PLS & CART

Marjete Vucinaj

2024-12-11

```
library(readr)
library(pls)
```

```
##
## Attaching package: 'pls'

## The following object is masked from 'package:stats':
##
##     loadings
```

```
library(magrittr)
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v purrr     1.0.2
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.1     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.1

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x tidyr::extract()   masks magrittr::extract()
## x dplyr::filter()    masks stats::filter()
## x dplyr::lag()       masks stats::lag()
## x purrr::set_names() masks magrittr::set_names()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(caret)
```

```
## Loading required package: lattice
##
## Attaching package: 'caret'
##
## The following object is masked from 'package:purrr':
##
##     lift
##
## The following object is masked from 'package:pls':
##
##     R2
```

```r
library(rpart)
library(doParallel)
```

```
## Loading required package: foreach
##
## Attaching package: 'foreach'
##
## The following objects are masked from 'package:purrr':
##
##     accumulate, when
##
## Loading required package: iterators
## Loading required package: parallel
```

Response Variable.: PH

```r
imputed_data <- read_csv("imputed_test_data.csv", show_col_types = FALSE)
```

```r
# Split
train_index <- createDataPartition(imputed_data$PH, p = 0.75, list = FALSE)
train_data <- imputed_data[train_index, ]
test_data <- imputed_data[-train_index, ]
```

```r
pls_model <- plsr(PH ~ ., data = train_data, ncomp = 10, validation = "CV")

optimal_components <- which.min(RMSEP(pls_model)$val[1, , -1])

pls_final_model <- plsr(PH ~ ., data = train_data, ncomp = optimal_components)

# Predictions and performance evaluation
predictions <- predict(pls_final_model, newdata = test_data, ncomp = optimal_components)
rmse <- sqrt(mean((test_data$PH - predictions)^2))
r_squared <- cor(test_data$PH, predictions)^2
mae <- mean(abs(test_data$PH - predictions))

cat("Optimal Components:", optimal_components, "\n")
```

```
## Optimal Components: 10
```

```r
cat("Test RMSE:", rmse, "\n")
```

```
## Test RMSE: 0.1414512
```

```r
cat("Test R²:", r_squared, "\n")
```

```
## Test R²: 0.3038614
```

```r
cat("Test MAE:", mae, "\n")
```

```
## Test MAE: 0.1111375
```

```
#also ran the mode wil ncomp= 7 for simplicity and r^2 was smaller
```

CART: Regression: response variable is numerical and continuous.

```
cl <- makeCluster(detectCores() - 1)
registerDoParallel(cl)

#tuning grid
tune_grid <- expand.grid(
  cp = seq(0.001, 0.05, by = 0.005)
)

optimized_train_control <- trainControl(
  method = "cv",
  number = 5,
  verboseIter = FALSE,
  allowParallel = TRUE
)

# Train  using caret
optimized_cart_model <- train(
  PH ~ .,
  data = train_data,
  method = "rpart",
  trControl = optimized_train_control,
  tuneGrid = tune_grid
)

best_hyperparameters <- optimized_cart_model$bestTune
cat("Best Hyperparameter (cp):\n")
```

```
## Best Hyperparameter (cp):
```

```
print(best_hyperparameters)
```

```
##      cp
## 1 0.001
```

```
final_predictions <- predict(optimized_cart_model, newdata = test_data)

cart_rmse <- sqrt(mean((test_data$PH - final_predictions)^2))
cart_r_squared <- cor(test_data$PH, final_predictions)^2
cart_mae <- mean(abs(test_data$PH - final_predictions))

cat("Optimized CART Test RMSE:", cart_rmse, "\n")
```

```
## Optimized CART Test RMSE: 0.1248095
```

```
cat("Optimized CART Test R²:", cart_r_squared, "\n")
```

```
## Optimized CART Test R²: 0.4827778
```

```r
cat("Optimized CART Test MAE:", cart_mae, "\n")
```

```
## Optimized CART Test MAE: 0.08998785
```

```r
stopCluster(cl)


#cp of  0.001   is the minimal error
```