

Evolving Event-driven Programs with SignalGP: Statistics

Here, we provide fully detailed statistical test results, including the R code used to generate them. Across experiments, we performed a Kruskal-Wallis rank sum test to determine if a set was significant, and if significant, we performed a post-hoc Dunn's test, applying a Bonferroni correction for multiple comparisons. See the paper for full context.

Each post-hoc Dunn's test was performed using the FSA package.

```
library(FSA)
```

```
## ## FSA v0.8.17. See citation('FSA') if used in publication.  
## ## Run fishR() for related website and fishR('IFAR') for related book.
```

We'll set our significance threshold to 0.05.

Changing Environment Problem

In this section, we give the statistical details for the Changing Environment Problem broken down by environment-size.

First, we'll load the fitness data.

```
ce_ff_data <- read.csv("../data/chg_env/mt_final_fitness.csv")
```

A note about how treatments are named within the data: treatment names describe the parameters and their values used when running the experiment. Parameters and their values are adjacent in the name, and parameter-value combinations are separated by underscores. For example, ED1_AS1_ENV2_TSK0 indicates that event-driven (ED) signals were enabled (1), active sensors (AS) were enabled (1), and there were two environments states. In other words, ED1_AS1_ENV2_TSK0 indicates the two-state environment combined treatment. The trailing TSK0 can be ignored.

Okay, next we'll partition out different subsets of the data.

```
# two-state environment  
# -- overall data --  
ce_ff_data_2 <- ce_ff_data[grepl("_ENV2_", ce_ff_data$treatment),]  
ce_ff_data_2 <- ce_ff_data_2[ce_ff_data_2$analysis == "fdom",]  
ce_ff_data_2$treatment <- relevel(ce_ff_data_2$treatment, ref="ED1_AS1_ENV2_TSK0")  
# -- teasing apart combined treatment data --  
ce_tff_data_2 <- ce_ff_data[grepl("ED1_AS1_ENV2_", ce_ff_data$treatment),]  
ce_tff_data_2$treatment <- relevel(ce_tff_data_2$treatment, ref="ED1_AS1_ENV2_TSK0")  
  
# four-state environment  
# -- overall data --  
ce_ff_data_4 <- ce_ff_data[grepl("_ENV4_", ce_ff_data$treatment),]  
ce_ff_data_4 <- ce_ff_data_4[ce_ff_data_4$analysis == "fdom",]  
ce_ff_data_4$treatment <- relevel(ce_ff_data_4$treatment, ref="ED1_AS1_ENV4_TSK0")  
# -- teasing apart combined treatment data --  
ce_tff_data_4 <- ce_ff_data[grepl("ED1_AS1_ENV4_", ce_ff_data$treatment),]  
ce_tff_data_4$treatment <- relevel(ce_tff_data_4$treatment, ref="ED1_AS1_ENV4_TSK0")  
  
# eight-state environment
```

```

# -- overall data --
ce_ff_data_8 <- ce_ff_data[grepl("_ENV8_", ce_ff_data$treatment),]
ce_ff_data_8 <- ce_ff_data_8[ce_ff_data_8$analysis == "fdom",]
ce_ff_data_8$treatment <- relevel(ce_ff_data_8$treatment, ref="ED1_AS1_ENV8_TSK0")
# -- teasing apart combined treatment data --
ce_tff_data_8 <- ce_ff_data[grepl("ED1_AS1_ENV8_", ce_ff_data$treatment),]
ce_tff_data_8$treatment <- relevel(ce_tff_data_8$treatment, ref="ED1_AS1_ENV8_TSK0")

# sixteen-state environment
# -- overall data --
ce_ff_data_16 <- ce_ff_data[grepl("_ENV16_", ce_ff_data$treatment),]
ce_ff_data_16 <- ce_ff_data_16[ce_ff_data_16$analysis == "fdom",]
ce_ff_data_16$treatment <- relevel(ce_ff_data_16$treatment, ref="ED1_AS1_ENV16_TSK0")
# -- teasing apart combined treatment data --
ce_tff_data_16 <- ce_ff_data[grepl("ED1_AS1_ENV16_", ce_ff_data$treatment),]
ce_tff_data_16$treatment <- relevel(ce_tff_data_16$treatment, ref="ED1_AS1_ENV16_TSK0")

```

Two-state Environment

Overall Results

```

# Run the Kruskal-Wallis rank sum test
kw_fit_ce_2 <- kruskal.test(fitness ~ treatment, data=ce_ff_data_2)
kw_fit_ce_2

```

```

##
##  Kruskal-Wallis rank sum test
##
## data:  fitness by treatment
## Kruskal-Wallis chi-squared = 283.26, df = 2, p-value < 2.2e-16

```

According to our Kruskal-Wallis rank sum test, at least one treatment is significantly different from the other treatments. Thus, we'll perform a post-hoc Dunn's test, applying a Bonferroni correction for multiple comparisons.

```

# Run the post-hoc Dunn's test.
dt_ce_2 <- dunnTest(fitness~treatment, data=ce_ff_data_2, method="bonferroni")
dt_ce_2

```

```
## Dunn (1964) Kruskal-Wallis multiple comparison
```

```
##  p-values adjusted with the Bonferroni method.
```

```

##              Comparison              Z      P.unadj
## 1 ED0_AS1_ENV2_TSK0 - ED1_AS0_ENV2_TSK0 -14.57562 4.014904e-48
## 2 ED0_AS1_ENV2_TSK0 - ED1_AS1_ENV2_TSK0 -14.57562 4.014904e-48
## 3 ED1_AS0_ENV2_TSK0 - ED1_AS1_ENV2_TSK0  0.00000 1.000000e+00
##              P.adj
## 1 1.204471e-47
## 2 1.204471e-47
## 3 1.000000e+00

```

Combined Treatment Re-evaluations

```
# Run the Kruskal-Wallis rank sum test
kw_ce_comb_2 <- kruskal.test(fitness ~ analysis, data=ce_tff_data_2)
kw_ce_comb_2
```

```
##
## Kruskal-Wallis rank sum test
##
## data: fitness by analysis
## Kruskal-Wallis chi-squared = 283.27, df = 2, p-value < 2.2e-16
```

According to our Kruskal-Wallis rank sum test, at least one treatment is significantly different from the other treatments. Thus, we'll perform a post-hoc Dunn's test, applying a Bonferroni correction for multiple comparisons.

```
# Run the post-hoc Dunn's test.
dt_ce_comb_2 <- dunnTest(fitness~analysis, data=ce_tff_data_2, method="bonferroni")
dt_ce_comb_2
```

```
## Dunn (1964) Kruskal-Wallis multiple comparison
## p-values adjusted with the Bonferroni method.
##
## Comparison Z P.unadj P.adj
## 1 fdom - no_sensors 0.00000 1.000000e+00 1.000000e+00
## 2 fdom - no_signals 14.57563 4.014092e-48 1.204228e-47
## 3 no_sensors - no_signals 14.57563 4.014092e-48 1.204228e-47
```

Four-state Environment

Overall Results

```
# Run the Kruskal-Wallis rank sum test
kw_fit_ce_4 <- kruskal.test(fitness ~ treatment, data=ce_ff_data_4)
kw_fit_ce_4
```

```
##
## Kruskal-Wallis rank sum test
##
## data: fitness by treatment
## Kruskal-Wallis chi-squared = 283.26, df = 2, p-value < 2.2e-16
```

According to our Kruskal-Wallis rank sum test, at least one treatment is significantly different from the other treatments. Thus, we'll perform a post-hoc Dunn's test, applying a Bonferroni correction for multiple comparisons.

```
# Run the post-hoc Dunn's test.
dt_ce_4 <- dunnTest(fitness~treatment, data=ce_ff_data_4, method="bonferroni")
dt_ce_4
```

```
## Dunn (1964) Kruskal-Wallis multiple comparison
## p-values adjusted with the Bonferroni method.
##
## Comparison Z P.unadj
## 1 ED0_AS1_ENV4_TSK0 - ED1_AS0_ENV4_TSK0 -14.57561 4.015039e-48
## 2 ED0_AS1_ENV4_TSK0 - ED1_AS1_ENV4_TSK0 -14.57561 4.015039e-48
## 3 ED1_AS0_ENV4_TSK0 - ED1_AS1_ENV4_TSK0 0.00000 1.000000e+00
## P.adj
## 1 1.204512e-47
```

```
## 2 1.204512e-47
## 3 1.000000e+00
```

Combined Treatment Re-evaluations

```
# Run the Kruskal-Wallis rank sum test
kw_ce_comb_4 <- kruskal.test(fitness ~ analysis, data=ce_tff_data_4)
kw_ce_comb_4
```

```
##
## Kruskal-Wallis rank sum test
##
## data: fitness by analysis
## Kruskal-Wallis chi-squared = 283.27, df = 2, p-value < 2.2e-16
```

According to our Kruskal-Wallis rank sum test, at least one treatment is significantly different from the other treatments. Thus, we'll perform a post-hoc Dunn's test, applying a Bonferroni correction for multiple comparisons.

```
# Run the post-hoc Dunn's test.
dt_ce_comb_4 <- dunnTest(fitness~analysis, data=ce_tff_data_4, method="bonferroni")
dt_ce_comb_4
```

```
## Dunn (1964) Kruskal-Wallis multiple comparison
## p-values adjusted with the Bonferroni method.
##
## Comparison Z P.unadj P.adj
## 1 fdom - no_sensors 0.00000 1.000000e+00 1.000000e+00
## 2 fdom - no_signals 14.57564 4.013687e-48 1.204106e-47
## 3 no_sensors - no_signals 14.57564 4.013687e-48 1.204106e-47
```

Eight-state Environment

Overall Results

```
# Run the Kruskal-Wallis rank sum test
kw_fit_ce_8 <- kruskal.test(fitness ~ treatment, data=ce_ff_data_8)
kw_fit_ce_8
```

```
##
## Kruskal-Wallis rank sum test
##
## data: fitness by treatment
## Kruskal-Wallis chi-squared = 273.26, df = 2, p-value < 2.2e-16
```

According to our Kruskal-Wallis rank sum test, at least one treatment is significantly different from the other treatments. Thus, we'll perform a post-hoc Dunn's test, applying a Bonferroni correction for multiple comparisons.

```
# Run the post-hoc Dunn's test.
dt_ce_8 <- dunnTest(fitness~treatment, data=ce_ff_data_8, method="bonferroni")
dt_ce_8
```

```
## Dunn (1964) Kruskal-Wallis multiple comparison
## p-values adjusted with the Bonferroni method.
```

```
##               Comparison              Z      P.unadj
## 1 ED0_AS1_ENV8_TSK0 - ED1_AS0_ENV8_TSK0 -14.2165237 7.235911e-46
## 2 ED0_AS1_ENV8_TSK0 - ED1_AS1_ENV8_TSK0 -14.4131139 4.279713e-47
## 3 ED1_AS0_ENV8_TSK0 - ED1_AS1_ENV8_TSK0  -0.1965902 8.441483e-01
##           P.adj
## 1 2.170773e-45
## 2 1.283914e-46
## 3 1.000000e+00
```

Combined Treatment Re-evaluations

```
# Run the Kruskal-Wallis rank sum test
kw_ce_comb_8 <- kruskal.test(fitness ~ analysis, data=ce_tff_data_8)
kw_ce_comb_8
```

```
##
## Kruskal-Wallis rank sum test
##
## data: fitness by analysis
## Kruskal-Wallis chi-squared = 290.92, df = 2, p-value < 2.2e-16
```

According to our Kruskal-Wallis rank sum test, at least one treatment is significantly different from the other treatments. Thus, we'll perform a post-hoc Dunn's test, applying a Bonferroni correction for multiple comparisons.

```
# Run the post-hoc Dunn's test.
dt_ce_comb_8 <- dunnTest(fitness~analysis, data=ce_tff_data_8, method="bonferroni")
dt_ce_comb_8
```

```
## Dunn (1964) Kruskal-Wallis multiple comparison
## p-values adjusted with the Bonferroni method.
##               Comparison              Z      P.unadj      P.adj
## 1 fdom - no_sensors 0.00000 1.000000e+00 1.000000e+00
## 2 fdom - no_signals 14.77117 2.247789e-49 6.743368e-49
## 3 no_sensors - no_signals 14.77117 2.247789e-49 6.743368e-49
```

Sixteen-state Environment

Overall Results

```
# Run the Kruskal-Wallis rank sum test
kw_fit_ce_16 <- kruskal.test(fitness ~ treatment, data=ce_ff_data_16)
kw_fit_ce_16
```

```
##
## Kruskal-Wallis rank sum test
##
## data: fitness by treatment
## Kruskal-Wallis chi-squared = 199.38, df = 2, p-value < 2.2e-16
```

According to our Kruskal-Wallis rank sum test, at least one treatment is significantly different from the other treatments. Thus, we'll perform a post-hoc Dunn's test, applying a Bonferroni correction for multiple comparisons.

```
# Run the post-hoc Dunn's test.
dt_ce_16 <- dunnTest(fitness~treatment, data=ce_ff_data_16, method="bonferroni")
dt_ce_16

## Dunn (1964) Kruskal-Wallis multiple comparison
## p-values adjusted with the Bonferroni method.
##
## Comparison Z P.unadj
## 1 ED0_AS1_ENV16_TSK0 - ED1_AS0_ENV16_TSK0 -12.1256145 7.727836e-34
## 2 ED0_AS1_ENV16_TSK0 - ED1_AS1_ENV16_TSK0 -12.3285843 6.355345e-35
## 3 ED1_AS0_ENV16_TSK0 - ED1_AS1_ENV16_TSK0 -0.2029699 8.391586e-01
## P.adj
## 1 2.318351e-33
## 2 1.906603e-34
## 3 1.000000e+00
```

Combined Treatment Re-evaluations

```
# Run the Kruskal-Wallis rank sum test
kw_ce_comb_16 <- kruskal.test(fitness ~ analysis, data=ce_tff_data_16)
kw_ce_comb_16
```

```
##
## Kruskal-Wallis rank sum test
##
## data: fitness by analysis
## Kruskal-Wallis chi-squared = 207.01, df = 2, p-value < 2.2e-16
```

According to our Kruskal-Wallis rank sum test, at least one treatment is significantly different from the other treatments. Thus, we'll perform a post-hoc Dunn's test, applying a Bonferroni correction for multiple comparisons.

```
# Run the post-hoc Dunn's test.
dt_ce_comb_16 <- dunnTest(fitness~analysis, data=ce_tff_data_16, method="bonferroni")
dt_ce_comb_16
```

```
## Dunn (1964) Kruskal-Wallis multiple comparison
## p-values adjusted with the Bonferroni method.
##
## Comparison Z P.unadj P.adj
## 1 fdom - no_sensors 0.00000 1.000000e+00 1.000000e+00
## 2 fdom - no_signals 12.46013 1.231736e-35 3.695207e-35
## 3 no_sensors - no_signals 12.46013 1.231736e-35 3.695207e-35
```

Distributed Leader Election Problem

In this section, we give the statistical details for the distributed leader election problem.

First, we'll load the fitness data.

```
election_data <- read.csv("../data/election/final_fitness.csv")

# Run the Kruskal-Wallis rank sum test.
elec_kw <- kruskal.test(max_fitness ~ treatment, data=election_data)
elec_kw
```

```
##
## Kruskal-Wallis rank sum test
##
## data: max_fitness by treatment
## Kruskal-Wallis chi-squared = 103.25, df = 2, p-value < 2.2e-16
# Run the post-hoc Dunn's test.
elec_dt <- dunnTest(max_fitness~treatment, data=election_data, method="bonferroni")
elec_dt

## Dunn (1964) Kruskal-Wallis multiple comparison
## p-values adjusted with the Bonferroni method.

## Comparison Z P.unadj
## 1 EventDriven_MsgForking - Imperative_MsgForking 9.681599 3.610247e-22
## 2 EventDriven_MsgForking - Imperative_MsgNonForking 7.512422 5.804340e-14
## 3 Imperative_MsgForking - Imperative_MsgNonForking -2.169177 3.006924e-02
## P.adj
## 1 1.083074e-21
## 2 1.741302e-13
## 3 9.020771e-02
```