Pertussis Challenge 2.1

Runqi Zhang

2024-11-21

```r
# Load required libraries
suppressPackageStartupMessages({
  library(dplyr)         # Data manipulation
  library(tidyr)         # Data reshaping
  library(readr)         # Reading TSV files
  library(glmnet)        # Regularized regression (LASSO, Ridge)
  library(agua)          # H2O AutoML integration
  library(tibble)
})
```

```
## Warning: package 'tidyr' was built under R version 4.3.3

## Warning: package 'readr' was built under R version 4.3.3

## Warning: package 'glmnet' was built under R version 4.3.3

## Warning: package 'agua' was built under R version 4.3.3

## Warning: package 'parsnip' was built under R version 4.3.3
```

```r
# Set working directory and initialize H2O
workDir <- "C:/Users/zhang/Desktop/cmi-pb-3rd-final/Runqi/CMI-PB"
options(readr.show_col_types = FALSE)
agua::h2o_start()
```

```
## Warning: JAVA not found, H2O may take minutes trying to connect.

## Warning in h2o.clusterInfo():
## Your H2O cluster version is (11 months and 1 day) old. There may be a newer version available.
## Please download and install the latest version from: https://h2o-release.s3.amazonaws.com/h2o/latest_
```

```r
# Function to read input files for a given year
read_data <- function(year, type = "LD") {
  list(
    pts = read_tsv(file.path(workDir, paste0("data/", year, type, "_subject.tsv"))),
    sample = read_tsv(file.path(workDir, paste0("data/", year, type, "_specimen.tsv"))),
    ab = read_tsv(file.path(workDir, paste0("data/", year, type, "_plasma_ab_titer.tsv"))),
    fcm = read_tsv(file.path(workDir, paste0("data/", year, type, "_pbmc_cell_frequency.tsv")))
  )
}

# Load datasets for 2020, 2021, 2022, and 2023
data2020 <- read_data("2020")
data2021 <- read_data("2021")
data2022 <- read_data("2022")
data2023 <- read_data("2023", type = "BD")
```

```r
# Extract percent_live_cell for Monocytes from FCM data
prepare_outcome <- function(fcm_data, sample_data, pts_data) {
  fcm_data %>%
    filter(cell_type_name == "Monocytes") %>%
    inner_join(sample_data, by = "specimen_id") %>%
    inner_join(pts_data, by = "subject_id") %>%
    filter(planned_day_relative_to_boost == 1) %>%
    dplyr::select(subject_id, percent_live_cell)
}

# Prepare outcome data for each year
yDF <- prepare_outcome(data2020$fcm, data2020$sample, data2020$pts)
y2DF <- prepare_outcome(data2021$fcm, data2021$sample, data2021$pts)
y3DF <- prepare_outcome(data2022$fcm, data2022$sample, data2022$pts)
```

```r
# Prepare predictors for Monocytes using baseline data
prepare_predictors <- function(fcm_data, sample_data, pts_data) {
  fcm_data %>%
    mutate(cell_type_name = make.names(cell_type_name)) %>%  # Clean column names
    inner_join(sample_data, by = "specimen_id") %>%          # Merge with sample data
    inner_join(pts_data, by = "subject_id") %>%             # Merge with patient data
    filter(planned_day_relative_to_boost == 0) %>%          # Filter baseline data
    dplyr::select(subject_id, cell_type_name, percent_live_cell) %>%
    pivot_wider(
      names_from = cell_type_name,
      values_from = percent_live_cell,
      values_fn = mean,  # Aggregate duplicates using the mean
      values_fill = 0    # Fill missing values with 0
    ) %>%
    column_to_rownames(var = "subject_id")                  # Set rownames
}


# Prepare predictors for each year
xDF <- prepare_predictors(data2020$fcm, data2020$sample, data2020$pts)
x2DF <- prepare_predictors(data2021$fcm, data2021$sample, data2021$pts)
x3DF <- prepare_predictors(data2022$fcm, data2022$sample, data2022$pts)
x4DF <- prepare_predictors(data2023$fcm, data2023$sample, data2023$pts)

xDF[is.na(xDF)] <- 0
x2DF[is.na(x2DF)] <- 0
x3DF[is.na(x3DF)] <- 0
x4DF[is.na(x4DF)] <- 0

# Align and scale predictors
```

```r
common_cols <- Reduce(intersect, list(colnames(xDF), colnames(x2DF), colnames(x3DF), colnames(x4DF)))
xDF <- scale(xDF[, common_cols, drop = FALSE])
x2DF <- scale(x2DF[, common_cols, drop = FALSE])
x3DF <- scale(x3DF[, common_cols, drop = FALSE])
x4DF <- scale(x4DF[, common_cols, drop = FALSE])
```

```r
# Combine predictors and response variable for training
trainDF <- rbind(xDF, x2DF, x3DF) %>%
  as.data.frame() %>%
  mutate(percent_live_cell = c(yDF$percent_live_cell, y2DF$percent_live_cell, y3DF$percent_live_cell))

# Train regression model using H2O AutoML
set.seed(3)
auto_fit <- auto_ml() %>%
  set_engine("h2o", max_runtime_secs = 5) %>%
  set_mode("regression") %>%
  fit(percent_live_cell ~ ., data = trainDF)
```

```r
# Predict on training data
train_predictions <- predict(auto_fit, new_data = trainDF)$.pred

# Calculate correlations
pearson_cor <- cor(train_predictions, trainDF$percent_live_cell, method = "pearson")
spearman_cor <- cor(train_predictions, trainDF$percent_live_cell, method = "spearman")

# Display correlation results
cat("Pearson Correlation: ", pearson_cor, "\n")
```

```
## Pearson Correlation:  0.7111069
```

```r
cat("Spearman Correlation: ", spearman_cor, "\n")
```

```
## Spearman Correlation:  0.6705863
```

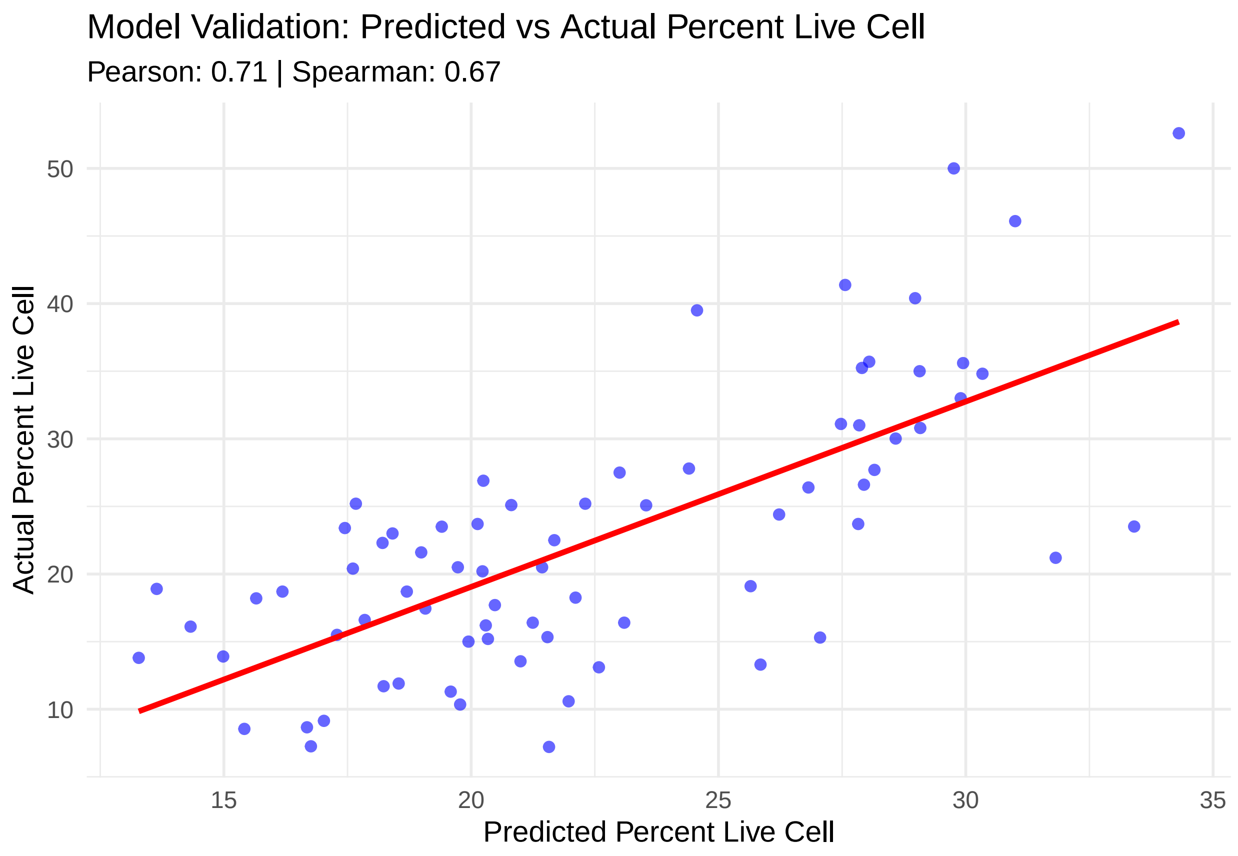
```r
# Create a correlation plot
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 4.3.3
```

```r
correlation_plot <- ggplot(data = data.frame(
  Predicted = train_predictions,
  Actual = trainDF$percent_live_cell
), aes(x = Predicted, y = Actual)) +
  geom_point(alpha = 0.6, color = "blue") +  # Scatter plot
  geom_smooth(method = "lm", color = "red", se = FALSE) +  # Regression line
  labs(
    title = "Model Validation: Predicted vs Actual Percent Live Cell",
    subtitle = paste0("Pearson: ", round(pearson_cor, 2),
                      " | Spearman: ", round(spearman_cor, 2)),
    x = "Predicted Percent Live Cell",
    y = "Actual Percent Live Cell"
  ) +
  theme_minimal()

# Display the plot
print(correlation_plot)
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



```
# Predict on 2023 data and rank predictions
yhat <- predict(auto_fit, new_data = x4DF)$.pred
rhat <- rank(-1 * yhat)  # Rank predictions in descending order
print(cbind(rownames(x4DF), rhat))
```

```
##             rhat
##  [1,] "146" "38"
##  [2,] "163" "16"
##  [3,] "124" "33"
##  [4,] "134" "23"
##  [5,] "170" "9"
##  [6,] "132" "40"
##  [7,] "140" "13"
##  [8,] "155" "6"
##  [9,] "172" "18"
## [10,] "148" "17"
## [11,] "135" "1"
## [12,] "123" "4"
## [13,] "128" "45"
## [14,] "164" "12"
## [15,] "159" "28"
## [16,] "167" "35"
## [17,] "158" "14"
## [18,] "150" "39"
## [19,] "133" "15"
## [20,] "126" "10"
## [21,] "125" "43"
## [22,] "130" "3"
## [23,] "145" "29"
## [24,] "122" "25"
## [25,] "138" "7"
## [26,] "157" "5"
## [27,] "119" "27"
## [28,] "136" "48"
## [29,] "165" "41"
## [30,] "131" "2"
## [31,] "169" "8"
## [32,] "160" "46"
## [33,] "168" "20"
## [34,] "154" "24"
## [35,] "147" "21"
## [36,] "121" "37"
## [37,] "120" "34"
## [38,] "144" "42"
## [39,] "143" "19"
## [40,] "171" "26"
## [41,] "127" "11"
## [42,] "129" "30"
## [43,] "162" "47"
## [44,] "166" "36"
## [45,] "152" "31"
## [46,] "161" "44"
## [47,] "137" "32"
## [48,] "139" "22"
```

```r
# Read submission template and update rankings
submission_file <- file.path(workDir, "3rdChallengeSubmissionTemplate_revised.tsv")
data <- read_tsv(submission_file)

ranking_df <- data.frame(
  SubjectID = as.numeric(rownames(x4DF)),
  `2.1) Monocytes-D1-Rank` = rhat,
  check.names = FALSE # Prevent automatic renaming of column names
)

data <- data %>%
  mutate(
    `2.1) Monocytes-D1-Rank` = ifelse(
      SubjectID %in% ranking_df$SubjectID,
      ranking_df$`2.1) Monocytes-D1-Rank`[match(SubjectID, ranking_df$SubjectID)],
      `2.1) Monocytes-D1-Rank`
    )
  )

# Save updated submission file
write_tsv(data, submission_file)
```

```
# End H2O session and display session info
agua::h2o_end()
sessionInfo()
```

```
## R version 4.3.1 (2023-06-16 ucrt)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 11 x64 (build 22631)
##
## Matrix products: default
##
##
## locale:
## [1] LC_COLLATE=English_United States.utf8
## [2] LC_CTYPE=English_United States.utf8
## [3] LC_MONETARY=English_United States.utf8
## [4] LC_NUMERIC=C
## [5] LC_TIME=English_United States.utf8
##
## time zone: America/Los_Angeles
## tzcode source: internal
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
## [1] ggplot2_3.5.1  tibble_3.2.1   agua_0.1.4     parsnip_1.2.1  glmnet_4.1-8
## [6] Matrix_1.5-4.1 readr_2.1.5    tidyr_1.3.1    dplyr_1.1.3
##
## loaded via a namespace (and not attached):
##  [1] tidyselect_1.2.1    timeDate_4041.110   farver_2.1.2
##  [4] bitops_1.0-9        fastmap_1.2.0       RCurl_1.98-1.16
##  [7] digest_0.6.37       rpart_4.1.19        timechange_0.3.0
## [10] lifecycle_1.0.4     yardstick_1.3.1     survival_3.5-5
## [13] magrittr_2.0.3      compiler_4.3.1      rlang_1.1.1
## [16] tools_4.3.1         utf8_1.2.3          yaml_2.3.10
## [19] data.table_1.16.2   knitr_1.49          labeling_0.4.3
## [22] bit_4.5.0           curl_6.0.1          DiceDesign_1.10
## [25] withr_3.0.2         purrr_1.0.2         workflows_1.1.4
## [28] h2o_3.44.0.3        nnet_7.3-19         grid_4.3.1
## [31] tune_1.2.1          fansi_1.0.4         colorspace_2.1-0
## [34] future_1.34.0       globals_0.16.3      scales_1.3.0
## [37] iterators_1.0.14    MASS_7.3-60         cli_3.6.1
## [40] crayon_1.5.3        rmarkdown_2.29      generics_0.1.3
## [43] rstudioapi_0.17.1   future.apply_1.11.3 tzdb_0.4.0
## [46] splines_4.3.1       dials_1.3.0         parallel_4.3.1
## [49] vctrs_0.6.3         hardhat_1.4.0       jsonlite_1.8.9
## [52] hms_1.1.3           bit64_4.5.2         listenv_0.9.1
## [55] foreach_1.5.2       gower_1.0.1         recipes_1.1.0
## [58] glue_1.6.2          parallelly_1.39.0   codetools_0.2-19
## [61] rsample_1.2.1       lubridate_1.9.3     shape_1.4.6.1
## [64] gtable_0.3.6        munsell_0.5.1       GPfit_1.0-8
## [67] pillar_1.9.0        furrr_0.3.1         htmltools_0.5.8.1
## [70] ipred_0.9-15        lava_1.8.0          R6_2.5.1
## [73] lhs_1.2.0           vroom_1.6.5         evaluate_1.0.1
## [76] lattice_0.21-8      class_7.3-22        Rcpp_1.0.11
## [79] nlme_3.1-162        prodlim_2024.06.25  mgcv_1.8-42
## [82] xfun_0.48           pkgconfig_2.0.3
```