Analysis

2024-11-30

```r
data <- read.csv("pandemic_data.csv")
head(data)
```

```
##         load soc.iso rate.vac quar.dur num.daily
## 1 0.80713787     0.0     0.00        0        15
## 2 0.64718361     0.5     0.00        0        15
## 3 0.27645898     1.0     0.00        0        15
## 4 0.78194120     0.0     0.01        0        15
## 5 0.73239290     0.5     0.01        0        15
## 6 0.07060294     1.0     0.01        0        15
```

```r
summary(data) # load is i+q
```

```
##       load            soc.iso        rate.vac        quar.dur     num.daily
##  Min.   :0.01002   Min.   :0.0   Min.   :0.00   Min.   : 0   Min.   :15
##  1st Qu.:0.27646   1st Qu.:0.0   1st Qu.:0.00   1st Qu.: 0   1st Qu.:15
##  Median :0.71534   Median :0.5   Median :0.01   Median : 7   Median :30
##  Mean   :0.55530   Mean   :0.5   Mean   :0.01   Mean   : 7   Mean   :30
##  3rd Qu.:0.83129   3rd Qu.:1.0   3rd Qu.:0.02   3rd Qu.:14   3rd Qu.:45
##  Max.   :0.93448   Max.   :1.0   Max.   :0.02   Max.   :14   Max.   :45
```

```r
# Anova
anova_model <- aov(load ~ soc.iso * rate.vac * quar.dur * num.daily, data = data)
summary(anova_model)
```

```
##                                     Df Sum Sq Mean Sq F value   Pr(>F)
## soc.iso                              1  5.185   5.185 208.373  < 2e-16 ***
## rate.vac                             1  0.081   0.081   3.269 0.075218 .
## quar.dur                             1  0.398   0.398  15.995 0.000165 ***
## num.daily                            1  0.889   0.889  35.732 1.06e-07 ***
## soc.iso:rate.vac                     1  0.001   0.001   0.049 0.825882
## soc.iso:quar.dur                     1  0.055   0.055   2.220 0.141049
## rate.vac:quar.dur                    1  0.000   0.000   0.010 0.920560
## soc.iso:num.daily                    1  0.009   0.009   0.343 0.560141
## rate.vac:num.daily                   1  0.018   0.018   0.711 0.402159
## quar.dur:num.daily                   1  0.001   0.001   0.023 0.880033
## soc.iso:rate.vac:quar.dur            1  0.020   0.020   0.799 0.374769
## soc.iso:rate.vac:num.daily           1  0.002   0.002   0.093 0.760777
## soc.iso:quar.dur:num.daily           1  0.086   0.086   3.439 0.068197 .
## rate.vac:quar.dur:num.daily          1  0.001   0.001   0.059 0.809529
## soc.iso:rate.vac:quar.dur:num.daily  1  0.015   0.015   0.585 0.447152
## Residuals                           65  1.617   0.025
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
# Regression and Box-Cox
lm_model <- lm(load ~ soc.iso * rate.vac * quar.dur * num.daily, data = data)
```

```
summary(lm_model)
```

```
##
## Call:
## lm(formula = load ~ soc.iso * rate.vac * quar.dur * num.daily,
##     data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.37987 -0.08997 -0.03031  0.10783  0.32196
##
## Coefficients:
##                                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)                       7.925e-01  1.833e-01   4.323 5.39e-05 ***
## soc.iso                          -6.914e-01  2.840e-01  -2.435   0.0177 *
## rate.vac                          5.016e-01  1.420e+01   0.035   0.9719
## quar.dur                         -8.013e-03  2.028e-02  -0.395   0.6941
## num.daily                         4.208e-03  5.657e-03   0.744   0.4596
## soc.iso:rate.vac                 -1.504e+01  2.200e+01  -0.684   0.4965
## soc.iso:quar.dur                 -2.527e-03  3.142e-02  -0.080   0.9362
## rate.vac:quar.dur                -1.458e+00  1.571e+00  -0.928   0.3569
## soc.iso:num.daily                 7.306e-03  8.764e-03   0.834   0.4075
## rate.vac:num.daily               -3.959e-02  4.382e-01  -0.090   0.9283
## quar.dur:num.daily                1.693e-04  6.260e-04   0.270   0.7876
## soc.iso:rate.vac:quar.dur         2.546e+00  2.434e+00   1.046   0.2995
## soc.iso:rate.vac:num.daily        2.709e-01  6.788e-01   0.399   0.6912
## soc.iso:quar.dur:num.daily       -5.629e-04  9.697e-04  -0.581   0.5636
## rate.vac:quar.dur:num.daily       3.615e-02  4.849e-02   0.745   0.4587
## soc.iso:rate.vac:quar.dur:num.daily -5.745e-02  7.512e-02  -0.765   0.4472
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1577 on 65 degrees of freedom
## Multiple R-squared:  0.807,  Adjusted R-squared:  0.7624
## F-statistic: 18.11 on 15 and 65 DF,  p-value: < 2.2e-16
```

```
confint(lm_model, level = 0.95)
```

```
##                                      2.5 %        97.5 %
## (Intercept)                      0.426375670   1.158537279
## soc.iso                         -1.258532695  -0.124272808
## rate.vac                       -27.854891763  28.858102605
## quar.dur                        -0.048522020   0.032496543
## num.daily                       -0.007088996   0.015506002
## soc.iso:rate.vac               -58.973056778  28.886336303
## soc.iso:quar.dur                -0.065283231   0.060230188
## rate.vac:quar.dur               -4.595731984   1.679938950
## soc.iso:num.daily               -0.010195869   0.024808151
## rate.vac:num.daily              -0.914690609   0.835510395
## quar.dur:num.daily              -0.001080830   0.001419457
## soc.iso:rate.vac:quar.dur       -2.315419112   7.406808494
## soc.iso:rate.vac:num.daily      -1.084815360   1.626584376
## soc.iso:quar.dur:num.daily      -0.002499653   0.001373775
## rate.vac:quar.dur:num.daily     -0.060689172   0.132982238
```

```
## soc.iso:rate.vac:quar.dur:num.daily  -0.207466891   0.092567566
# Box-Cox Transformation
boxcox_result <- boxcox(lm_model, lambda = seq(-2, 2, by = 0.1))
```
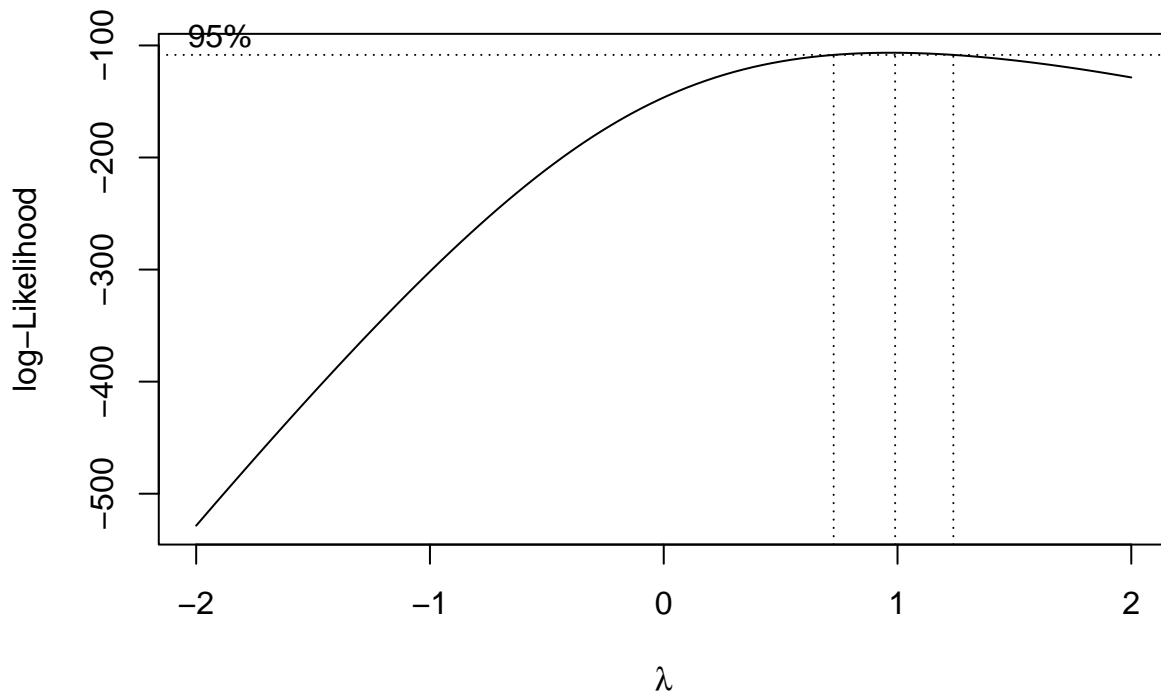


```
optimal_lambda <- boxcox_result$x[which.max(boxcox_result$y)]
optimal_lambda # no transformation
```
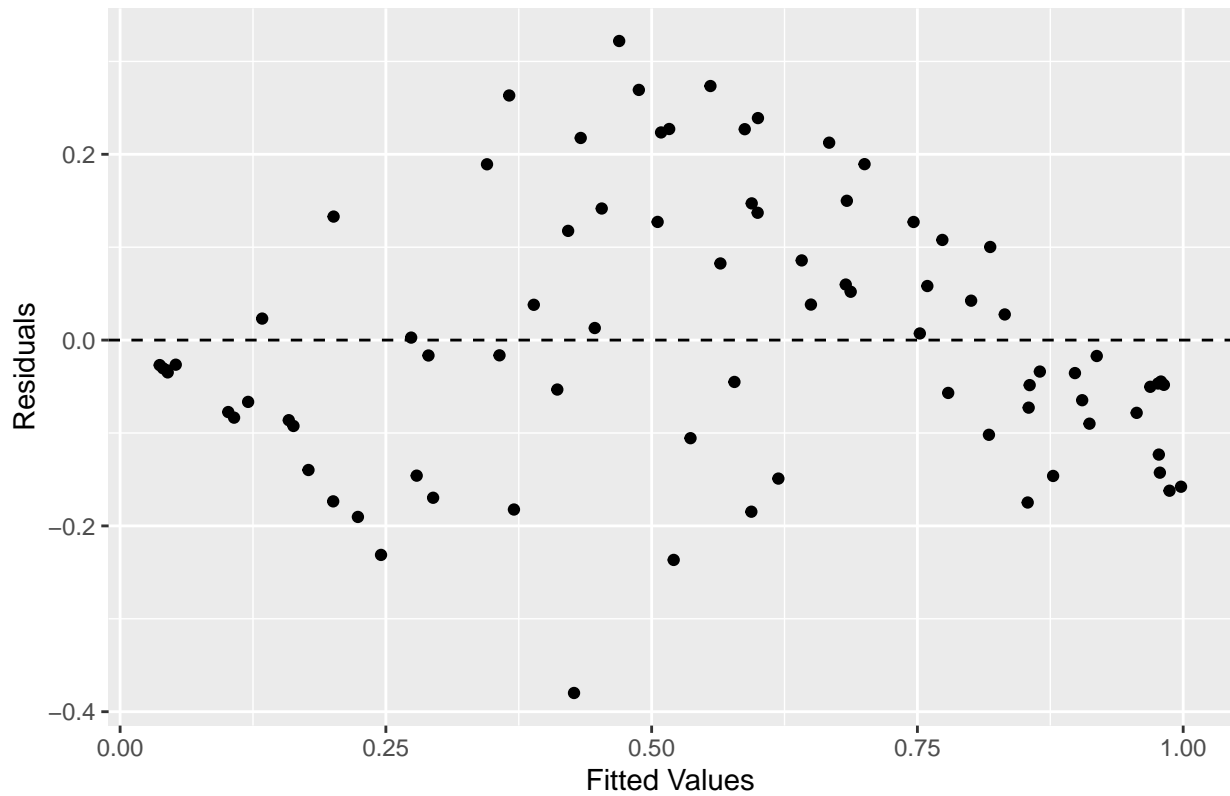
```
## [1] 0.989899
```

```
# Residual plots
residuals_plot <- ggplot(data.frame(fitted = lm_model$fitted.values,
                                    residuals = lm_model$residuals),
                         aes(x = fitted, y = residuals)) +
  geom_point() +
  geom_hline(yintercept = 0, linetype = "dashed") +
  labs(title = "Residuals vs Fitted Values", x = "Fitted Values", y = "Residuals")
print(residuals_plot)
```

Residuals vs Fitted Values


```
# Q-Q Plot
qqnorm(lm_model$residuals, main = "Normal Q-Q Plot")
qqline(lm_model$residuals)
```
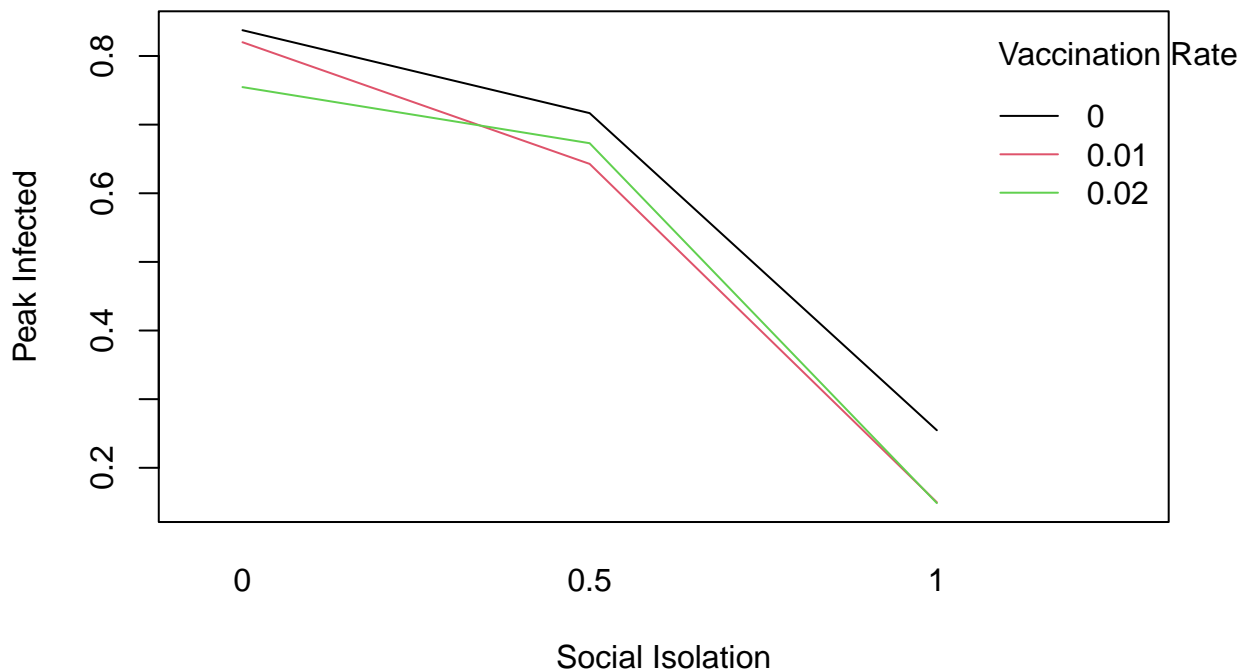
Normal Q–Q Plot
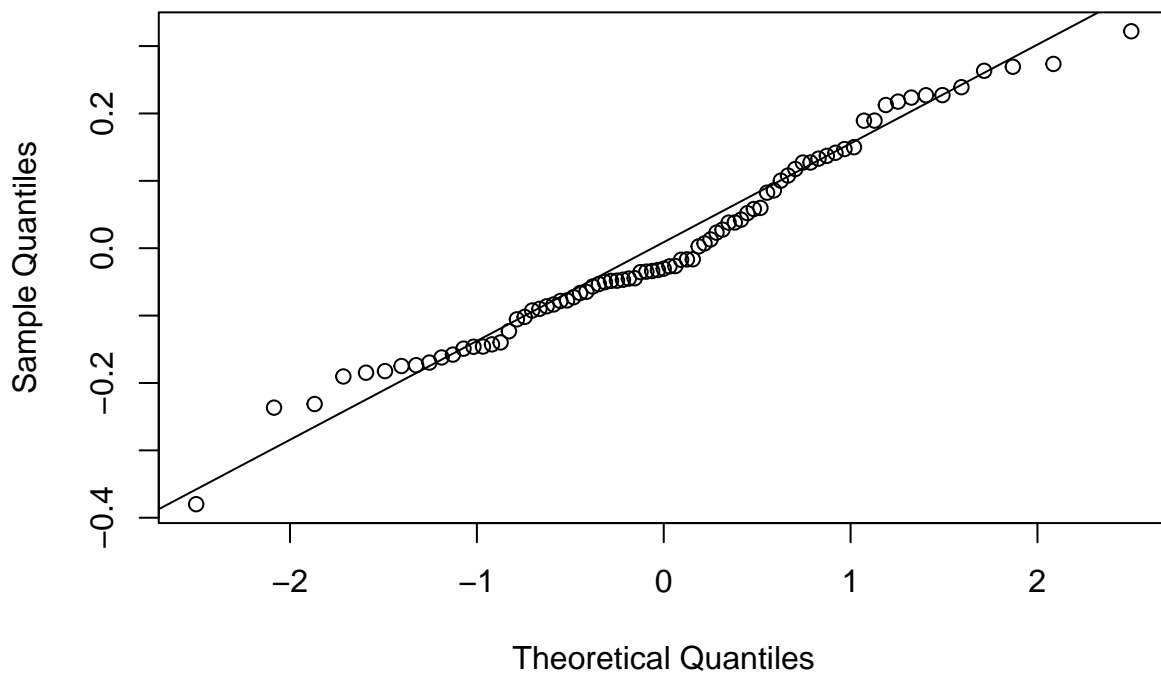
```
# Graphics
# Interaction plots
interaction.plot(
  x.factor = data$soc.iso,
  trace.factor = data$rate.vac,
  response = data$load,
  col = 1:4,
  lty = 1,
  main = "Interaction Plot: Social Isolation x Vaccination Rate",
  xlab = "Social Isolation",
  ylab = "Peak Infected",
  trace.label = "Vaccination Rate"
)
```

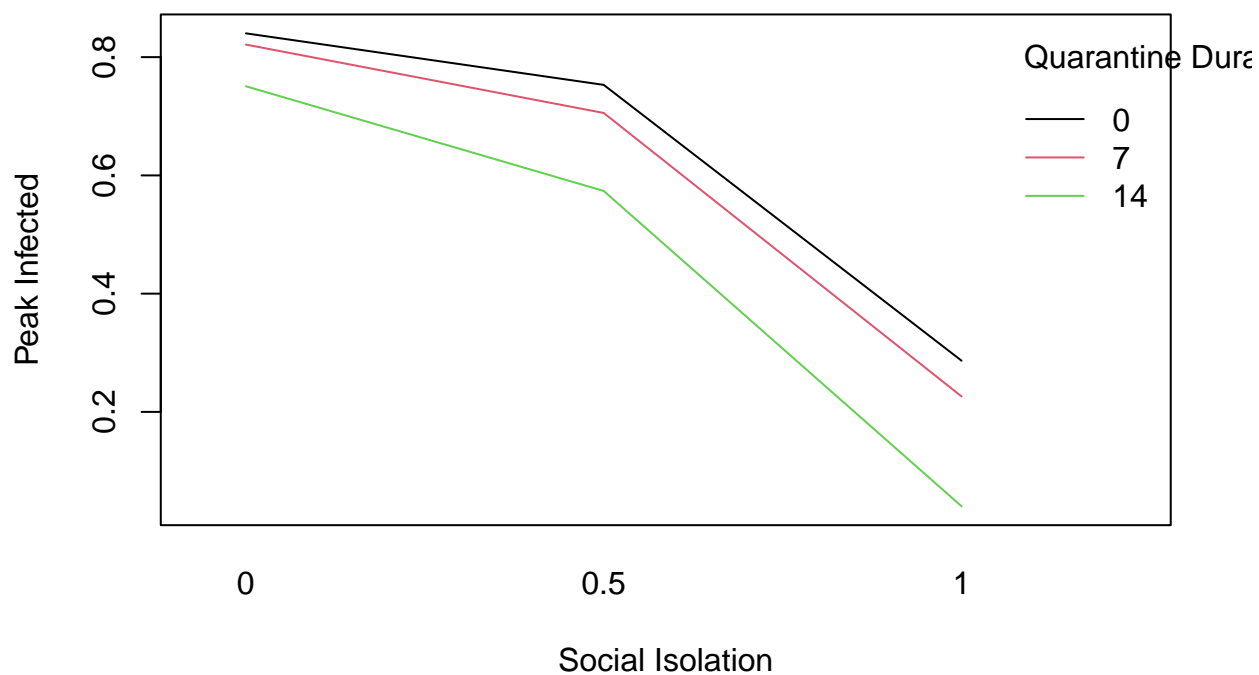**Interaction Plot: Social Isolation x Vaccination Rate**


```
interaction.plot(
  x.factor = data$soc.iso,
  trace.factor = data$quar.dur,
  response = data$load,
  col = 1:4,
  lty = 1,
  main = "Interaction Plot: Social Isolation x Quarantine Duration",
  xlab = "Social Isolation",
  ylab = "Peak Infected",
  trace.label = "Quarantine Duration"
)
```
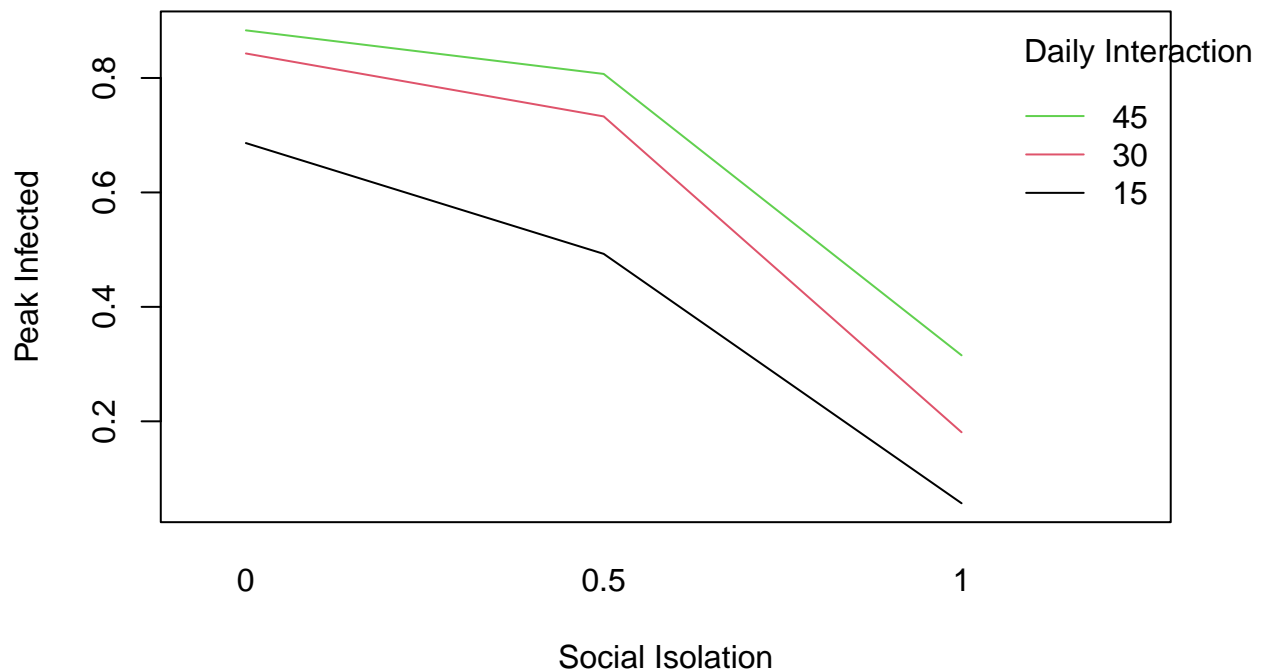
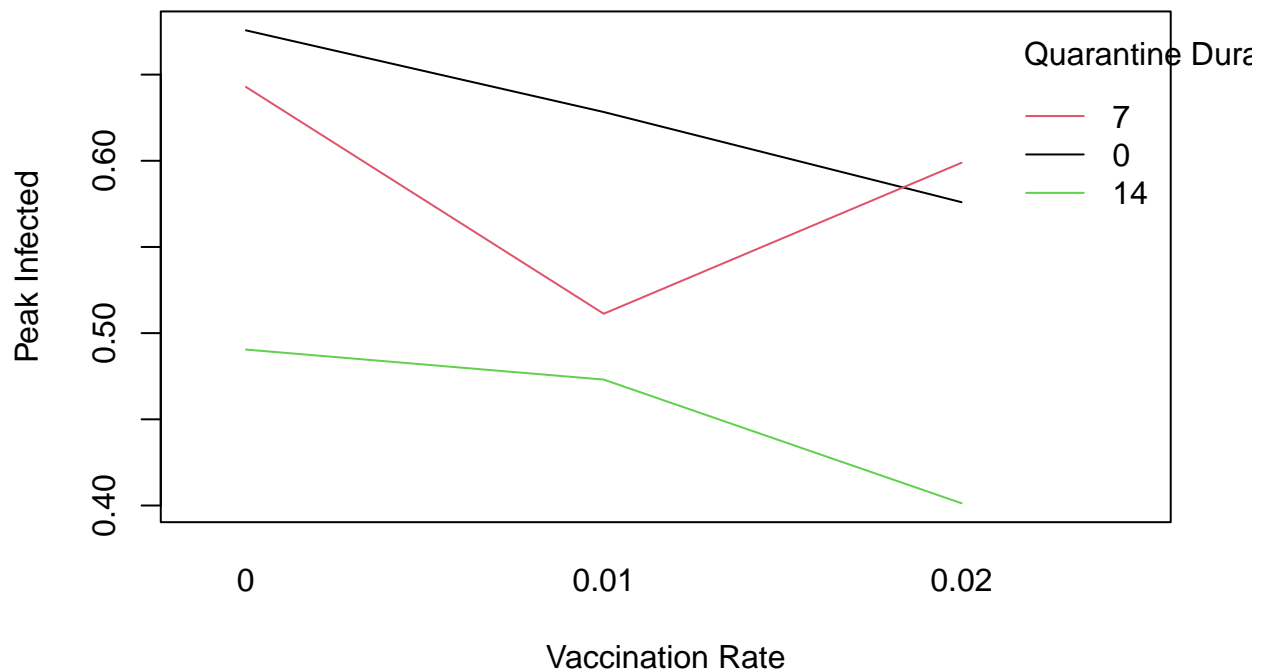**Interaction Plot: Social Isolation x Quarantine Duration**


```
interaction.plot(
  x.factor = data$soc.iso,
  trace.factor = data$num.daily,
  response = data$load,
  col = 1:4,
  lty = 1,
  main = "Interaction Plot: Social Isolation x Daily Interactions",
  xlab = "Social Isolation",
  ylab = "Peak Infected",
  trace.label = "Daily Interactions"
)
```

Interaction Plot: Social Isolation x Daily Interactions


```
interaction.plot(
  x.factor = data$rate.vac,
  trace.factor = data$quar.dur,
  response = data$load,
  col = 1:4,
  lty = 1,
  main = "Interaction Plot: Vaccination Rate x Quarantine Duration",
  xlab = "Vaccination Rate",
  ylab = "Peak Infected",
  trace.label = "Quarantine Duration"
)
```

Interaction Plot: Vaccination Rate x Quarantine Duration

Quarantine Dura

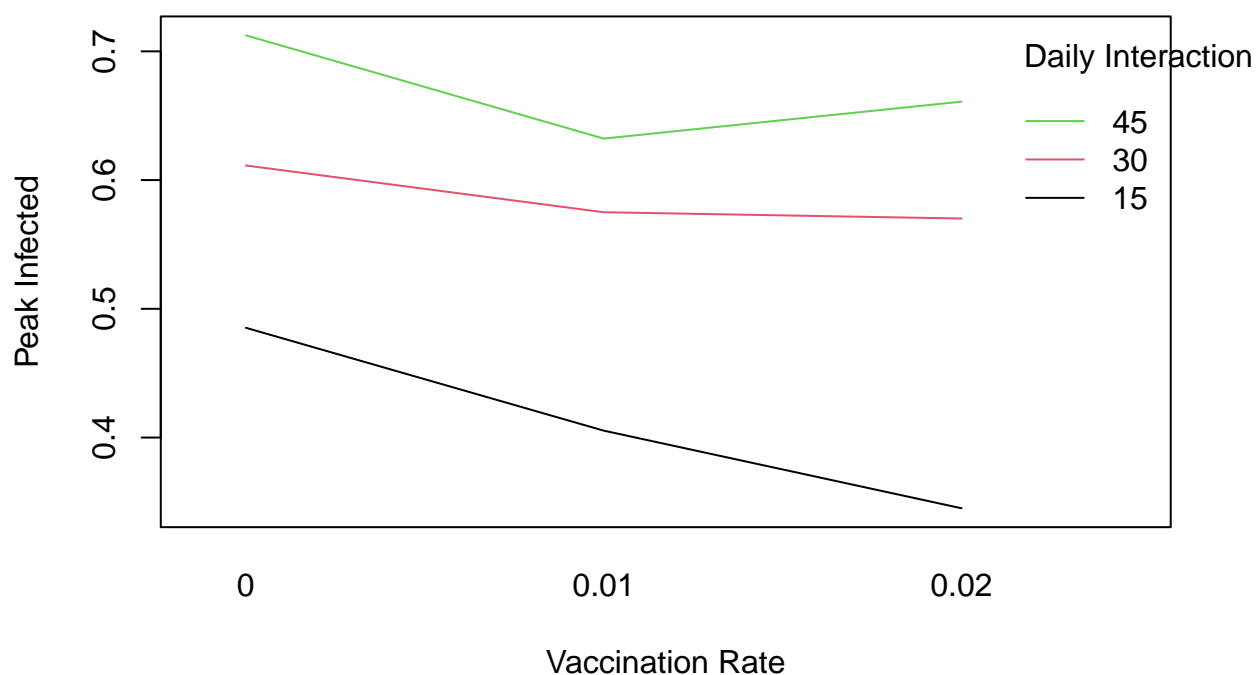|	
7	
0	
14	

Peak Infected

Vaccination Rate

```
interaction.plot(
  x.factor = data$rate.vac,
  trace.factor = data$num.daily,
  response = data$load,
  col = 1:4,
  lty = 1,
  main = "Interaction Plot: Vaccination Rate x Daily Interactions",
  xlab = "Vaccination Rate",
  ylab = "Peak Infected",
  trace.label = "Daily Interactions"
)
```

Interaction Plot: Vaccination Rate x Daily Interactions


```r
interaction.plot(
  x.factor = data$quar.dur,
  trace.factor = data$num.daily,
  response = data$load,
  col = 1:4,
  lty = 1,
  main = "Interaction Plot: Quarantine Duration x Daily Interactions",
  xlab = "Quarantine Duration",
  ylab = "Peak Infected",
  trace.label = "Daily Interactions"
)
```

Interaction Plot: Quarantine Duration x Daily Interactions
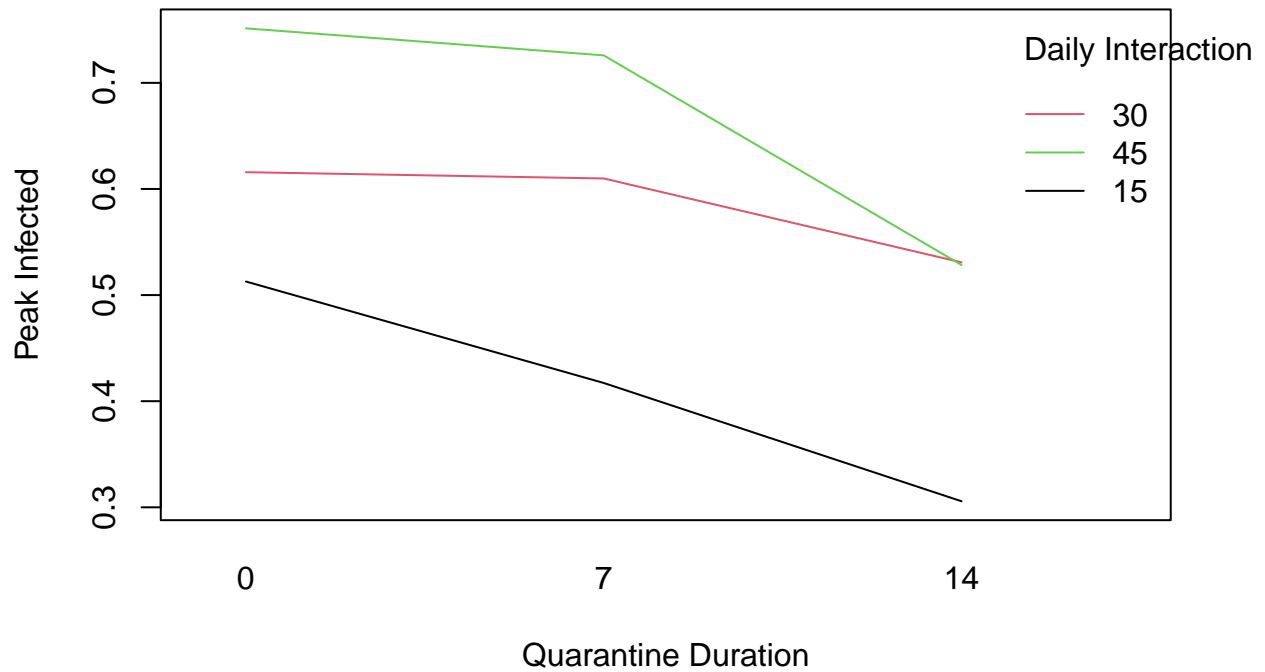

```
# Boxplots
ggplot(data, aes(x = as.factor(soc.iso), y = load)) +
  geom_boxplot() +
  labs(title = "Effect of Social Isolation on Peak Infected",
       x = "Social Isolation Level", y = "Peak Infected")
```

Effect of Social Isolation on Peak Infected
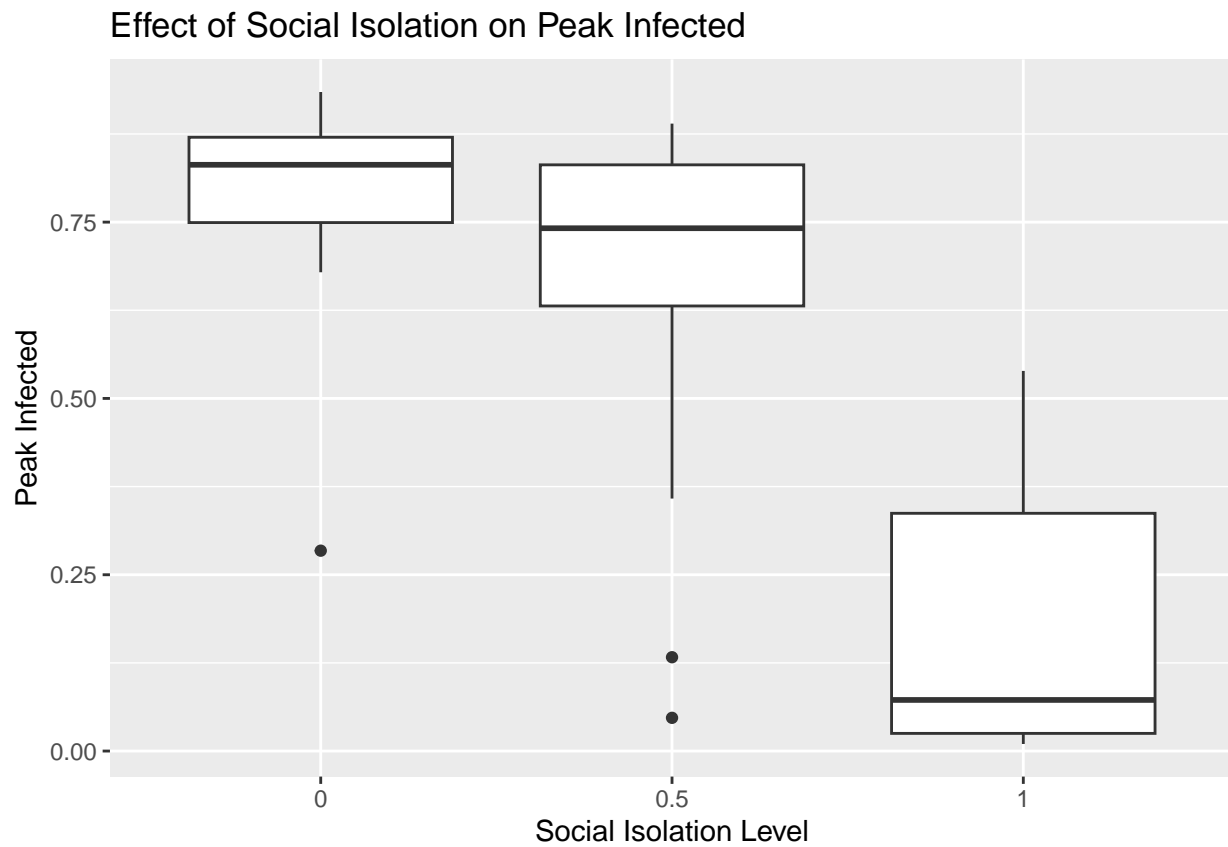

```r
ggplot(data, aes(x = as.factor(rate.vac), y = load)) +
  geom_boxplot() +
  labs(title = "Effect of Vaccination Rate on Peak Infected",
       x = "Vaccination Rate", y = "Peak Infected")
```

Effect of Vaccination Rate on Peak Infected
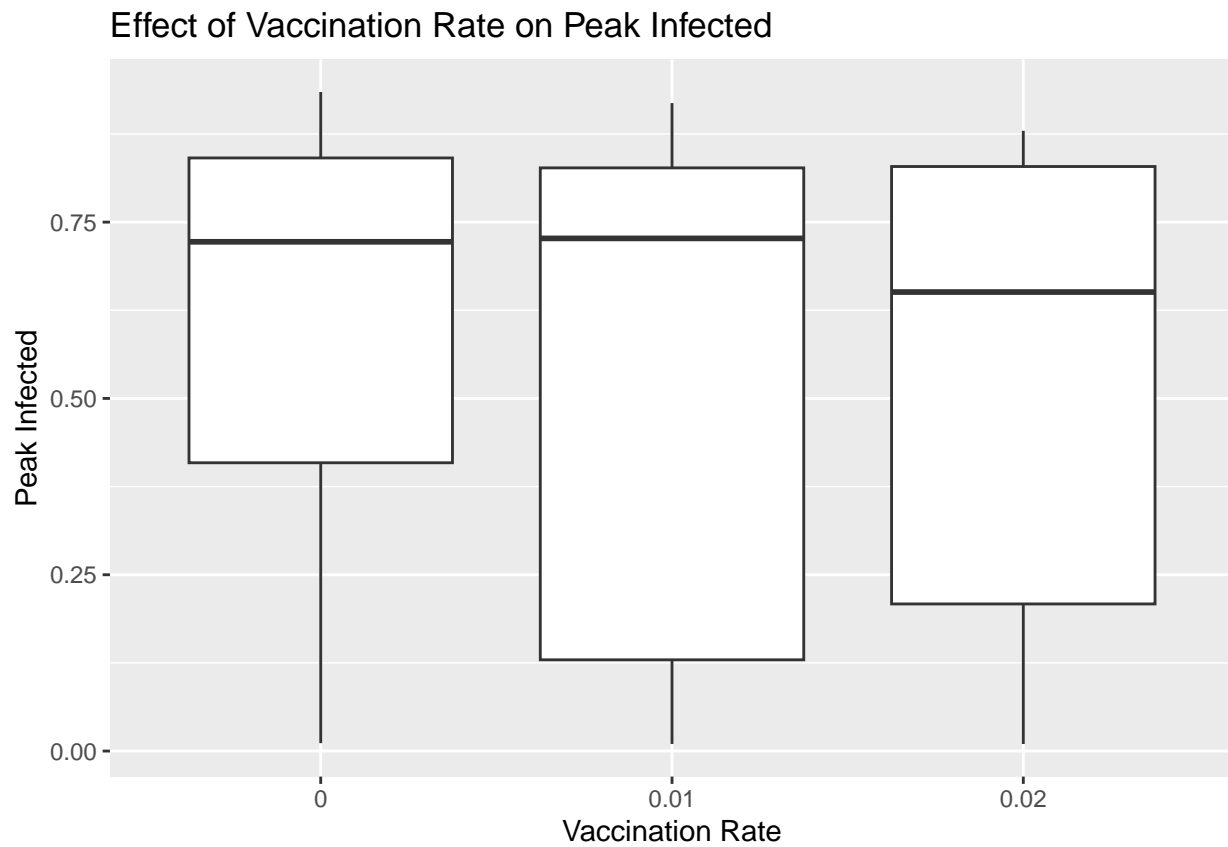

```
ggplot(data, aes(x = as.factor(quar.dur), y = load)) +
  geom_boxplot() +
  labs(title = "Effect of Quarantine Duration on Peak Infected",
    x = "Quarantine Duration (Days)", y = "Peak Infected")
```
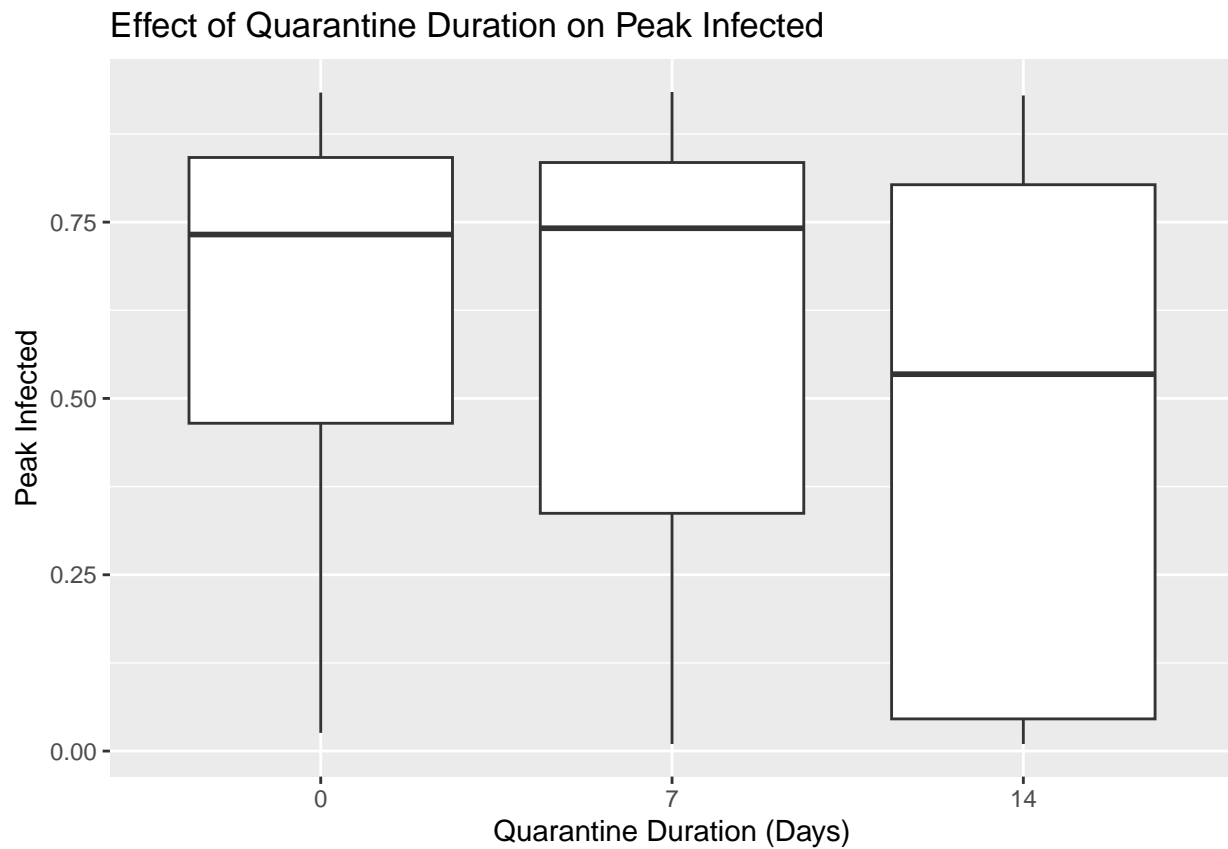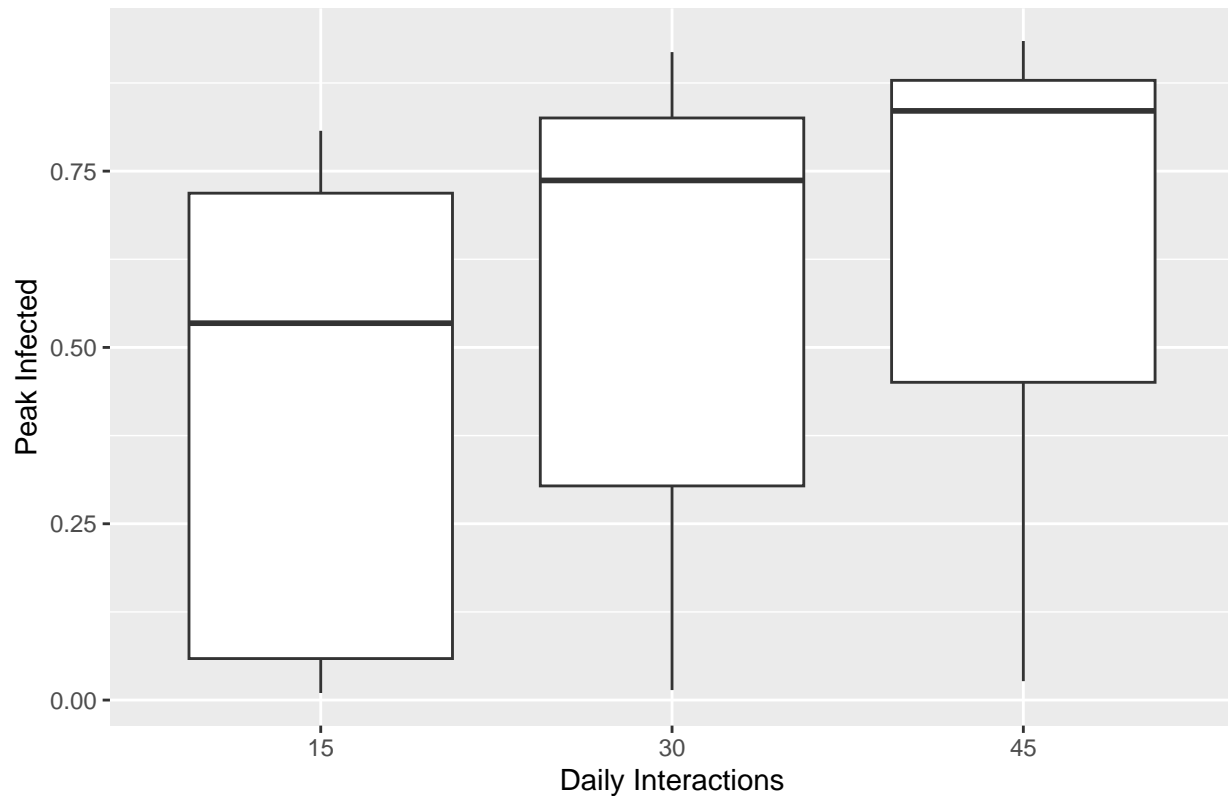
Effect of Quarantine Duration on Peak Infected


```
ggplot(data, aes(x = as.factor(num.daily), y = load)) +
  geom_boxplot() +
  labs(title = "Effect of Daily Interactions on Peak Infected",
    x = "Daily Interactions", y = "Peak Infected")
```

Effect of Daily Interactions on Peak Infected


```r
# Optimal Combination
aggregated_results <- aggregate(load ~ soc.iso + rate.vac + quar.dur + num.daily, data = data, mean)
aggregated_results
```

```
##    soc.iso rate.vac quar.dur num.daily       load
## 1      0.0     0.00        0        15 0.80713787
## 2      0.5     0.00        0        15 0.64718361
## 3      1.0     0.00        0        15 0.27645898
## 4      0.0     0.01        0        15 0.78194120
## 5      0.5     0.01        0        15 0.73239290
## 6      1.0     0.01        0        15 0.07060294
## 7      0.0     0.02        0        15 0.67890619
## 8      0.5     0.02        0        15 0.59468481
## 9      1.0     0.02        0        15 0.02584835
## 10     0.0     0.00        7        15 0.71533767
## 11     0.5     0.00        7        15 0.75729810
## 12     1.0     0.00        7        15 0.07245189
## 13     0.0     0.01        7        15 0.75941389
## 14     0.5     0.01        7        15 0.04711413
## 15     1.0     0.01        7        15 0.02426370
## 16     0.0     0.02        7        15 0.73927788
## 17     0.5     0.02        7        15 0.62933531
## 18     1.0     0.02        7        15 0.01001565
## 19     0.0     0.00       14        15 0.72209633
## 20     0.5     0.00       14        15 0.35802719
## 21     1.0     0.00       14        15 0.01119957
## 22     0.0     0.01       14        15 0.68808198
```

```
## 23    0.5    0.01    14    15 0.53437658
## 24    1.0    0.01    14    15 0.01008588
## 25    0.0    0.02    14    15 0.28410682
## 26    0.5    0.02    14    15 0.13305129
## 27    1.0    0.02    14    15 0.01031777
## 28    0.0    0.00     0    30 0.90158964
## 29    0.5    0.00     0    30 0.74241129
## 30    1.0    0.00     0    30 0.45948728
## 31    0.0    0.01     0    30 0.82188088
## 32    0.5    0.01     0    30 0.72693603
## 33    1.0    0.01     0    30 0.18810326
## 34    0.0    0.02     0    30 0.84029662
## 35    0.5    0.02     0    30 0.73679191
## 36    1.0    0.02     0    30 0.12473303
## 37    0.0    0.00     7    30 0.86265759
## 38    0.5    0.00     7    30 0.74130782
## 39    1.0    0.00     7    30 0.27348081
## 40    0.0    0.01     7    30 0.83129222
## 41    0.5    0.01     7    30 0.82880476
## 42    1.0    0.01     7    30 0.01420269
## 43    0.0    0.02     7    30 0.85977938
## 44    0.5    0.02     7    30 0.74372227
## 45    1.0    0.02     7    30 0.33378771
## 46    0.0    0.00    14    30 0.73141670
## 47    0.5    0.00    14    30 0.63285640
## 48    1.0    0.00    14    30 0.15678004
## 49    0.0    0.01    14    30 0.91874011
## 50    0.5    0.01    14    30 0.79141668
## 51    1.0    0.01    14    30 0.05388418
## 52    0.0    0.02    14    30 0.81755784
## 53    0.5    0.02    14    30 0.65085486
## 54    1.0    0.02    14    30 0.02371587
## 55    0.0    0.00     0    45 0.93368412
## 56    0.5    0.00     0    45 0.84295894
## 57    1.0    0.00     0    45 0.47015082
## 58    0.0    0.01     0    45 0.91876582
## 59    0.5    0.01     0    45 0.88128431
## 60    1.0    0.01     0    45 0.53283670
## 61    0.0    0.02     0    45 0.87793650
## 62    0.5    0.02     0    45 0.87354649
## 63    1.0    0.02     0    45 0.43096527
## 64    0.0    0.00     7    45 0.93447981
## 65    0.5    0.00     7    45 0.88966512
## 66    1.0    0.00     7    45 0.53901621
## 67    0.0    0.01     7    45 0.83540620
## 68    0.5    0.01     7    45 0.83357275
## 69    1.0    0.01     7    45 0.42716914
## 70    0.0    0.02     7    45 0.85387802
## 71    0.5    0.02     7    45 0.87948887
## 72    1.0    0.02     7    45 0.34045949
## 73    0.0    0.00    14    45 0.92958326
## 74    0.5    0.00    14    45 0.83899166
## 75    1.0    0.00    14    45 0.03333219
## 76    0.0    0.01    14    45 0.82501809
```

```
## 77     0.5      0.01        14          45 0.40905894
## 78     1.0      0.01        14          45 0.02683887
## 79     0.0      0.02        14          45 0.84012290
## 80     0.5      0.02        14          45 0.81461598
## 81     1.0      0.02        14          45 0.03734338
```

```r
# Optimal Levels for each level of daily interaction
optimal_levels <- aggregated_results %>%
  group_by(num.daily) %>%
  slice_min(load, n = 1, with_ties = FALSE)
print(optimal_levels)
```

```
## # A tibble: 3 x 5
## # Groups:   num.daily [3]
##   soc.iso rate.vac quar.dur num.daily   load
##     <dbl>    <dbl>    <int>     <int>  <dbl>
## 1       1     0.02        7        15 0.0100
## 2       1     0.01        7        30 0.0142
## 3       1     0.01       14        45 0.0268
```

```r
optimal_combination <- aggregated_results[which.min(aggregated_results$load), ]
optimal_combination
```

```
##    soc.iso rate.vac quar.dur num.daily       load
## 18       1     0.02        7        15 0.01001565
```

```r
# Treatment Contrasts
data$soc.iso <- as.factor(data$soc.iso)
data$rate.vac <- as.factor(data$rate.vac)
data$quar.dur <- as.factor(data$quar.dur)
data$num.daily <- as.factor(data$num.daily)
head(data)
```

```
##         load soc.iso rate.vac quar.dur num.daily
## 1 0.80713787       0        0        0        15
## 2 0.64718361     0.5        0        0        15
## 3 0.27645898       1        0        0        15
## 4 0.78194120       0     0.01        0        15
## 5 0.73239290     0.5     0.01        0        15
## 6 0.07060294       1     0.01        0        15
```

```r
contrasts(data$soc.iso) <- contr.treatment(levels(data$soc.iso), base = 1)
contrasts(data$rate.vac) <- contr.treatment(levels(data$rate.vac), base = 1)
contrasts(data$quar.dur) <- contr.treatment(levels(data$quar.dur), base = 1)
contrasts(data$num.daily) <- contr.treatment(levels(data$num.daily), base = 1)

# Fit linear model with treatment contrasts
lm_treatment <- lm(load ~ soc.iso + rate.vac + quar.dur + num.daily, data = data)
summary(lm_treatment)
```

```
##
## Call:
## lm(formula = load ~ soc.iso + rate.vac + quar.dur + num.daily,
##     data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
```

```
## -0.49819 -0.05650 -0.00510  0.08971  0.20259
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.77971    0.04188  18.619  < 2e-16 ***
## soc.iso0.5  -0.12662    0.03419  -3.703 0.000414 ***
## soc.iso1    -0.61974    0.03419 -18.125  < 2e-16 ***
## rate.vac0.01 -0.06547   0.03419  -1.915 0.059522 .
## rate.vac0.02 -0.07763   0.03419  -2.270 0.026189 *
## quar.dur7   -0.04233    0.03419  -1.238 0.219780
## quar.dur14  -0.17171    0.03419  -5.022 3.59e-06 ***
## num.daily30  0.17361    0.03419   5.077 2.89e-06 ***
## num.daily45  0.25664    0.03419   7.505 1.28e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1256 on 72 degrees of freedom
## Multiple R-squared:  0.8644, Adjusted R-squared:  0.8493
## F-statistic: 57.35 on 8 and 72 DF,  p-value: < 2.2e-16
```

```r
confint(lm_treatment, level = 0.95)
```

```
##                  2.5 %       97.5 %
## (Intercept)   0.6962321  0.863196380
## soc.iso0.5   -0.1947790 -0.058453286
## soc.iso1     -0.6878982 -0.551572446
## rate.vac0.01 -0.1336279  0.002697842
## rate.vac0.02 -0.1457888 -0.009463062
## quar.dur7    -0.1104902  0.025835597
## quar.dur14   -0.2398682 -0.103542466
## num.daily30   0.1054475  0.241773230
## num.daily45   0.1884727  0.324798488
```

```r
coef_lm <- coef(lm_treatment)
coef_lm
```

```
##  (Intercept)    soc.iso0.5      soc.iso1 rate.vac0.01 rate.vac0.02     quar.dur7
##   0.77971424  -0.12661617   -0.61973533  -0.06546504  -0.07762594  -0.04232729
##   quar.dur14  num.daily30  num.daily45
##  -0.17170535   0.17361035   0.25663561
```

```r
confint_lm <- confint(lm_treatment)
confint_lm
```

```
##                  2.5 %       97.5 %
## (Intercept)   0.6962321  0.863196380
## soc.iso0.5   -0.1947790 -0.058453286
## soc.iso1     -0.6878982 -0.551572446
## rate.vac0.01 -0.1336279  0.002697842
## rate.vac0.02 -0.1457888 -0.009463062
## quar.dur7    -0.1104902  0.025835597
## quar.dur14   -0.2398682 -0.103542466
## num.daily30   0.1054475  0.241773230
## num.daily45   0.1884727  0.324798488
```

```r
# Polynomial Contrasts
data$soc.iso <- ordered(data$soc.iso)
```

```
data$rate.vac <- ordered(data$rate.vac)
data$quar.dur <- ordered(data$quar.dur)
data$num.daily <- ordered(data$num.daily)
contrasts(data$soc.iso) <- contr.poly(levels(data$soc.iso))
contrasts(data$rate.vac) <- contr.poly(levels(data$rate.vac))
contrasts(data$quar.dur) <- contr.poly(levels(data$quar.dur))
contrasts(data$num.daily) <- contr.poly(levels(data$num.daily))
lm_poly <- lm(load ~ soc.iso + rate.vac + quar.dur + num.daily, data = data)
summary(lm_poly)
```

```
##
## Call:
## lm(formula = load ~ soc.iso + rate.vac + quar.dur + num.daily,
##     data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.49819 -0.05650 -0.00510  0.08971  0.20259
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.55530    0.01396  39.780  < 2e-16 ***
## soc.iso.L   -0.43822    0.02418 -18.125  < 2e-16 ***
## soc.iso.Q   -0.14962    0.02418  -6.188 3.33e-08 ***
## rate.vac.L  -0.05489    0.02418  -2.270   0.0262 *
## rate.vac.Q   0.02176    0.02418   0.900   0.3711
## quar.dur.L  -0.12141    0.02418  -5.022 3.59e-06 ***
## quar.dur.Q  -0.03554    0.02418  -1.470   0.1460
## num.daily.L  0.18147    0.02418   7.505 1.28e-10 ***
## num.daily.Q -0.03698    0.02418  -1.530   0.1305
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1256 on 72 degrees of freedom
## Multiple R-squared:  0.8644, Adjusted R-squared:  0.8493
## F-statistic: 57.35 on 8 and 72 DF,  p-value: < 2.2e-16
```

```
anova(lm_poly)
```

```
## Analysis of Variance Table
##
## Response: load
##           Df Sum Sq Mean Sq F value    Pr(>F)
## soc.iso    2 5.7894 2.89472 183.397 < 2.2e-16 ***
## rate.vac   2 0.0941 0.04707   2.982   0.05699 .
## quar.dur   2 0.4321 0.21606  13.689 9.155e-06 ***
## num.daily  2 0.9261 0.46303  29.336 4.803e-10 ***
## Residuals 72 1.1364 0.01578
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```