1. Times were recorded at which 41 vehicles passed a fixed point on the M1 motorway in Bedfordshire, England on March 23, 1985.1 The times were subtracted to form 40 intervals between successive cars. These interarrival times, rounded to the nearest second, are:

12, 2, 6, 2, 19, 5, 34, 4, 1, 4, 8, 7, 1, 21, 6, 11, 8, 28, 6, 4, 5, 1, 18, 9, 5, 1, 21, 1, 1, 5, 3, 14, 5, 3, 4, 5, 1, 3, 16, 2

* 1. A common model for interarrival times is a random sample from an exponential distribution. Do you think an exponential distribution provides a good model for the interarrival times? Justify your answer.

Exponential models are adequate for interarrival times. This can be demonstrated with a quantile-quantile plot comparing the observed data with a theoretical exponential distribution.

![A close up of a map

Description automatically generated]()

Alternatively, a simple plot of the sorted values can also be plotted against a fitted exponential curve, and it can be shown to be a relatively good fit.  
![A picture containing map

Description automatically generated]()

interarrival\_times <- c(12, 2, 6, 2, 19, 5, 34, 4, 1, 4, 8, 7, 1, 21, 6, 11, 8, 28, 6, 4, 5, 1, 18, 9, 5, 1, 21, 1, 1, 5, 3, 14, 5, 3, 4, 5, 1, 3, 16, 2)

arrival\_interval=diff(interarrival\_times)

exponential.quantiles = qexp(ppoints(length(interarrival\_times)))

# q-q plot

qqplot(exponential.quantiles, interarrival\_times,main="Exponential Q-Q Plot of Arrival Intervals",

xlab = "Theoretical Exponential Quantiles", ylab = "Empirical Quantiles")

lines(exponential.quantiles,exponential.quantiles\*mean(interarrival\_times))

# standard plot

x <-c(1:40)

y <-sort(interarrival\_times)

plot(x,y,main='Exponential Plot',xlab='Sorted Position',ylab='Time')

data.df <- data.frame(x=x,y=y)

model\_expo <- lm(y ~ x + I(x^2), data = data.df)

result <- data.frame(x = seq(1, 40, by = 1))

result$model\_expo <- predict(model\_expo)

result <- melt(result,

id.vars = "x",

variable.name = "model",

value.name = "fitted")

lines(x,result$fitted)

* 1. When interarrival times are randomly sampled from an exponential distribution, the counts of events per unit time are a random sample from a Poisson distribution. Using a time unit of 15 seconds, find the number of cars passing in each 15-second block of time after the initial car. (The initial car is used to bound the recording interval, so the total car count in your data set should be 40.) Do you think a Poisson distribution provides a good model for the count data? Justify your answer.

A Poisson distribution fits the data, but in this case the numbers of both possible counts and observed counts are low enough to be uncertain.

![A close up of a logo

Description automatically generated]()

actual\_counts <- c(2, 2, 1, 1, 0, 4, 3, 0, 2, 1, 1, 1, 4, 0, 2, 2, 3, 2, 4, 3, 2)

interval\_hist <- c(3,5,7,3,3)

counts <- c(0:4)

barplot(interval\_hist,

main='Histogram of Arrivals per Period',

ylab='Count of Cars',

xlab='Count of Periods with Y cars',

names.arg=counts)

data.df <- data.frame(period\_counts=counts,car\_counts=interval\_hist)

print(data.df)

poisson <- dpois(0:5, mean(actual\_counts))

poisson <- poisson \* sum(interval\_hist)

print(poisson)

lines(poisson)

* 1. Assume that Λ, the rate parameter of the Poisson distribution has a discrete uniform prior distribution on 20 equally spaced values between (0.2, 0.4, ..., 3.8, 4.0) cars per 15-second interval. Find the posterior distribution after observing the first 10 observations of car counts in 15 second intervals. Find the posterior mean, standard deviation, median and 95th percentile of Λ given the first 10 observations.

![A screenshot of a cell phone

Description automatically generated]()

* Posterior mean: 1.700
* Standard deviation: 0.412
* Median: 1.6
* 95th percentile: 2.4

lambda <- seq(length=20,from=0.2,to=4)

priors <- array(1/20,20)

barplot(priors,

main="Prior Dist",

xlab="Counts of Cars per Period",

ylab="Prior Probability",

names.arg=lambda,

border="darkblue",

col="lightblue",

ylim=c(0,0.25))

#print(lambda)

#print(priors)

lik <- array(1,length(lambda)) # Initialize likelihood as a constant

for (i in 1:10) {

lik <- lik\*dpois(actual\_counts[i],lambda) # Multiply by Likelihood

}

# The posterior distribution

postDist <- priors\*lik # Prior times likelihood

postDist <- postDist/sum(postDist) # Normalize to sum to 1

barplot(postDist,

main="Posterior Dist",

xlab="Counts of Cars per Period",

ylab="Probability",

names.arg=lambda,

border="darkblue",

col="lightblue",

ylim=c(0,0.25))

# define a function to return medians and percentiles

# the formula returns the index+1 of the last value smaller

# than the target percentile, while will always be argmin(f(x)>q)

# shown on page 19 of unit 2 notes

dist\_percentile <- function(distr,percentile){

postDistCumu <- cumsum(postDist)

index <- 0

for (i in 1:length(postDistCumu)) {

if (postDistCumu[i] < percentile){

index <- i

}

}

index<-index+1

return(index)

}

postMean <- sum(lambda\*postDist)

postVar <- sum((lambda-postMean)^2\*postDist)

postSD <- sqrt(postVar)

postMedian <- lambda[dist\_percentile(postDist,.5)]

post95th <- lambda[dist\_percentile(postDist,0.95)]

* 1. Using the posterior distribution from part c as the prior distribution, find the new posterior distribution after observing the remaining observations. Find the posterior mean, standard deviation, median and 95th percentile of Λ given all observations.

![A screenshot of a cell phone

Description automatically generated]()

* Posterior mean: 1.950
* Standard deviation: 0.312
* Median: 2
* 95th percentile: 2.4

The two lines below that are commented out can be replaced by the lines immediately following them; they provide the same functionality either way.

#lik <- postDist # Initialize likelihood as a constant

lik <- array(1,length(lambda))

for (i in 11:20) {

lik <- lik\*dpois(actual\_counts[i],lambda) # Multiply by Likelihood

}

#postDist <- priors\*lik # Prior times likelihood

postDist <- postDist\*lik

postDist <- postDist/sum(postDist) # Normalize to sum to 1

barplot(postDist,

main="2nd Posterior Dist",

xlab="Counts of Cars per Period",

ylab=" Probability",

names.arg=lambda,

border="darkblue",

col="lightblue",

ylim=c(0,0.3))

postMean <- sum(lambda\*postDist)

postVar <- sum((lambda-postMean)^2\*postDist)

postSD <- sqrt(postVar)

postMedian <- lambda[dist\_percentile(postDist,.5)]

post95th <- lambda[dist\_percentile(postDist,0.95)]

* 1. Find the predictive distribution for the number of cars in the next 15-second interval. Find the predictive probability that 0, 1, 2, 3, 4, and more than 4 cars will pass the point in the next 15 seconds.

![A screenshot of a cell phone

Description automatically generated]()

* 0: 5.328478e-14
* 1: 6.045425e-02
* 2: 8.943108e-01
* 3: 4.521689e-02
* 4: 1.811001e-05

I may have misunderstood this component; if it is incorrect please provide feedback pointing me toward creating a predictive distribution, as I merely summed the posterior distribution.

* 1. Discuss what your results mean in terms of traffic on this motorway.

If this flow rate of traffic is typical of this motorway, it is a relatively busy motorway, with just under two cars passing during each 15 second period. However, given that motorways typically experience patterns of traffic that fluctuate with several orders of seasonality, commenting on the traffic from such a limited span from an unknown time and day is not necessarily useful.