

# Engenharia de Variáveis: Escolhendo Features



Plataforma completa de aprendizado  
contínuo em programação.

**#BoostingPeople**

[rocketseat.com.br](https://rocketseat.com.br)

Todos os direitos reservados © Rocketseat S.A.

# O que é engenharia de variável?

Se o modelo fosse um carro, as features seriam o combustível. Qual você está colocando no tanque?

# O que é engenharia de variável?

**Feature Engineering:** Processo de criar, transformar e selecionar variáveis que alimentam os modelos.

# Por que isso importa tanto?

- **Variáveis irrelevantes:** causam ruído, reduzem performance.
- **Variáveis mal construídas:** confundem o modelo, prejudicam a generalização.
- **Variáveis bem pensadas:** aumentam acurácia, interpretabilidade e robustez.

# Tipos de features

- Numéricas (ex: idade, salário)
- Categóricas (ex: estado civil, região)
- Ordinais (ex: nível de escolaridade)
- Temporais (ex: dia da semana, mês, sazonalidade)
- Texto (usado em NLP, ex: sentimentos, contagem de palavras)

# Técnicas de criação

1. Transformações básicas: Transformações diretas em colunas existentes.
2. Binning (faixas ou discretização): Converte variáveis contínuas em categorias.
3. Variáveis temporais: Criar features a partir de datas ou tempos.
4. Codificação de variáveis categóricas: Converter texto para números de forma útil.
5. Interações entre variáveis: Multiplicar ou combinar variáveis.
6. Estatísticas por grupo: Criar variáveis com base em estatísticas locais.
7. Encoding avançado (para ML): Técnicas mais específicas para modelos.

# Transformações básicas

Técnica	Exemplo prático	Quando usar
Soma, subtração, média	$\text{ticket\_médio} = \text{receita} / \text{número de compras}$	Criar métricas simples a partir de colunas base
Diferença de datas	$\text{dias\_desde\_ultima\_compra} = \text{hoje} - \text{última compra}$	Variáveis temporais
Proporções e razões	$\text{porcentagem\_desconto} = \text{desconto} / \text{preço original}$	Avaliar intensidade ou impacto relativo
Agregações por grupo	$\text{compra\_média\_por\_cliente, total por cidade}$	Comparações entre grupos

# Binning (faixas ou discretização)

Técnica	Exemplo	Quando usar
Faixas fixas	Idade: 18–25, 26–35, etc.	Simplificar variáveis contínuas
Quartis	Dividir em quartis, decis, percentis	Detectar comportamento por grupo
Binarização	Ex: renda > R\$ 5.000 = "alta"	Para modelos que lidam melhor com categorias

# Variáveis temporais

Técnica	Exemplo	Quando usar
Extraír partes da data	dia_da_semana, mês, hora	Detectar padrões sazonais ou horários de pico
Lags	vendas_mês_anterior	Modelos com memória temporal
Diferença entre datas	dias_entre_compras	Comportamento de usuário, recorrência
Rolling averages / médias móveis	média_3_meses_de_vendas	Suavização de séries temporais

# Codificação de variáveis categóricas

Técnica	Exemplo	Quando usar
One-hot encoding	Gera colunas binárias por categoria	Quando poucas categorias
Label encoding	Transforma categorias em números	Quando há ordenação implícita
Target encoding	média do target por categoria	Quando há forte relação entre categoria e target
Frequency encoding	Número de vezes que aparece	Catégoricos com muitos níveis

# Interações entre variáveis

Técnica	Exemplo	Quando usar
Produto de variáveis	preço * quantidade	Reflete valor total de transações
Combinação de categorias	plano_tipo + canal_venda = novo grupo	Descobrir interações que afetam o resultado
Razões cruzadas	vendas_por_cliente / vendas_total	Medir proporções entre dimensões

# Estatísticas por grupo

Técnica	Exemplo	Quando usar
Média por grupo	salário_médio_por_cargo	Comparar indivíduo com grupo
Desvio padrão por grupo	variabilidade_dentro_da_cidade	Medir dispersão contextual
Ranking	ranking_de_vendas_no_estado	Ver performance relativa

# Encoding avançado (para ML)

Técnica	Exemplo	Quando usar
Embeddings (NLP, Deep Learning)	Vetores de palavras ou categorias	Modelos de aprendizado profundo
Polynomial features	$x$ , $x^2$ , $x^3$ , etc.	Regressões não lineares
PCA / Redução de dimensão	Transformar muitas variáveis correlacionadas	Evitar overfitting e melhorar performance

# Exemplo: Churn em telecom

Problema: Prever quais clientes vão cancelar o serviço.

✗ Sem feature engineering:

- Usar apenas variáveis como idade, plano atual e tempo de contrato.

✓ Com feature engineering:

- Criada a variável tempo\_desde\_ultimo\_contato: dias desde o último atendimento ao cliente.
- Criada a variável qtde\_reclamacoes\_ultimos\_3m: número de reclamações recentes.

Por que melhora?

- Clientes que entram em contato com frequência e reclamam bastante têm maior probabilidade de churn. Sem essas variáveis, o modelo ignora sinais comportamentais críticos.

# Exemplo: Previsão de vendas

Problema: Estimar a venda diária de uma loja.

✗ Sem feature engineering:

- Usa data da venda, código da loja, região.

✓ Com feature engineering:

- Criada a variável `é_feriado`: flag indicando se é feriado.
- Criada a variável `dias_ate_fim_do_mes`: número de dias até encerrar o mês.

Por que melhora?

- Vendas tendem a crescer perto do fim do mês (salário) e cair em feriados. Com essas variáveis, o modelo entende melhor o comportamento temporal.

# Exemplo: Crédito bancário

Problema: Prever inadimplência de clientes.

✗ Sem feature engineering:

- Só usa renda, idade, valor da dívida.

✓ Com feature engineering:

- Criada a variável `comprometimento_renda = valor_dívida / renda_mensal`.
- Criada `historico_pagamentos_atrasados`: conta atrasos anteriores.

Por que melhora?

- Apenas saber o valor da dívida não é suficiente. A relação dívida/renda captura a pressão financeira real. O histórico também é um preditor essencial de comportamento futuro.

# Exemplo: Pedidos fraudulentos

Problema: Identificar pedidos suspeitos de fraude em E-commerce

✗ Sem feature engineering:

- Usa valor do pedido, estado de entrega, data.

✓ Com feature engineering:

- Criada  $\text{tempo\_conta\_criada} = \text{data\_pedido} - \text{data\_criacao\_conta}$ .
- Criada  $\text{dispositivo\_diferente} = \text{dispositivo\_último\_login} \neq \text{dispositivo\_atual}$ .

Por que melhora?

- Fraudes geralmente vêm de contas recém-criadas e com comportamentos diferentes do padrão. Essas variáveis capturam isso.

# Exemplo: Prever internações

Problema: Prever risco de internação nos próximos 6 meses.

✗ Sem feature engineering:

- Só usa idade, histórico de doenças, consultas anteriores.

✓ Com feature engineering:

- Criada `media_intervalo_consultas`: tempo médio entre visitas médicas.
- Criada `qtde_medicamentos_diferentes`: número de remédios distintos em us

Por que melhora?

- Frequência médica anormal ou polifarmácia (muitos medicamentos) são preditores de piora clínica. Isso não é evidente sem essas novas variáveis.

# Passo-a-passo de feature engineering

## 1. Entenda profundamente o problema de negócio

Antes de qualquer linha de código, faça perguntas como:

- O que estou tentando prever ou explicar?
- Quais fatores potencialmente influenciam esse resultado?
- Como os dados foram gerados?

Exemplo: Se você está prevendo churn, entenda o que costuma causar a saída dos clientes.

# Passo-a-passo de feature engineering

## 2. Explore os dados com atenção

Use técnicas de EDA (Análise Exploratória de Dados):

- Verifique tipos de variáveis, nulos, distribuição, valores extremos.
- Use `.describe()`, `.info()`, `.value_counts()`, histogramas, scatter plots, etc.

Exemplo: Descubra se salário tem outliers, se estado civil tem categorias pouco frequentes.

# Passo-a-passo de feature engineering

## 3. Crie novas variáveis com base em lógica de negócio

Busque agregar inteligência ao modelo com variáveis mais explicativas:

- Utilize as técnicas citadas anteriormente

Boas features são aquelas que melhor traduzem comportamentos e padrões.

# Passo-a-passo de feature engineering

## 4. Normalize ou padronize variáveis quando necessário

Especialmente útil para:

- Modelos lineares e regressão logística.
- Algoritmos baseados em distância (KNN, SVM, PCA, clustering).

# Passo-a-passo de feature engineering

## 5. Elimine ou combine variáveis colineares

- Use VIF (Variance Inflation Factor).
- Evite deixar variáveis muito correlacionadas juntas, elas causam instabilidade e dificultam interpretação.

# Passo-a-passo de feature engineering

## 6. Faça feature selection (seleção de variáveis relevantes)

- **Filter Methods (pré-modelo):** Seleccionam com base em estatísticas simples, sem usar modelo.
  - Correlação, Chi-quadrado, ANOVA...
- **Wrapper Methods:** usam o modelo para testar variáveis
  - Recursive Feature Elimination (RFE) (removendo features e testando performance) e Sequential Feature Selection (SFS) (Adiciona ou remove uma a uma)
- **Embedded Methods:** integrado ao modelo
  - Árvore de decisão
- **Análise de Colinearidade:** VIF

# Passo-a-passo de feature engineering

## 7. Valide o impacto das features na performance

- Compare o modelo com e sem certas features.
- Acompanhe métricas como  $R^2$ , RMSE, AUC, precisão, F1.
- Faça validação cruzada (dividir os dados em múltiplos subconjuntos)

Se a nova feature não melhora performance ou interpretabilidade, **remova**.

# Passo-a-passo de feature engineering

 **Dica de bônus: envolva o time de negócio!**

Eles têm conhecimento sobre as variáveis e podem sugerir insights que não aparecem no dado cru...

# Conclusão

- Modelos aprendem com os dados que você entrega
- Boas features capturam conhecimento do negócio