1 e C–S = 2; one-hot: todas $\sqrt{2} \approx 1{,}41$. Os inteiros afirmam que Q fica **entre** C e S e que S está duas vezes mais longe de C. (c) As três colunas somam 1 em toda linha, exatamente a coluna do intercepto: os pesos deixam de ser únicos. Descartando a primeira, **C vira a referência** e cada peso se lê "em comparação com Cherbourg". (d) Ordem alfabética: alto = 0, baixo = 1, médio = 2: **baixo fica entre alto e médio**. Declare a ordem: `OrdinalEncoder(categories=[['baixo', 'médio', 'alto']])`.

× **Cuidado:** one-hot com `drop='first'` num k-NN deixa a categoria de referência e uma desconhecida com o mesmo vetor de zeros; para distâncias, mantenha todas as colunas.

5. (a) Média 6; variância observada $(9 + 1 + 1 + 9)/4 = 5$; depois de preencher, $20/8 = 2{,}5$: **razão 0,5**, a fração observada (4/8). Os valores imputados não somam nada aos desvios, mas somam à contagem. (b) $14{,}53 \times \sqrt{713/890} \approx 14{,}53 \times 0{,}895 \approx$ **13,00 anos** (é o que a aula mede). (c) Distâncias na tarifa até £9: 21, 1, 17 e 1; os vizinhos são os de £8 e £10, com 20 e 24 anos: **22 anos**. A média das idades dá 33,5. (d) A ausência carrega sinal (quem não tem idade sobreviveu menos); a imputação apaga quais valores foram inventados e **o indicador devolve isso ao modelo**.

× **Cuidado:** preencher pela média deixa as idades mais homogêneas do que são; nenhum imputador recupera a informação perdida na coleta.

6. Uma boa resposta. (a) `PassengerId`: **sai** (identificador). `Pclass`: one-hot (ou ordinal com a ordem declarada). `Sex`: uma coluna 0/1 ou one-hot. `Age`: imputar com a mediana do treino (ou por grupo, já que é MAR) com indicador; depois escalonar. `SibSp`, `Parch`: escalonar. `Fare`: $\log(1 + x)$, depois escalonar. `Embarked`: moda do treino (ou "desconhecido") e one-hot com `handle_unknown='ignore'`. `Ticket`: sai (one-hot criaria 681 colunas). `Cabin`: sai a coluna bruta; no máximo extraia o deck, com "desconhecido" como categoria. Tudo ajustado só no treino. (b) Árvore: **escala e forma deixam de importar** e códigos ordinais servem; as árvores do scikit-learn aceitam até `NaN`. (c) A maioria dos bilhetes aparece uma ou duas vezes, então o valor codificado de um passageiro é, em boa parte, **o próprio rótulo**: o modelo lê a resposta. Direito: `TargetEncoder` com

`fit_transform` só no treino (cross-fitting: cada parte codificada com médias das outras) e `transform` no teste, dentro de um pipeline.

D3 Pipelines

1. (a) Mediana de 4, 8, 12 e 16: **10**. Coluna imputada: 4, 10, 8, 12, 16; média **10**; desvio $\sqrt{(36+0+4+4+36)/5} = \sqrt{16} = \mathbf{4}$. (b) NaN $\to 10 \to (10-10)/4 = \mathbf{0}$; $22 \to (22-10)/4 = \mathbf{3}$: três desvios acima da média do treino, uma informação que o modelo usa. (c) Nas duas linhas novas, a mediana vira 22, as duas linhas viram 22 e o scaler as leva a **0 e 0**; numa linha só, **0**. É o bug 1: o distrito fora do comum vira "perfeitamente médio".

× **Cuidado:** o `predict` de um pipeline nunca chama `fit`; quem reajusta no lote que prevê joga fora o que aprendeu.

2. (a) 1, S; (b) 1, S; (c) 2, E; (d) 2, S; (e) 3, S; (f) 3, E. (a) é o bug 1 no teste inteiro: 4.128 distritos sorteados têm quase as estatísticas do treino. (c) é o imputador esquecido; (d), o scaler esquecido. (g) **(b), (d) e (e)** erram mais que os US$ 92.894 da média. O mais perigoso é **(a)**: parece certo no notebook e passa pela revisão, mas explode em produção, onde as linhas chegam em lotes ou uma a uma.

3. (a) Ajustado em todas as linhas, o "valor médio da zona" de um distrito de teste sozinho na zona é **o próprio valor dele** (um pouco suavizado): o modelo lê a resposta na entrada. (b) Sim: de 60.383 para 42.552, **cerca de US$ 17.800 a menos**; só não é mágica. (c) O scaler **não olha o alvo** e a média e o desvio de 16.512 ou de 20.640 distritos quase não mudam; o encoder é feito do alvo. (d) Não dá para saber o tamanho de um vazamento sem medir as duas versões; **pondo todo passo dentro do pipeline, não há nada a medir**.

× **Cuidado:** "o scaler vazou pouco" não autoriza ajustar passos fora do pipeline: quanto mais o passo usa o alvo, maior o vazamento possível.

4. (a) **16.512 linhas, sempre**: nenhum passo vê o teste durante o `fit`. Cada transformador é ajustado e logo transforma, porque o passo seguinte precisa ser ajustado na saída *já transformada* do anterior (o scaler aprende com os valores imputados). (b) Preencher 163 buracos baixou a média, então **a mediana (437) é menor que a média**: a coluna tem cauda longa à direita. (c) No `predict`, o teste é medido com a média e o desvio **do treino**, então não sai exatamente 0. Uma média 0,00 no teste seria o sintoma do bug 1 (o scaler reajustado no teste).

5. (a) Ficam de fora `PassengerId`, `Name`, `Ticket` **e** `Cabin` (`remainder='drop'`). Chegam $3+1+3+2+3 =$ **12 colunas**. (b) Idade: $28{,}5 \to (28{,}5 - 29{,}556)/13{,}016 \approx -0{,}08$; SibSp: $(2 - 0{,}493)/1{,}060 \approx 1{,}42$; Parch: $(0 - 0{,}390)/0{,}838 \approx -0{,}47$. Porto novo: **zeros nas três colunas de** `Embarked`, em vez de um erro. (c) `prep__cat__simpleimputer__strategy` e `knn__n_neighbors`. (d) As quatro colunas descartadas chegariam cruas ao modelo, inclusive texto (`Name`), e o k-NN quebraria ou leria um identificador como medida.

× **Cuidado:** as médias e desvios do teste vêm do treino; recalculá-los na linha nova é o bug 1.

6. (a) (1) e (2): `FunctionTransformer`, nada a aprender. (3) e (4): **classe com** `fit` (percentis e categorias frequentes vêm do treino). (5): classe com `fit` **que usa o alvo**: precisa de cross-fitting (`TargetEncoder`). (b) Uma resposta:

```
class AgrupaRaras(TransformerMixin,
                  BaseEstimator):
  def __init__(self, min_count=10):
    self.min_count = min_count
  def fit(self, X, y=None):
    n = pd.Series(np.ravel(X)).value_counts()
    self.frequentes_ = sorted(
        n[n >= self.min_count].index)
    return self
  def transform(self, X):
    s = pd.Series(np.ravel(X))
    ok = s.isin(self.frequentes_)
    return s.where(ok, 'Rara').to_frame()
```

(c) Recontar no `transform` usaria as linhas que estão sendo previstas: uma categoria rara no treino e comum num lote viraria "frequente" (o bug 1). (d) **A cadeia inteira com tudo o que cada passo aprendeu**: medianas, médias, categorias e o k-NN. Quem carrega o arquivo não tem como esquecer um passo, reajustá-lo ou reordenar colunas.

U1 Redução de dimensionalidade

1. (a) PCA (uma multiplicação de matrizes, determinística, com `transform`); **(b) t-SNE ou UMAP**, com PCA antes porque d é grande (olhar e gerar hipóteses); **(c) UMAP** (tem `transform`; o t-SNE teria de refazer o mapa inteiro); **(d) UMAP** (é o papel dele no BERTopic, antes do HDBSCAN); **(e) PCA** (descarta as direções de baixa variância, onde fica o ruído).

2. Variâncias: 0,50; 0,25; 0,15; 0,05; 0,033; 0,017. Acumulada: 0,50; 0,75; **0,90 (3 componentes para 90%)**; **0,95 (4 para 95%)**; 0,983; 1. (b) **Erro** $= 0{,}9 + 0{,}3 + 0{,}2 + 0{,}1 = 1{,}5$, 25% do total 6: exatamente a variância descartada. (c) A curva acumulada: o que atrapalha distâncias são colunas sem estrutura. Se poucas direções guardam 95%, a dimensão intrínseca é pequena (aqui, 4 de 6) e as distâncias ainda têm contraste.

× **Cuidado:** contar colunas não mede a maldição; uma folha 2D girada para 1.000 colunas não perde contraste nenhum.

3. (a) Média $(5,4)$; centralizados $(-3,-2)$, $(-1,0)$, $(0,0)$, $(1,2)$, $(3,0)$. Variância de x: $20/4 = 5$; de y: $8/4 = 2$; covariância $8/4 = 2$: $C = \left(\begin{smallmatrix} 5 & 2 \\ 2 & 2 \end{smallmatrix}\right)$. (b) Traço 7, determinante 6: $\lambda^2 - 7\lambda + 6 = 0$, $\lambda_1 = 6$, $\lambda_2 = 1$; **PC1 explica** $6/7 = 85{,}7\%$. (c) $(5-6)w_1 + 2w_2 = 0 \Rightarrow w_2 = w_1/2$: $w_1 = (2,1)/\sqrt{5} \approx (0{,}894;\ 0{,}447)$. (d) Escore $= (1\cdot 2 + 2\cdot 1)/\sqrt{5} = 4/\sqrt{5} \approx 1{,}789$; reconstrução $(5,4) + \frac{4}{5}(2,1) = (6{,}6;\ 4{,}8)$; **erro** $0{,}6^2 + 1{,}2^2 = 1{,}8$. (e) A soma dos erros dividida por $n-1$ é λ_2: **soma** $= 4 \times 1 = 4$.

× **Cuidado:** o `np.linalg.eigh` devolve os autovalores em ordem crescente; quem pega a primeira coluna sem ordenar guarda a pior direção.

4. (a) **A unidade da prolina**: centenas contra unidades; a PC1 é a prolina sozinha. Não diz nada sobre química. (b) Padronizar é fazer o PCA na matriz de correlação, cujo traço é 13 (vezes 178/177, porque o `StandardScaler` divide por n e o PCA por $n-1$). (c) **O scaler e o PCA foram ajustados com todos os vinhos, teste incluído**: a projeção viu o teste. Correto: dividir primeiro e usar `make_pipeline(StandardScaler(), PCA(2), KNeighborsClassifier())`, com `fit` só em `X_tr`.

5. **Só (d) se sustenta**: a perda do t-SNE pune muito separar vizinhos de verdade. (a) Não: o tamanho é artefato da densidade (11,6 vezes vira 1,0). (b) Não: distâncias entre grupos não valem (10 vira 1,3). (c) Não: são seis manchas; até ruído vira grupos (0,09 vira 0,42). Peça grupos medidos nos dados originais, estabilidade entre sementes e perplexidades e uma comparação com colunas embaralhadas. (e) Não: não há `transform` honesto e o mapa foi feito para os olhos. (f) Não com t-SNE; com UMAP, sim (`reducer.transform`).

6. (a) $8/12 = 2/3$, $3/12 = 1/4$ **e** $1/12$. (b) x − média $= (2,-2,2)$: $z_1 = 0$, $z_2 = 2$, $z_3 = 4/\sqrt{2} = 2\sqrt{2} \approx 2{,}83$; $0 + 4 + 8 = 12 = \|(2,-2,2)\|^2$. (c) $k = 1$: $(5,5,2)$, **erro 12**; $k = 2$: $(5,5,4)$, **erro 8**. (d) $8/12 = 2/3$ **do afastamento está na PC3**. A variância explicada é uma média sobre o conjunto; o resíduo de um ponto pode morar justamente na direção "pouco importante".

× **Cuidado:** guardar 11/12 da variância do conjunto não garante reconstruir bem cada ponto.

U2 Agrupamento

1. (a) Ponto 0: $a = (2+3)/2 = 2{,}5$, $b = (7+8+10)/3 = 8{,}33$, $s = 0{,}70$. Ponto 3: $a = (3+1)/2 = 2$, $b = (4+5+7)/3 = 5{,}33$, $s = 0{,}625$. (b) Em 5: $a = (5+3)/2 = 4$, $b = (2+3+5)/3 = 3{,}33$, $s = -0{,}17$: em média, ele está mais perto do outro grupo. (c) Para que s fique entre -1 e 1, qualquer que seja a escala das distâncias.

2. Iteração 0 (d^2 a c_1 e a c_2): P_1 0 e 5 (grupo 1); P_2 5 e 0 (2); P_3 5 e 2 (2); P_4 25 e 40 (1); P_5 13 e 26 (1); P_6 26 e 49 (1). **Inércia** $0 + 0 + 2 + 25 + 13 + 26 = 66$; P_1 caiu com os três da direita. Atualizar: $c_1' = (5;\ 1{,}5)$, $c_2' = (0{,}5;\ 1{,}5)$. Iteração 1: P_1 11,25 e 4,5, P_1 **troca para o grupo 2**; os demais mantêm o grupo; **inércia 13,25**. Novos centros $(6,2)$ e $(1,1)$: ninguém troca, **convergiu com inércia 8** (cada grupo soma 4).

× **Cuidado:** o erro de um ponto se corrige quando os centros andam; um erro de centros inteiros (dois centros numa bolha) não se corrige: por isso `n_init`.

3. (a) Quedas: 186,3; 79,0; 67,8; 28,3; 17,2; 16,1. **Cotovelo em** $k = 3$; **os três índices escolhem** $k = 2$ (0,532; 482,2; 0,714). (b) Não erraram: medem compacidade e separação. Juntar as duas espécies parecidas dá uma partição de fato mais coesa e separada; "quão separada" não é "a partição que eu quero". O cotovelo acertou por sorte deste conjunto. (c) **Rodar o mesmo pipeline com cada coluna embaralhada** e relatar as duas silhuetas lado a lado: só a diferença diz se há grupos.

× **Cuidado:** 0,4 de silhueta, sozinho, não prova que existem grupos.

4. (a) Brutas: $d(P,Q) = \sqrt{64 + 9 + 625 + 10.000} \approx 103$; $d(P,R) = \sqrt{1 + 0{,}25 + 4 + 250.000} \approx 500$: **Q parece mais perto**, porque **a massa decide**. (b) Padronizadas: $d(P,Q) = \sqrt{1{,}455^2 + 1{,}5^2 + 1{,}786^2 + 0{,}125^2} \approx 2{,}75$; $d(P,R) = \sqrt{0{,}18^2 + 0{,}25^2 + 0{,}14^2 + 0{,}625^2} \approx 0{,}71$: **R é o vizinho**. (c) Sem escala, o k-means corta faixas de peso, que são bem separadas: silhueta alta, espécies erradas. A silhueta compara partições no mesmo espaço; mudar a escala muda a régua. Decide-se pelas unidades: grama e milímetro não se somam.

5. A, C, D: 4, centrais. B, E: 6, centrais. F: 5, central. G: 3 (G, F, H), **borda** (alcança F). H: 2 (H, G), **ruído**. I, J: 2, **ruído**. (a) **Um grupo (A a G);** -1 **para H, I e J**. (b) Grupos só crescem através de pontos centrais; G é borda, então H fica de fora. (c) **Com 2: todos viram centrais; dois grupos (A a H; I e J), nenhum ruído.**

× **Cuidado:** estar a menos de `eps` de um ponto do grupo não basta: esse ponto tem de ser central.

6. (a) Silhueta: bolhas, densidades, esticados, luas,

círculos. ARI: bolhas, esticados, densidades, luas, círculos. **Não concordam**: em *densidades* o k-means juntou os dois grupos compactos e cortou o frouxo; isso dá grupos compactos e separados, a mesma ideia de grupo da silhueta. (b) Cada grupo do k-means é uma célula convexa em torno de um centro; um anel não é convexo. O defeito está no critério, não no k nem na semente. (c) **Dos 45 pontos de ruído**: o ARI conta -1 como erro. (d) Por exemplo: "Padronizamos as 13 medidas e particionamos os vinhos em 3 grupos com k-means; $k = 3$ veio do cotovelo e dos índices internos, mas a silhueta só o separa de $k = 2$ por 0,017. A silhueta de 0,285 se compara a 0,068 com as colunas embaralhadas."

R1 Regressão linear

1. (a) **Entre apartamentos de mesma idade, cada m² a mais soma R$ 3.200 ao preço previsto.** (b) $50.000 + 224.000 - 15.000 =$ **R$ 259.000**. (c) **O mais velho sai R$ 7.500 mais barato** (5×1.500). (d) Como 1 km² $= 10^6$ m², o coeficiente viraria 3,2 bilhões por km² sem que nada no modelo mudasse. **Não dá**: o tamanho do coeficiente depende da unidade. Para comparar, padronize os atributos; mesmo assim, coeficiente grande quer dizer resposta íngreme, não importância. (e) O preço de um apartamento novo de 0 m²: uma extrapolação sem sentido. O intercepto só posiciona a reta.

× **Cuidado:** ler o 3.200 sem o "com a idade fixa". Se área e idade andam juntas nos dados, a relação bruta entre área e preço mistura os dois efeitos; o coeficiente os separa.

2. $\bar{x} = 5$, $\bar{y} = 7$. Desvios de x: $-3, -1, 1, 3$; de y: $-4, 0, -1, 5$. Produtos 12, 0, -1, 15 (soma 26); quadrados 9, 1, 1, 9 (soma 20). $\hat{w}_1 = 26/20 = 1{,}3$ **e** $\hat{w}_0 = 7 - 1{,}3 \cdot 5 = 0{,}5$**.** Previsões 3,1; 5,7; 8,3; 10,9; resíduos $-0{,}1$; 1,3; $-2{,}3$; 1,1. $\sum e_i = 0$ **e** $\sum e_i x_i = -0{,}2 + 5{,}2 - 13{,}8 + 8{,}8 = 0$: são as equações normais, uma por parâmetro (a do intercepto e a de w_1). Valem para todo ajuste OLS com intercepto.

3. (a) $\sum |e_i| = 4{,}8$, **MAE** $= 1{,}2$; $\sum e_i^2 = 8{,}2$, **MSE** $= 2{,}05$ **e RMSE** $\approx 1{,}43$; SST $= 16 + 0 + 1 + 25 = 42$, $R^2 = 1 - 8{,}2/42 \approx 0{,}80$ (com um atributo, é a correlação ao quadrado). (b) MAE 3 nos dois; **RMSE 3 para A e** $\sqrt{225/5} \approx 6{,}71$ **para B**. Se um atraso grande custa mais, **prefira A**: o RMSE vê a diferença que o MAE esconde. (c) $R^2 = 1 - 50/40 = -0{,}25$: o modelo erra mais que prever sempre a média. (d) O R^2 de treino nunca cai ao acrescentar uma coluna (o ajuste pode dar peso 0 a ela); a subida não prova nada. **Compare em dados que o modelo não viu.**

× **Cuidado:** escolher entre modelos pelo R^2 de treino.

4. (a) **1: funil** (o erro cresce com $\hat{y}$): modelar o log de y ou usar mínimos quadrados ponderados. **2: ondas no tempo** (erros vizinhos parecidos): um modelo de série temporal; as barras de erro do OLS não valem. **3: faixa sem estrutura**: nada a corrigir. **4: curva em ∩**: a relação não é linear; acrescentar termos polinomiais ou transformar x. (b) **Pode**: é um ponto de alta alavancagem. Longe de $\bar{x}$, ele puxa a reta para si com tanta eficácia que o próprio resíduo fica pequeno. A regra pega outliers, não alavancagem. Investigue (ajuste com e sem ele); não apague em silêncio.

× **Cuidado:** confiar num R^2 alto sem olhar os resíduos.

5. (a) **X é 3×2: não é quadrada e não tem inversa.** Também não há solução exata: os pontos $(0, 1)$ e $(1, 3)$ exigem $w_0 = 1$ e $w_1 = 2$, que preveem 5 para $x = 2$, não 2. (b) $X^\top X = \begin{pmatrix} 3 & 3 \\ 3 & 5 \end{pmatrix}$, $X^\top y = \begin{pmatrix} 6 \\ 7 \end{pmatrix}$; de $3w_0 + 3w_1 = 6$ e $3w_0 + 5w_1 = 7$ sai $w_1 = 0{,}5$ **e** $w_0 = 1{,}5$. (c) $\hat{y} = 1{,}5;\ 2;\ 2{,}5$ e $e = -0{,}5;\ 1;\ -0{,}5$; $X^\top e = (0;\ 0 + 1 - 1) = (0; 0)$. $\|e\|^2 = 1{,}5$, menor que os 2 da reta $w = (1, 1)$ (resíduos 0, 1, -1). (d) **Zero nas duas coordenadas**: $-\frac{2}{3}\sum e_i = 0$ e $-\frac{2}{3}\sum e_i x_i = 0$. A forma fechada é exatamente o ponto em que o gradiente se anula.

× **Cuidado:** tentar "inverter X". Inverte-se (em teoria) $X^\top X$, que é quadrada; as bibliotecas resolvem o sistema sem inversa nenhuma.

6. (a) $\hat{w}_1 = r\, s_2/s_1 = 0{,}6$ (os desvios são iguais). (b) Piores: $6 + 0{,}6\,(3{,}4 - 6) =$ **4,44**; melhores: $6 + 0{,}6\,(8{,}6 - 6) =$ **7,56**. Observados: 4,9 e 7,6. (c) Da subida de 1,5 ponto, **1,04 viria sem reforço nenhum**; sobram uns 0,46 (e mesmo esse número é uma estimativa). Os melhores caíram o previsto (7,56 contra 7,6): **ninguém se acomodou**. (d) Uma força teria um sentido; aqui o achatamento aparece também "de trás para a frente" e a prova 2 não pode puxar a prova 1. É aritmética: uma nota extrema é sinal mais sorte e a sorte não se repete. (e) **Sortear**, entre os alunos de nota baixa, quem faz o reforço e comparar com os que não fizeram (grupo de controle).

× **Cuidado:** atribuir ao tratamento a melhora de um grupo escolhido justamente por ter sido extremo.

R2 Gradiente descendente e regularização

1. (a) Em $w = 0$ o resíduo é o próprio y: $J = (4+1+16+25)/4 = \mathbf{11{,}5}$; $\sum e_i = 12$ e $\sum e_i x_i = 0+1+8+15 = 24$, logo $\partial J/\partial w_0 = -\frac{2}{4}\cdot 12 = -6$ **e** $\partial J/\partial w_1 = -\frac{2}{4}\cdot 24 = -12$. (b) $w_0 = 0 - 0{,}05\,(-6) = 0{,}3$ e $w_1 = 0 - 0{,}05\,(-12) = 0{,}6$; previsões 0,3; 0,9; 1,5; 2,1; resíduos 1,7; 0,1; 2,5; 2,9; $J = 17{,}56/4 = 4{,}39$. (c) $\sum e_i = 7{,}2$ e $\sum e_i x_i = 13{,}8$: **gradiente** $(-3{,}6;\ -6{,}9)$, ainda negativo nas duas coordenadas e menor. O próximo passo sobe os dois pesos de novo, um pouco menos. (d) $\bar{x} = 1{,}5$, $\bar{y} = 3$, $\hat{w}_1 = 6/5 = 1{,}2$ e $\hat{w}_0 = 3 - 1{,}2\cdot 1{,}5 = 1{,}2$. **Lá o gradiente é zero**: o gradiente descendente chega, por outro caminho, à mesma resposta das equações normais.

× **Cuidado:** somar o gradiente em vez de subtraí-lo ($w + \eta\nabla J$ sobe a ladeira) ou esquecer o $\frac{2}{n}$.

2. (a) $w - \eta\cdot 2w = (1-2\eta)\,w$. (b) $\eta = 0{,}1$: fator 0,8; 3,2; 2,56; 2,048: **converge devagar**. $\eta = 0{,}5$: fator 0; 0; 0; 0: **acerta de uma vez**. $\eta = 0{,}75$: fator $-0{,}5$; -2; 1; $-0{,}5$: **oscila e converge**. $\eta = 1$: fator -1; -4; 4; -4: **fica batendo entre** ± 4. $\eta = 1{,}25$: fator $-1{,}5$; -6; 9; $-13{,}5$: **diverge**. Diverge sempre que $|1-2\eta| > 1$, ou seja, $\eta > 1$.

× **Cuidado:** achar que um η maior é sempre mais rápido. Aqui isso vale só até 0,5.

3. (a) $e = (1,2,3)$: $\partial J/\partial w_0 = -\frac{2}{3}\cdot 6 = -4$ e $\partial J/\partial w_1 = -\frac{2}{3}\cdot 68 \approx -45{,}3$, **cerca de 11 vezes maior**: cada resíduo entra multiplicado por $x \approx 11$. Um η que não faça w_1 pular longe deixa w_0 andando a passos minúsculos: o vale é estreito e o caminho ziguezagueia. (b) $x' = (-1,0,1)$, $\sum e_i x'_i = 2$ e $\partial J/\partial w_1 = -\frac{4}{3} \approx -1{,}33$, da ordem de $\partial J/\partial w_0$. (c) $\hat{y} = 2 + (x-11) = x - 9$: **a mesma reta** ($w_0 = -9$, $w_1 = 1$). O centramento mudou o formato da paisagem (o vale vira tigela) e, com ele, o caminho até o mínimo; não mudou o mínimo. (d) O gradiente de um ponto é uma estimativa barata e ruidosa do gradiente médio: cada passo custa pouco, mas o caminho treme em volta do mínimo.

4. (a) **Grau 1: underfitting** (treino e validação altos e parecidos); **grau 2: adequado**; **grau 30: overfitting** (o menor treino e uma validação de 10,43). Escolho o grau 2. (b) $52{,}16/5 \approx 10{,}43$. **O desvio entre as dobras (13,46)**: o grau 30 vai bem em três dobras e explode numa (37,29). Overfitting não é errar sempre; é errar de forma imprevisível. (c) O ruído de desvio 3 não depende de x: nenhum modelo o prevê. É um piso; o grau 2 já chega perto dele. (d) São 11 parâmetros para 10 pontos: a curva passa por todos eles. **Não é bom sinal**: o modelo memorizou, sem aprender a forma da curva. Complexidade é relativa à quantidade de dados.

× **Cuidado:** escolher o grau pela coluna de treino: ela escolheria justamente o grau 30.

5. (a) Ridge (divide por 1,6): **1,0; 0,625;** $-0{,}31$**; 0,125**. Lasso (tira 0,3 do tamanho e corta em zero): **1,3; 0,7;** $-0{,}2$**; 0**. (b) Só o **0,2** (todo $|\hat{w}| \le \alpha/2 = 0{,}3$ some). Para zerar os quatro, $\alpha/2 \ge 1{,}6$: $\alpha \ge 3{,}2$. O ridge não zera nenhum: dividir nunca dá zero. (c) A solução é onde a elipse do erro, crescendo a partir do ótimo do OLS, toca a região pela primeira vez. O losango tem quinas **exatamente sobre os eixos**, onde um peso é zero, e uma curva que cresce em direção a uma região pontuda costuma tocar uma quina. O círculo não tem quinas: tocar um eixo é coincidência.

× **Cuidado:** dizer que o lasso "encolhe mais": ele subtrai uma quantidade fixa, então o peso 1,0 encolhe menos no lasso (0,7) do que no ridge (0,625).

6. (a) **Multicolinearidade**: os dois termômetros dizem quase a mesma coisa, então muitas divisões do crédito com a mesma soma (2,2) dão quase as mesmas previsões. Os dados fixam a soma, não a divisão; cada coeficiente sozinho não quer dizer nada. (b) **O ridge divide o peso por igual** (perto de 1,1 e 1,1) e estabiliza os dois. **O lasso tende a ficar com um e zerar o outro** (qual deles fica é quase arbitrário). (c) $\alpha = 1$ (validação 12,3, a menor). Overfitting em α pequeno (treino 10,1 e validação 14,8); underfitting em α grande (30,2 e 30,5: altos e parecidos). (d) O mesmo efeito custa $2{,}2^2 = 4{,}84$ na coluna em °C e $1{,}22^2 \approx 1{,}49$ na coluna em °F: **a penalidade empurra o peso para o termômetro em °F** só por causa da unidade. Padronize as colunas, dentro do pipeline, antes de regularizar.

× **Cuidado:** regularizar atributos em escalas diferentes: a penalidade passa a depender da unidade, não da informação.

C1 Regressão logística

1. (a) **Probabilidades ficam entre 0 e 1**; $-0{,}23$ e $1{,}18$ não dizem "quão seguro" o modelo está. (b) Os mínimos quadrados querem a reta perto de *todos* os pontos, inclusive os distantes: **a inclinação achata e o cruzamento de 0,5 anda de 6,67 h para 8,66 h**. Quem passou com 7,5 a 8,5 horas vira "reprovado". (c) Fica abaixo de 0,5 (cerca de 0,48): **"reprovado", um erro**. (d) Para quem estudou 25 a 45 horas a curva já está na parte plana, com $\hat{p} \approx 1$: **não há o que corrigir e esses pontos quase não puxam o ajuste**.

× **Cuidado:** achar que pontos fáceis e bem rotulados nunca atrapalham. Atrapalham a reta, que é puxada por eles.

2. (a) $z = -3;\ -1{,}5;\ 0;\ 1{,}5;\ 3$ e $\hat{p} = 0{,}047;\ 0{,}182;\ 0{,}5;\ 0{,}818;\ 0{,}953$ (por exemplo, $1/(1+4{,}48) = 0{,}182$ e $1/(1+1/4{,}48) = 4{,}48/5{,}48 = 0{,}818$). A fronteira é $z = 0$: **6 horas** (9/1,5). Com 5 horas, **reprovado**; com 7, **aprovado**. (b) $e^z = 2{,}718;\ 1;\ 0{,}368$, soma 4,086: $\hat{p} = 0{,}665;\ 0{,}245;\ 0{,}090$. Somar 5 a todos não muda nada: numerador e denominador ganham o mesmo fator e^5.

× **Cuidado:** trocar o sinal do expoente: $\hat{p} = 1/(1+e^{-z})$, então z negativo dá $\hat{p}$ abaixo de 0,5.

3. (a) **4; 1; 0,25**; chance 3 é $p = 3/4 = 0{,}75$. (b) A chance é e^z: **0,223 com 5 h, 1 com 6 h e 4,48 com 7 h**; cada hora **multiplica a chance por** $e^{1{,}5} \approx 4{,}48$. Em probabilidade, de 5 para 6 horas são **+0,318**; de 7 para 8, só **+0,135**. (c) **0,068 vez**; $1/0{,}068 \approx 14{,}7$ **vezes**. (d) Probabilidades: $0{,}601/0{,}093 \approx 6{,}5$. Chances: $(0{,}601/0{,}399)/(0{,}093/0{,}907) \approx 1{,}51/0{,}103 \approx 14{,}7$, o mesmo $e^{2{,}685}$. **A razão de chances é a mesma para todo passageiro**; a de probabilidades depende de onde cada um parte.

× **Cuidado:** ler e^w como razão de probabilidades: "a probabilidade de um homem é 0,068 da de uma mulher" está errado.

4. (a) Probabilidades dadas à verdade: 0,8; 0,9; 0,3; 0,4. Custos: 0,223; 0,105; 1,204; 0,916; soma 2,448; **média** $\approx 0{,}612$. É melhor que os $\ln 2 \approx 0{,}693$ de dizer sempre 0,5, mas por pouco. (b) **O exemplo 3**: era da classe 1 e recebeu só 0,3. Sozinho, é metade da perda. (c) Erro quadrático 0,980; **entropia cruzada 4,605**. Forças: entropia cruzada $0{,}01 - 1 = -0{,}99$; erro quadrático $2(-0{,}99)(0{,}01)(0{,}99) \approx -0{,}0196$, **cerca de 50 vezes mais fraco**. No erro quadrático, quanto pior o erro, mais fraco o empurrão, porque a sigmoide é plana ali.

× **Cuidado:** usar $\hat{p}$ em vez de $1 - \hat{p}$ nos exemplos da classe 0.

5. (a) $2x_1 + x_2 = 4$: **a reta por** $(0, 4)$ **e** $(2, 0)$; a classe 1 fica do lado de cima e à direita (onde $z > 0$). (b) $(3,1)$: $z = 3$, $\hat{p} = 20{,}1/21{,}1 \approx 0{,}953$. $(1,1)$: $z = -1$, $\hat{p} = 1/(1+2{,}718) \approx 0{,}269$. (c) $z = \ln 9$: $2x_1 + x_2 \approx 6{,}2$**, uma reta paralela à fronteira**, por $(0;\ 6{,}2)$ e $(3{,}1;\ 0)$. (d) Toda reta deixa pedaços do anel dos dois lados. Com os quadrados, $z = 0$ quando $16{,}83\,x_1^2 + 16{,}00\,x_2^2 = 9{,}70$: **quase um círculo de raio** $\sqrt{9{,}70/16{,}4} \approx 0{,}77$, entre o disco e o anel; dentro dele $z > 0$, classe 1. O modelo continua linear nos pesos; a curva só aparece nas medidas originais.

6. (a) **0,3**: sinaliza A a F; 0 doentes perdidos, 2 alarmes falsos (C e E), acurácia $6/8 = 0{,}75$. **0,5**: A, B e C; 2 perdidos (D e F), 1 alarme, acurácia $5/8 = 0{,}625$. **0,7**: A e B; 2 perdidos, 0 alarme, acurácia 0,75. (b) Custos $5 \times$ perdidos $+$ alarmes: 2, 11 e 10. **Limiar 0,3**. (c) A acurácia conta os dois erros como iguais: **0,3 e 0,7 têm a mesma acurácia e custos 2 e 10**. (d) **Não**: o limiar é aplicado depois do modelo e as probabilidades não mudam. Recomendação: "usar 0,3: nos oito casos, acha os quatro doentes ao preço de dois alarmes falsos; confirmar em mais pacientes antes de adotar".

× **Cuidado:** deixar o 0,5 do `predict` decidir sem perguntar quanto custa cada erro.

C2 k-vizinhos mais próximos (k-NN)

1. (a) d^2: A 2, B 1, C 9, D 5, E 8, F 13, G 10. Ordem: B, A, D, E, C, G, F. (b) $k = 1$: só B vota, **gostou (1,0)**. $k = 3$: B e D sim, A não, **gostou** $(2/3 \approx 0{,}67)$. $k = 5$: entram E e C, os dois "não": **não gostou** $(2/5 = 0{,}4)$. $k = 7$: todos votam, **gostou** $(4/7 \approx 0{,}57)$. (c) Com $k = n = 7$, toda consulta recebe os mesmos sete vizinhos: **a resposta é sempre "gostou" (a classe majoritária)**, seja quem for o assinante.

× **Cuidado:** achar que um k maior é sempre mais seguro: no limite, o modelo deixa de olhar para o ponto novo.

2. (a) P: $\sqrt{300^2 + 28^2} \approx 301{,}3$; Q: $\sqrt{1.200^2 + 1^2} \approx 1.200{,}0$. **O 1-NN é P: "não compra"**, embora P tenha 28 anos a mais. (b) P: $\sqrt{0{,}15^2 + 2^2} \approx 2{,}01$; Q: $\sqrt{0{,}6^2 + 0{,}07^2} \approx 0{,}60$. **Agora o vizinho é Q: "compra"**. (c) P: $\sqrt{0{,}3^2 + 28^2} \approx 28{,}0$; Q: $\sqrt{1{,}2^2 + 1^2} \approx 1{,}56$: **Q**. A resposta do k-NN bruto muda com a unidade em que alguém anotou a coluna: ele não está medindo semelhança. (d) **Só nas linhas de treino**, dentro de um pipeline, e aplicadas sem reajuste à validação e ao teste.

3. (a) **Com** $k = 1$**, o 100% de treino é garantido por construção**: cada linha de treino é a sua própria vizinha, a distância 0. Não diz nada sobre dados novos. (b) **Escolher** k **pelo teste** faz do teste uma validação: o 0,900 fica otimista. E uma linha vale 1 ponto: $k = 5$, 15 e 45 diferem por uma linha, puro ruído. (c) **O escalonador foi ajustado em todas as linhas antes da divisão**: médias e desvios já viram o

teste (vazamento). (d) Dividir em treino, validação e teste; ajustar `make_pipeline(StandardScaler(), KNeighborsClassifier(k))` no treino; escolher k na validação; reajustar em treino + validação e medir o teste uma vez.

× **Cuidado:** ajustar o escalonador antes de dividir: o pipeline existe justamente para impedir isso.

4. (a) $k = 73$ (validação 0,910). Overfitting em $k = 1$ (treino 1,000 e validação 0,850); underfitting em $k = 451$ (0,772 e 0,765: baixos e próximos). (b) $0{,}015 \times 200 = 3$ **linhas**: não convence. De $k = 15$ a 73 é um platô e qualquer valor dele é defensável. (c) O 73 foi escolhido *porque* foi o melhor naquelas 200 linhas, então parte dos 0,910 é sorte. **O teste, usado uma vez, é a estimativa honesta**. Voltar e trocar o k agora faria do teste uma segunda validação.

5. (a) Diferenças 3, 4 e 12: **Manhattan 19, euclidiana** $\sqrt{9+16+144} = 13$**, Chebyshev 12**. (b) Manhattan: U 6, V 5, **V**. Euclidiana: U ≈ 4,24, V 5, **U**. Chebyshev: U 3, V 5, **U**. A régua escolhe o vizinho. (c) Pesos $1/d$: B 1; A 0,707; D 0,447; E 0,354; C 0,333. $k = 3$: gostou $1 + 0{,}447 = 1{,}447$ contra 0,707, $P \approx 0{,}67$**, gostou** (não muda). $k = 5$: gostou 1,447 contra $0{,}707 + 0{,}354 + 0{,}333 = 1{,}394$, $P \approx 0{,}51$**, gostou**: muda, porque os "não" de E e C estão longe e contam menos.

× **Cuidado:** somar pesos $1/d^2$ quando a regra é $1/d$: a tabela do exercício 1 tem d^2, tire a raiz.

6. (a) Em 1.000 dimensões o mais próximo está só 12% mais perto que o mais distante. **Todos ficam quase igualmente longe**: os k "mais próximos" mal se destacam do resto. (b) A distância soma uma parcela por coluna, **todas com o mesmo peso**: com 1.000 colunas de ruído, quase toda a distância vem do ruído. A regressão logística aprende um peso por coluna e pode deixar o do ruído perto de zero (aqui ajudada pela penalidade L2 padrão do scikit-learn). **Padronizar não resolve**: dá a cada coluna o mesmo peso, inclusive às inúteis. Remédio: escolher os atributos ou comprimir com PCA (análise de componentes principais). (c) **(i) k-NN**: fronteira curva com poucas colunas e muitas linhas (nos círculos, 0,993 contra 0,493). **(ii) Regressão logística**: muitas colunas irrelevantes e previsão rápida (o k-NN calcula uma distância por linha guardada a cada previsão). **(iii) k-NN**: "estes são os casos mais parecidos com o seu".

× **Cuidado:** confiar no escalonador para tirar o peso de colunas inúteis.

C3 Métricas de classificação

1. (a) revocação; (b) VPN; (c) precisão; (d) especificidade; (e) acurácia. (b) lê a coluna dos liberados; (d) a linha dos saudáveis.

× **Cuidado:** (b) não é a especificidade: "dos saudáveis, quantos libero" é a linha; "dos que libero, quantos estão bem" é a coluna.

2. (a) TP: P01–P04, P06, P07, P09; FP: P05, P08; FN: P10, P12, P15; TN: os outros 8. **Matriz** `[[7, 3], [2, 8]]`: TP 7 e FN 3 na linha dos doentes; FP 2 e TN 8 na dos saudáveis. (b) Sem `labels`, a ordem é 0, 1 e a matriz é `[[TN, FP], [FN, TP]]`: **8 é o TN**, 2 o FP, 3 o FN e 7 o TP. Com `labels=[1, 0]` sai o layout padrão.

× **Cuidado:** ler a matriz do scikit-learn sem `labels` como se fosse o layout padrão troca TP com TN e FP com FN em todas as contas.

3. (a) TP 15, FN 5, FP 10, TN 170. (b) Acurácia $185/200 = 0{,}925$ (matriz inteira); precisão $15/25 = 0{,}600$ (coluna dos marcados); revocação $15/20 = 0{,}750$ (linha das fraudes); especificidade $170/180 \approx 0{,}944$ e FPR ≈ 0,056 (linha das legítimas); VPN $170/175 \approx 0{,}971$ (coluna dos não marcados); F1 $= 30/45 \approx 0{,}667$. (c) Nunca marcar acerta $180/200 = 0{,}900$: **o modelo ganha só 2,5 pontos da baseline**, embora encontre 3 de cada 4 fraudes. A acurácia quase não enxerga o que importa aqui.

4. Só o P01: precisão 1,0, revocação 0,1, média 0,550, F1 ≈ 0,182. Limiar 0,5: 0,778; 0,700; 0,739; 0,737. Os 20: 0,5; 1,0; 0,750; 0,667. (b) **A média põe "marcar todo mundo" na frente; o F1 põe o limiar 0,5**. A harmônica só fica alta quando os dois lados ficam altos: não dá para comprar um bom F1 abandonando um deles. (c) $F_1 = 2 \cdot 0{,}02/1{,}02 \approx 0{,}04$.

× **Cuidado:** achar que o F1 é o mínimo dos dois: ele é puxado para o menor, mas não é igual a ele.

5. 0,3: 13 marcados, TP 9, FP 4, FN 1, TN 6, precisão $9/13 \approx 0{,}692$, revocação 0,900, F1 $18/23 \approx 0{,}783$. 0,5: 9; 7, 2, 3, 8; 0,778; 0,700; $14/19 \approx 0{,}737$. 0,7: 6; 5, 1, 5, 9; $5/6 \approx 0{,}833$; 0,500; $10/16 = 0{,}625$. (a) A precisão sobe e a revocação cai, **com o mesmo modelo**: só a regra de decisão mudou. (b) 0,750; 0,857; 0,833. Subir o limiar só tira pacientes do grupo marcado, então os TP só diminuem. A precisão depende de quem sai: tirar o P08 (saudável) a sobe; tirar o P07 (doente) a derruba. (c) **0,3**: só 1 escape, ao preço de 2 alarmes a mais que em 0,5. Na prática, a escolha seria feita num conjunto de validação.

6. (a) 50%: 5.000 doentes, TP 3.500, FP 1.000, precisão 0,778, VPN 0,727, acurácia 0,750. 10%: 1.000; 700; 1.800; 0,280; 0,960; 0,790. 1%: 100; 70; 1.980; **0,034**; 0,996; 0,799. A acurácia balanceada é 0,750 nas três. Com 1%, são $1.980/70 \approx 28$ alarmes falsos por

doente encontrado. (b) **Revocação, especificidade e acurácia balanceada não mudam**: as duas primeiras são proporções dentro de uma classe (uma linha só) e a terceira é a média delas. Precisão, VPN e acurácia misturam as duas linhas; quando a dos saudáveis fica 99 vezes maior, até uma FPR de 0,20 afoga os TP (e a acurácia *sobe*, puxada pelos TN). (c) Que 0,778 vale para 50% de doentes; no rastreamento com 1% ela seria 0,034. Reporte a precisão junto com a prevalência dos dados em que foi medida. (d) $F_2 = 35/49 \approx 0{,}714$, abaixo do F1: o F_2 pesa mais a revocação, que é o lado fraco deste modelo (0,700 contra 0,778).

× **Cuidado:** levar uma precisão (ou um F1) medida num conjunto para outro com prevalência diferente.

C4 ROC, PR e dados desbalanceados

1. (a) Cada doente sobe 0,1 e cada saudável anda 0,1 para a direita: quatro degraus para cima (P01–P04), direita (P05), dois para cima, direita (P08), dois para cima, direita (P11), cima, duas vezes direita (P13, P14), cima (P15) e cinco vezes direita até (1, 1). (b) O limiar 0,5 marca até o P09: **(FPR; TPR) = (0,2; 0,7)**. (c) O perfeito sobe direto até (0, 1) e depois vai para a direita; o que sorteia escores fica na diagonal. (d) São 21 pontos (os 20 limiares mais o canto (0, 0)). O `roc_curve` descarta os pontos no meio de um trecho reto (como P02 e P03, entre P01 e P04): a escada é a mesma.

2. (a) 4, 6, 8, 9, 9, 10, 10, 10, 10, 10: **86 de $10 \times 10 = 100$ pares, AUC 0,86**. (b) No passo para a direita de cada saudável, a altura da curva é a fração dos doentes já marcados, ou seja, o número da tabela dividido por 10. Cada retângulo tem largura 1/10: a área é a soma da tabela dividida por 100. (c) 0,86; 0,86; **0,14**: as duas primeiras preservam a ordem; a terceira a inverte. (d) Triângulo até (0,2; 0,7) mais trapézio até (1, 1): $0{,}2 \cdot 0{,}7/2 + 0{,}8 \cdot (0{,}7 + 1)/2 = 0{,}07 + 0{,}68 = 0{,}75$, que é $(\text{TPR} + 1 - \text{FPR})/2$, a média do recall e da especificidade: **a acurácia balanceada em 0,5**, não a AUC do modelo.

× **Cuidado:** passar o `predict` (rótulos) ao `roc_auc_score`: não dá erro, só um número errado.

3. (a) Posições 1, 2, 3, 4, 6, 7, 9, 10, 12, 15; precisões 1, 1, 1, 1, 5/6, 6/7, 7/9, 8/10, 9/12, 10/15 (0,833; 0,857; 0,778; 0,800; 0,750; 0,667). (b) **AP $= 8{,}685/10 = 0{,}8685$**. Ao acaso, a precisão fica na prevalência em qualquer limiar: AP ≈ 0,5. (c) A AP: ligar dois pontos PR por uma reta descreve classificadores que não existem e o trapézio pode ser otimista; a AP soma retângulos.

4. (a) $\pi = 0{,}5$: $0{,}40/(0{,}40 + 0{,}06) \approx 0{,}87$; $\pi = 0{,}2$: $0{,}16/(0{,}16 + 0{,}096) = 0{,}625$; $\pi = 0{,}01$: $0{,}008/(0{,}008 + 0{,}119) \approx 0{,}06$, **uns 15 alarmes falsos por doente**. (b) **O ponto ROC não muda**: TPR divide pelos doentes e FPR pelos saudáveis, cada um dentro da sua classe. A precisão divide pelos marcados, que misturam as classes na proporção da população: a PR afunda. (c) O de 1%: 0,25 é **25 vezes** a AP do acaso (0,01); com 50% de positivos, é metade do acaso. (d) A PR (e a AP), lida contra a prevalência: é ela que mostra o custo dos alarmes falsos que o radiologista paga.

× **Cuidado:** ler a AP como se 0,5 fosse o acaso, como na AUC: o acaso da AP é a prevalência.

5. 0,50: 7, 2, 3; 0,778; 0,7; 0,737; 0,5; 32. 0,45: 8, 2, 2; 0,800; 0,8; 0,800; 0,6; 22. 0,35: 9, 3, 1; 0,750; 0,9; 0,818; 0,6; 13. 0,20: 10, 5, 0; 0,667; 1,0; 0,800; 0,5; 5. 0,10: 10, 7, 0; 0,588; 1,0; 0,741; 0,3; 7. (a) **F1: 0,35. J: empate 0,45 e 0,35. Recall $\geq 0{,}9$: 0,35. Custo: 0,20.** (b) $t^* = 1/(1 + 10) \approx 0{,}09$: o 0,10 custa 7, perto do mínimo mas não nele, porque os escores não são perfeitamente calibrados e 20 pacientes são poucos. (c) A ordenação é a mesma; cada critério declara um custo diferente para os erros. Quem decide é a aplicação, antes de olhar o teste.

× **Cuidado:** escolher o limiar olhando o conjunto de teste: ele deixa de ser honesto.

6. (a) $w_0 = 12.000/(2 \cdot 11.767) \approx 0{,}510$ e $w_1 = 12.000/(2 \cdot 233) \approx 25{,}75$: **cada fraude vale umas 50 transações legítimas** ($11.767/233 \approx 50{,}5$). (b) Os pesos **moveram o ponto de operação**, não a ordenação: a AUC ficou igual. Cortar o modelo sem pesos num limiar mais baixo chega ao mesmo ponto, sem retreinar e podendo mudar depois. (c) **Vazamento:** cada fraude virou umas 50 cópias antes de dividir, então quase toda fraude do "teste" tem cópias idênticas no treino e o 5-NN as acha. **Mundo errado:** o "teste" tem 50% de fraude, não 2%, e a precisão depende dessa proporção. **Certo:** dividir primeiro; sobreamostrar só o treino, dentro de um pipeline do `imblearn`; avaliar na validação e no teste com a prevalência real; escolher o limiar na validação.

× **Cuidado:** achar que reamostrar ensina algo novo ao modelo: a AUC e a AP quase não mudam; o que muda é onde o 0,5 corta.

C5 Máquinas de vetores de suporte (SVM)

1. $\|w\| = 5$. A: $f = 8$, $m = 8$, distância 1,6, hinge 0. B: -5, -5, 1,0, 6. C: 0, 0, 0, 1. D: 2, 2, 0,4, 0. E: -1, 1, 0,2, 0. **Rua** $2/5 = 0{,}4$. (b) **B está do lado errado** ($m < 0$); C está sobre a fronteira; E está exatamente sobre a borda ($m = 1$). A hinge ignora A, D e E ($m \geq 1$).

× **Cuidado:** usar $f(x)$ como distância: é preciso dividir por $\|w\|$.

2. (a) Os negativos mais próximos são B e C ($x_1 + x_2 = 1$); o positivo, D ($x_1 + x_2 = 4$). A fronteira fica no meio: $x_1 + x_2 = 2{,}5$. (b) $4a + b = 1$ e $a + b = -1$: $a = \frac{2}{3}$, $b = -\frac{5}{3}$. (c) $\|w\| = \frac{2}{3}\sqrt{2} \approx 0{,}943$: **rua** $\approx 2{,}121 = 3/\sqrt{2}$, a distância entre as retas $x_1 + x_2 = 1$ e $= 4$. Bordas: $x_1 + x_2 = 1$ e $x_1 + x_2 = 4$. (d) m: A, E e F $\frac{5}{3}$; B, C e D 1. **Vetores de suporte: B, C e D.**

3. (a) $-\frac{2}{9}(1,0) - \frac{2}{9}(0,1) + \frac{4}{9}(2,2) = (\frac{2}{3}, \frac{2}{3})$. (b) **Nada muda**: E tem $m > 1$ e $\alpha = 0$; não entra em w nem em b. (c) Os positivos mais próximos passam a ser E e F ($x_1 + x_2 = 5$): fronteira $x_1 + x_2 = 3$, $5a + b = 1$ e $a + b = -1$ dão $w = (\frac{1}{2}, \frac{1}{2})$, $b = -\frac{3}{2}$, **rua** $4/\sqrt{2} \approx 2{,}828$. (d) Sim, um pouco: na log loss todo ponto tem peso não nulo.

4. m: $\frac{7}{3} \approx 2{,}333$; 1; $\frac{1}{3}$; $-\frac{2}{3}$. Hinge: 0; 0; **0,667**; 1,667. Perda 0-1: 0; 0; 0; 1. (a) (1,5; 1,5): está do lado certo, mas dentro da rua ($0 < m < 1$). (b) A log loss (0,093): ela nunca é zero, então todo ponto puxa a solução. A hinge é exatamente zero para $m \geq 1$: esses pontos têm $\alpha = 0$ e somem, daí a solução depender só dos vetores de suporte.

× **Cuidado:** achar que a hinge só cobra erros: ela cobra também os acertos dentro da rua.

5. (a) Violar a margem fica caro: o otimizador aceita um $\|w\|$ maior (rua mais estreita) para tirar pontos de dentro dela; sobram menos sobre a borda ou dentro. (b) $C = 1$ (CV 0,886). Em $C = 0{,}01$ treino e CV são os mais baixos (0,886 e 0,863): underfitting. Em $C = 100$ o treino segue subindo (0,962) e a CV cai (0,852), vão de 0,110 contra 0,061 em $C = 1$: overfitting. (c) O treino sobe sem parar com o C: escolher por ele leva sempre ao maior C.

× **Cuidado:** ler o número de vetores de suporte como qualidade: ele mede quão regularizado está o modelo.

6. (a) Nenhum limiar acerta mais que 4 de 5 (0,8). Em (x, x^2): positivos em $x^2 = 4$, negativos em $x^2 \leq 1$; a fronteira de margem máxima é $x^2 = 2{,}5$ ($w = (0, \frac{2}{3})$, $b = -\frac{5}{3}$), que em x vira **dois limiares,** $\pm\sqrt{2{,}5} \approx \pm 1{,}58$. (b) $\phi(u) = (4, 2\sqrt{2}, 1)$, $\phi(v) = (1, \sqrt{2}, 1)$: $4 + 4 + 1 = 9$; $u \cdot v = 3$ e $3^2 = 9$. Com 34 atributos e grau 3, o mapa tem 7.770 colunas; o kernel custa um produto escalar de 34 termos. (c) 0,607 e 0,011; com $\gamma = 2$, 0,135: só pontos muito próximos contam e a fronteira contorna pontos isolados. (d) **Overfitting**: cada vetor de suporte vira uma ilha; um ponto novo fica longe de todos, recebe $f \approx b$ e cai quase sempre na classe maior.

C6 Validação cruzada e vazamento de dados

1. (a) Cada fold treina em 364 linhas e valida em 91; **5 ajustes, cada linha validada exatamente uma vez.** (b) São descartados: a CV avaliou a receita. Vai para produção a mesma receita reajustada uma vez nas 455 linhas. (c) $13 \times 5 = 65$ ajustes mais 1 reajuste: **66.** A repetida faz $5 \times 10 = 50$ ajustes.

× **Cuidado:** ficar com o modelo do melhor fold: é escolher pelo ruído da divisão.

2. **(a)** `StratifiedKFold`**; (b)** `GroupKFold` **(grupo = paciente); (c)** `TimeSeriesSplit`**; (d)** `KFold(shuffle=True)`**; (e) nenhum resolve**: a coluna é registrada depois do momento da previsão; o vazamento está nas colunas, não na divisão, e a coluna tem de sair. Sem embaralhar, o arquivo é cortado em blocos: **8, 0, 0, 0, 0**; em quatro folds o recall nem é definido.

3. (a) 4, 1, 0, 2, 1 tumores; um tumor vale $1/34 \approx 0{,}029$. (b) Média 0,953; desvio $\sqrt{0{,}00804/5} \approx 0{,}040$: **recall** $0{,}953 \pm 0{,}040$. (O `pandas` divide por 4 e dá 0,045.) (c) Não: é o mesmo modelo e os mesmos dados; a diferença é quais tumores difíceis caíram em cada fold, a sorte da divisão. Com 34 positivos por fold, um tumor move 3 pontos. (d) Treino 0,971 contra validação 0,953: menos de 2 pontos de gap, **sem overfitting relevante**; os dois são altos, então também não há underfitting.

× **Cuidado:** reportar só a média: $0{,}95 \pm 0{,}005$ e $0{,}95 \pm 0{,}05$ são afirmações muito diferentes.

4. (a) Em A o seletor aprende com as 100 linhas, inclusive os rótulos das que depois validam; em B, em cada fold, só com as 80 de treino. **O honesto é 0,53** (o acaso). (b) Entre 5.000 colunas, algumas se correlacionam com os rótulos por acaso, inclusive com os de validação; o seletor escolhe justamente essas e cada fold "confirma" um padrão que ajudou a escolher: é a maldição do vencedor escondida no pré-processamento. (c) Não prova. O scaler só aprende média e desvio das colunas (nunca olha o y) e 569 linhas fixam bem esses números; um passo que olha o y (seleção, *target encoding*, SMOTE) ou um conjunto pequeno pode vazar muito. Só se sabe medindo: com o passo no pipeline, a pergunta desaparece.

× **Cuidado:** concluir de um caso em que o vazamento foi pequeno que o procedimento é seguro.

5. (a) A gêmea está em algum dos outros folds, que são de treino: **cerca de 4/5.** Aí o 1-NN acha a gêmea e

copia o rótulo: acerta sempre. Senão, chuta: acerta metade das vezes. (b) $0{,}8 \cdot 1 + 0{,}2 \cdot 0{,}5 = \mathbf{0{,}90}$; sem gêmeas, uns 0,5 (a página mede 0,902 e 0,487). (c) `GroupKFold` com o par como grupo (mede 0,515) ou remover as duplicatas antes de dividir. Aparece depois de aumento de dados, sobreamostragem e junções de tabelas.

6. (a) Pelas médias ± desvio os intervalos quase se sobrepõem por inteiro. Diferenças: 0,018; 0,040; 0,040; 0,023; 0,061: média $+0{,}036 \pm 0{,}015$, a logística vence nos 5 folds: **vitória**. Os desvios grandes vêm dos folds (o fold 4 é difícil para os dois) e a subtração fold a fold cancela essa dificuldade comum. (b) $-0{,}023$; 0,010; 0,056; $-0{,}022$; $-0{,}018$: média $+0{,}001 \pm 0{,}030$ e o sinal troca: **empate**. Fica o mais simples: a logística de grau 2 (28 números e uma previsão por produto escalar, contra 8.946 linhas percorridas a cada previsão). (c) O 0,712 é a melhor de 35 estimativas ruidosas, medida nos mesmos folds que a escolheram: **otimista**. A CV aninhada faz a busca dentro de cada fold externo e mede o vencedor num fold que a busca não viu.

× **Cuidado:** comparar duas médias ± desvio em vez das diferenças fold a fold: a dificuldade comum dos folds esconde a vitória.

M1 Árvores de decisão

1. (8, 0): **Gini 0, entropia 0, erro 0**. (4, 4): **0,5; 1 bit; 0,5**. (6, 2): $1 - 0{,}75^2 - 0{,}25^2 = \mathbf{0{,}375}$; $0{,}75 \cdot 0{,}415 + 0{,}25 \cdot 2 \approx \mathbf{0{,}811}$; erro **0,25**. (2, 1, 1): $1 - 0{,}25 - 0{,}0625 - 0{,}0625 = \mathbf{0{,}625}$; $0{,}5 + 0{,}5 + 0{,}5 =$ **1,5 bits**; erro **0,5**. O 0,5 é o máximo com *duas* classes; com K classes equilibradas o máximo do Gini é $1 - 1/K$.

× **Cuidado:** esquecer o sinal de menos da entropia: $\log_2 p$ é negativo para $p < 1$ e a entropia nunca é negativa.

2. (a) 5 de 10 cancelaram: **Gini 0,5**. (b) `bill` ≤ 65: 3/0 e 2/5, Gini 0 e 0,408, ponderado $0{,}7 \cdot 0{,}408 = 0{,}286$. `bill` ≤ 85: 4/1 e 1/4, Gini 0,32 e 0,32, ponderado 0,32. `tenure` ≤ 10: 1/5 e 4/0, Gini 0,278 e 0, ponderado $0{,}6 \cdot 0{,}278 = 0{,}167$. **A árvore escolhe** `tenure` ≤ 10**, queda** $0{,}5 - 0{,}167 = 0{,}333$ (é também o melhor dos 18 cortes possíveis). (c) Qualquer limiar entre 60 e 70 separa as mesmas linhas; o CART testa um por intervalo, **o ponto médio**.

× **Cuidado:** somar os Gini dos filhos sem ponderar: um filho puro de 4 clientes não conta o mesmo que um misturado de 6.

3. (a) Uma reta vertical em `tenure` $= 10$ e, só à esquerda dela, uma horizontal em `bill` $= 60$. **3 folhas, acurácia de treino 10/10**. (b) **P: fica, P(cancelar) = 0** (cai na folha só com B); **Q: fica, 0**; **R: cancela, 1**. (c) Sobra o toco `tenure` ≤ 10: 2 folhas, acurácia 9/10; P e R recebem P(cancelar) $= 5/6 \approx 0{,}83$ e Q continua com 0. O segundo corte existe para isolar *um* cliente (B): padrão ou acaso? Com um exemplo só não há como saber.

× **Cuidado:** achar que a probabilidade 0 de P é confiável: ela vem de uma folha com um único cliente de treino.

4. (a) Nadadeira $\leq 206{,}5$ e bico $\leq 43{,}35$ de comprimento: Adelie. Nadadeira $\leq 206{,}5$ e bico mais comprido: Chinstrap. Nadadeira $> 206{,}5$ e bico $\leq 17{,}55$ de profundidade: Gentoo. Nadadeira $> 206{,}5$ e bico mais fundo: Chinstrap. (b) **X: Chinstrap,** $5/7 \approx 0{,}71$; **Y: Adelie,** $107/111 \approx 0{,}96$ (a profundidade de Y nem é perguntada). (c) $4 + 5 + 0 + 2 =$ **11 erros, acurácia de treino** $245/256 \approx 0{,}957$. A última folha: só 7 aves, uma mistura de Adelie e Chinstrap.

× **Cuidado:** ler a classe da folha como certeza: a folha só sabe as proporções dos exemplos de treino que caíram nela.

5. (a) Nó `bill` ≤ 60: $R(t) = 0{,}6 \cdot 0{,}278 = 0{,}167$, folhas puras ($R(T_t) = 0$), 2 folhas: $\alpha_{\text{ef}} = 0{,}167$. Raiz: $R = 0{,}5$, 3 folhas: $\alpha_{\text{ef}} = 0{,}5/2 = 0{,}25$. (b) **O nó** `bill` ≤ 60 **cai primeiro, em** $\alpha = 0{,}167$: some o corte que isolava B. Na árvore de 2 folhas ($R = 0{,}167$) a raiz sai em $\alpha = 0{,}5 - 0{,}167 = 0{,}333$. (c) $\alpha = 0{,}2$: 2 folhas; $\alpha = 0{,}4$: 1 folha. O α é um hiperparâmetro, **escolhido por validação cruzada**.

6. (a) Média 4,5; MSE $(12{,}25 + 6{,}25 + 2{,}25 + 2{,}25 + 6{,}25 + 12{,}25)/6 \approx \mathbf{6{,}917}$. (b) ≤ 45: 2,417; ≤ 60: esquerda (1, 2, 3) e direita (6, 7, 8), cada uma com MSE 2/3: **0,667**; ≤ 75: 2,417. **Vence área** ≤ 60, que remove $(6{,}917 - 0{,}667)/6{,}917 \approx 90\%$ do MSE. (c) **7; 7; 7** (mil R$). Nem uma árvore funda acerta: ela só devolve médias de aluguéis vistos, no máximo 8. Num alvo que sobe com o tempo, todo caso novo além da faixa recebe o valor da última folha.

× **Cuidado:** esperar que mais profundidade resolva a extrapolação: mais folhas só fazem uma escada mais fina dentro da faixa de treino.

M2 Random forest

1. (a) $0{,}5 \cdot 0{,}09 = 0{,}045$ mais $0{,}045/n$: **0,09; 0,0495; 0,04545; limite 0,045**. De 10 para 100 árvores quase nada muda: o piso domina. (b) $0{,}2 \cdot 0{,}12 + 0{,}8 \cdot 0{,}12/100 = 0{,}024 + 0{,}00096 \approx$ **0,025; limite 0,024**, cerca de metade do anterior. **Valeu**: o sorteio age em ρ, o termo que nenhuma quantidade de árvores corta.

× **Cuidado:** achar que mais árvores resolvem tudo: com $\rho = 0{,}5$, a média de infinitas árvores ainda tem metade da variância de uma.

2. (a) Linha 1: 2 vezes; 2: 1; 3: 3; 6: 1; 8: 1; **fora da sacola: 4, 5 e 7**. (b) $3/8 = 0{,}375$ contra $(7/8)^8 \approx 0{,}344$: perto, com 8 linhas o sorteio oscila muito. Com 1.000 linhas, $(1 - 1/1000)^{1000} \approx 0{,}368$ ficam de fora: **uns 632 distintos**, alguns repetidos.

3. (a) e1: S, S, L → S, certa. e2: L, L, S → L, certa. e3: L, L, S → **L, errada**. e4: L, certa. e5: S, certa. e6: S, S, L → **S, errada**. **Erro OOB** $2/6 \approx 0{,}33$. (b) Nenhum e-mail foi votado por uma árvore que treinou nele: para quem vota, o e-mail é dado novo. (c) **e4**: um único voto. Com 5 árvores, cada linha tem pouquíssimos votantes OOB; a "floresta" que avalia cada linha é minúscula e bem pior que a floresta inteira. Leia o OOB só com algumas dezenas de árvores.

× **Cuidado:** incluir na contagem os votos das árvores que treinaram no e-mail; aí a estimativa deixa de ser honesta.

4. (a) A variância da floresta é $\rho\sigma^2 + (1-\rho)\sigma^2/B$: com 200 árvores sobra quase só $\rho\sigma^2$. **A correlação cai de 0,304 para 0,198** e isso compensa árvores piores: o erro da floresta cai de 0,053 para 0,023. (b) Com 1 atributo por divisão a correlação ainda cai (0,133), mas cada árvore erra 37,5%: σ^2 cresce mais do que ρ cai. (c) **Não**: a floresta rasa estacionou em 8,7–8,9% (de 200 para 400 árvores a diferença é de uns 7 e-mails, ruído). O erro de uma árvore de profundidade 3 é viés e a média não remove viés.

× **Cuidado:** concluir que `max_features` menor é sempre melhor: há um ótimo, largo, numa pequena fração dos atributos.

5. (a) 3: $0{,}7^3 + 3 \cdot 0{,}7^2 \cdot 0{,}3 = 0{,}343 + 0{,}441 =$ **0,784**. 5: $0{,}7^5 + 5 \cdot 0{,}7^4 \cdot 0{,}3 + 10 \cdot 0{,}7^3 \cdot 0{,}3^2 \approx 0{,}168 + 0{,}360 + 0{,}309 =$ **0,837**. (b) Os 3 que copiam já são maioria: o júri decide sempre como o jornal, **0,70**, nada melhor que um jurado. (c) O "jornal" é o que todas as árvores compartilham: as mesmas linhas de treino e, sobretudo, um atributo forte pelo qual todas começariam. O **bootstrap** dá a cada árvore uma amostra diferente e o **sorteio de** `max_features` tira o atributo forte de muitos nós.

6. (a) No fundo das árvores os nós têm poucas linhas e uma coluna com 1.000 valores oferece centenas de limiares: algum sempre separa essas linhas por acaso. A MDI é medida *no treino*, onde esse ruído ajustado "reduz a impureza". **Confie na permutação, medida em dados separados**: lá `random_id` não faz falta ($0{,}002 \pm 0{,}007$) e `checking_status` é a primeira. (b) Com uma gêmea, embaralhar uma cópia deixa a outra fazer o trabalho: **a importância se divide entre as duas**. O erro: concluir que um atributo não importa quando ele só tem um substituto. Permute o grupo de colunas correlacionadas junto.

× **Cuidado:** ler a MDI como "o que importa para o problema": ela favorece colunas com muitos valores distintos.

M3 Gradient boosting

1. (a) O erro de um toco é **viés**: qualquer bootstrap leva mais ou menos ao mesmo corte, então a média de 200 tocos ainda é quase um degrau (0,572). O erro da árvore profunda é **variância** (treino 0, teste 0,509): cada uma persegue o ruído do seu bootstrap e a média cancela boa parte disso (0,369). (b) 0,25 é a variância do ruído: a parte de y que não depende de x, que nenhum modelo prevê.

× **Cuidado:** achar que mais tocos na floresta resolveriam: a média de cópias de um modelo enviesado continua enviesada.

2. (a) $F_0 = 25/5 = 5$; resíduos $-4, -3, -2, 1, 8$. (b) Médias e SSE: $x < 1{,}5$: -4 / 1, 74; $x < 2{,}5$: $-3{,}5$ / 2,33, 53,2; $x < 3{,}5$: -3 / 4,5, 26,5; $x < 4{,}5$: -2 / 8, 14. **Vence** $x < 4{,}5$: o toco gasta o corte no último ponto. $F_1 = 5 + 0{,}5 \cdot (-2) = 4$ nos quatro primeiros e $5 + 4 = 9$ no último: $F_1 = (4, 4, 4, 4, 9)$. (c) Resíduos $-3, -2, -1, 2, 4$; $x < 3{,}5$: folhas -2 / 3, SSE 4; $x < 4{,}5$: -1 / 4, SSE 14. Vence $x < 3{,}5$: $F_2 = (3; 3; 3; 5{,}5; 10{,}5)$. MSE de treino: **18,8; 6,8; 2,3**. (d) $F_1 = (3, 3, 3, 3, 13)$: o passo inteiro ajusta o último ponto exatamente, com todo o ruído que ele tiver. Com η pequeno, uma correção só vale por inteiro se várias árvores seguidas a pedirem.

× **Cuidado:** ajustar o toco a y em vez dos resíduos; ou somar a folha sem multiplicar por η.

3. (a) $\eta = 0{,}5$**: 5 árvores;** $\eta = 0{,}05$**: 200**. Depois do pico a AP cai (0,336 e 0,408 com 1.000): as árvores seguintes ajustam ruído. Numa random forest, mais árvores nunca pioram, só estabilizam a média. (b) Passos pequenos chegam ao sinal devagar e ajustam menos ruído pelo caminho: o vale é mais fundo e mais largo. (c) Com uma **fatia separada dos dados de treino** (`validation_fraction`); usar o teste para escolher o número de árvores o transformaria em validação e a nota final deixaria de ser honesta.

4. (a) **Não**: a duração só existe depois que a ligação aconteceu e o modelo serve para decidir para quem ligar. É uma coluna que conhece o futuro. (b) Idade,

saldo, resultado da campanha anterior e número de ligações da campanha anterior: **existem**; **duração: não**. (c) **Não**: o vazamento mora na coluna, não na divisão; ele está presente em todo fold. Só a pergunta "este valor existe na hora da previsão?" o detecta.

× **Cuidado:** comemorar o salto de 0,797 para 0,934: métricas boas demais para ser verdade pedem que se examine cada coluna.

5. (a) $p_0 = 0{,}6$, $F_0 = \ln 1{,}5 \approx 0{,}405$; pseudo-resíduos $-0{,}6; -0{,}6; 0{,}4; 0{,}4; 0{,}4$. **Corte** $x < 2{,}5$ (SSE 0). (b) $p_0(1-p_0) = 0{,}24$. Esquerda: $-1{,}2/0{,}48 = -2{,}5$; direita: $1{,}2/0{,}72 \approx 1{,}667$. $F_1 = 0{,}405 - 1{,}25 = -0{,}845$ e $0{,}405 + 0{,}833 = 1{,}239$; $p_1 \approx 0{,}30$ **e** $0{,}775$. A log loss cai de 0,673 para 0,296. (c) Esquerda $-1{,}2/1{,}48 \approx -0{,}81$; direita $1{,}2/1{,}72 \approx 0{,}70$. O λ puxa os valores das folhas para zero, mais forte nas folhas com poucos pontos: é a ideia do Ridge dentro da árvore.

× **Cuidado:** somar probabilidades em vez de log-odds: a soma poderia sair de $[0, 1]$.

6. (a) Diferenças: 2,3; 1,3; 1,3; 1,4; 1,4; **média** $\approx +1{,}5 \pm 0{,}4$ (sem arredondar os RMSE, $+1{,}6 \pm 0{,}4$). O boosting vence em todos os folds e a média é quatro vezes o desvio: **vantagem real**, ainda que pequena. (b) A floresta é difícil de estragar (faixa de 0,064, padrões perto do topo); o boosting tem uma faixa três vezes maior: bem ajustado vence, mal ajustado (por exemplo, η pequeno com poucas árvores) perde para qualquer floresta. (c) **(i) a floresta**: quase sem ajuste, com erro OOB de graça. **(ii) o boosting**, com parada antecipada e uma busca nos mesmos folds; ele também é muito mais rápido de treinar aqui (0,5 s contra 11,5 s).

M4 Seleção de modelos

1. (a) Distâncias: 9; 91; 11.069; 67.266. **Viés alto: a regressão linear**: erra quase o mesmo no treino e na validação (66 mil) e cortou só 42% do erro do baseline. **Variância alta: a árvore sem limites**: treino 0, validação 67.266. O k-NN tem alguma variância (11 mil). (b) k-NN (59.967), linear (66.460), árvore (67.266). A ordem não diz *por que* cada um erra, logo não diz o que tentar a seguir: a distância diz.

× **Cuidado:** ler o treino 0 da árvore como "perfeito". Treino 0 num modelo sem limite é memorização; só a validação conta.

2. (a) A: média 1,30, viés $-0{,}70$, viés^2 0,49, variância $(0{,}01 + 0{,}01 + 0 + 0{,}04 + 0{,}04)/5 = 0{,}02$. B: média 2,00, viés^2 0, variância $(0{,}36 + 0{,}64 + 0{,}09 + 0{,}25 + 0{,}16)/5 = 0{,}30$. (b) **A:** $0{,}49 + 0{,}02 + 0{,}04 = 0{,}55$**; B:** $0 + 0{,}30 + 0{,}04 = 0{,}34$. O ruído $\sigma^2 = 0{,}2^2 = 0{,}04$ é o piso dos dois. (c) **A está em underfitting** (erra do mesmo jeito em todo conjunto). B acerta na média, mas cada versão isolada pode errar muito: é o que muda com outro conjunto de treino.

× **Cuidado:** calcular a variância em torno da verdade (2) em vez da média do próprio modelo; isso mistura viés e variância.

3. (a) **A = k-NN, B = árvore, C = linear.** B tem treino 0 em todo tamanho; C tem as duas curvas encontradas em 66,5 mil desde umas 2.600 linhas; A guarda uma distância de 11 mil, com a validação ainda caindo (61,3 → 60,0). (b) **Ajudariam A e B** (validação ainda descendo na borda direita); **para C, nada**: as curvas convergiram num erro alto. (c) C: mais capacidade (atributos polinomiais com Ridge: 57.473 na aula); B: regularizar (`min_samples_leaf` = 10: 57.796); A: um k maior ($k = 10$: 59.182).

× **Cuidado:** "mais dados sempre ajudam". Num modelo com viés alto, as duas curvas já se encontraram e ficam planas.

4. Ridge, `alpha` $= 0{,}01$**: variância** (distância de 14 mil) → aumentar `alpha`. `alpha` $= 10^6$**: viés** (distância 42, perto do baseline) → diminuir. $k = 1$**: variância** (treino 0) → aumentar k. `min_samples_leaf` $= 100$**: viés** (distância pequena, validação 5.700 acima do fundo da família) → diminuir. **Grau 1: viés** (treino e validação altos e próximos) → aumentar o grau. Giram ao contrário `alpha`, `n_neighbors` e `min_samples_leaf`: maior é mais simples.

5. (a) $r = 0{,}0001/(0{,}0001 + 0{,}0004) = 0{,}2$; $D = 0{,}04$, então **uns** $0{,}80 + 0{,}2 \cdot 0{,}04 \approx 0{,}81$: quatro quintos da vantagem eram sorte dos folds. (b) $r = 0{,}8$: esperado 0,832; com pouco ruído, a vantagem quase toda se mantém. (c) Não o 0,84, que foi usado para escolher: **o número medido uma vez no teste trancado** (ou o da CV aninhada, que mede o procedimento "buscar

e escolher”). (d) **Não existe almoço grátis**: cada família vence onde a sua aposta combina com o problema (logística no hiperplano, k-NN nas luas, a família das árvores na regra de limiares, 0,938 a 0,948 contra 0,686 da SVM). Nem o ensemble serve sempre: no hiperplano de 40 dimensões a floresta e o boosting ficam em 0,735 e 0,720, contra 0,820 da logística, porque cada árvore corta ao longo de um eixo só. Em cada projeto, compare as famílias nos mesmos folds; num empate dentro do ruído, fique com a mais simples.

× **Cuidado:** repetir a CV com outros folds sobre as mesmas linhas para “confirmar” o vencedor: a sorte de quais linhas foram coletadas continua a mesma.

E1 Agrupamento hierárquico

1. (a) AB 2, AC 5, AD 11, AE 18, BC 3, BD 9, BE 16, CD 6, CE 13, DE 7. (b) Entre pontos isolados, as quatro ligações dão a própria distância; a menor é AB $= 2$. (c) **Simples: C 3, D 9, E 16. Completa: C 5, D 11, E 18. Média: C 4, D 10, E 17.**

2. (a) **P e U: 13; Q e R: 1**: a distância cofenética. (b) **Não**: V e P só se juntam no topo (24); R e U em 13. Cada barra pode girar: $2^6 = 64$ desenhos. (c) **Três grupos:** $\{P, Q, R\}$, $\{S, T, U\}$ **e** $\{V\}$. (d) $k = 2$: de 13 a 24 (11); $k = 3$: de 4 a 13 (9); $k = 4$: de 3 a 4 (1). $k = 2$ **e** $k = 3$ **têm apoio**; $k = 4$ não.

× **Cuidado:** folhas vizinhas no desenho não são vizinhas nos dados; leia a altura em que se encontram.

3. (a) **Simples**: ABC 3 (AB–C $= \min(5, 3)$); ABCD 6 (C–D); ABCDE 7 (D–E). **Completa**: ABC 5 (contra C–D 6); DE 7 (ABC–D seria 11); tudo em 18. **Média**: ABC 4; DE 7 (ABC–D seria $26/3 \approx 8{,}67$); tudo em $73/6 \approx 12{,}17$. (b) **Simples:** $\{A, B, C, D\}$ **e** $\{E\}$**; completa e média:** $\{A, B, C\}$ **e** $\{D, E\}$. (c) A completa: dois grupos valem para qualquer corte entre 7 e 18 (salto 11); na simples, dois grupos só valem entre 6 e 7 (salto 1): a cadeia cresce de espaço em espaço e isola o último ponto. (d) AB+C: $\frac{2 \cdot 1}{3}(5-1)^2 = 10{,}67$; C+D: $36/2 = 18$; D+E: $49/2 = 24{,}5$. **Funde ABC na altura** $\sqrt{21{,}33} \approx 4{,}62$**.**

× **Cuidado:** na ligação média, a distância entre grupos é a média de *todos* os pares cruzados, não a média das duas distâncias aos centros.

4. (a) Faixas: $k = 3$: 30,3 a 32,3 (2,0); $k = 4$: 9,7 a 30,3 (20,6); $k = 5$: 9,2 a 9,7 (0,5); $k = 6$: 8,9 a 9,2 (0,3); $k = 7$: 8,7 a 8,9 (0,2). **Os saltos destacam** $k = 2$ **e** $k = 4$: duas escalas (dois pares; duas bolhas em cada par). (b) A silhueta é máxima em $k = 2$ e cai sem parar: prefere a divisão grossa e não vê a segunda escala. (c) **Cortes da mesma árvore são aninhados**: os 4 grupos são os 2 partidos ao meio. Duas rodadas do k-means não têm essa obrigação.

5. (a) d: 1, 4, 6, 3, 5, 2; c: 1, 3, 3, 3, 3, 2. (b) $\bar{d} = 3{,}5$, $\bar{c} = 2{,}5$; $\sum(d - \bar{d})(c - \bar{c}) = 6{,}5$; $\sum(d - \bar{d})^2 = 17{,}5$; $\sum(c - \bar{c})^2 = 3{,}5$; $r = 6{,}5/\sqrt{61{,}25} \approx 0{,}83$. (c) **A fidelidade da árvore às distâncias, não a qualidade dos grupos.** É o critério quando o produto é a própria árvore (uma taxonomia, a ordem de um mapa de calor).

6. (a) A–B $= \sqrt{224} \approx 15{,}0$; A–C $= \sqrt{8} \approx 2{,}8$; B–D $= \sqrt{200} \approx 14{,}1$. **Euclidiana:** $\{A, C\}$ **e** $\{B, D\}$ (pequenos e grandes). (b) B $= 5\times$ A: $r = 1$**, distância 0**; C é A ao contrário: $r = -1$**, distância 2. Correlação:** $\{A, B\}$ **e** $\{C, D\}$ (crescentes e decrescentes). (c) A de correlação para “gasta cada vez mais”; a euclidiana para tamanho. Nenhum índice decide: decide a pergunta. (d) O custo de Ward usa centroides e distâncias euclidianas ao quadrado; com outra distância ele não tem sentido. (e) $50.000 \times 49.999/2 \approx 1{,}25$ bilhão de distâncias $\times$ 8 bytes ≈ 10 **GB**.

× **Cuidado:** a distância euclidiana mede quanto; a de correlação, o formato. Escolher uma é escolher a pergunta.

E2 Representação de texto

1. (a) **time, não, venceu, time, empatou, de, novo**: “O” e “o” somem (uma letra) e a pontuação também. **6 colunas** (de, empatou, novo, não, time, venceu). (b) Sai “de”: **5 colunas**; “não” sobrevive porque não está na lista. (c) **6 bigramas**: time não, não venceu, venceu time, time empatou, empatou de, de novo. **“venceu time”** junta palavras separadas por “: o”. (d) **“venc” é stemming; “vencer” é lematização.**

× **Cuidado:** a lista de stop words do inglês do scikit-learn tem “not” e “no”: ela apaga negações.

2. (a) d1: bom 1, filme 1. d2: bom 2, filme 1, muito 2. d3: livro 1, ruim 1. d4: do 1, filme 1, livro 1, ruim 1. “é” e “o” têm uma letra só. (b) $\cos(d1, d2) = 3/(\sqrt{2} \cdot 3) \approx$ **0,707**; $\cos(d1, d4) = 1/(\sqrt{2} \cdot 2) \approx$ **0,354**; $\cos(d1, d3) =$ **0** (nenhuma palavra em comum). (c) **Cosseno 1, distância** $\sqrt{2} \approx 1{,}41$: o mesmo conteúdo, mais longo. Para texto, o cosseno. (d) **Só “filme” = 1**: “ótimo” não estava no vocabulário do `fit` e foi descartada.

3. (a) df: filme 3; bom, livro, ruim 2; do, muito 1. **idf: filme** $\ln 1{,}25 + 1 = 1{,}2231$**; bom, livro, ruim** 1,5108**; do, muito** 1,9163. “filme” está em 3 das 4 e distingue pouco. (b) tf×idf: bom 1,5108; filme 1,2231; comprimento $\sqrt{2{,}2825 + 1{,}4960} = 1{,}9438$: **bom 0,777; filme 0,629**. (c) Só “filme” em comum: $0{,}629 \times 0{,}392 \approx 0{,}247$, contra 0,354 nas contagens: a palavra compartilhada é a mais comum e o TF-IDF a desconta. (d) d1: $0{,}629^2 = 0{,}396$; d3: $0{,}777 \times 0{,}707 = 0{,}550$; d4: $0{,}629 \times 0{,}392 + 0{,}777 \times 0{,}484 = 0{,}623$. **Ordem: d4, d3,**

d1.

4. (a) **Sinônimos dão zero**; **mesmo sentido com outras palavras** recebe a mesma nota que **uma grafia com dois sentidos** (0,21 nas duas); **a ordem é invisível**. O embedding separa as linhas 2 e 3, mas também quase não vê a ordem (0,98). (b) Os cossenos de embeddings ocupam uma escala comprimida e própria de cada modelo: vale a ordenação e um limiar precisa ser medido para o modelo. (c) (1) **esparso**: token raro e exato, idf alto; (2) **denso**: textos curtos sem palavras em comum; (3) **denso**: poucos rótulos (55,7% contra 81,9% com 5 por classe); (4) **denso multilíngue**: o espaço é compartilhado entre línguas.

× **Cuidado:** com muitos rótulos o TF-IDF empata (90,4% contra 90,1%) e custa uma fração do tempo: comece por ele.

5. (a) (carro, o): $\log_2(2 \times 126/(6 \times 22)) = \log_2 1{,}91 \approx$ **0,93**; (carro, gasolina): $\log_2(126/24) = \log_2 5{,}25 \approx$ **2,39**. **"gasolina"**: "o" aparece ao lado de tudo e quase não surpreende. (b) Contagens: o artigo "o", comum a todo substantivo, domina o cosseno. PPMI: conta só o que excede o acaso e tira o artigo, mas fica literal (leite e água são colunas diferentes). SVD: comprime as colunas e junta contextos parecidos, então gato e cachorro ficam próximos. (c) Antônimos aparecem nos mesmos contextos ("a sopa está quente/fria"): a hipótese distribucional mede relação, não sinonímia. Armadilha em análise de sentimento. (d) **Um vetor por palavra**, dominado pelo sentido mais frequente. Embeddings contextuais (sentence embeddings) dão a cada ocorrência um vetor que depende da frase.

6. (a) Senha: F1 0,96; F2 0,48; F3 0,36: **F1**. Desisti: F1 0,60; F2 0,856; F3 0,928: **F3**, embora não haja palavra em comum. (b) **0 com as três**: nenhum token em comum. (c) Híbrida: F1 $0{,}5 \times 0{,}467 + 0{,}5 \times 0{,}96 \approx$ **0,71**; F2 0,24; F3 0,18: **a ordem não muda**. O TF-IDF decide em consultas com tokens raros e exatos (códigos, nomes, números). (d) Avisar que não tem certeza ou mostrar as duas opções, em vez de responder como se tivesse achado. (e) **Cosseno 1**: as duas frases têm as mesmas palavras, e a média não vê a ordem.

× **Cuidado:** nota alta de embedding é relativa: 0,93 contra 0,86 é uma margem pequena.

E3 Modelagem de tópicos e BERTopic

1. (a) $p = 0{,}75\,\varphi_{\text{cul}} + 0{,}25\,\varphi_{\text{esp}}$: **pão 0,375; forno 0,2375; massa 0,0875; foguete 0,1375; órbita 0,075; lua 0,0875**; soma 1. (b) $0{,}25 \times 20 = 5$ **palavras**. A história sorteia cada palavra de forma independente: só as contagens importam (bag-of-words). (c) $\alpha = 0{,}1$: documentos concentrados num tópico. (d) Com 53 palavras há pouca evidência e o ajuste para onde o ponto de partida aleatório o leva.

2. (a) **Tópico 1: futebol (gol, time, jogo); tópico 2: espaço (foguete, lua).** d1 é futebol, d2 espaço, d3 meio a meio, d4 sobretudo espaço com um pouco de futebol. (b) **d3: (1; 0,5; 0,5; 1; 0,5); d4: (0,4; 0,2; 0,2; 1,2; 0,6).** (c) "Futebol *menos* espaço": uma direção que separa os dois temas e não descreve nenhum. Na NMF, com $W, H \geq 0$, um documento só se monta somando tópicos. (d) **Nenhum modelo vence sempre**: depende do corpus; leia os tópicos (aqui, um tópico-lixo de assinatura).

× **Cuidado:** os pesos de W não são probabilidades: uma linha não precisa somar 1.

3. (a) $P(a) = 0{,}1$, $P(b) = 0{,}05$, $P(a,b) = 0{,}04$; razão $0{,}04/0{,}005 = 8$; **NPMI** $= 2{,}079/3{,}219 \approx 0{,}65$. (b) Com 10: $P(a,b) = 0{,}005 = P(a)P(b)$, **NPMI 0** (independentes). Com 0: -1. (c) $10/12 \approx 0{,}83$. "gente" é uma palavra corriqueira que escapou da lista de stop words e se repete entre tópicos. (d) Não pela perplexidade; varra k, procure um cotovelo na coerência (com várias sementes) e **leia os tópicos** em dois ou três candidatos: é uma escolha de granularidade.

4. (a) $A = \lfloor 45/3 \rfloor = 15$. time: $f = 4$, $\ln(1 + 15/4) = 1{,}558$, $4/20 \times 1{,}558 \approx$ **0,312**. lua: $f = 2$, $\ln 8{,}5 = 2{,}140$, $2/10 \times 2{,}140 \approx$ **0,428**. forno: $f = 3$, $\ln 6 = 1{,}792$, $3/15 \times 1{,}792 \approx$ **0,358**. (b) que: $f = 14$, peso 0,728: **0,218; 0,218; 0,243**. Aqui não vence, mas chega perto: o peso a desconta, a frequência a sustenta. Num grupo grande e variado, as palavras funcionais vencem e nomeiam o tópico; a cura é a lista de stop words no vetorizador. (c) **Não**: as palavras são calculadas depois do agrupamento.

5. (a) **Embeddings** (textos → vetores de 384 números) → **UMAP** (384 → 5 dimensões) → **HDBSCAN** (rótulos de grupo, -1 para outliers) → **c-TF-IDF** (palavras de cada grupo). (b) Textos curtos quase não dividem palavras: contagens ortogonais, nada para a NMF agrupar. Embeddings põem perto textos de mesmo sentido. (c) Em 384 dimensões as distâncias se concentram e a densidade se borra; 2 dimensões espremeriam demais: o objetivo é agrupar, não desenhar. (d) O número depende da configuração e muda em degraus; **relate os tópicos que sobrevivem a várias sementes e perturbações** e evite um `min_cluster_size` na beira do degrau (como 10 aqui).

× **Cuidado:** fixar a semente torna a rodada repetível, não torna os tópicos verdadeiros.

6. (a) Coerência e diversidade: **NMF**; homogeneidade: **BERTopic**; estabilidade: NMF. O BERTopic é medido só nos 80% que manteve; com todos os posts, 0,822 a 0,882, ainda à frente. (b) A NPMI premia palavras que coocorrem em muitos documentos, o que a NMF reconstrói; o c-TF-IDF escolhe palavras *distintivas*, mais raras. (c) Uma boa resposta: **BERTopic**, porque os textos são curtos, com um modelo multilíngue (`paraphrase-multilingual-MiniLM-L12-v2`), stop words em português no vetorizador e `min_cluster_size` longe de um degrau, com semente fixa. Relatar: a configuração, a fração de outliers, como foram tratados e a estabilidade entre sementes.

E4 Redes neurais

1. (a) A: $z = 0$, $yz = 0 \leq 0$: erro, $w = (2,1)$, $b = 1$. B: $z = 2+2+1 = 5$, $yz = -5$: erro, $w = (2,1) - (1,2) = (1,-1)$, $b = 0$. C: $z = -1$, $yz = 1$: certo. D: $z = 3-2 = 1$, $yz = 1$: certo. (b) Época 2: $z = 1, -1, -1, 1$ para A, B, C, D, todos do lado certo: **nenhum erro, o perceptron para**. (c) $x_1 - x_2 = 0$**, a reta** $x_2 = x_1$; a classe $+1$ fica onde $x_1 > x_2$ (w aponta para ela).

× **Cuidado:** calcular z com os pesos já atualizados ou achar que $z = 0$ é acerto: um ponto sobre a reta conta como erro.

2. (a) $w^\top = w_2^\top W_1 = (2\cdot 1 - 1\cdot 3;\ 2\cdot 2 - 1\cdot(-1)) = (-1, 5)$ e $b = w_2^\top b_1 + b_2 = 2 + 0{,}5 = 2{,}5$. Em $x = (1,1)$: $h = (4, 2)$, $u = 8 - 2 + 0{,}5 = 6{,}5$; direto, $-1 + 5 + 2{,}5 = 6{,}5$. $u = -x_1 + 5x_2 + 2{,}5$. (b) (i) $21\cdot 16 + 17\cdot 1 = 353$; (ii) $21\cdot 8 + 9\cdot 8 + 9\cdot 1 = 249$: **a camada única de 16 tem mais** (a conta cresce com o produto das larguras vizinhas). (c) **Só retas**: a rede inteira é a regressão logística.

× **Cuidado:** esquecer os vieses: $20\cdot 16 + 16\cdot 1 = 336$ conta só os pesos.

3. (a) $h_1 = 0,0,1,0$; $h_2 = 0,1,0,0$; saída $= 0,1,1,0$: **o XOR**. h_1 detecta "x_1 ligado e x_2 desligado"; h_2, o contrário. (b) $(0,0)$ e $(1,1)$ caem os dois em $(0,0)$ (classe 0); $(1,0)$ vai para $(1,0)$ e $(0,1)$ para $(0,1)$ (classe 1). As duas diagonais, impossíveis de separar no espaço de entrada, viram um canto contra dois: **a reta** $h_1 + h_2 = 0{,}5$ **separa**. (c) A derivada do degrau é 0 em todo lugar (e não existe em 0): o gradiente não diz para onde mexer os pesos.

4. (a) Uma unidade traça uma reta só; a saída é uma função dessa reta. (b) **Com 100 unidades e** `alpha` **= 0,0001**: treino 0,997 contra CV 0,857, uma distância de 0,14. (c) **100 unidades com** `alpha` **= 1** (CV 0,917, a maior, e treino 0,914: a distância se fechou). Com `alpha` = 10 a penalidade encolhe tanto os pesos que a rede vira quase uma reta: underfitting, 0,830 como a logística.

× **Cuidado:** escolher pelo treino (0,997): a escolha é sempre pela CV.

5. (a) $z_1 = 0{,}5 + 0{,}5 - 0{,}5 = 0{,}5$ e $z_2 = -1 + 1 - 0{,}5 = -0{,}5$; $h = (0{,}5;\ 0)$; $u = 2\cdot 0{,}5 = 1$; $\hat{y} = 0{,}731$; $L = 0{,}313$. (b) $\delta_y = -0{,}269$; $\partial L/\partial v_1 = -0{,}269\cdot 0{,}5 \approx -0{,}134$ e $\partial L/\partial v_2 = 0$ ($h_2 = 0$). $\delta_{h_1} = -0{,}269\cdot 2\cdot 1 = -0{,}538$ e $\delta_{h_2} = 0$. $\partial L/\partial w_{11} = -0{,}538$, $\partial L/\partial w_{12} = -1{,}076$, $\partial L/\partial b_{h_1} = -0{,}538$. (c) $z_2 < 0$: a ReLU está desligada e sua derivada é 0, então $\delta_{h_2} = 0$ e todo gradiente que passa por h_2 é zero. (d) Gradientes negativos fazem os pesos subirem: $v_1 \approx 2{,}013$, $w_{11} \approx 0{,}554$, $w_{12} \approx 0{,}358$, viés de $h_1 \approx -0{,}446$, $b_y \approx 0{,}027$. **Sim**: $\hat{y}$ subiu em direção ao alvo 1, então $L = -\ln \hat{y}$ caiu (de 0,313 para cerca de 0,170).

× **Cuidado:** somar $\eta\cdot$ gradiente em vez de subtrair: o passo vai contra o gradiente.

6. (i) Não é sorte: o gradiente descendente sofre com atributos em escalas muito diferentes (um vale longo e estreito). **O** `StandardScaler` **fica no pipeline**, ajustado em cada fold. (ii) A perda de uma rede **não é convexa**: inícios diferentes param em mínimos diferentes (metade chegou a 0,94). Use mais unidades (que dão mais caminhos a um bom mínimo), fixe a semente e confira conclusões em várias sementes. (iii) Média 0,008, desvio $\approx$ 0,009: o boosting vence 4 dos 5 folds, mas por menos de um desvio em média. **"Neste conjunto o boosting ficou um pouco à frente, com menos trabalho: uma vantagem, não um nocaute."** O boosting não precisou de escala, de arquitetura, de `alpha` nem de solver; por isso, em tabelas, ele é a primeira escolha e a rede precisa pagar o próprio custo.

× **Cuidado:** concluir "a rede não serve" por uma diferença menor que o desvio entre folds.

E5 Explicabilidade

1. (a) `conta_corrente`: média 0,660, **queda 0,140**; `finalidade`: média 0,798, queda 0,002. **O modelo depende de** `conta_corrente`; a queda de `finalidade` está dentro do ruído dos embaralhamentos (de $-0{,}01$ a $+0{,}01$). Uma nota acima da original é esse ruído: um embaralhamento que, por acaso, ajudou um pouco. (b) A MDI vem das divisões do treino e um ruído contínuo, com muitos valores distintos, atrai divisões (a moeda tem só dois valores). No treino, a floresta decorou linhas em parte por ele, então embaralhá-lo custa 0,016. Em linhas novas ele não carrega nada: **vai para o relatório a permutação no teste,** $-0{,}000$.

× **Cuidado:** calcular a importância no treino ou confiar na MDI: as duas premiam o que o modelo decorou.

2. (a) D1 e D4: 160 e 340, ganho $+180$; D2 e D3: 190 e 460, ganho $+270$. $\mathrm{PD}(2) = 175$ **e** $\mathrm{PD}(8) = 400$. A renda de cada linha é *trocada* por v; só o resto da linha (`antigo`) fica. (b) **Em leque**: $+180$ nas casas novas, $+270$ nas antigas; o PDP ($+225$, a média) esconde a interação renda × antigo. (c) Fixar a latitude mantendo a longitude de cada linha cria distritos que não existem (no oceano). Olhe as curvas ICE e o alcance dos dados; desenhe o PDP do par junto.

× **Cuidado:** tirar a média das previsões das linhas que têm renda perto de v: o PDP fixa v em *todas* as linhas.

3. (a) Embaralhando uma cópia, as árvores que dividem pela outra continuam funcionando: a queda parece pequena. Separadas, somam 0,392, metade do que a informação vale; juntas, a informação some de vez. **Reporte o grupo: "renda" vale 0,748**. (b) $0{,}825 - 1{,}330 \approx -0{,}5$: R^2 negativo é pior que prever a média. Latitudes embaralhadas com longitudes reais caem no oceano ou no deserto, onde o modelo extrapola mal; o número mistura a importância com o comportamento fora do mapa. **Trate latitude e longitude como um atributo só, "localização", e embaralhe o par junto**.

4. (a) Previsões do fundo: 0,10; 0,30; 0,20; 0,10, então $v(\varnothing) = 0{,}175$. Com atraso = 1: 0,30; 0,30; 0,60; 0,30, então $v(\{\text{atraso}\}) = 0{,}375$. Com jovem = 1: 0,20; 0,60; 0,20; 0,20, então $v(\{\text{jovem}\}) = 0{,}30$. $v(\{\text{atraso, jovem}\}) = \hat{f}(1,1) = 0{,}60$. (b) $\phi_{\text{atraso}} = \frac{1}{2}(0{,}200) + \frac{1}{2}(0{,}300) =$ **0,25**; $\phi_{\text{jovem}} = \frac{1}{2}(0{,}125) + \frac{1}{2}(0{,}225) =$ **0,175**; $0{,}175 + 0{,}25 + 0{,}175 = 0{,}60$. (c) $\hat{f}(1,0) = 0{,}30$: cairia 0,30. O SHAP compara com a média do fundo (onde um em quatro já é jovem) e divide a interação ao meio; não é "trocar o valor". E **não**: a parcela descreve o modelo, não o efeito de envelhecer; idade ainda é um atributo protegido.

× **Cuidado:** ler uma parcela SHAP como o efeito de mudar o atributo, no modelo ou no mundo.

5. (a) **Não se contradizem**. O SHAP diz quanto a renda *real* deste distrito afasta a previsão da do distrito médio; a renda é comum, então afasta pouco ($+2{,}6$): responde "por que é cara?". O LIME dá a inclinação local, que é íngreme: responde "o que mudaria o preço?". (b) A explicação descreve o modelo: domicílios lotados são um marcador de bairros mais pobres, que o modelo usa. **Não é causal**; para isso, só um experimento ou inferência causal. (c) A largura da vizinhança (a inclinação muda de 38,2 a 54,8 com ela) e a unidade (por desvio padrão).

6. (a) $0{,}710 - 0{,}305 = 0{,}405$; as seis somam 0,347, então **as outras 14 somam cerca de 0,058** (0,057 com os valores sem arredondar). Na carta: duração, conta corrente e poupança, motivos que o banco defende. **Não**: estado civil com sexo e idade, características protegidas. (b) O escore não precisa dele (permutação $-0{,}001$; retreinar dá um empate), mas as previsões o usam: **sai**, sem custo de acurácia. Tirar a coluna não remove *proxies*: outras colunas podem carregar a mesma informação; é preciso auditar (por exemplo, prever `personal_status` a partir das outras). (c) **"Quando esse valor é registrado?"** Cartas de cobrança são enviadas depois que o empréstimo dá errado: é vazamento. A validação cruzada não o pega, porque ele está em todas as linhas. (d) "O modelo pode tratar grupos de forma diferente porque só a coluna `personal_status` foi retirada, não seus proxies; quem usa deve comparar as taxas de recusa por grupo antes de aplicá-lo."

× **Cuidado:** concluir que um atributo é inofensivo porque sua importância por permutação é zero: ela mede se o *escore* precisa dele, não se as *decisões* o usam.

E6 AutoML

1. (a) Uniforme: $(1-0{,}01)/(100-0{,}01) \approx$ **1%**. Na escala log, os sorteios abaixo de 1 ocupam 2 das 4 décadas: **50%**. (b) **Taxa de aprendizado, `alpha` e número de folhas**: o que importa nelas é a razão entre valores. A fração de atributos varia numa faixa estreita, em que somar 0,1 pesa igual em toda parte: escala linear.

× **Cuidado:** sortear uniformemente uma taxa de aprendizado: quase todo o orçamento cai na década mais alta.

2. (a) **A grade testa 4 valores de x_1; os sorteios, 64.** (b) O pior caso é c no meio de dois valores da grade ou numa ponta: **1/8 = 0,125**. Em média, **1/16 ≈ 0,063**, oito vezes a distância dos sorteios (0,008). (c) $5^5 = 3.125$ avaliações, com **só 5 valores de cada hiperparâmetro**: a grade paga cada eixo inútil multiplicando o custo.

3. (a) A: $z = 1$; termos 0,0084 e 0,0024; EI = 0,0108. B: $z = -0{,}5$; termos −0,0093 e 0,0211; EI = 0,0118. C: $z = -0{,}5$; termos −0,0031 e 0,0070; EI = 0,0040. **O próximo é B**, mesmo com média abaixo de f^+. Em A trabalha a média (exploração local); em B, a incerteza (exploração). (b) Com z fixo, EI $= \sigma\,[z\,\Phi(z) + \varphi(z)]$ é proporcional a σ: $\sigma_B = 3\,\sigma_C$. (c) Depois de algumas avaliações planas perto do pico, o substituto prevê ali um escore plano e *certo* e não dá melhoria esperada nenhuma: **o pico estreito nunca é avaliado**. Um substituto errado decide errado com confiança.

× **Cuidado:** achar que um candidato com média abaixo do melhor tem EI zero: o termo $\sigma\varphi(z)$ premia a incerteza.

4. (a) 81×5, 27×15, 9×45, 3×135, 1×405: cada rodada custa 405, **2.025 árvores ao todo**, contra $81 \times 405 = 32.805$ (cerca de 6%, o preço de 5 avaliações completas). (b) Taxa de aprendizado e número de árvores se compensam: com poucas árvores, quem aprende devagar mal começou e parece pior, mesmo terminando melhor. **Quando a unidade de orçamento interage com um hiperparâmetro buscado, a ordem inicial é enviesada**; use linhas como orçamento ou parada antecipada.

5. (a) **O modelo sem a coluna** (0,804 contra 0,782 no uso real). Nenhum método de busca: o vazamento está nos dados e toda verificação rodou na tabela histórica, onde a coluna está preenchida. Só quem sabe quando cada coluna é registrada pega o problema. (b) Relatado +0,016, entregue +0,004: $0{,}012/0{,}016 = 75\%$ **do ganho é otimismo**. Com mais linhas, cada escore de CV é menos ruidoso, então a busca acha configurações boas em vez de sortudas.

× **Cuidado:** acreditar no escore mais alto da busca sem perguntar se cada coluna existe na hora da previsão.

6. (a) Num **canto**: a maior taxa, o maior número de folhas, a menor folha. A grade era estreita naquela direção; seria preciso estendê-la. (b) As diferenças entre os métodos (até 0,003) são menores que o desvio entre folds (0,0042): **nenhum venceu**; todos ganham de 0,003 a 0,006 dos valores padrão. (c) Não: os valores padrão também caíram, de 0,9468 para 0,9374; essas linhas de teste são mais difíceis. Contra os valores padrão, nas mesmas linhas, **o ajuste vale** +0,008. (d) $0{,}9499 - 0{,}9472 \approx 0{,}003$ de otimismo, do tamanho de todo o ganho da busca sobre os valores padrão. (e) "O modelo ajustado marca 0,9455 de ROC-AUC no teste trancado, 0,008 acima dos valores padrão nas mesmas linhas; o 0,9528 da CV é o melhor escore de quatro buscas e, por isso, otimista."

× **Cuidado:** divulgar o `best_score_` da busca como desempenho esperado.

E7 MLOps

1. Semana A: termos $0{,}05 \cdot 0{,}182 = 0{,}009$ (duas vezes) e $(-0{,}05)(-0{,}223) = 0{,}011$ (duas vezes): **PSI = 0,041, estável**. Semana B: $(-0{,}20)(-1{,}609) = 0{,}322$; $(-0{,}10)(-0{,}511) = 0{,}051$; $0{,}05 \cdot 0{,}182 = 0{,}009$; $0{,}25 \cdot 0{,}693 = 0{,}173$: **PSI = 0,555, mudança significativa**. Se $a_i > e_i$, os dois fatores são positivos; se $a_i < e_i$, os dois são negativos: o produto nunca é negativo.

× **Cuidado:** esquecer o sinal de $a_i - e_i$ e subtrair termos: todo termo soma.

2. (a) Nos valores 2, 4, 5, 6, 7, 8, 9, 11: $F_{\text{ref}} = 0{,}25;\ 0{,}5;\ 0{,}5;\ 0{,}75;\ 0{,}75;\ 1;\ 1;\ 1$ e $F_{\text{cur}} = 0;\ 0;\ 0{,}25;\ 0{,}25;\ 0{,}5;\ 0{,}5;\ 0{,}75;\ 1$. Os saltos chegam a 0,5 (em 4, 6 e 8): $D = 0{,}5$. (b) **Não**: a diferença é real, mas minúscula. O p-valor só diz se existe *alguma* diferença; com um milhão de linhas, sempre existe; o PSI mede o tamanho dela e fica perto de zero.

3. (a) **I, prior shift**: a taxa real caiu para 0,200 e a prevista média ficou em 0,339; a acurácia quase não mudou (0,794), mas a precisão caiu para 0,491. O PSI moderado (0,152) vem de haver menos positivos: o x de cada classe é o mesmo. **II, sem mudança**: PSI 0,004 e p 0,695, acurácia 0,816. **III, concept drift**: entradas iguais (PSI 0,003, p 0,340), mas acurácia 0,660 e precisão 0,329. **IV, covariate shift**: PSI 3,505 e p 0,000; a acurácia caiu para 0,728 porque o modelo é usado onde viu poucos dados. (b) O concept drift: só os rótulos o revelam (métricas por lote quando os desfechos chegam ou uma amostra auditada).

× **Cuidado:** achar que um PSI baixo garante que o modelo está bem: o concept drift não move as entradas.

4. R1 **rejeitar** (abaixo do limite do negócio, 18). R2 **avisar** (fora da faixa de treino, mas possível). R3 **avisar**: com o tempo, `pdays` cresce além do ano de treino. R4 **rejeitar** (categoria desconhecida). R5 **rejeitar** (tipo errado). R6 **rejeitar** (campo inesperado). Sem o schema, R1 é padronizada como qualquer número e R4 vira "tudo zero" no one-hot: **as duas recebem uma probabilidade com cara de confiável, sem aviso**. R6: a duração da ligação só existe depois da ligação; se o sistema cliente a manda, alguém pode estar planejando usá-la, um vazamento de um atributo que conhece o futuro.

5. (a) **Não**: a ordenação é carregada pelos outros atributos e a ROC-AUC quase não se mexe, mas 17,2% das decisões mudam. Pegam o problema a verificação de faixa do schema e o monitor da fração sinalizada. (b) $18/250 = 7{,}2\%$. O placar é o mesmo, mas a decisão de cada pessoa não: **fixar a semente torna cada resposta reprodutível** (sem escolher a semente pelo teste). (c) $10.000 \times 1{,}7$ ms $= 17$ s, contra uns 6 ms numa só chamada: **batch**.

× **Cuidado:** monitorar só a ROC-AUC: ela mede a ordem, não as decisões nem as probabilidades.

6. (a) 2009Q2: avisos e PSI do atributo. 2009Q3, 2010Q2 e 2010Q4: avisos, PSI do atributo, PSI do escore e calibração (desvios $-0{,}217$, $-0{,}384$ e $-0{,}253$). A regra da ROC-AUC (limite 0,569) nunca dispara. (b) **Não**: as duas mudanças têm explicação (o tempo e uma mudança de registro); o escore (PSI 0,071) e a calibração ($-0{,}043$) estão bem. Drift não é dano. (c) **As probabilidades**: o modelo segue prevendo uns 16% onde até 55% contratam, enquanto a ROC-AUC (0,584 a 0,749) fica perto da validação: a ordem se sustentou. É sobretudo **prior shift**. (d) A aposta de que o passado recente se parece com o futuro próximo; o tamanho da janela é o tamanho da aposta. Antes de substituir: o portão de avaliação (vencer o atual em dados recentes rotulados, por mais que o ruído), empacotar como nova versão com metadados, liberar em shadow ou canary e manter o rollback pronto.

× **Cuidado:** ler um alerta de drift como prova de que o modelo quebrou ou a ausência de alerta de ROC-AUC como prova de que nada mudou.

E8 A fronteira

1. Softmax, a regressão logística multiclasse (C1); gradiente descendente em mini-batch com a penalidade do Ridge, o weight decay (R2); k-NN (C2) sobre embeddings (E2); regressão logística sobre a diferença de escores (C1); vazamento (C6); a composição de E3, com peças de U1, U2 e E2; matriz de confusão, precisão e recall (C3).

× **Cuidado:** achar que a pilha nova dispensa o curso: cada peça dela é uma das aulas, em escala maior.

2. (a) $2 \cdot 2048 \cdot 16 = 65.536$ números, contra $2048^2 = 4.194.304$: **cerca de 1,6%**. (b) $\sigma(1{,}5 - 0{,}5) = \sigma(1) =$ **0,731**. Se a pessoa preferiu A, a perda é $-\ln 0{,}731 = 0{,}313$; se preferiu B, $-\ln(1 - 0{,}731) = 1{,}313$: errar com confiança custa caro, como na regressão logística.

3. S1 **clássico**: em tabelas médias e grandes, o gradient boosting lidera. S2 **fundação**: linguagem livre com poucos rótulos (few-shot ou embeddings congelados e uma regressão logística). S3 **clássico**: coeficientes e valores SHAP podem ser mostrados a um auditor; um prompt não. S4 **clássico**: custo e latência. S5 **clássico por padrão, mas teste o desafiante**: em tabelas pequenas, um modelo de fundação tabular (TabPFN) compete com o boosting; compare nos mesmos folds. S6 **os dois**: o modelo de fundação transcreve e lê; o boosting combina o que ele extrai com a tabela do cliente (roteamento).

4. (a) Matriz $[[45, 15], [10, 130]]$. **Acurácia** $175/200 = 0{,}875$**; precisão** $45/55 \approx 0{,}82$**; recall** $45/60 = 0{,}75$. Sempre "não é cobrança" acerta $140/200 = 0{,}70$. (b) **Vazamento**: o prompt já viu 5 tíquetes do teste. Escolha os exemplos fora do conjunto de teste (ou tire esses 5 do teste) e meça de novo. (c) **Não**: $0{,}75 \times 60 = 45$ contra $0{,}70 \times 60 = 42$, só 3 tíquetes de diferença, dentro do ruído de 60 positivos (o erro-padrão de um recall de 0,75 com 60 casos é cerca de 0,056) e o teste do prompt ainda está contaminado.

× **Cuidado:** avaliar um prompt nos mesmos exemplos que ele mostra ou declarar vencedor por 3 casos.

5. P1: etapa 1, **certo** (F2). P2: etapa 2, errado: dividir primeiro e explorar só o treino (D1). P3: etapa 3, errado: imputação e escala dentro de um pipeline ajustado só no treino (D2, D3, C6). P4: etapa 3, errado: a coluna só existe depois do cancelamento, **vazamento** (C6; a importância por permutação a denunciaria, E5). P5: etapa 4, errado: k-NN sem escala deixa a coluna de números maiores decidir; com $k = 1$ o treino dá 1,00 por construção; escolha k por CV (C2). P6: etapa 5, errado: dizer "ninguém cancela" já acerta 0,92; use recall, precisão ou PR-AUC (C3, C4). P7: etapa 5, errado: escolher pelo

teste é a **maldição do vencedor**; escolha por CV e abra o teste uma vez (C6, M4, E6). P8: etapa 5, errado: **o SHAP não é causal**; frequência baixa pode ser sintoma, não causa (E5). P9: etapa 6, errado: salve o pipeline inteiro com metadados e monitore entradas e escores todo mês, métricas quando os rótulos chegarem (E7).

× **Cuidado:** achar só o erro mais visível: neste plano, oito dos nove passos têm problema.

6. Um exemplo. **Formular** (F2): no dia 1 de cada mês, ordenar os restaurantes pelo risco de reprovar na próxima inspeção, só com o que se sabe antes da visita; com 100 visitas por mês, a métrica é a precisão entre os 100 primeiros; deixar passar um restaurante perigoso custa mais que uma visita perdida. **Preparar** (D2, D3, E2): o texto vira TF-IDF ou embeddings de sentenças (ou um LLM extrai categorias: roteamento, E8); bairro em one-hot e imputação num pipeline; nada registrado depois da inspeção (C6). **Treinar** (C1, M3): baseline que ordena pelas reprovações anteriores; regressão logística e gradient boosting. **Avaliar** (C4, C6): treino até 2023, validação em 2024, teste trancado em 2025; PR-AUC e precisão no topo, lidas contra a prevalência de 0,12 e contra o baseline. **Explicar e implantar** (E5, E7): para cada restaurante, as parcelas SHAP que o fiscal entende; cuidado com o bairro como proxy; pipeline empacotado; PSI das entradas e do escore todo mês; e um laço de realimentação: só se sabe o resultado dos restaurantes visitados, então uma fração das visitas deve ser sorteada.