Newton Raphson

```
covid = read_csv("./covid_final.csv") %>%
  mutate(date = lubridate::parse_date_time(x = date,
                                           orders = c("m/d/y", "m-d-Y")),
         t = as.numeric(difftime(date, min(date), units = "days"))) %>%
  filter(country_region == "US", province_state == "New York") %>%
  filter(date > "2020-03-15") %>% filter(date < "2020-03-31")
```

```
## Parsed with column specification:
## cols(
##   province_state = col_character(),
##   country_region = col_character(),
##   lat = col_double(),
##   long = col_double(),
##   date = col_character(),
##   confirmed_cases = col_double(),
##   fatalities = col_double()
## )
```

#Log-likelihood function

```
logisticfunc = function(t, y, betavec) {
  a = betavec[1]
  b = betavec[2]
  c = betavec[3]
  # Expu
  expu = exp(b*(t - c))

  # Log-likelihood
  loglik = sum(log(a) - log(1 + expu))

  #Loss function
  #pred_y = a / (1 + exp(-b * (t - c)))
  #loss = sum((pred_y - y)^2) / length(y)
  #Gradient for loss function
  #grad_loss = 2*sum(pred_y - y)^2/length #(?)

  # Gradient for log-likelihood function
  grad = rep(0,3)
  grad[1] = sum(1/a)
  grad[2] = sum((t-c) / (1 + expu))
  grad[3] = sum(-b / (1 + expu))

  # Hessian Matrix
  hess = matrix(data = 0, nrow = 3, ncol = 3)
  hess[1,1] = sum(-1/a^2)
  hess[2,2] = sum(-((t-c)^2)*expu/(expu+1)^2)
  hess[3,3] = sum(-(b^2)*expu/(expu+1)^2)
  hess[1,2] = 0
```

```r
  hess[1,3] = 0
  hess[2,1] = 0
  hess[2,3] = sum((b*(t-c)*expu)/((expu+1)^2)-1/(expu+1))
  hess[3,1] = 0
  hess[3,2] = sum((b*(t-c)*expu)/((expu+1)^2)-1/(expu+1))
  return(list(loglik = loglik, grad = grad, Hess = hess))
}
```

```r
logisticfunc(covid$confirmed_cases, covid$t, c(10000,1.5,100))
```

```
## $loglik
## [1] -Inf
##
## $grad
## [1]       0.0001 -29742.6164   -517.7699
##
## $Hess
##         [,1] [,2] [,3]
## [1,] -1e-08    0    0
## [2,]  0e+00  NaN  NaN
## [3,]  0e+00  NaN  NaN
```

```r
#NewtonRaphson(covid$confirmed_cases, covid$t, logisticstuff, c(10000,1.5,200))
```

Loss function

We define our loss function as:

$$f_{LOSS}(t_{ij}) = \frac{1}{n_j}\Sigma_{i=1}^{n_j}(\frac{a}{1+exp(-b(t_{ij}-c))} - y_{ij})^2$$

where n_j is the number of observed days since first infection for region j, y_{ij} is the number of new infections on day i in region j.

Then, the gradient can be obtained as follows:

$$\bigtriangledown f_{LOSS}(t_{ij}) = (\frac{\partial f_{LOSS}}{\partial a}, \frac{\partial f_{LOSS}}{\partial b}, \frac{\partial f_{LOSS}}{\partial c})$$

where

$$\frac{\partial f_{LOSS}}{\partial a} = \frac{2}{n_j}\Sigma_{i=1}^{n_j}\frac{\frac{a}{expu+1} - y_{ij}}{expu+1},$$

and

$$\frac{\partial f_{LOSS}}{\partial b} = \frac{2}{n_j}\Sigma_{i=1}^{n_j}\frac{a(t_{ij}-c)expu(\frac{a}{expu+1} - y_{ij})}{(expu+1)^2},$$

and

$$\frac{\partial f_{LOSS}}{\partial c} = -\frac{2}{n_j}\Sigma_{i=1}^{n_j}\frac{(ab)expu\frac{a}{expu+1} - y_{ij}}{(expu+1)^2}$$

where $expu = exp(-b(t_{ij}-c))$