Principles of statistical inference project - Part 1

Carmel Gafa'

March 9, 2025

1 Question 1

Consider a random variable X that follows an exponential distribution with scale parameter λ.

The **Exponential Distribution** is a continuous probability distribution that represents the time intervals between consecutive events in a Poisson process, where events happen independently and at a constant average rate. It is defined by a single parameter, λ, referred to as the rate parameter.

1.1 Give reference to a publication in which the exponential distribution has been used in practise. Explain the context in which this distribution has been used in this publication.

Mahmud et al. presented a study where they analyzed and estimated response times to questions on Twitter [1]. The authors developed predictive models to estimate response wait times, exploring three different approaches:

- Personalized wait time models: These models estimate the wait time for a specific user based on their individual history of response wait times. They assume each response event for a user occurs continuously and independently at a constant average rate, modelled by an exponential distribution. Each user's rate parameter λ is estimated as the inverse of their average past response wait times.

  These models demonstrated a promising ability to estimate response times on Twitter. They generally outperformed generalized models and showed reasonable accuracy, especially for an hour or more time limits. The choice of cut-off probability (a threshold used to determine whether a user is considered sufficiently likely to respond to a question on Twitter within a given period) significantly influenced the precision and recall of the predictions.

- Generalized wait time models: Instead of individual models, a single model is built using the previous responses of all users in the dataset, again using the exponential distribution. The rate parameter λ is estimated from the responses of all users. This model underperformed compared to the personalized models in estimating response times on Twitter.

- Time-sensitive wait time models: These models incorporate sensitivity to the time of day or day of the week when questions are sent for both generalized and personalized models by calculating the rate parameter based on responses to questions sent during a specific day or hour. Personalized time-sensitive models only considered users with at least five responses during the modelled time interval. Incorporating time sensitivity had a modest positive impact on the generalized models but did not consistently improve the performance of the personalized models.

1.2 State the mean and the variance of X.

As X follows an exponential distribution with scale parameter, $X \sim Exp(X)$, the expected value or mean is:

$$E[X] = \frac{1}{\lambda} \tag{1}$$

and the variance is:

$$Var(X) = \frac{1}{\lambda^2} \tag{2}$$

1.3 Derive the moment estimator of λ.

The p.d.f. for an exponential distribution is:

$$f(x) = \lambda e^{-\lambda x}, x \leq 0, \lambda > 0 \tag{3}$$

The first moment, or expected value:

$$\begin{aligned} E[X] &= \int_0^\infty x f(x) dx \\ &= \int_0^\infty x \lambda e^{-\lambda x} dx \\ &= \lambda \int_0^\infty x e^{-\lambda x} \end{aligned}$$

Integrating by parts, we let:

$u = x \qquad du = (1)dx$

$v = -\frac{e^{-\lambda x}}{\lambda} \qquad dv = e^{-\lambda x} dx$

As $\int udv = uv - \int vdu$:

$$
\begin{aligned}
E[X] &= \lambda\left(\left[-\frac{xe^{-\lambda x}}{\lambda}\right]_0^\infty - \int_0^\infty -\frac{e^{-\lambda x}}{\lambda}(1)dx\right) \\
&= \left[-xe^{-\lambda x}\right]_0^\infty + \int_0^\infty -e^{-\lambda x}dx
\end{aligned}
$$

Let us consider $\left[xe^{-\lambda x}\right]_0^\infty$:

- $-xe^{-\lambda x} = 0$, when $x = 0$
- $\lim\limits_{x\to\infty} xe^{-\lambda x} = 0$, as exponential decay dominates polynomial growth

So the first term is removed;

$$
\begin{aligned}
E[X] &= \int_0^\infty -e^{-\lambda x}dx \\
&= \left[-\frac{1}{\lambda} - e^{-\lambda x}\right]_0^\infty \\
&= 0 - \left(-\frac{1}{\lambda}\right) \\
&= \frac{1}{\lambda}
\end{aligned}
$$

As in the method of moments the sample mean is equal to theoretical expectation;

$$E[X] = \frac{1}{\lambda} = \overline{X}$$

and solving for λ

$$\hat{\lambda} = \frac{1}{\overline{X}} \tag{4}$$

1.4 Use the second moment to obtain another estimator of λ

For an exponential distribution with rate parameter λ, the second moment,

$$
\begin{aligned}
E[X^2] &= \int_0^\infty x^2 f(x)dx \\
&= \int_0^\infty x^2\lambda e^{-\lambda x}dx \\
&= \lambda\int_0^\infty x^2 e^{-\lambda x}
\end{aligned}
$$

We let:

$u = x^2 \qquad du = 2xdx$

$v = -\frac{e^{-\lambda x}}{\lambda} \qquad dv = e^{-\lambda x}dx$

$$\begin{aligned} E[X^2] &= \lambda \left(\left[-\frac{x^2 e^{-\lambda x}}{\lambda} \right]_0^\infty - \int_0^\infty -\frac{e^{-\lambda x}}{\lambda} 2xdx \right) \\ &= \left[-x^2 e^{-\lambda x} \right]_0^\infty + \int_0^\infty 2xe^{-\lambda x}dx \end{aligned}$$

Let us consider $\left[x^2 e^{-\lambda x}\right]_0^\infty$:

- $-x^2 e^{-\lambda x} = 0$, when $x = 0$
- $\lim_{x \to \infty} x^2 e^{-\lambda x} = 0$, as exponential decay dominates polynomial growth

Then

$$E[X^2] = \int_0^\infty 2xe^{-\lambda x}dx$$

We let:

$u = x \qquad du = dx$

$v = -\frac{e^{-\lambda x}}{\lambda} \qquad dv = e^{-\lambda x}dx$

$$E[X^2] = 2 \left(\left[-\frac{xe^{-\lambda x}}{\lambda} \right] - \int_0^\infty e^{-\lambda x}dx \right)$$

We have seen previously that the first term will equate to zero.

$$\begin{aligned} E[X^2] &= 2 \int_0^\infty \frac{e^{-\lambda x}}{\lambda} dx \\ &= \left[-\frac{2e^{-\lambda x}}{\lambda^2} \right]_0^\infty \\ &= 0 - \left(-\frac{2}{\lambda^2} \right) \\ &= \frac{2}{\lambda^2} \end{aligned}$$

The variance of the exponential distribution is given by:

$$\begin{aligned} Var(X) &= E[X^2] - E[X]^2 \\ &= \frac{2}{\lambda^2} - \left(\frac{1}{\lambda} \right)^2 \\ &= \frac{1}{\lambda^2} \end{aligned}$$

We can estimate the sample variance

$$\hat{\sigma}^2 = \frac{1}{\hat{\lambda}^2}$$

and solving for λ

$$\hat{\lambda} = \frac{1}{\sqrt{\hat{\sigma}^2}} \tag{5}$$

1.5 Comment on the unbiasedness and consistency of the moment estimator for λ derived in Q1iii. State any assumption/s that need to be made to check for unbiasedness and consistency.

1.6 Use R software to generate 1000 data points from an exponential distributed random variable using any admissible parameter value for λ

The R script generates 1000 points from an exponentially distributed random variable with a rate parameter λ of 1.5. It then plots a histogram of these points and overlays the theoretical density function of the exponential distribution. The result is shown in Figure 1.

```
library(ggplot2)
library(glue)

set.seed(50)
lambda <- 1.5
x <- rexp(n = 1000, rate = lambda)

data <- data.frame(x = x)

p <- ggplot(data,
                aes(x = x)) +
        geom_histogram(
                aes(y = after_stat(density)),
                bins = 50, fill = "blue",
                color = "black",
                alpha = 0.6) +
        stat_function(
                fun = function(x) lambda * exp(-lambda * x),
                color = "red",
                size = 1) +
        labs(title = glue("Histogram of exponentially distributed
        random variable with lambda = {lambda}"),
                x = "x", y = "Density") +
        theme_minimal() +
        theme(plot.title = element_text(hjust = 0.5))
```

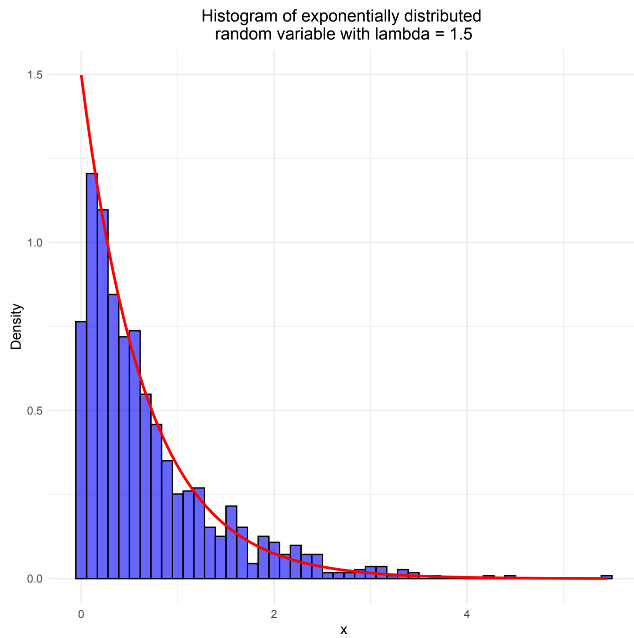

Figure 1: Exponential distributed random variable; histogram of 1000 generated points and theoretical distribution

1.6.1 Write down the log-likelihood function for this exponentially distributed random variable.

For sample $\mathbf{x} = (x_1, \ldots, x_n)^T$ obtained on an exponential distributed random variable X, with parameter vector $\theta = (\lambda)$, the likelihood

$$\begin{aligned} L(\mathbf{x}, \theta) &= \prod_{i=1}^{n} f(x_i, \theta) \\ &= \prod_{i=1}^{n} \lambda e^{-\lambda x_i} \\ &= \lambda^n e^{\sum_{i=1}^{n} \lambda x_i} \end{aligned}$$

The log-likelihood is then

$$l(\mathbf{x}, \theta) = nlog(\lambda) - \lambda \sum_{i=0}^{n} x_i$$

Taking the derivative with respect to λ,

$$\frac{\partial l(\mathbf{x}, \theta)}{\partial \lambda} = \frac{n}{\lambda} - \sum_{i=0}^{n} x_i$$

for maximum $\frac{\partial l(\mathbf{x},\theta)}{\partial \lambda}$

$$L\frac{\partial l(\mathbf{x},\theta)}{\partial \lambda} = 0$$
$$\frac{n}{\lambda} - \sum_{i=0}^{n} x_i = 0$$
$$\frac{n}{\lambda} = \sum_{i=0}^{n} x_i$$

So that

$$\hat{\lambda} = \frac{n}{\sum_{i=0}^{n} x_i} = \frac{1}{\bar{x}} \tag{6}$$

1.6.2 Evaluate the log-likelihood function for the generated data as a function of λ, and plot the resulting log-likelihood function against different values of λ. Present the plot together with the answers.

The follwoing listing calculates and plots the log-likelihood values for an exponential distribution with varying λ values. The resulting plot can be examined in Figure 2

```
        lambda_values <- seq(0.1, 5, by = 0.01)
        log_likelihood_values <- sapply(lambda_values,
        function(lambda) LL_exponential(lambda, x))

        df <- data.frame(lambda_values, log_likelihood_values)

        p <- ggplot(df,
                aes(x = lambda_values,
                y = log_likelihood_values)) +
        geom_point(
                color = "blue",
                alpha = 0.6) +
                labs(
                title = "Log-Likelihood with varying
                 lambda values",
                x = "Lambda",
                y = "Log-Likelihood") +
        theme_bw()

        quartz()
        print(p)
```

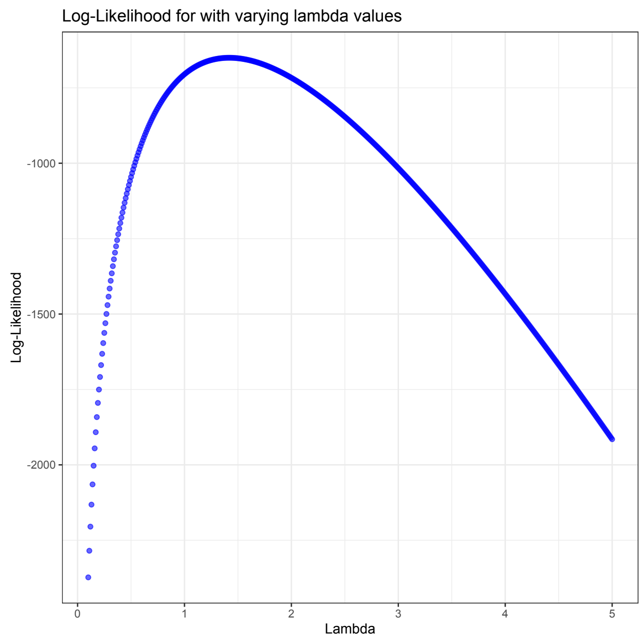

Figure 2: Log-likelihood plot for an exponential distribution with varying λ

1.6.3 Using the plot or otherwise, which estimate for λ is the MLE? Give a reason for your answer.

As we have seen in the previous question, the maximum likelihood estimation is the value for which the derivative of the log-likelihood with respect to lambda is zero, that is the peak of the curve shown in Figure 2. This can be easily calculated using the code below. The value obtained for $\hat{\lambda}$ was **1.43**.

```
max_ll <- max(log_likelihood_values)
max_lambda <- lambda_values[log_likelihood_values == max_ll]
print(glue("Lambda value for maximum log-likelihood is
    {max_lambda}"))
```

2 title

3 title

4 Question 4 - Jackknife and bootstrap

Consider 50 observations of bivariate pair (X, Y) in resampling.xlsx. Use the nls command in R to estimate the nonlinear regression $Y = \frac{aX}{b+X} + \epsilon$.

The code that estimated the nonlinear regression is below. The plots obtained are shown in Figure 3. The estimated values of a is $\hat{a} = 14.56$ and b is $\hat{b} = 7.10$

```
library(openxlsx)
library(ggplot2)

# load file
script_dir <- getwd()
file_path <- file.path(script_dir, "resampling.xlsx")
df <- read.xlsx(file_path, colNames = TRUE)

print(c("number of rows: ", nrow(df)))

# estimate the parameters of the model
init_a <- 1
init_b <- 1

nls_model <- nls(y ~ (a * x) / (b + x),
data = df,
start = list(a = init_a, b = init_b))


estimated_params <- coef(nls_model)
a_hat <- estimated_params["a"]
b_hat <- estimated_params["b"]
cat("Estimated a:", a_hat, "\n")
cat("Estimated b:", b_hat, "\n")

# predict the values of y
df$Predicted <- predict(nls_model)
print(head(df))

# plot the data
p <- ggplot(df, aes(x = x  , y = y)) +
geom_point(color = "blue", alpha = 0.5) +
geom_line(aes(
        y = Predicted),
        color = "red",
        linewidth = 1) +
labs(title = "Nonlinear Regression: Y = (aX) / (b+X)",
x = "X", y = "Y") +
theme_minimal()

print(p)
```

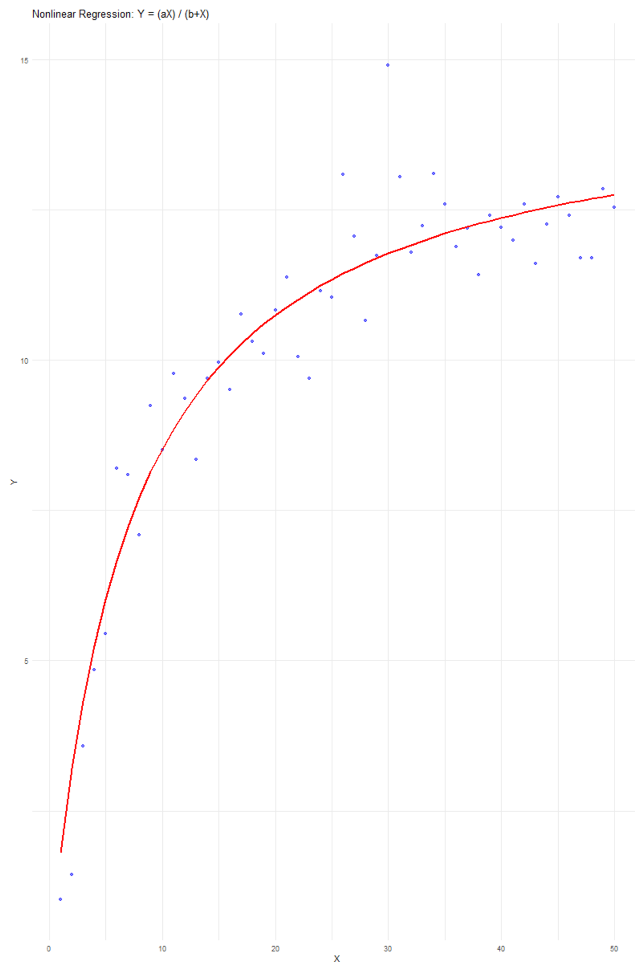

Figure 3: Nonlinear regression fit: observed vs. predicted values

4.1 Construct a computer code in R to find the Jackknife and Bootstrap estimators of a and b. In the case of Jackknife, section randomly the sampling into 5 partitions of size 10. In the case of Bootstrap, generate 1000 samples of size 100 with replacement.

The code below executes the following steps on the data.

1. **Shuffle the dataset:** The data is randomly shuffled to remove any ordering bias:
2. **Divide the data into m Jackknife partitions:** The dataset is split

into $m = 5$ partitions, each missing a unique subset of 5 elements. Note that the number of partitions was maintained as a parameter

3. **Fit the NLS model for each Jackknife sample:** A nonlinear regression model is fitted to each Jackknife sample using Nonlinear Least Squares (NLS) to estimate parameters a and b. The model is defined as:

$$Y = \frac{aX}{b + X} \tag{7}$$

   and is refitted for each sample S_{-a}, which excludes partition P_a.

4. **Compute Jackknife bias-corrected estimates:** The Jackknife estimate for each parameter is calculated using the bias correction formula:

$$\hat{\theta}_{\text{jack}} = m\hat{\theta} - (m-1)\hat{\theta}_{(-a)} \tag{8}$$

   where:

   - $m = 5$ is the number of partitions.
   - $\hat{\theta}$ is the parameter estimate from the full dataset.
   - $\hat{\theta}_{(-a)}$ is the parameter estimate from the jackknife sample with partition a removed.

5. **Compute final Jackknife estimates for a and b:** The final Jackknife estimates for a and b are obtained by averaging the bias-corrected values across all jackknife samples:

$$\hat{a}_{jk} = \frac{1}{m}\sum_{a=1}^{m} \hat{a}_{\text{jack},a}, \quad \hat{b}_{jk} = \frac{1}{m}\sum_{a=1}^{m} \hat{b}_{\text{jack},a} \tag{9}$$

```
# -------------------------
# Jackknife
# -------------------------


number_partitions <- 5

#shuffle the dataframe
set.seed(123)
df_shuf <- df[sample(nrow(df)), ]

# Generate jackknife samples by removing each fold of 5 elements
jackknife_samples <- lapply(1:number_partitions,
function(a) df_shuf[-((number_partitions * (a - 1) +
    1):(number_partitions * a)), ])


# we have calculated theta_hat_m before
theta_hat_m <- estimated_params

```

```
theta_m_a <- function(data) {
        model <- nls(y ~ (a * x) / (b + x),
        data = data,
        start = list(a = theta_hat_m["a"],
        b = theta_hat_m["b"]))
        return(coef(model))
}

# jackknife estimator for each partition
nlsjk <- sapply(jackknife_samples, function(y_a) number_partitions * theta_hat_m - (number_partitions - 1) * theta_m_a(y_a))

#evaluating the jackknife estimator of the parametrs
jackknife_estimates <- rowMeans(nlsjk)

a_hat_jk <- jackknife_estimates["a"]
b_hat_jk <- jackknife_estimates["b"]


df$predicted_jk <- (a_hat_jk * df$x) / (b_hat_jk + df$x)


# Plot with Jackknife predictions and legend
p_jk <- ggplot(df, aes(x = x, y = y)) +
geom_point(color = "blue", alpha = 0.5, size = 3) +
geom_line(aes(y = Predicted, color = "Full Sample Prediction"),
linewidth = 1, linetype = "dashed") +
geom_line(aes(y = predicted_jk, color = "Jackknife Prediction"),
linewidth = 1.2) +
labs(title = "Nonlinear Regression: Full Sample vs. Jackknife",
x = "X", y = "Y", color = "Legend") +
theme_minimal() +
scale_color_manual(values = c("Full Sample Prediction" = "red",
"Jackknife Prediction" = "green"))

# Print the plot
print(p_jk)
```

The resulting plot is shown below in Figure 4

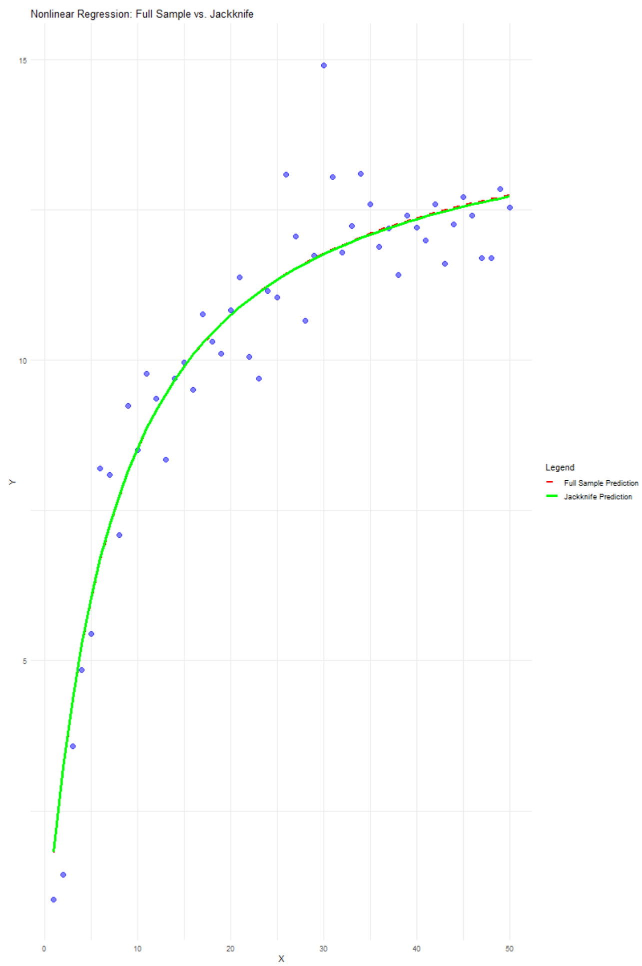

Figure 4: Nonlinear regression fit: observed vs. predicted values including Jackknife predictions

References

[1] Jalal Mahmud, Jilin Chen, and Jeffrey Nichols. When will you answer this? estimating response time in twitter. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 7, pages 697–700, 2013.