Spatial Statistics pre-course quiz

Question 1

Download a shapefile (`.shp`) of New Zealand's road network from the LINZ data service (link here) choosing the EPSG: 2193 projection. Using the R package `sf` read the shapefile into `R`.

1. Plot the complete network
2. Plot only `route_name "State Highway"`.
3. Download a shapefile of NZ using `nz <- raster::getData("GADM", country = "NZ", level = 0)`, project this onto the same projection as the road network above and plot both objects together (HINT you may find the following `sf` functions useful `st_as_sf()`, `st_transform()`, `st_crs()`).

Question 2

A Poisson process counts the number of events occurring in a fixed time or space when events occur independently and at a constant average rate, λ.

For $X \sim \text{Poisson}(\lambda)$,

$$f_X(x) = P(X = x) = \frac{\lambda^x}{x!}e^{-\lambda}$$

for $x = 0, 1, 2, \ldots$.

Suppose that $x_1, \ldots, x_n$ are iid observations from a Poisson distribution with unknown parameter λ then the likelihood function is

$$L(\lambda\,; x_1, \ldots, x_n) = Ke^{-n\lambda}\,\lambda^{n\overline{x}}\,,$$

where $\overline{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$, and $K = \prod_{i=1}^{n} \frac{1}{x_i\,!}$ is a constant that doesn't depend on λ.

What is the MLE for λ?

Given the vector of counts given by `x` below, which we assume are Poisson random variables, use `optimise()` in `R` to estimate λ.

```
x <- c(0, 38, 27, 0, 78, 99, 92, 96, 100, 105, 103, 104, 96, 108, 99, 85, 88, 90, 97, 101, 93)
```

Question 3

Consider the Gamma distribution with the density function

$$f(x) = \frac{b^a}{\Gamma(a)}\exp(-bx)x^{a-1}$$

for $x, a, b > 0$. Using the following quantities

$$\log(f(x)) = (a-1)\log(x) - bx + C$$

for some constant C, and

$$\frac{d\log(f(x))}{dx} = \frac{a-1}{x} - b,$$

and

$$\frac{d^2\log(f(x))}{d^2x} = -\frac{a-1}{x^2}$$

solve $\frac{d\log(f(x))}{dx} = 0$ to get $x^* = ...$ Using this value evaluate $-1/\frac{d^2\log(f(x))}{d^2x}$ at x^* to get ${\sigma^2}^* = ...$ This is in fact the procedure to obtain the Laplace approximation of the Gamma distribution. Hence

$$\text{Gamma}(a, b) \approx \text{Normal}(x^*, {\sigma^2}^*) = \text{Normal}(..., ...).$$

For $\text{Gamma}(5, 2)$ plot the density function and compare it to the Normal approximation. Do you think the approximation performs well in this instance? When might you expect it to perform better/worse?

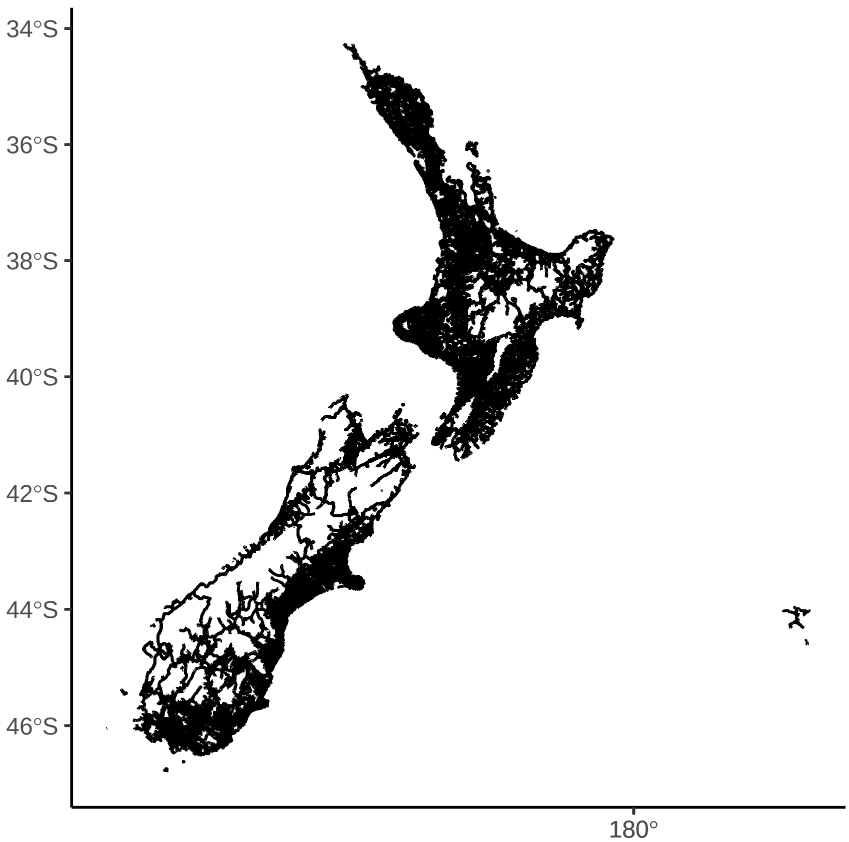
Answers

Q1

```
library(sf)
library(tidyverse)
## read in from unzipped folder in current working directory
path <- "lds-nz-roads-addressing-SHP/nz-roads-addressing.shp"
network <- st_read(path)
```

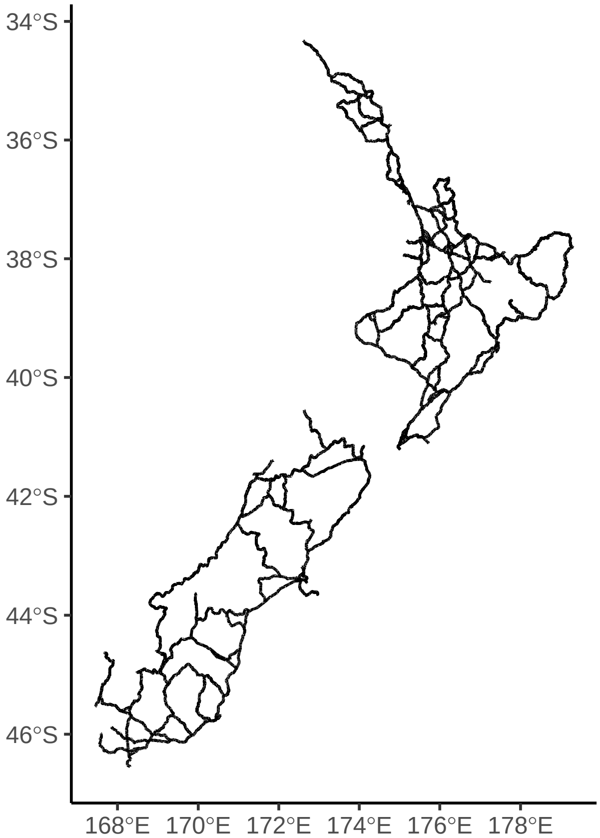
```
## Reading layer `nz-roads-addressing' from data source
##   `/home/cjon911/Desktop/lds-nz-roads-addressing-SHP/nz-roads-addressing.shp'
##   using driver `ESRI Shapefile'
## Simple feature collection with 79424 features and 17 fields
## Geometry type: MULTILINESTRING
## Dimension:     XY
## Bounding box:  xmin: 1114407 ymin: 4793178 xmax: 2467495 ymax: 6190214
## proj4string:   +proj=tmerc +lat_0=0 +lon_0=173 +k=0.9996 +x_0=1600000 +y_0=10000000 +ellps=GRS80 +uni
```

```
## complete
network %>%
  ggplot() +
  geom_sf() +
  theme_classic()
```

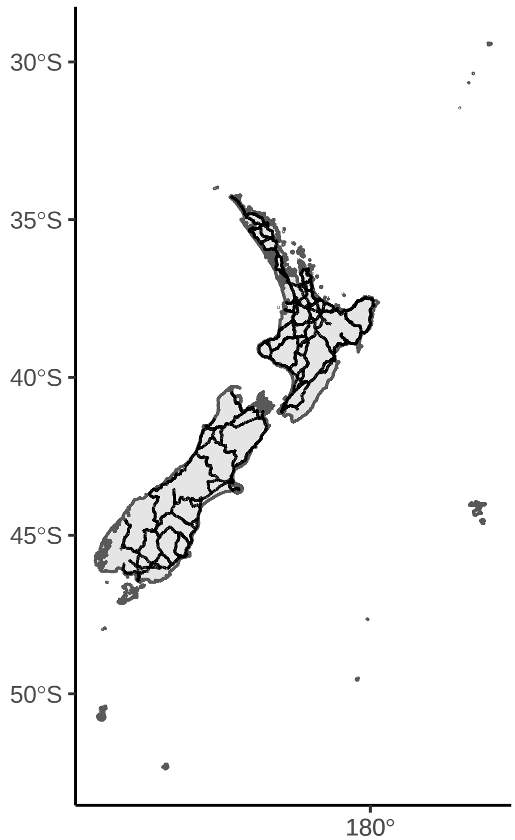



```
## State Highway
network %>%
  filter(route_name == "State Highway") %>%
```

```r
  ggplot() +
  geom_sf() +
  theme_classic()
```




```r
## NZ outline
nz <- raster::getData("GADM", country = "NZ", level = 0)
nz <- st_as_sf(nz) %>%
  st_transform(., st_crs(network))
network %>%
  filter(route_name == "State Highway") %>%
  ggplot() +
  geom_sf(data = nz) +
  geom_sf() +
  theme_classic()
```



Q2

We differentiate $L(\lambda\,; x_1, \ldots, x_n)$ and set to 0 to find the MLE:

$$\begin{aligned} 0 &= \frac{\delta}{\delta\lambda} L(\lambda\,; x_1, \ldots, x_n) \\ &= K\left(-ne^{-n\lambda}\,\lambda^{n\overline{x}} + n\overline{x}e^{-n\lambda}\,\lambda^{n\overline{x}-1}\right) \\ &= Ke^{-n\lambda}\lambda^{n\overline{x}-1}\,(-n\lambda + n\overline{x}) \end{aligned}$$

$\rightarrow \lambda = \infty, \lambda = 0,$ or $\lambda = \overline{x}$.

If we know that $L(\lambda\,; x_1, \ldots, x_n)$ reaches a unique maximum in $0 < \lambda < \infty$ then the MLE is $\overline{x}$.

So the maximum likelihood estimator is

$$\hat{\lambda} = \overline{x} = \frac{x_1 + \ldots + x_n}{n}\;.$$

```r
## log likelihood for sum of poisson random variables
## where observations are the observed number of counts
## for some unkown parameter lambda
#' @param lambda unknown parameter lambda
#' @param obvs vector of observations
log_lik_poisson <- function(lambda, obvs){
  llh <- sum(dpois(obvs, lambda, log = TRUE))
  return(llh)
}
# optimise the function above to find the MLE of  unknown parameter lambda
optimise(log_lik_poisson, c(10,127) , obvs = x, maximum = TRUE)
```

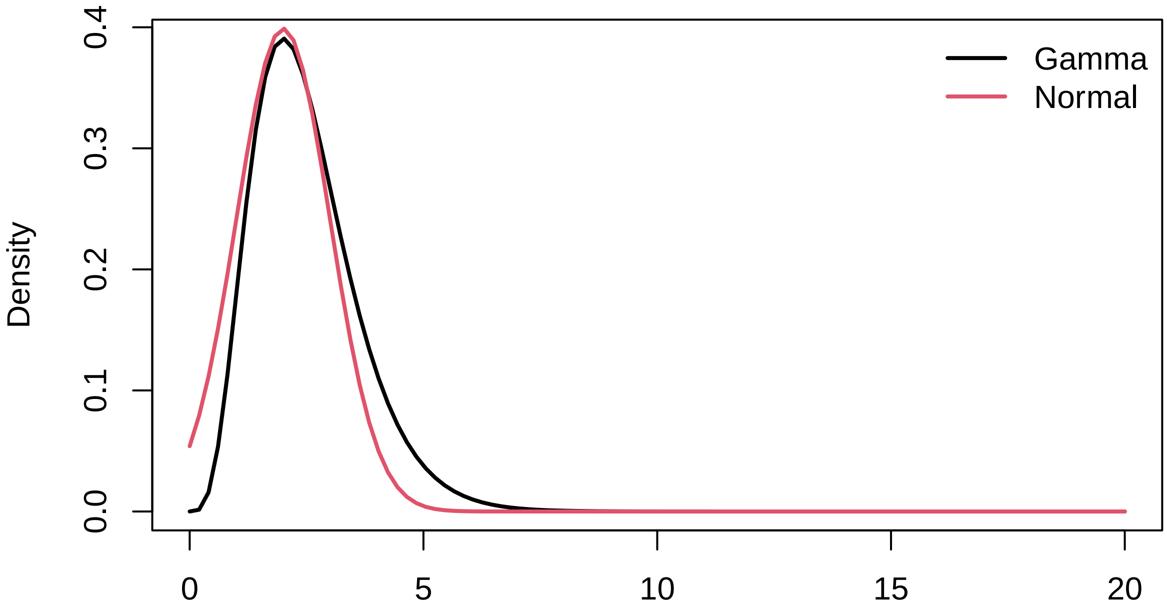
```
## $maximum
## [1] 80.90475
##
## $objective
## [1] -288.0552
```

Q3

Setting $\frac{d\log(f(x))}{dx} = 0$ we get $x^* = \frac{a-1}{b}$. Evaluating $-1/\frac{d^2\log(f(x))}{d^2x}\Big|_{x^*}$ we get $\sigma^{2^*} = \frac{a-1}{b^2}$. Therefore

$$\text{Gamma}(a, b) \approx \text{Normal}(x^*, \sigma^{2^*}) = \text{Normal}(\frac{a-1}{b}, \frac{a-1}{b^2}).$$

```r
a <- 5
b <- 2
x <- seq(0, 20, length.out = 100)
plot(x, dgamma(x, a, b), type = "l", xlab = "", ylab = "Density", lwd = 2)
lines(x, dnorm(x, (a-1)/b, (a-1)/b^2), col = 2, lwd = 2)
legend("topright", bty = "n", legend = c("Gamma", "Normal"), lty = 1, lwd = 2, col = 1:2)
```



Approximation works fairly well in this instance. Would imagine worse performance the less symmetrical the Gamma distribution we are attempting to approximate.