Teoria do Aprendizado Estatístico

Atividade 2

Luiz Henrique Barretta Francisco - 202100155302

abril/2025

Funções de perda no algoritmo KNN

Usualmente, as funções de perda associadas ao algoritmo **KNN (K-Vizinhos Mais Próximos)** são distintas dependendo da característica de seus dados:

A função de perda utilizada no KNN para classificação é a função de perda 0-1, definida como:

$$L(y_i, \hat{y}_i) = \begin{cases} 1, & \text{se } y_i \neq \hat{y}_i \\ 0, & \text{se } y_i = \hat{y}_i \end{cases}$$

Essa função penaliza qualquer erro de classificação com custo 1 e acertos com custo 0.

No caso da regressão com KNN, o modelo realiza a média dos valores das K observações mais próximas. A função de perda comumente associada é a **perda quadrática (L2)**:

$$L(y_i, \hat{y}_i) = (y_i - \hat{y}_i)^2$$

Proposta de nova função de perda para o KNN: perda de Huber

Nesta proposta, substituímos a função de perda quadrática tradicional utilizada no KNN por uma função mais robusta: a **função de perda de Huber**. Essa função combina os benefícios da perda quadrática (boa suavidade para pequenos erros) com a perda absoluta (robustez contra outliers).

A função de Huber é definida da seguinte forma:

$$L(y_i, \hat{y}_i) = \begin{cases} \frac{1}{2}(y_i - \hat{y}_i)^2, & \text{se } |y_i - \hat{y}_i| \leq \delta \\ \delta \cdot (|y_i - \hat{y}_i| - \frac{1}{2}\delta), & \text{caso contrário} \end{cases}$$

Esse tipo de função de perda é amplamente utilizado em regressão robusta por permitir que erros pequenos sejam tratados de maneira suave, enquanto evita penalizações excessivas para observações distantes (outliers).

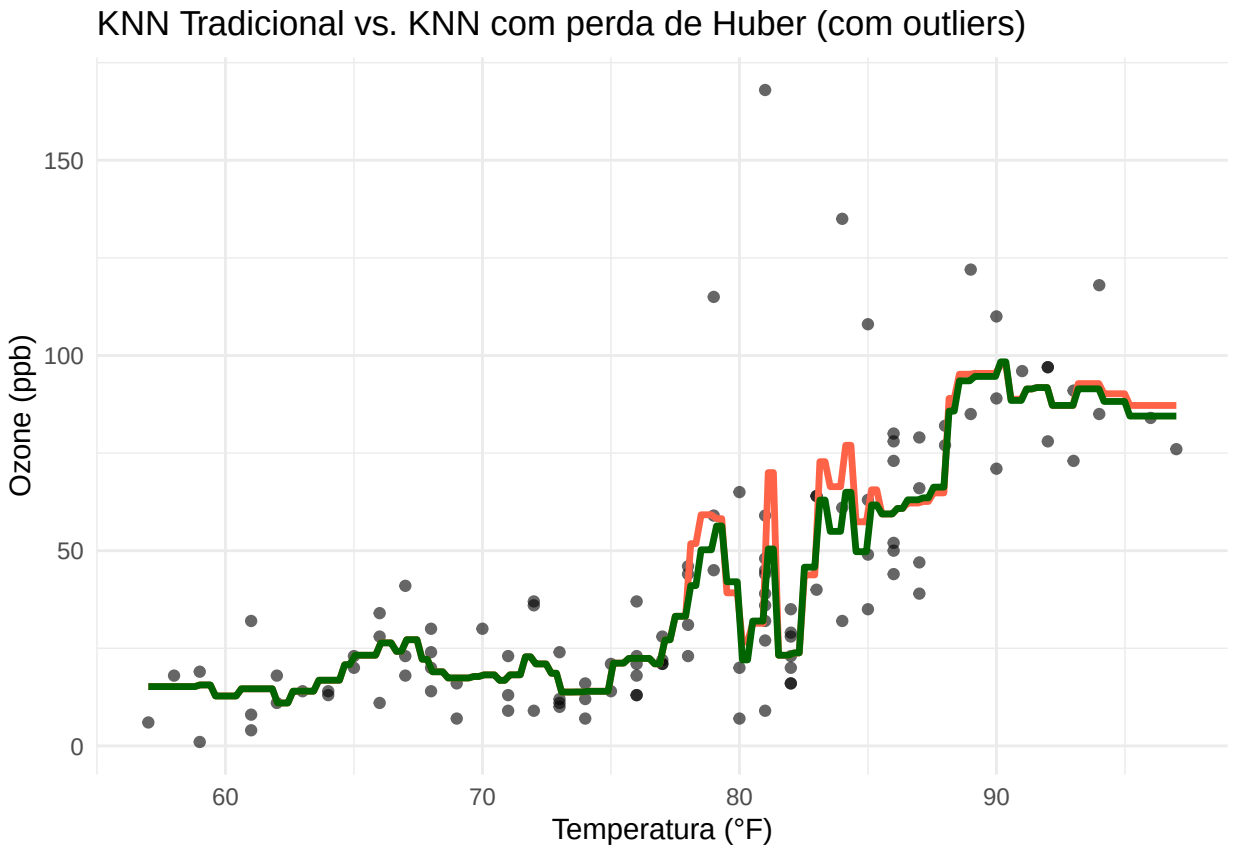
Adaptando essa perda ao KNN, realizamos a predição $\hat{y}$ que minimiza a soma das perdas de Huber nos K vizinhos mais próximos:

$$\hat{y} = \arg\min_{y \in \mathbb{R}} \sum_{i \in \mathcal{N}_x} L(y_i, y)$$

Implementação

```
library(ggplot2)
library(FNN)
library(dplyr)
library(tidyr)
library(patchwork)

dados <- na.omit(airquality[, c("Temp", "Ozone")])
colnames(dados) <- c("x", "y")

k <- 5
delta <- 20
```

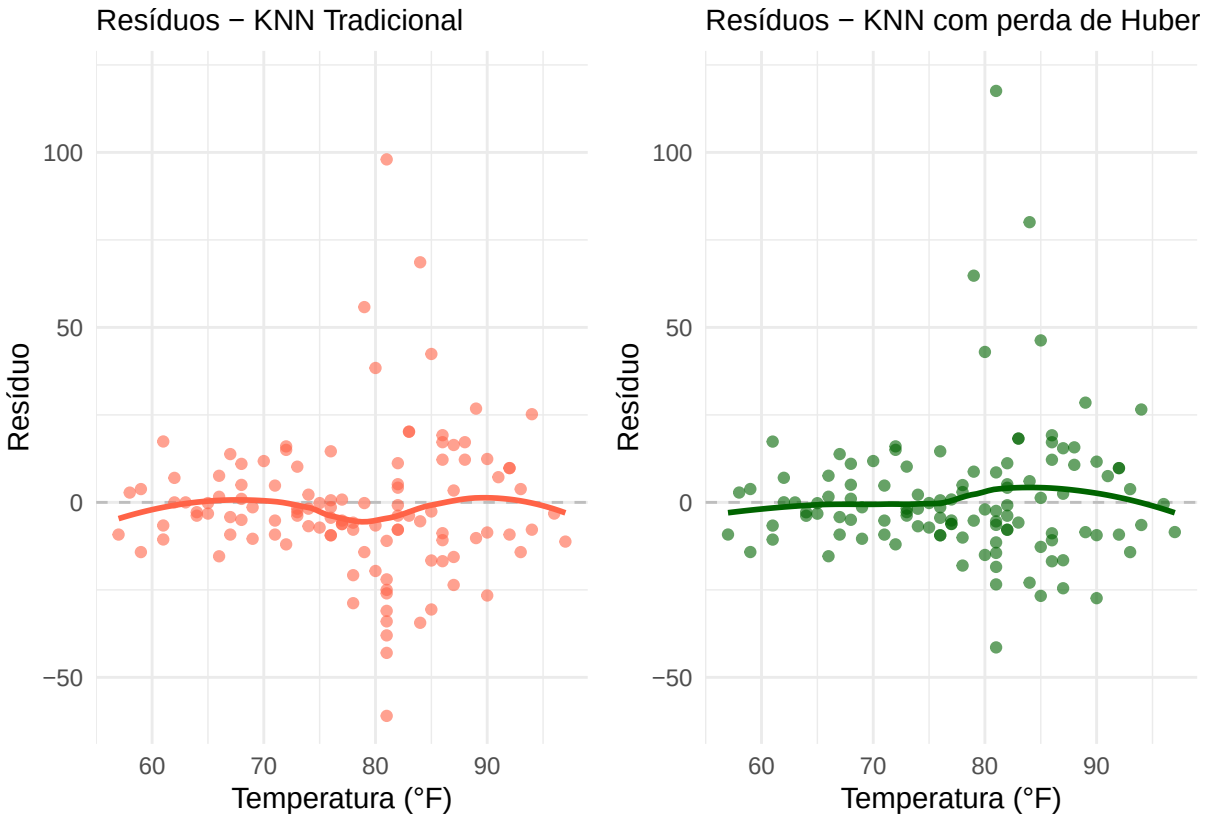
```r
huber_loss <- function(y_i, y_hat, delta) {
  diff <- y_i - y_hat
  if (abs(diff) <= delta) return(0.5 * diff^2)
  else return(delta * (abs(diff) - 0.5 * delta))}

knn_tradicional <- function(x_teste, dados, k) {
  nn <- get.knnx(data = matrix(dados$x), query = matrix(x_teste), k = k)
  apply(nn$nn.index, 1, function(idx) mean(dados$y[idx]))}

knn_huber <- function(x_teste, dados, k, delta) {
  nn <- get.knnx(data = matrix(dados$x), query = matrix(x_teste), k = k)
  sapply(1:length(x_teste), function(i) {
    idx <- nn$nn.index[i,]
    y_vizinhos <- dados$y[idx]
    candidato_y <- seq(min(y_vizinhos) - 20, max(y_vizinhos) + 20, length.out = 1000)
    perdas <- sapply(candidato_y, function(yhat) {
      sum(sapply(y_vizinhos, function(y_i) huber_loss(y_i, yhat, delta)))})
    candidato_y[which.min(perdas)]})}

x_grid <- seq(min(dados$x), max(dados$x), length.out = 200)
y_trad <- knn_tradicional(x_grid, dados, k)
y_huber <- knn_huber(x_grid, dados, k, delta)

df_plot <- data.frame(x = x_grid, Tradicional = y_trad, Huber = y_huber)
ggplot() + geom_point(data = dados, aes(x, y), alpha = 0.6) +
           geom_line(data = df_plot, aes(x, Tradicional), color = "tomato", size = 1.2) +
           geom_line(data = df_plot, aes(x, Huber), color = "darkgreen", size = 1.2) +
           labs(title = "KNN Tradicional vs. KNN com perda de Huber (com outliers)",
           y = "Ozone (ppb)", x = "Temperatura (°F)") + theme_minimal()
```



O gráfico mostra as predições dos modelos KNN tradicional (em vermelho) e KNN com perda de Huber (em verde) sobre os dados de ozônio em função da temperatura, na presença de outliers. Observa-se que, nas regiões com maior concentração de valores extremos (notadamente entre 78°F e 85°F), o KNN tradicional apresenta oscilações abruptas, refletindo forte influência dos outliers. Em contraste, o KNN com perda de Huber produz uma curva de predição mais suave e estável, demonstrando maior robustez ao ruído. Assim, o modelo com perda de Huber é mais adequado para manter a fidelidade à tendência central dos dados mesmo quando há observações atípicas.

```
pred_trad <- knn_tradicional(dados$x, dados, k)
pred_huber <- knn_huber(dados$x, dados, k, delta)

df_erros <- dados %>%
            mutate(erro_trad = (y - pred_trad)^2,
                   erro_huber = sapply(1:n(), function(i) huber_loss(y[i], pred_huber[i], delta)),
                   residuo_trad = y - pred_trad, residuo_huber = y - pred_huber)

limites_residuos <- c(-60, 120)

g1 <- ggplot(df_erros, aes(x = x, y = residuo_trad)) +
      geom_hline(yintercept = 0, linetype = "dashed", color = "gray") +
      geom_point(color = "tomato", alpha = 0.6) +
      geom_smooth(se = FALSE, method = "loess", color = "tomato") +
      labs(title = "Resíduos - KNN Tradicional", x = "Temperatura (°F)", y = "Resíduo") +
      coord_cartesian(ylim = limites_residuos) + theme_minimal() +
      theme(plot.title = element_text(size = 11))

g2 <- ggplot(df_erros, aes(x = x, y = residuo_huber)) +
```

```
      geom_hline(yintercept = 0, linetype = "dashed", color = "gray") +
      geom_point(color = "darkgreen", alpha = 0.6) +
      geom_smooth(se = FALSE, method = "loess", color = "darkgreen") +
      labs(title = "Resíduos - KNN com perda de Huber", x = "Temperatura (°F)", y = "Resíduo") +
      coord_cartesian(ylim = limites_residuos) +
      theme_minimal() + theme(plot.title = element_text(size = 11))
g1 + g2
```



Conclusão

A análise dos gráficos de resíduos mostra que o modelo KNN tradicional (à esquerda) apresenta maior sensibilidade a valores extremos (outliers), evidenciada pela presença de resíduos muito distantes de zero, especialmente em temperaturas entre 75°F e 85°F. Isso indica que algumas observações estão influenciando desproporcionalmente as previsões do modelo, comprometendo a robustez do ajuste. Além disso, o padrão dos resíduos sugere certa tendência sistemática, o que pode indicar um viés no modelo ou sensibilidade a ruídos.

Já o modelo KNN com perda de Huber (à direita) apresenta uma distribuição de resíduos mais concentrada em torno de zero, com menos influência de outliers. Apesar de ainda existirem alguns resíduos altos, o comportamento geral do gráfico revela um ajuste mais robusto, com menor variabilidade nas previsões. A perda de Huber, por suavizar penalizações para erros grandes, torna o modelo mais resistente a observações discrepantes. Essa robustez é desejável especialmente em contextos com dados reais, onde outliers podem distorcer a inferência. Assim, a escolha pelo KNN com perda de Huber é justificada pela melhora na estabilidade e pela redução da sensibilidade a ruídos nos dados. Portanto, essa nova abordagem de função de perda para o KNN pode ser interessante em alguns casos.

Referências

HUBER, Peter J. *Robust estimation of a location parameter*. The Annals of Mathematical Statistics, v. 35, n. 1, p. 73–101, 1964. DOI: https://doi.org/10.1214/aoms/1177703732.

FIX, Evelyn; HODGES JR., Joseph L. *Discriminatory analysis. Nonparametric discrimination: consistency properties*. Berkeley: USAF School of Aviation Medicine, Randolph Field, Texas, 1951. (Technical Report 4, Project 21-49-004, USAF School of Aviation Medicine).