Department of Mathematics and Computer Science
University of Southern Denmark, Odense

December 13, 2025
jarys25

ST523 – Statistical Modelling

Answers to the Take-home 2 Assignment, Winter 2025

Task 1

subtask 1

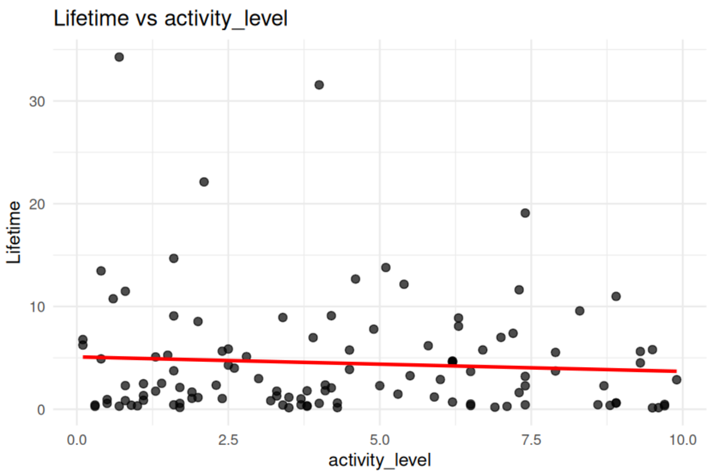
The dataset contains 113 observations across 9 variables. Summary statistics are presented below:

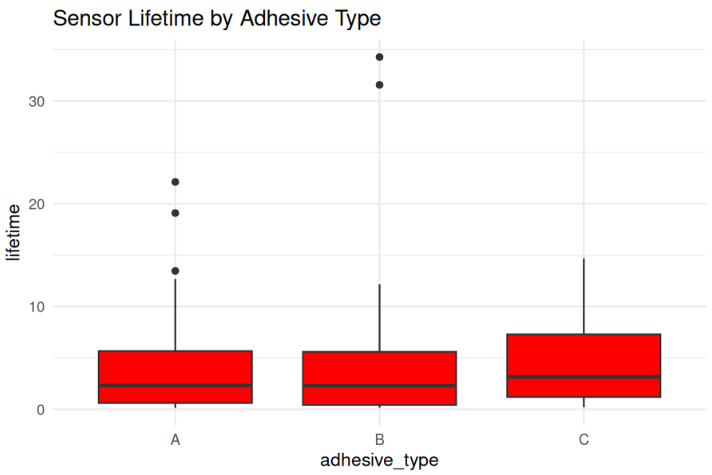
```
    lifetime        skin_temp        humidity      activity_level
 Min.   : 0.140   Min.   :31.10   Min.   :20.70   Min.   :0.100  
 1st Qu.: 0.610   1st Qu.:32.70   1st Qu.:39.00   1st Qu.:1.700  
 Median : 2.340   Median :34.30   Median :56.10   Median :4.000  
 Mean   : 4.478   Mean   :34.18   Mean   :54.06   Mean   :4.345  
 3rd Qu.: 5.800   3rd Qu.:35.70   3rd Qu.:71.80   3rd Qu.:6.500  
 Max.   :34.270   Max.   :37.00   Max.   :83.00   Max.   :9.900  
   sweat_rate  calibration_error  patient_bmi      experience   
 Min.   : 2.3   Min.   : 0.30     Min.   :15.60   Min.   :1.000  
 1st Qu.:46.8   1st Qu.: 5.80     1st Qu.:24.70   1st Qu.:3.000  
 Median :58.4   Median :12.90     Median :27.60   Median :4.000  
 Mean   :57.5   Mean   :19.27     Mean   :27.61   Mean   :3.965  
 3rd Qu.:69.4   3rd Qu.:27.60     3rd Qu.:31.20   3rd Qu.:5.000  
 Max.   :99.5   Max.   :78.30     Max.   :39.60   Max.   :9.000  
```

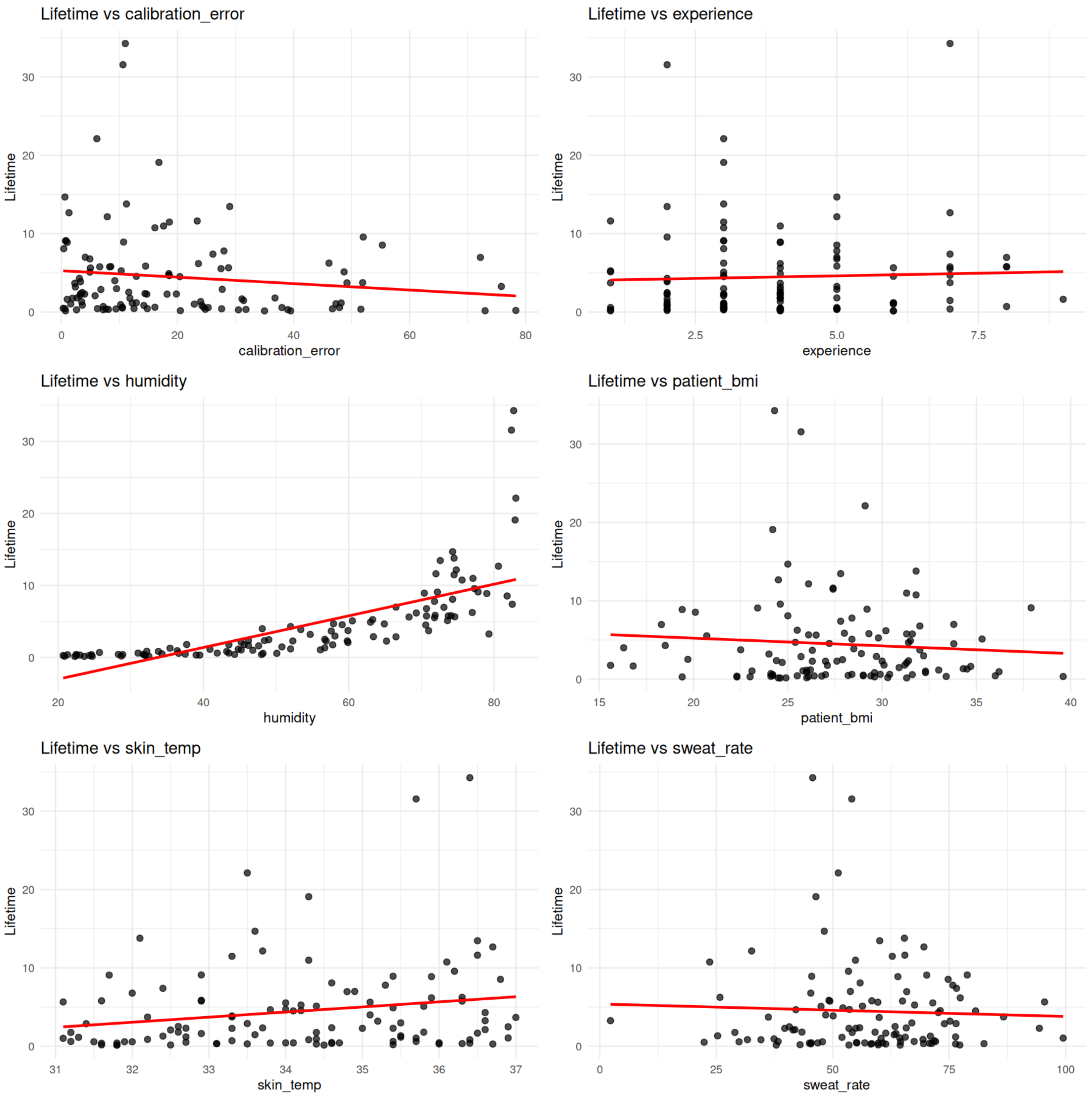
For the adhesive, as it is a factor variable, we only show the overall amount of data entries per type:

```
 A  B  C 
52 35 26 
```

Relationship with Lifetime plots:







It is worth noting that on the plots we can see 4 points which significantly higher lifetime stands out from the rest

---

---

subtask 2

Since lifetime is strictly positive and continuous, I considered two GLM families:

- Gamma distribution with log and canonical (inverse) links
- Inverse Gaussian distribution with log link

Log link function is chosen as the relationship between lifetime and humidity seems exponential - other continuous predictors appear linear on the plots. The canonical link for Inverse Gaussian caused convergence issues, so it was excluded.

I do not consider Poisson distribution as it is designed for count data, not continuous responses. Binomial on the other hand is mainly used for proportion data which is not present in our example.

We compare candidate models based on AIC:

```
      Gamma_Log Inv_Gaussian_Log  Gamma_Canonical
       296.7400         376.2993         423.0153
```

and residual deviance:

```
Inv_Gaussian_Log        Gamma_Log  Gamma_Canonical
        16.71100         16.83515         49.09622
```

The Gamma distribution with log linking was selected based on lowest AIC. Even though Inverse Gaussian had slightly lower residual deviance, the substantial AIC difference ($\Delta \approx 80$) and the excessive cubic mean-variance relationship (μ^3) were not necessary given the visually observed variance structure on the plots. It is worth bearing in mind the existance of cathegorical predictor in the model - `adhesive_type` - during the model selectio the p-value is low enough to consider it statistically significant.

Full fit model can be tested for interaction using `(...)^2` syntax, I performed sensible pairwise tests for the predictors including their interaction terms, ie. relating `activity_level` to, for instance, `patient_bmi`, `sweat_rate` and `humidity`, relating `experience:humidity` or `experience:calibration_error` etc, yielded no statistically significant interaction terms between the predictors (their p-values were > 0.05)

We consider 2 different predictor selection methods:

- Stepwise, based on AIC using `step()`:

```
Model_1 <- step(gamma_model_log, trace=0)
```

- P-value based selection (threshold: $p > 0.05$):

```
                  Estimate Std. Error  t value Pr(>|t|)
(Intercept)       -4.35185    0.91555 -4.75326  0.00001
skin_temp          0.06358    0.02335  2.72232  0.00761
humidity           0.06568    0.00206 31.88794  0.00000
activity_level    -0.03876    0.01342 -2.88731  0.00474
sweat_rate         0.00144    0.00237  0.60703  0.54516
calibration_error -0.01536    0.00212 -7.23654  0.00000
patient_bmi       -0.01141    0.00859 -1.32773  0.18720
experience         0.01279    0.02267  0.56413  0.57389
adhesive_typeB     0.10686    0.08740  1.22257  0.22428
adhesive_typeC     0.22620    0.09717  2.32781  0.02188
```

Initial full model coefficients were deemed:

- Significant ($p < 0.05$): humidity, calibration_error, skin_temp, activity_level
- Non-significant : sweat_rate (p=0.545), experience (p=0.574), patient_bmi (p=0.187)
- Adhesive type C showed significance (p=0.022), therefore all cathegorical predictors of `adhesive_type` retained.

Comparison between two models:

- Model 1 - step() best.
- Model 2 - p-value based choice.

```
Analysis of Deviance Table

Model 1: lifetime ~ skin_temp + humidity + activity_level + calibration_error +
    patient_bmi + adhesive_type
Model 2: lifetime ~ skin_temp + humidity + activity_level + calibration_error +
    adhesive_type
  Resid. Df Resid. Dev Df Deviance
1       105     16.948
2       106     17.338 -1 -0.38946
```

After the stepwise model selection based on Akaike Information Criterion and selection based on p values. We end up with two models, different in one predictor - patient_bmi. We perform `anova()` test in order to choose one of them.

The comparison shows insignificant difference in residual deviance therefore Model 2 is selected following parsimony principle - patient_bmi does not significantly improve the fit.

---

subtask 3

Final model is chosen to be Gamma GLM with log link with following predictors:

```
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)        -4.5080     0.7712 -5.8456   0.0000
skin_temp           0.0621     0.0225  2.7581   0.0068
humidity            0.0659     0.0021 31.9304   0.0000
activity_level     -0.0371     0.0132 -2.8041   0.0060
calibration_error  -0.0146     0.0021 -6.8801   0.0000
adhesive_typeB      0.1142     0.0885  1.2903   0.1998
adhesive_typeC      0.1876     0.0973  1.9289   0.0564
```

Since BMI and experience are not in the final model, they do not affect the prediction. Using activity_level=4 and sample means for remaining predictors (`adhesive_type` = A as reference):

```
(Intercept)
   2.111193
```

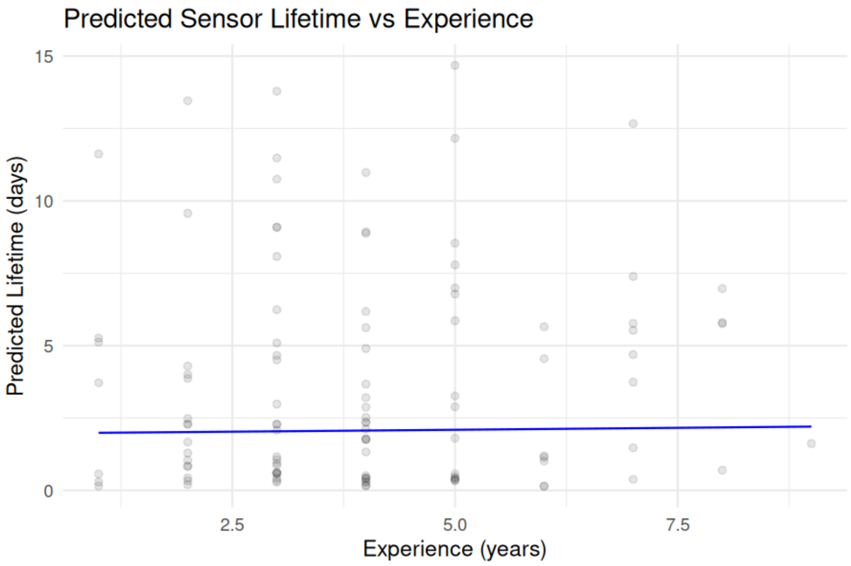
calculated by:

$$\hat{\mu} = \exp(\beta_0 + \beta_1 \cdot \text{skin temp} + \beta_2 \cdot \text{humidity} + \beta_3 \cdot \text{activity level} + \beta_4 \cdot \text{calibration error} + \text{adhesive effect})$$

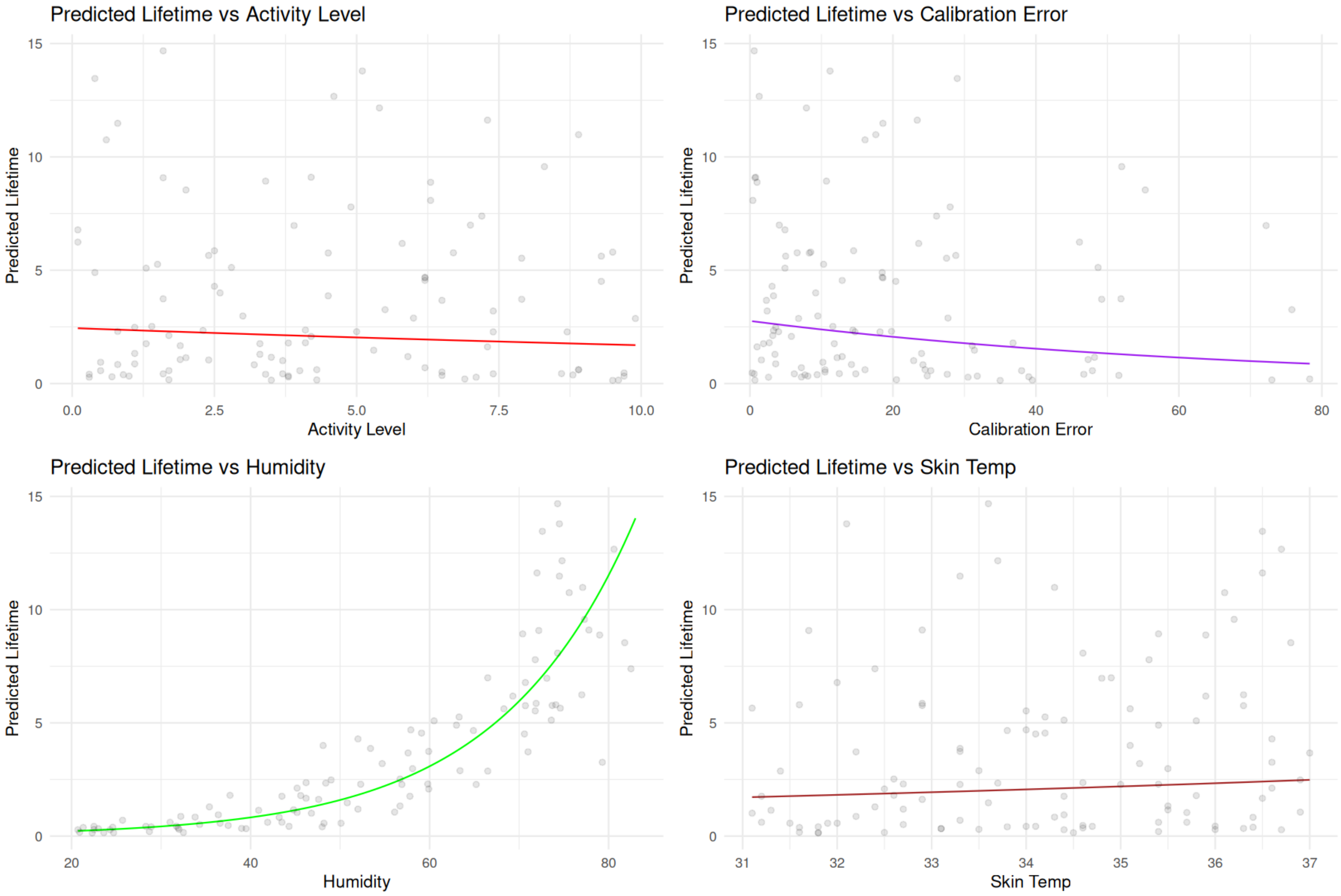
adhesive effect is set to 0 here as we take the default, A type, as the ”mean”

Since experience was removed from the final model, changing experience by 1 year produces no significant change in expected lifetime.

Plots presented here deliberatly cut off the lifetime >15 outlier points (only 4 of them) to increase visibility. As experience is not included as predictor in our final model following plot is made using default, full-fit model including all predictors.



As we can see, user experience does not seem to impact the liftime in a meaningfull way compared to our chosen predictors, which effect is clearly visible on the following plots:



Final Model Deviance:

```
[1] 17.33769
```

Final Model AIC:

```
[1] 294.1473
```

Final Model's Dispersion parameter is:

```r
summary(final_model)$dispersion
```

```
[1] 0.160626
```

This indicates underdispersion. While less problematic than overdispersion, this suggests the variance is smaller than expected under the Gamma model. The model fit quality seems enough, though a potential quasi-Gamma approach could be considered if precision is critical.

Sensor lifetime is significantly influenced by environmental and technical factors: higher skin temperature and higher humidity increases lifetime, while higher calibration_error and activity_level decrease it. Humidity shows the strongest effect, followed by calibration_error, while skin temperature and activity level have more modest impacts. User experience does not significantly impact sensor lifetime and was excluded from the final model. BMI and sweat rate were also deemed non-significant as predictors.

Task 2

Because our transformation g_α is monotonically increasing (trivial checks for $y \geq 0$ and $y < 0$)

$$P(Y_i < y_i) = P(g_\alpha(Y_i) < g_\alpha(y_i))$$

And since $g_\alpha(Y_i)$ is normally distributed, we can use the standard normal CDF Φ with normalized Y_i with its mean and standard error:

$$P(Y_i < y_i) = \Phi\left(\frac{g_\alpha(y_i) - \mathbf{x}_i^\top \beta}{\sigma}\right)$$

Taking the derivative with respect to y_i gives us:

$$\frac{d}{dy_i} P(Y_i < y_i) = \phi\left(\frac{g_\alpha(y_i) - \mathbf{x}_i^\top \beta}{\sigma}\right) \cdot \frac{\dot{g}_\alpha(y_i)}{\sigma}$$

where ϕ is the standard normal density and $\dot{g}_\alpha(y_i)$ being the derivative of $g_\alpha(y_i)$ defined as:

$$\dot{g}_\alpha(y_i) = \begin{cases} (y_i + 1)^{\alpha - 1}, & y \geq 0 \\ (1 - y_i)^{1 - \alpha}, & y < 0 \end{cases}$$

The final density of Y_i is:

$$\begin{aligned} f_{Y_i}(y_i) =& \phi\left(\frac{g_\alpha(y_i) - \mathbf{x}_i^\top \beta}{\sigma}\right) \cdot \frac{\dot{g}_\alpha(y_i)}{\sigma} = \\ =& \frac{1}{\sqrt{2\pi}} \exp\left\{ -\frac{1}{2}\left(\frac{g_\alpha(y_i) - \mathbf{x}_i^\top \beta}{\sigma}\right)^2 \right\} \cdot \frac{\dot{g}_\alpha(y_i)}{\sigma} \end{aligned}$$

$$\ell(\alpha, \beta, \sigma^2; y_1, \dots, y_n) = \sum_{i=1}^{n} \log f_{Y_i}(y_i)$$

For independent Y_i with density f_{Y_i}, this becomes:

$$\ell(\alpha, \beta, \sigma^2; y_1, \dots, y_n) = \log \left[\prod_{i=1}^{n} \frac{1}{\sqrt{2\pi}} \exp\left\{ -\frac{1}{2} \left(\frac{g_\alpha(y_i) - \mathbf{x}_i^\top \beta}{\sigma} \right)^2 \right\} \cdot \frac{\dot{g}_\alpha(y_i)}{\sigma} \right]$$

The estimates derived from equating the derivative of loglikelihood with respect to β and σ to 0 (as we did on the lecture) are:

$$\hat{\beta} = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top g_\alpha(\mathbf{Y})$$

$$\hat{\sigma}^2 = \frac{1}{n} RSS_g$$

Where

$$RSS_g = \sum_{i=1}^{n} \left(g_\alpha(y_i) - \mathbf{x}_i^\top \hat{\beta} \right)^2$$

$$l(\alpha; y_1, \dots, y_n) = \log \left[\prod_{i=1}^{n} \frac{1}{\sqrt{2\pi}} \exp\left\{ -\frac{1}{2} \left(\frac{g_\alpha(y_i) - \mathbf{x}_i^\top \hat{\beta}}{\hat{\sigma}} \right)^2 \right\} \cdot \frac{\dot{g}_\alpha(y_i)}{\hat{\sigma}} \right]$$

Which equals to:

$$= -\frac{n}{2} \log(2\pi) - \frac{1}{2\hat{\sigma}^2} \sum_{i=1}^{n} \left(g_\alpha(y_i) - \mathbf{x}_i^\top \hat{\beta} \right)^2 + \sum_{i=1}^{n} \log \dot{g}_\alpha(y_i) - n \log(\hat{\sigma})$$

Reduced by substituting $RSS_g = \sum_{i=1}^{n} \left(g_\alpha(y_i) - \mathbf{x}_i^\top \hat{\beta} \right)^2$ and $\hat{\sigma}^2 = \frac{1}{n} RSS_g$:

$$-\frac{n}{2} \log(2\pi) - \frac{n}{2} + \sum_{i=1}^{n} \log \dot{g}_\alpha(y_i) - n \log(\sqrt{\frac{RSS_g}{n}})$$

And further into:

$$l(\alpha; y_1, \dots, y_n) = \sum_{i=1}^{n} \log \dot{g}_\alpha(y_i) - \frac{n}{2} \log(\frac{RSS_g}{n}) + const.$$

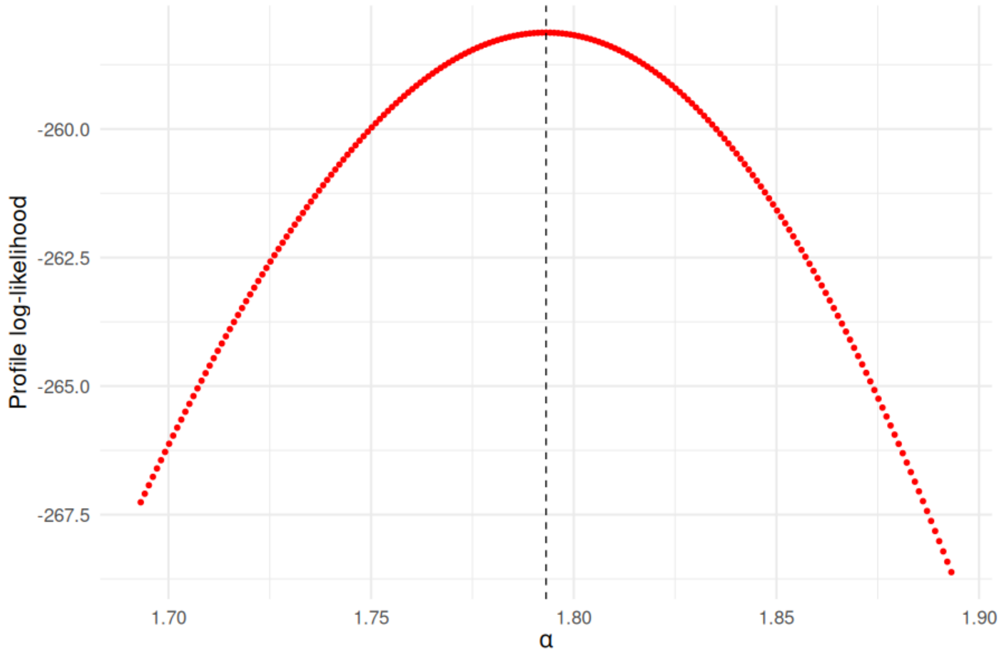
Implementation of those functions in R looks as follows:

```r
g = function(y, alpha) {
  out <- numeric(length(y))

  # case 1: y >= 0, alpha != 0
  in_1 = (y >= 0 & alpha != 0)
  out[in_1] = (((y[in_1] + 1)^(alpha - 1)) / alpha)

  # case 2: y >= 0, alpha == 0
  in_2 = (y >= 0 & alpha == 0)
  out[in_2] = log(y[in_2] + 1)

  # case 3: y < 0, alpha != 2
  in_3 = (y < 0 & alpha != 2)
  out[in_3] = ( - ((((-y[in_3] + 1)^(2 - alpha)) - 1) / (2 - alpha)))

  # case 4: y < 0, alpha == 2
  in_4 = (y < 0 & alpha == 2)
  out[in_4] = ( - log(-y[in_4] + 1))

  return(out)
}

g_prim = function(y, alpha) {
  out <- numeric(length(y))

  # case 1: y >= 0
  in_1 = (y >= 0)
  out[in_1] = (y[in_1] + 1)^(alpha - 1)

  # case 2: y < 0
  in_2 = (y < 0)
  out[in_2] = (1 - y[in_2])^(1 - alpha)

  return(out)
}

profile_log_likelihood = function(Y, X, alpha) {
  n = length(Y)

  # transformed Y
  g_Y <- g(Y, alpha)

  # B_hat estimate
  B_hat = solve(t(X) %*% X) %*% t(X) %*% g_Y

  # Residual Sum of Squares
  RSS = sum((g_Y - X %*% B_hat)^2)

  # Constants
  constants = (-n/2)*log(2*pi) - (n/2)

  # profile log likehood = sum of log(g_prim(y)) -n/2 log(sigma_hat^2) + constants
  out = sum(log(g_prim(Y, alpha))) - ((n/2)*log((1/n) * RSS)) + constants

  return(out)
}
```

We then load the data, add intercept term and find the ML estimate using `optim()`

```
load("Data_Assignment2_Ex2_E2025.rdata")
df_x <- cbind(1, x)
df_y <- Y

out <- optim(
  par = 0,
  fn = function(a) {-profile_log_likelihood(df_y, df_x, a)}
)
```

We plot the values of loglik funciton spread in $\pm$ 0.1 around from `out$par` (which is the result of the optimization) with step of 0.001:



As we can see $\hat{\alpha} \approx 1.8$

Final transformation is then (by substituting $\alpha = 1.8$):

$$g_\alpha(Y) = \begin{cases} \dfrac{(Y+1)^{1.8}-1}{1.8}, & y \geq 0 \\ -\dfrac{(-Y+1)^{0.2}-1}{0.2}, & y < 0 \end{cases}$$