Predictions of the SARS-CoV-2 Omicron Variant (B.1.1.529) Spike Protein Receptor-Binding Domain Structure and Neutralizing Antibody Interactions: HADDOCK results

Colby T. Ford, Denis Jacob Machado, Daniel A. Janies

December 10, 2021

Contents

Preamble

```r
# Load libraries:
library(ggplot2) # For the plots
library(ggsignif) # To perform Wilcoxon/Mann-Whitney tests on vectors of data
library(ggpubr) # For the Kruskal-Wallis test
library(gridExtra) # Miscellaneous Functions for ``Grid'' Graphics

# Load data:
df <- read.csv("haddock_stacks.csv")

# Filter out Beta and Delta:
df <- df[df$variant != "Beta", ]
df <- df[df$variant != "Delta", ]

# Change levels in the vectors (affects ordering):
df$analysis <- factor(df$analysis,
  levels = c("Reference", "AlphaFold2", "RoseTTAFold"))
```

```r
df$variant <- factor(df$variant,
  levels = c("Alpha","Omicron (AlphaFold2)", "Omicron (RoseTTAFold)"))
```

Functions

```r
# Significance fucntion for the boxplots:
sigFunc = function(x){
  if(x < 0.001){"***"}
  else if(x < 0.01){"**"}
  else if(x < 0.05){"*"}
  else{NA}
}

# Kruskal-Wallis test:
kTest = function(DATA){
  kruskal.test(value ~ variant, data = DATA)
}

# Color pallete:

my_colors = c("#23a455", "#2459A3", "#8425A1")

# Boxplots:
boxes = function(DATA, NAME){
  ggplot(data=DATA, aes(x=variant, y=value, fill=analysis)) +
  geom_boxplot(alpha = 0.5) +
  geom_signif(test="wilcox.test",
              comparisons = combn(levels(DATA$variant),
                                  2,
                                  simplify = F),
              step_increase = 0.15,
              vjust = 0.5,
              map_signif_level = sigFunc,
              textsize = 6) +
  scale_fill_manual(values = my_colors) +
  geom_jitter(alpha = 0.5, width = 0.25) +
  stat_compare_means(test = "kruskal.test", size = 2.5, vjust = 32) +
  scale_y_continuous(expand = c(0.2,0.2)) +
  labs(y = NAME) +
  theme_classic() +
  theme(axis.title.x=element_blank(),
    legend.title = element_blank(),
    axis.text.x=element_text(size=7.5),
    axis.text.y=element_text(size=7.5),
    legend.position = "none")
}

boxes2 = function(DATA, NAME){
  ggplot(data=DATA, aes(x=variant, y=value, fill=analysis)) +
  geom_boxplot(alpha = 0.5) +
  geom_signif(test="wilcox.test",
              comparisons = combn(levels(DATA$variant),
```

```
                                        2,
                                 simplify = F),
                 step_increase = 0.2,
                 map_signif_level = function(p) sprintf("p = %.2g", p),
                 textsize = 3) +
    stat_compare_means(test="kruskal.test",
                       textsize = 3,
                       vjust = 16,
                       hjust = 0.5) +
    scale_fill_manual(values = my_colors) +
    geom_jitter(alpha = 0.5, width = 0.25) +
    scale_y_continuous(expand = c(0.2,0.2)) +
    labs(y = NAME) +
    theme_classic() +
    theme(axis.title.x=element_blank(),
      legend.title = element_blank(),
      legend.position = "bottom")
}

get_legend<-function(PLOT){
    tmp <- ggplot_gtable(ggplot_build(PLOT))
    leg <- which(sapply(tmp$grobs, function(x) x$name) == "guide-box")
    legend <- tmp$grobs[[leg]]
    return(legend)
    }
```

HADDOCK values

```
# Filter:
subset <- df[df$variable == "haddock", ]

# Kruskal-Wallis test:
had_kw <- kTest(subset)
print(had_kw)

##
##  Kruskal-Wallis rank sum test
##
## data:  value by variant
## Kruskal-Wallis chi-squared = 7.7308, df = 2, p-value = 0.02095
```
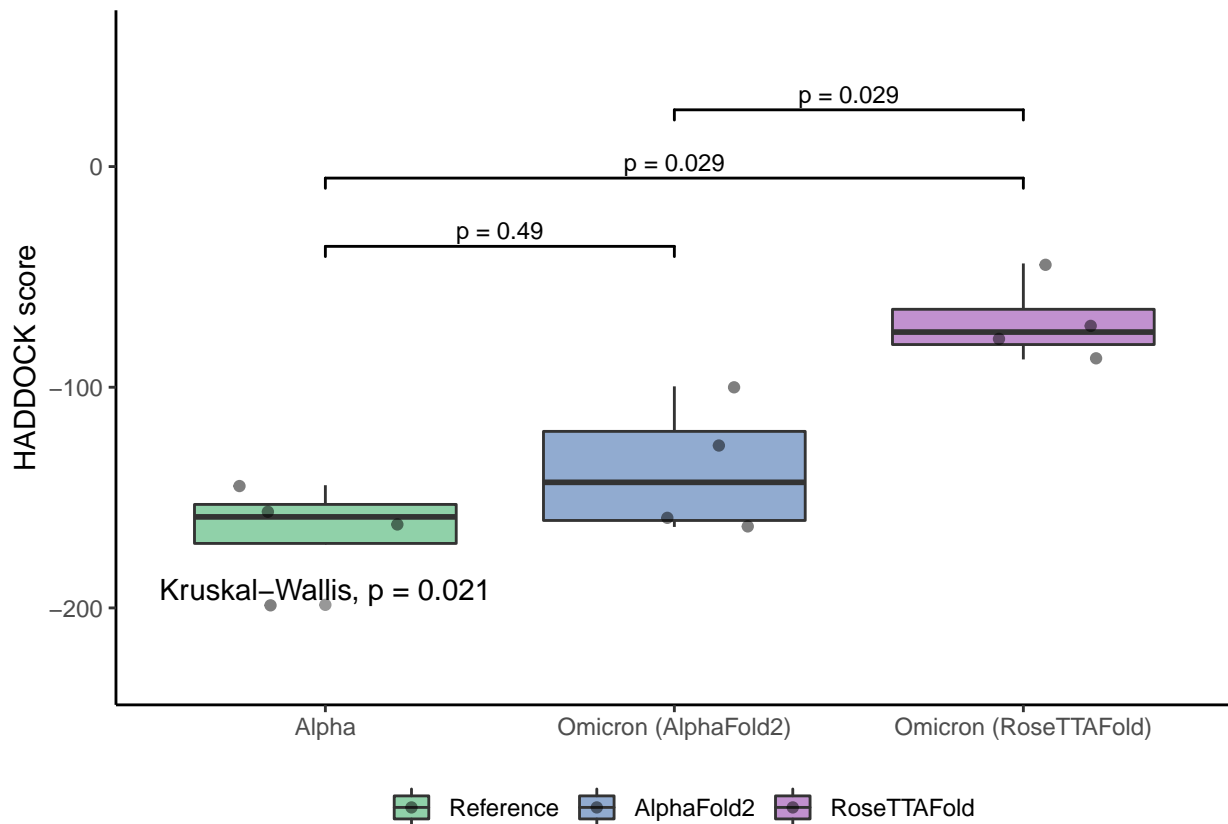```
# Boxplots:
had_plot <- boxes2(subset, "HADDOCK score")

Warning: Ignoring unknown parameters: test, textsize
```
```
print(had_plot)
```

## Van der Waals energy

```
# Filter:
subset <- df[df$variable == "Van_der_Waals",]

# Kruskal-Wallis test:
vdw_kw <- kTest(subset)
print(vdw_kw)
```
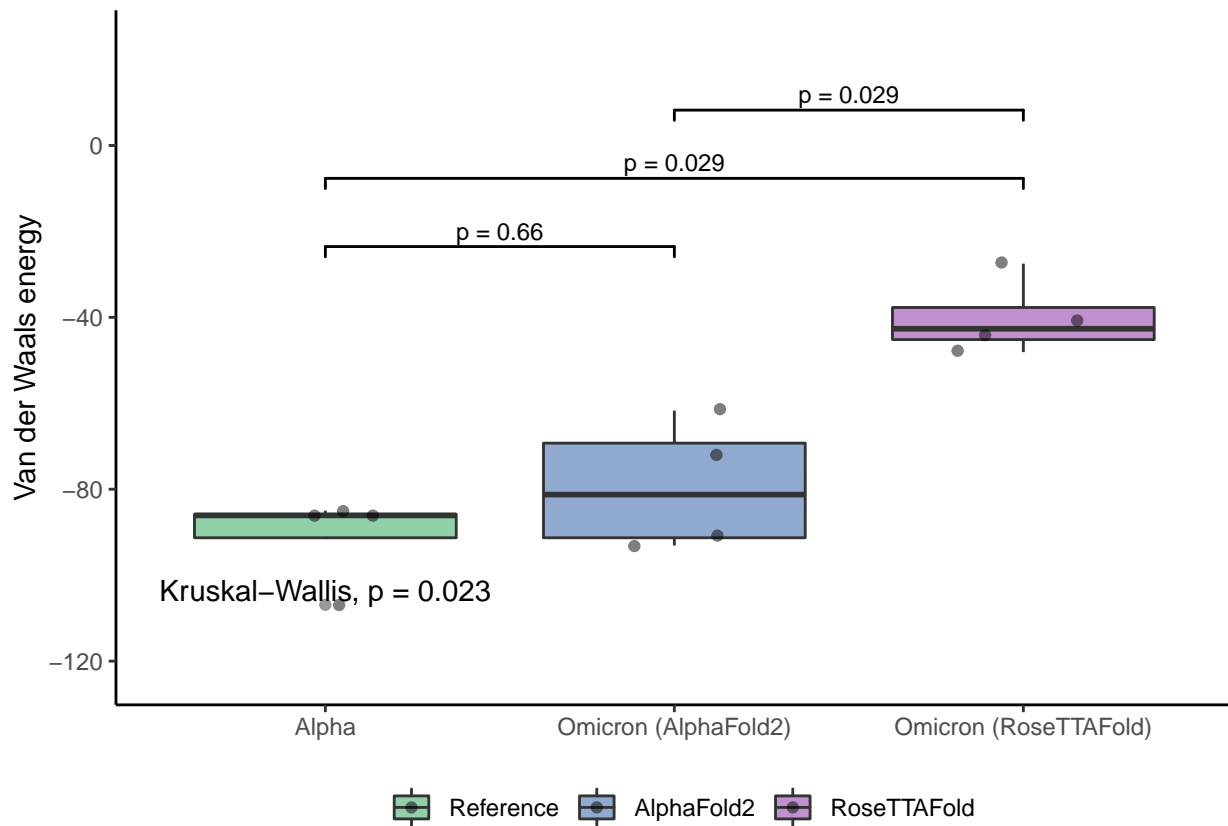
```
##
Kruskal-Wallis rank sum test
##
data: value by variant
Kruskal-Wallis chi-squared = 7.5649, df = 2, p-value = 0.02277
```

```
# Boxplots:
vdw_plot <- boxes2(subset, "Van der Waals energy")
```

```
## Warning: Ignoring unknown parameters: test, textsize
```

```
print(vdw_plot)
```

```
## Warning in wilcox.test.default(c(-106.9, -86.1, -86.1, -85), c(-90.7, -93.1, :
## cannot compute exact p-value with ties
```

```
## Warning in wilcox.test.default(c(-106.9, -86.1, -86.1, -85), c(-27.5, -48.1, :
## cannot compute exact p-value with ties
```
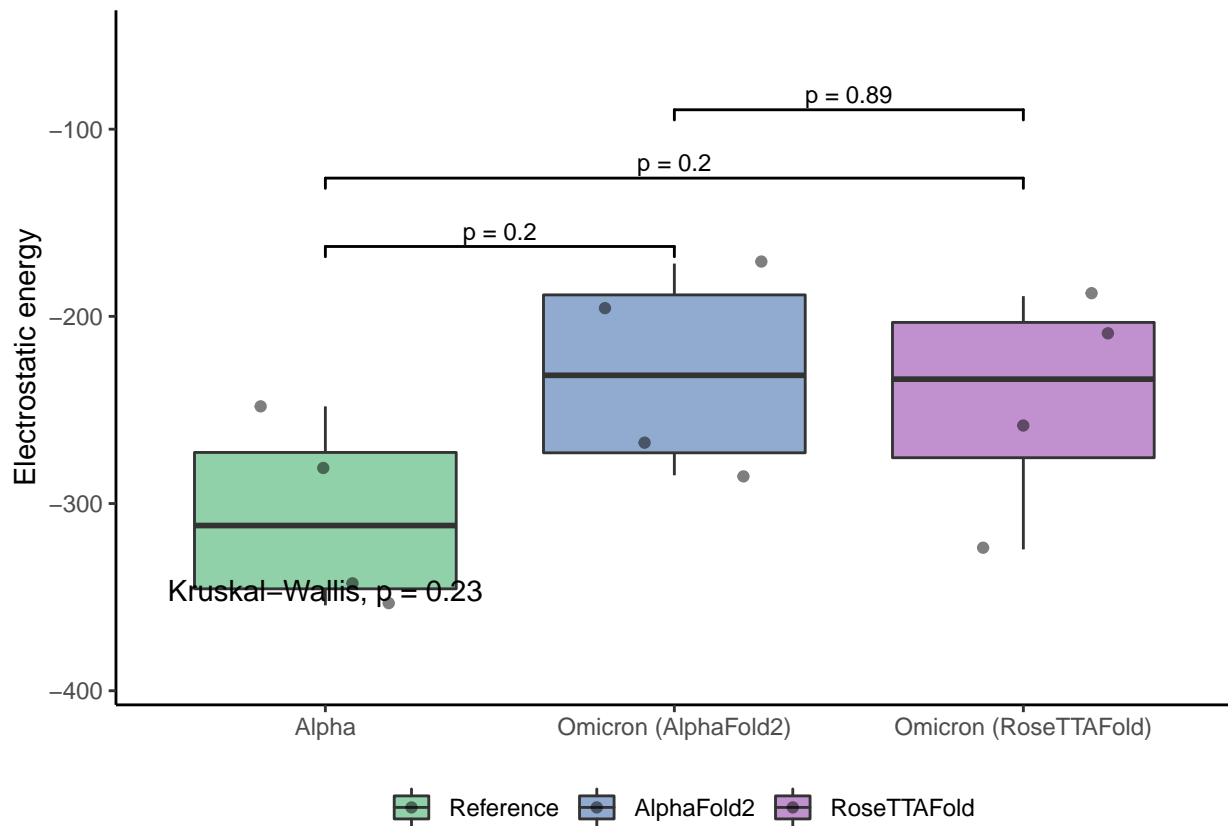
Electrostatic energy

```r
# Filter:
subset <- df[df$variable == "electrostatic", ]

# Kruskal-Wallis test:
elt_kw <- kTest(subset)
print(elt_kw)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  value by variant
## Kruskal-Wallis chi-squared = 2.9231, df = 2, p-value = 0.2319
```

```r
# Boxplots:
elt_plot <- boxes2(subset, "Electrostatic energy")
```

```
## Warning: Ignoring unknown parameters: test, textsize
```

```r
print(elt_plot)
```
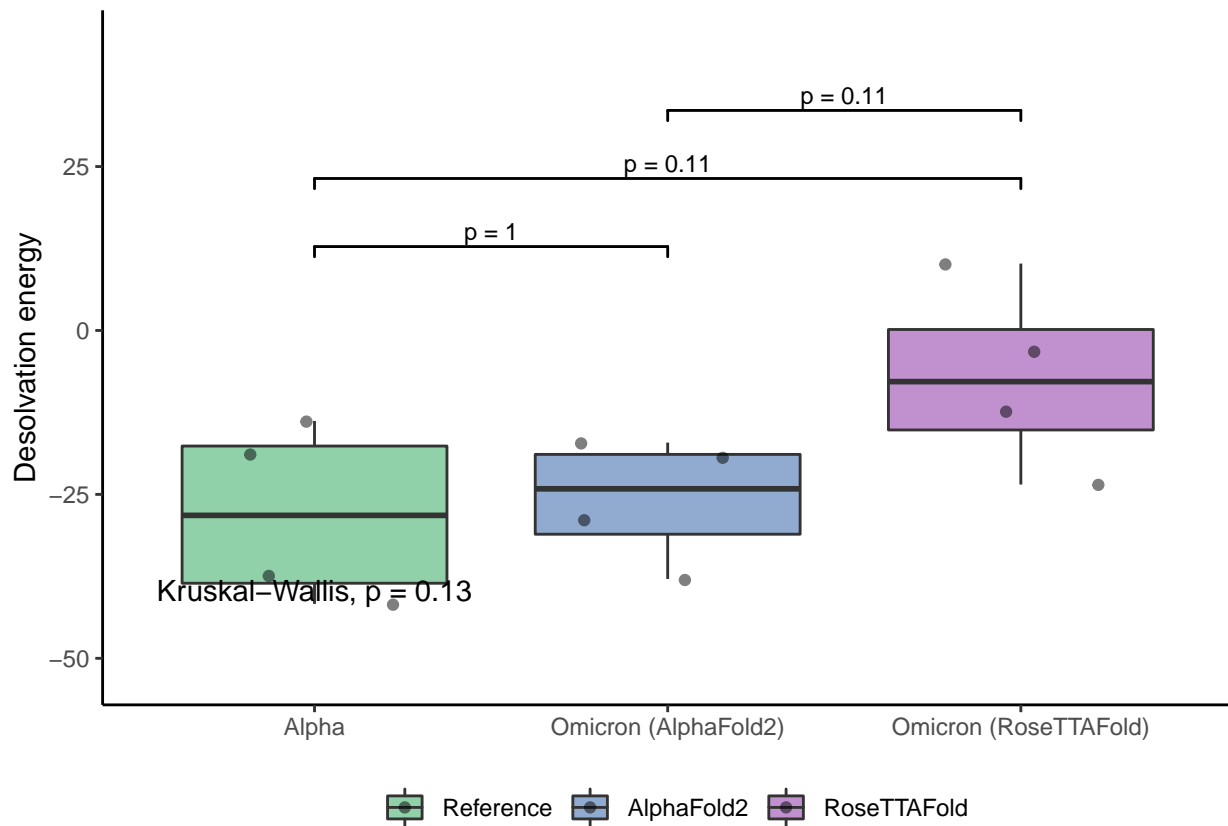
Desolvation energy

```
# Filter:
subset <- df[df$variable == "desolvation", ]

# Kruskal-Wallis test:
des_kw <- kTest(subset)
print(des_kw)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  value by variant
## Kruskal-Wallis chi-squared = 4.1538, df = 2, p-value = 0.1253
```

```
# Boxplots:
des_plot <- boxes2(subset, "Desolvation energy")
```

```
## Warning: Ignoring unknown parameters: test, textsize
```

```
print(des_plot)
```
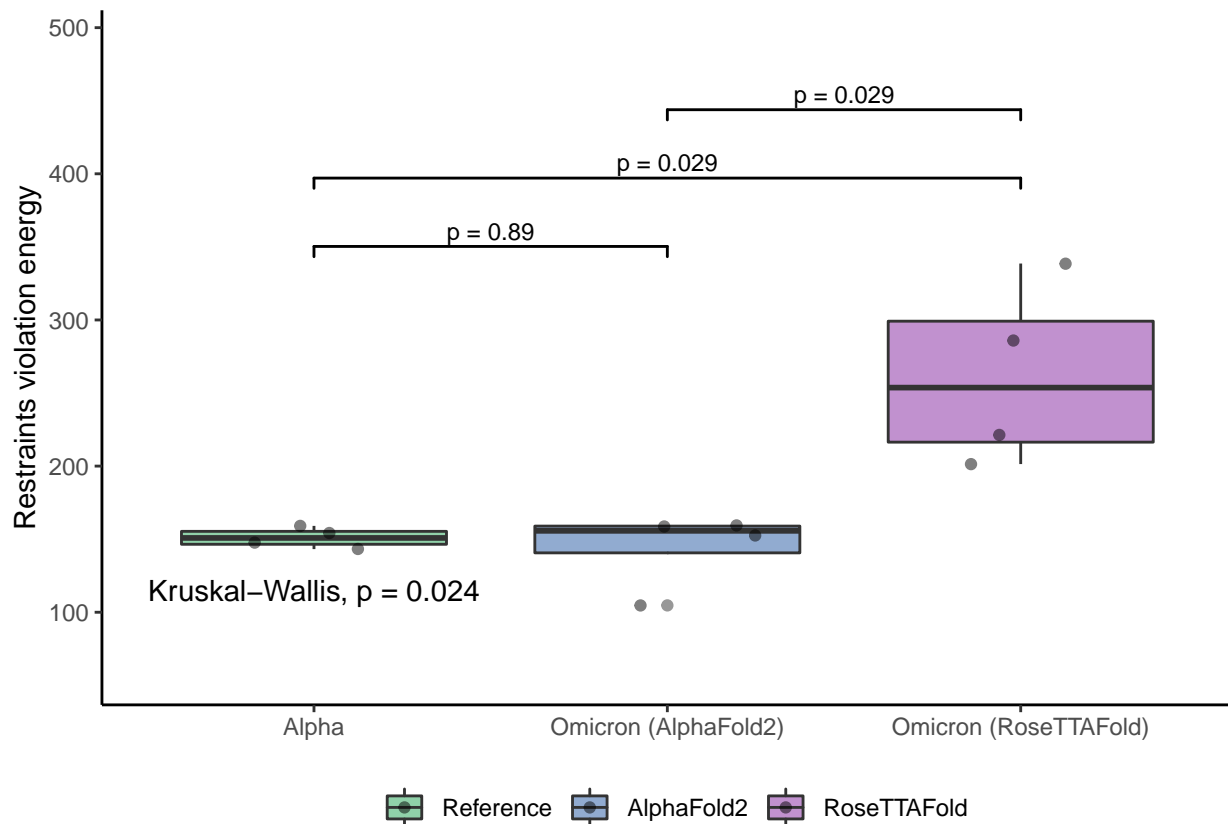
Restraints violation energy

```r
# Filter:
subset <- df[df$variable == "restraints", ]

# Kruskal-Wallis test:
res_kw <- kTest(subset)
print(res_kw)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  value by variant
## Kruskal-Wallis chi-squared = 7.4231, df = 2, p-value = 0.02444
```

```r
# Boxplots:
res_plot <- boxes2(subset, "Restraints violation energy")
```

```
## Warning: Ignoring unknown parameters: test, textsize
```

```r
print(res_plot)
```
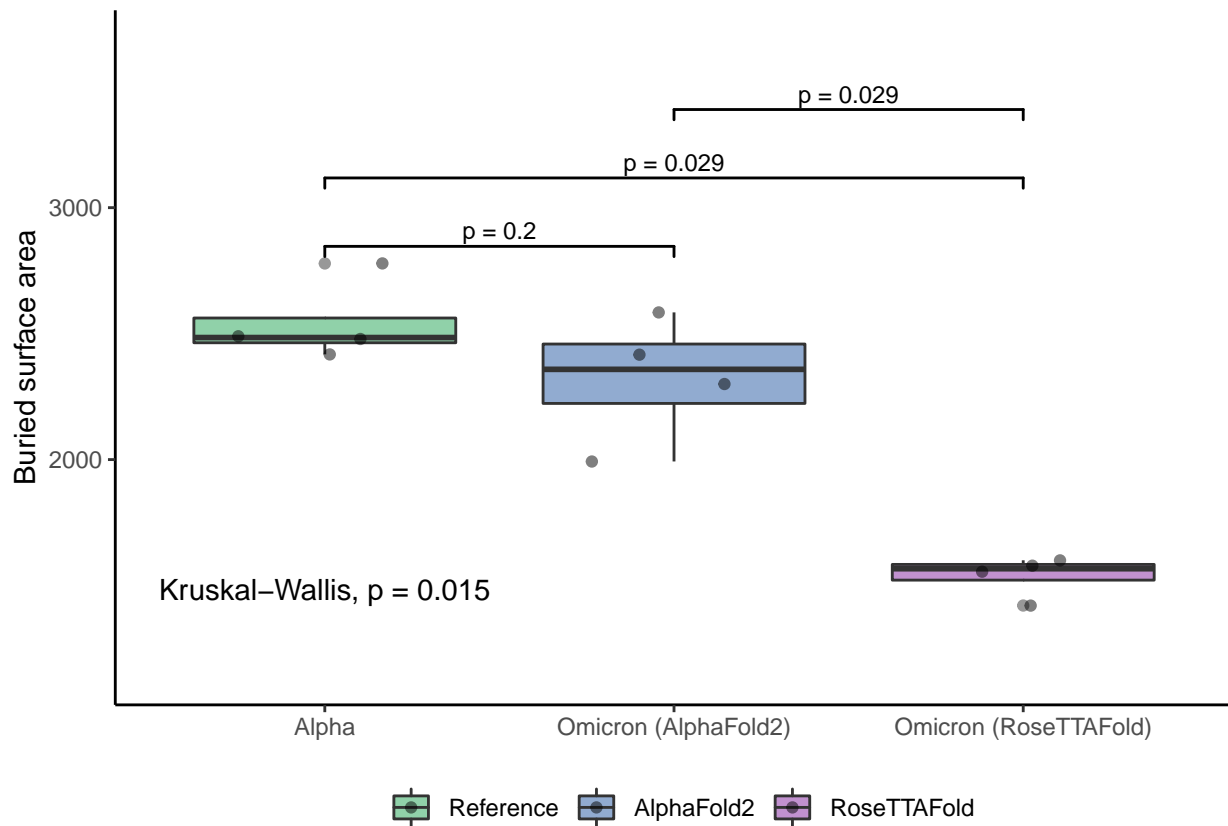
Buried surface area

```r
# Filter:
subset <- df[df$variable == "buried", ]

# Kruskal-Wallis test:
bur_kw <- kTest(subset)
print(bur_kw)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  value by variant
## Kruskal-Wallis chi-squared = 8.3462, df = 2, p-value = 0.0154
```

```r
# Boxplots:
bur_plot <- boxes2(subset, "Buried surface area")
```

```
## Warning: Ignoring unknown parameters: test, textsize
```

```r
print(bur_plot)
```

Multi panel plot

```r
# grid.arrange(had_plot, vdw_plot, elt_plot, des_plot, res_kw, bur_plot, ncol=2, top="HADDOCK results")
myGrid <- grid.arrange(
  boxes(df[df$variable == "haddock", ],        "HADDOCK score"),
  boxes(df[df$variable == "Van_der_Waals", ], "Van der Waals energy"),
  boxes(df[df$variable == "electrostatic", ], "Electrostatic score"),
  boxes(df[df$variable == "desolvation", ],    "Desolvation energy"),
  boxes(df[df$variable == "restraints", ],     "Restraints violation energy"),
  boxes(df[df$variable == "buried", ],         "Buried surface area"),
  get_legend(boxes2(df[df$variable == "haddock", ], "HADDOCK score")),
  layout_matrix=rbind(c(1, 2),
                      c(1, 2),
                      c(1, 2),
                      c(1, 2),
                      c(1, 2),
                      c(1, 2),
                      c(1, 2),
                      c(3, 4),
                      c(3, 4),
                      c(3, 4),
                      c(3, 4),
                      c(3, 4),
                      c(3, 4),
                      c(3, 4),
                      c(5, 6),
```

```
                    c(5, 6),
                    c(5, 6),
                    c(5, 6),
                    c(5, 6),
                    c(5, 6),
                    c(5, 6),
                    c(7, 7)),
  ncol=2, top="HADDOCK results")
```

```
## Warning: Ignoring unknown parameters: test
```

```
## Warning: Ignoring unknown parameters: test
```

```
## Warning: Ignoring unknown parameters: test
```

```
## Warning: Ignoring unknown parameters: test
```

```
## Warning: Ignoring unknown parameters: test
```

```
## Warning: Ignoring unknown parameters: test
```

```
## Warning: Ignoring unknown parameters: test, textsize
```

```
## Warning: Removed 3 rows containing missing values (geom_signif).
```

```
## Warning in wilcox.test.default(c(-106.9, -86.1, -86.1, -85), c(-90.7, -93.1, :
## cannot compute exact p-value with ties
```

```
## Warning in wilcox.test.default(c(-106.9, -86.1, -86.1, -85), c(-27.5, -48.1, :
## cannot compute exact p-value with ties
```

```
## Warning: Removed 3 rows containing missing values (geom_signif).
```
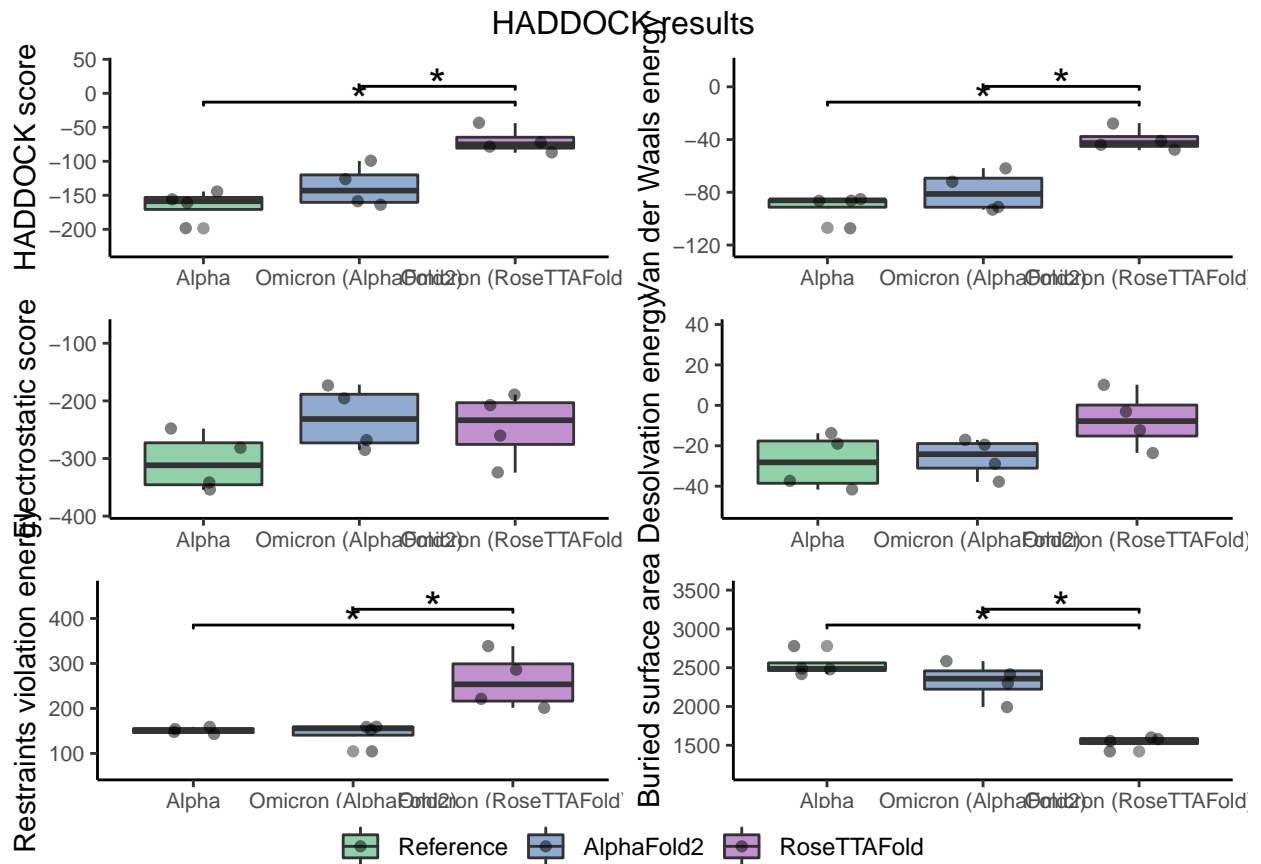
```
## Warning: Removed 9 rows containing missing values (geom_signif).
```

```
## Warning: Removed 9 rows containing missing values (geom_signif).
```

```
## Warning: Removed 3 rows containing missing values (geom_signif).
```

```
## Warning: Removed 3 rows containing missing values (geom_signif).
```

```
ggsave(filename="haddock_grid.pdf", plot=myGrid, width=6, height=12, units="in")
```