Analysis of the Decay of the Radioactive Element Volatilium

David Chattaway, Dovudkhon Abdubokiev, Huangyu Yang, Timothy Everett, Edward Bogaciu

Assessed Practical 2024 - Sam Power

August 7, 2025

Contents

1 Describing the Problem

In this paper we are interested in analyzing the fictitious element Volatilium. Specifically, we are interested in modelling the rate of decay of this element. We have a data frame, 'surv', which contains the results of 70 experiments. Each experiment involves a specific initial amount of Volatilium left for a set duration and the final amount of Volatilium after the duration. This paper will look at several different types of linear models as suggested by our various colleagues. We shall analyze these models by finding their log-likelihoods, confidence intervals and any residuals in order to assess which linear model fits our data set the best.

2 Henri's Model

2.1 The Model

It has been suggested by Henri that we use a log-linear model of the form:

$$\log n_i^{\text{fin}} = \log n_i^{\text{init}} + \phi_1 + \phi_2 \cdot t_i + \epsilon_i \tag{1}$$

where we will denote n_i^{init} and n_i^{fin} as the initial and final number of particles measured with t_i being the number of hours for which the sample was left to decay. Our ϵ_i is some iid $\mathcal{N}(0, \sigma^2)$ noise for some unknown $\sigma^2 > 0$ and ϕ_1, ϕ_2 are unknown real-valued parameters with the maximum number of entries being $N = 70$.

Firstly we find the negative log-likelihood function, which will allow us to minimize our function using the optim function in R a bit later. In order to do this we must first find our function for the residuals, we can use this to find our negative log-likelihood as our residuals are normally distributed as well as the variable $(\log n_i^{\text{fin}} - \log n_i^{\text{init}} - \phi_1 - \phi_2 \cdot t_i)$. Based on this we set our:

$$\text{Residuals}(\epsilon) = \log n_i^{\text{fin}} - \log n_i^{\text{init}} - \phi_1 - \phi_2 \cdot t_i \tag{2}$$

Using this we can compute our negative log-likelihood function to be:

$$\ell(\phi_1, \phi_2, \sigma^2) = \frac{n}{2} \log 2\pi + \log \sigma^2 + \frac{1}{2\sigma^2} \sum \epsilon^2 \tag{3}$$

Now that we have our negative log-likelihood function we can apply the "optim" function in R to minimize the data set, see appendix for full code. Doing this allows us to compute the MLE's for $\ell(\phi_1, \phi_2, \sigma^2)$ which in turn will allow us to visualize the uncertainty of our data set using 95% confidence intervals. Taking the "optim" function give us:

$$\begin{aligned} \phi_1 &= 0.1441563 \\ \phi_2 &= -0.2764727 \\ \sigma &= 0.08886572 \end{aligned}$$

to seven significant figures. Bellow, in 1, shows the data acquired by the MLE with a 95% confidence interval. We can also visualize the nature of our residuals by the QQ plot as seen in 2.

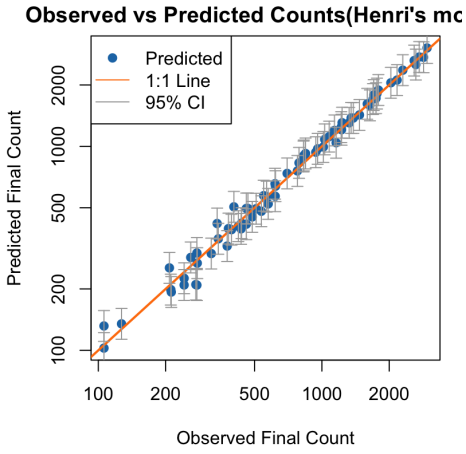

Figure 1: 95% confidence intervals for Henri's model

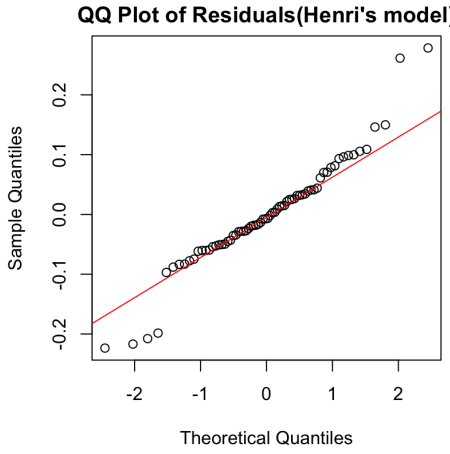

Figure 2: QQ plot for the residuals in Henri's model

2.2 Results

By looking at our two plots we can comment on the quality of Henri's model. We have computed that our estimated variance is approximately 0.09, thus six of our data points, as seen in 2 differ by at least twice this variance from zero so we categorize these as outliers. This could suggest that the model may not be as accurate in lower predictions as it is in the higher predictions, where it is mostly accurate. This is further backed up by our confidence intervals 1 as our predictions tend to the expected values as we increase our counts, within our confidence intervals. Thus, we can conclude that the model is sufficiently accurate for large number of counts.

3 Pierre's Remarks

Pierre claims that if the Volatilium sample is not left alone(i.e. if the time between measurements is zero), then the number of particles should not change at all. We will try to evaluate this claim using hypothesis testing. Pierre's claim corresponds to: $n_i^{fin} = n_i^{init}$ when $t_i = 0$. By taking logs, we obtain: $log(n_i^{fin}) = log(n_i^{init})$. The model proposed by Henri is

$$log(n_i^{fin}) = log(n_i^{init}) + \phi_1 + \phi_2 t_i + \epsilon_i.$$

At time $t_i = 0$, the term $\phi_2 t_i$ becomes zero, so Pierre's claim is $\phi_1 = 0$. We formulate the following hypothesis test to evaluate Pierre's claim:

$$H_0 : \phi_1 = 0$$
$$H_1 : \phi_1 \neq 0$$

To formulate the test, we calculate the Wald test statistic:

$$Z = \frac{\hat{\phi_1} - 0}{SE(\hat{\phi_1})},$$

where $SE(\hat{\phi_1})$ is the standard error and can be obtained from the Hessian matrix. We then calculate the p-value in the following way as this is a two-tailed test:

$$p = 2(1 - \Phi(|Z|),$$

where $\Phi(|Z|)$ is the cumulative standard normal function.
By performing the test in R, we get a p-value of $7.098043e(-9)$ which is very small. Hence, we reject H_0 and conclude from the data that Pierre's claim is wrong.

4 Marie's alternative Model

Marie has proposed an alternative model with a similar structure, claiming that the particle decay process is as follows:

$$n_i^{\text{fin}} = n_i^{\text{init}} \cdot (\theta_1 \cdot \exp(-\theta_2 \cdot t_i) + \delta_i) \tag{4}$$

where δ_i is iid $\mathcal{N}(0, s^2)$ noise, for some unknown $s^2 > 0$ which is normally distributed noise with mean 0 and variance s^2. θ_1, θ_2 are unknown parameters, with $\theta_1 \geq 0$ and θ_2 with real value.

4.1 The Model

Firstly, we compute the final count that the model predicts, defined bellow as:

$$\text{pred}_i = n_i^{\text{init}}(\theta_1 \exp\ (-\theta_2 t_i\ )) \tag{5}$$

with Residuals

$$r_i = n_i^{\text{fin}} - \text{pred}_i \tag{6}$$

We use this function to obtain the ideal parameter estimation to derive the negative log-likelihood, we use this to simplify the computations and to obtain the optimal optimizations. Thus, our negative log-likelihood is given by:

$$\ell(\theta_1, \theta_2, s^2) = \sum_{i=1}^{n} \left\{ \frac{1}{2}\log(2\pi) + \frac{1}{2}\left[\log(s^2) + 2\log(n_i^{\text{init}})\right] + \frac{r_i^2}{2s^2(n_i^{\text{init}})^2} \right\} \tag{7}$$

Using the `optim` function in R for parameter estimation, we set initial settings for the new model based on the parameter estimations from Henri's model in section 2. We set our initial values as $\theta_1, \theta_2, \log(s^2)$. Our predicted value, of the estimated parameters is:

$$\hat{n}_i^{\text{fin}} = n_i^{\text{init}} \left(\hat{\theta}_1 \exp\{-\hat{\theta}_2 t_i\} \right). \tag{8}$$

Our Confidence interval can be calculated as,

$$\hat{n}_i^{\text{fin}} \pm 1.96\, \hat{s}\, n_i^{\text{init}}. \tag{9}$$

4.2 Results

Using the methods described above, we compute the maximum likelihood estimates of Marie's model parameters in R.

$$\hat{\theta}_1 = 1.12578$$
$$\hat{\theta}_2 = 0.2697043$$
$$\hat{s} = 0.01536631$$

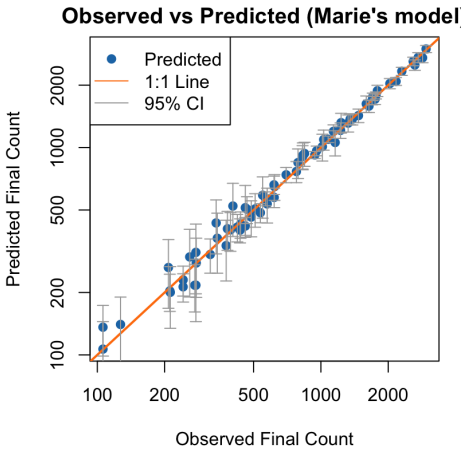

Figure 3: Observed vs. Predicted

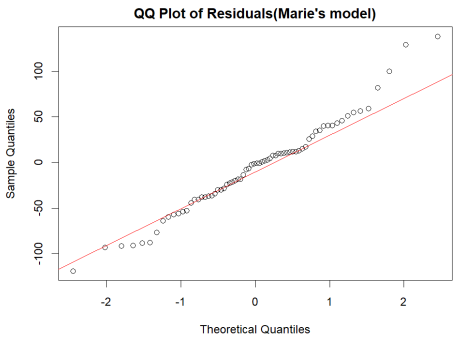

Figure 4: QQ Plot

The plot shown in 3 shows our 95% confidence intervals and follows a similar pattern to Henri's model where our data points are erratic around smaller counts but tend towards the normal distribution as the count increases, as seen by the intervals getting smaller and our points converging

to the orange line. This implies that the model is less accurate at lower counts but becomes increasingly more accurate as we increase the number of counts. Analysis of 4 would suggest at least 10 outliers, as our variance was computed as 0.015, which corroborates the notion that our data is behaving a bit erratically, especially at the lower counts where there are significantly more outliers. This further emphasizes the models strength around larger counts.

5 Comparison between Henri's and Marie's Model

We tested this using the Akaike Information Criterion (AIC). AIC is a metric that helps compare the two models by measuring how well they fit the data and penalizing for complexity, with a lower score seen as a better output. AIC is computed as:

$$AIC = 2k - 2ln(L)$$

where k = number of parameters and L = maximum likelihood of the model. Computing the AIC gives:

$$AIC_{Henri} = -131.2367$$
$$AIC_{Marie} = 731.0751$$

As the AIC in Henri's model is smaller, this suggests his model is more suited to fit the data. This is further supported when looking at the plots of the two models. In Marie's model, the error term is not independent. This is illustrated in the confidence intervals around the fitted values. For small final counts, the confidence intervals are large, which suggests the same thing. On the other hand, the error term in Henri's model is independent and this can be seen from the confidence intervals in 1. Furthermore, QQ plot of residuals in Henri's model shows that they follow normal distribution, with a little deviation from the straight line and a few outliers. However, in Marie's model the residuals deviate more from the straight line and there are many outliers. These suggest that Henri's model is a better fit for the data.

6 Ernest's Problematic Model

Ernest proposes the following 'multi-rate' model:

$$n_i^{fin} = n_i^{init} \cdot \left(\left[\sum_{j=1}^{7} \theta_{1,j} \cdot \exp(-\theta_{2,j} \cdot t_i) \right] + \delta_i \right),$$

where we now have $\theta_{1,j} \geq 0$ and $\theta_{2,j} \in \mathbb{R}$ for every $j \in \{1, 2, \ldots 7\}$.

Marie's model is a special case of Ernest's model, where only $\theta_{1,j}$ and $\theta_{2,j}$ is non-zero. Choosing between these models is a nested model selection problem. Wilk's theorem can be used to test if Ernest's model is better. However, even before attempting to fit this model, we can see that it introduces several difficulties. First of all, Ernest's model has 14 parameters in total. This increases the **computational cost** of finding the optimal values. Secondly, this makes the model highly flexible, which creates the risk of **overfitting** the data. The model may achieve very low error on the training data, but perform poorly on unseen data. Furthermore, as the number of parameters is large, different combinations of the parameters may produce same fits to the data. This means that it is **difficult to find unique estimates** for the parameters. Finally, as the number of parameters is large, it becomes **difficult to interpret the results**. Even if the model works well, it will be hard to understand the meanings of the parameters.

7 Concluding Thoughts

In conclusion, this report has explored various models to try and discover, given the data set provided, which model provides the best representation of said data set, given our residuals. This was done by minimizing our negative log-likelihood and plotting 95% confidence intervals; as well as using AIC to compare both models. Finding that Henri's provided the best overall model for the data set provided. A third model was suggested by Ernest but this was quickly rejected due restrictions with the computational cost and the risk of overfitting the data.

A Full Code

```
##############################################
#START OF CODE

# Loading data
surv_data <- read.csv("surv.csv")

# Extract some important data
init <- surv_data$init
fin <- surv_data$fin
times <- surv_data$times

log_fin <- log(surv_data$fin)
log_init <- log(surv_data$init)


# Question 1: Henri's model
neg_log_likelihood_Henri <- function(params, log_fin, log_init, times) {
  phi1 <- params[1]
  phi2 <- params[2]
  log_sigma_sq <- params[3]  # Log-variance to make sure of positivity
  sigma_sq <- exp(log_sigma_sq)

  residuals <- log_fin - log_init - phi1 - phi2 * times
  n <- length(log_fin)

  nll <- (n/2) * (log(2 * pi) + log_sigma_sq) + (1/(2 * sigma_sq)) * sum(residuals^2)
  return(nll)
}

# The optim function will be used to find the optimizing
# phi1, phi2 and sigma for the model given the data.
initial_params <- c(0, 0, 0)
result_Henri <- optim(
  par = initial_params,
  fn = neg_log_likelihood_Henri,
  log_fin = log_fin,
  log_init = log_init,
  times = times,
  method = "BFGS",
  hessian = TRUE
)

# Final estimates and printing the outcomes
phi1_mle <- result_Henri$par[1]
phi2_mle <- result_Henri$par[2]
sigma_sq_mle <- exp(result_Henri$par[3])
sigma_mle <- sqrt(sigma_sq_mle)  # Standard deviation

cat("MLE Estimates for Henri's model:\n",
    "phi1 =", phi1_mle, "\n",
    "phi2 =", phi2_mle, "\n",
    "sigma =", sigma_mle, "\n")

# Calculate predictions, for later usage
predicted_log_fin_Henri <- log_init + phi1_mle + phi2_mle * times
predicted_fin_Henri <- exp(predicted_log_fin_Henri)

# Calculate 95% confidence intervals
se <- sigma_mle  # Standard error of the residuals
lower_bound_Henri <- exp(predicted_log_fin_Henri - 1.96 * se)  # Lower bound (original scale)
upper_bound_Henri <- exp(predicted_log_fin_Henri + 1.96 * se)  # Upper bound (original scale)

# Creating needed plots
par(mfrow = c(1, 1), mar = c(4.5, 4.5, 2, 1))

# Observed vs Predicted final counts, including a line of the best fit
plot(fin, predicted_fin_Henri,
     xlab = "Observed Final Count", ylab = "Predicted Final Count",
     main = "Observed vs Predicted Counts(Henri's model)",
     pch = 19, col = "#1f77b4", log="xy" )
abline(a = 0, b = 1, col = "#ff7f0e", lwd = 2)

# Add 95% confidence intervals
arrows(fin, lower_bound_Henri, fin, upper_bound_Henri,
       length = 0.05, angle = 90, code = 3, col = "darkgray")

# Add legend
legend("topleft",
       legend = c("Predicted", "1:1 Line", "95% CI"),
       col = c("#1f77b4", "#ff7f0e", "darkgray"),
       pch = c(19, NA, NA),
```

```
       lty = c(NA, 1, 1))

# Residuals vs Fitted, their log values
residuals_log_Henri <- log_fin - predicted_log_fin_Henri
plot(predicted_log_fin_Henri, residuals_log_Henri,
     xlab = "Fitted Values(log)", ylab = "Residuals(log)",
     main = "Residuals vs Fitted(Henri's model)",
     pch = 19, col = "#2ca02c")
abline(h = 0, col = "#d62728", lty = 2, lwd = 2)

# QQ-plot of residuals
qqnorm(residuals_log_Henri, main = "QQ Plot of Residuals(Henri's model)")
qqline(residuals_log_Henri, col = "red")

# Question 2: Hypothesis test for Pierre's claim
# H0: phi1 = 0, H1 = phi1!= 0.
# Extract standard error of phi1 from the inverse Hessian matrix

se_phi1 <- sqrt(solve(result_Henri$hessian)[1, 1])  # First diagonal element is variance of phi1

# Compute Wald test statistic
Z_phi1 <- phi1_mle / se_phi1

# Compute two-tailed p-value
p_value <- 2 * (1 - pnorm(abs(Z_phi1)))

# Print results
cat("Wald Test Statistic for H0: phi1 = 0 is ", Z_phi1, "\n",
    "p-value:", p_value, "\n")

# Decision rule
if (p_value < 0.05) {
  cat("Reject H0: phi1 is significantly different from 0. Pierre's claim is not supported.\n")
} else {
  cat("Fail to reject H0: No significant evidence against Pierre's claim. His hypothesis holds.\n")
}

#Question 3: Marie's Model

#Define negative log-likelihood function of the residuals.
#Params is a numeric vector of length 3 that holds (theta1, theta2, log_s_sq).
#fin, init, and times are vectors of the same number of observations.
#Extract three variables from the parameter vector.
neg_log_likelihood_Marie <- function(params, fin, init, times) {
  theta1   <- params[1]
  theta2   <- params[2]
  log_s_sq <- params[3]

  #Convert log_s_sq to s^2 by exponentiating
  s_sq <- exp(log_s_sq)

  #Compute the predicted final count under the model
  pred <- init * (theta1 * exp(-theta2 * times))

  #Calculate residuals (observed - predicted)
  resid <- fin - pred

  #Compute the negative log-likelihood contribution.
  nll <- sum(
    0.5 * log(2 * pi) + 0.5 * (log_s_sq + 2 * log(init)) + 0.5 * (resid^2) / (s_sq * init^2)
  )

  return(nll) #Return the total negative log-likelihood
}

# Use Q1-based MLE to initialise Q3 parameters
theta1_init <- exp(phi1_mle)
theta2_init <- -phi2_mle
log_s_sq_init <- log(sigma_mle^2)

initial_params_Marie <- c(theta1_init, theta2_init, log_s_sq_init)

# Optimising Marie's model
# The optim function will be used to find the optimizing
# theta1, theta2 and log_s_sq for the model given the data.
result_Marie <- optim(
  par   = initial_params_Marie,
  fn    = neg_log_likelihood_Marie,
  fin   = fin,
  init  = init,
  times = times,
  method = "BFGS",
  hessian = TRUE
)
```

```
#Extract maximum likelihood estimates from the optimization output
theta1_mle <- result_Marie$par[1]
theta2_mle <- result_Marie$par[2]
s_sq_mle   <- exp(result_Marie$par[3]) #The estimate for s^2 is exp(log_s_sq).
s_mle      <- sqrt(s_sq_mle) #The standard deviation s is the square root of s^2.

#print the estimated data
cat("MLE Estimate for Marie's Module:\n",
    "theta1 =", theta1_mle, "\n",
    "theta2 =", theta2_mle, "\n",
    "s      =", s_mle, "\n")

#Calculate the predicted final counts under Marie's model
predicted_fin_Marie <- init * (theta1_mle * exp(-theta2_mle * times))

#Compute the 95% confidence intervals
lower_bound_Marie <- predicted_fin_Marie - 1.96 * s_mle * init
upper_bound_Marie <- predicted_fin_Marie + 1.96 * s_mle * init

#Plot Observed vs. Predicted final counts, including a line of the best fit
par(mfrow = c(1, 1), mar = c(4.5, 4.5, 2, 1))
plot(fin, predicted_fin_Marie,
     xlab = "Observed Final Count",
     ylab = "Predicted Final Count",
     main = "Observed vs Predicted (Marie's model)",
     pch = 19, col = "#1f77b4", log = "xy")
abline(a = 0, b = 1, col = "#ff7f0e", lwd = 2)
arrows(fin, lower_bound_Marie,
       fin, upper_bound_Marie,
       length = 0.05, angle = 90, code = 3, col = "darkgray")
#Draw vertical error bars from lower_bound to upper_bound at each observed value.

legend("topleft",
       legend = c("Predicted", "1:1 Line", "95% CI"),
       col    = c("#1f77b4", "#ff7f0e", "darkgray"),
       pch    = c(19, NA, NA),
       lty    = c(NA, 1, 1))

#Residuals vs Fitted
residuals_Marie = fin - predicted_fin_Marie
plot(predicted_fin_Marie, residuals_Marie,
     xlab = "Fitted Values",
     ylab = "Residuals",
     main = "Residuals vs Fitted (Marie's model)",
     pch = 19, col = "#2ca02c")

#Draw a horizontal line at y=0
abline(h = 0, col = "#d62728", lty = 2, lwd = 2)

# QQ-plot of residuals
qqnorm(residuals_Marie, main = "QQ Plot of Residuals(Marie's model)")
qqline(residuals_Marie, col = "red")

# Q4
##AIC

# AIC = 2k - 2*log(L)
AIC_Henri <- 2 * length(result_Henri$par) + 2 * result_Henri$value
AIC_Marie <- 2 * length(result_Marie$par) + 2 * result_Marie$value


cat("AIC:\n",
    "Henri's Model:", AIC_Henri, "\n",
    "Marie's Model:", AIC_Marie, "\n")

#END OF CODE
################################################
```