Solutions Lecture 10

Exercise 1

Question 1

The derivative is:

$$f'(x) = 3x^2 - 2$$

To understand where the function is increasing and where is decreasing we can study the sign of the first derivative:

$$3x^2 - 2 > 0$$
$$x > \sqrt{\frac{2}{3}}, x < -\sqrt{\frac{2}{3}}$$

The derivative is negative in $(-\sqrt{\frac{2}{3}}, \sqrt{\frac{2}{3}})$. Therefore, f(x) is increasing in $(-\infty, -\sqrt{\frac{2}{3}})$ and $(\sqrt{\frac{2}{3}}, \infty)$.

Question 2

There is no global minimum since $\lim_{x \to -\infty} f(x) = -\infty$. There is no global maximum since $\lim_{x \to \infty} f(x) = \infty$.

Question 3

To see if $x = \pm\sqrt{\frac{2}{3}}$ are a local minima or local maxima we can study the sign of the derivative. In Question 1 we have shown that the function increases until $-\sqrt{\frac{2}{3}}$ then decreases until $\sqrt{\frac{2}{3}}$ Therefore $-\sqrt{\frac{2}{3}}$ must be a local maximum. The function increases after $\sqrt{\frac{2}{3}}$ so this must be a local minimum.

Question 4

```r
f <- function(x){
  x^3 -2*x + 5
}
```

First we can check if $-\sqrt{\frac{2}{3}}$ is a local *maxima*. In order to do so we need to specify `maximum = TRUE` in the `optimize` arguments (the default is `maximum = FALSE`).

```r
optimize(f, c(-5, 0), maximum = T)
```

```
## $maximum
## [1] -0.8165118
##
## $objective
## [1] 6.088662
```

For $\sqrt{\frac{2}{3}}$:

```
optimize(f, c(0, 5))
```

```
## $minimum
## [1] 0.8165118
##
## $objective
## [1] 3.911338
```

Exercise 2

Question 1

We assume that $(x_1, x_2, ..., x_n)$ are i.i.d:

$$L(\lambda|x_1, x_2, ..., x_n) = \prod_{i=1}^{n} \frac{\lambda^{x_i} e^{-\lambda}}{x_i!}$$
$$L(\lambda|x_1, x_2, ..., x_n) = e^{-n\lambda} \frac{\lambda^{\sum_{i=1}^{n} x_i}}{\prod_{i=1}^{n} x_i!}$$

Question 2

$$\ell(\lambda|x_1, x_2, ..., x_n) = -n\lambda + \left(\sum_{i=1}^{n} x_i\right) \ln(\lambda) - \ln(\prod_{i=1}^{n} x_i!)$$

Question 3

As usual, we study the sign of the derivative:

$$\ell'(\lambda|x_1, x_2, ..., x_n) = -n + \frac{\left(\sum_{i=1}^{n} x_i\right)}{\lambda}$$
$$-n + \frac{\left(\sum_{i=1}^{n} x_i\right)}{\lambda} > 0$$
$$\lambda < \frac{\sum_{i=1}^{n} x_i}{n}$$

The function increases until $\frac{\sum_{i=1}^{n} x_i}{n}$ and then decreases, so it must be a global maximum. Therefore the maximum likelihood estimation is:

$$\lambda = \frac{\sum_{i=1}^{n} x_i}{n}$$

Question 4

```r
l <- function(lambda, x){
  -length(x)*lambda + sum(x)*log(lambda) - log(prod(factorial(x)))
}
```

Question 5 and 6

```r
x1 = c(9, 7, 7, 8, 10, 5, 8, 4, 3, 5, 7, 7, 9, 6)
set.seed(10)
x2 = rpois(300, lambda = 3)
```

We can use `optimize` function to answer this question. It can handle further arguments (`...`) passed to the function that is being optimized, in this case we can specify `x1` as `x` argument.

```r
# The upper limit is rather arbitrary.
# Notice maximum TRUE.

optimize(l, c(0, 10), x = x1, maximum = T)
```

```
## $maximum
## [1] 6.785712
##
## $objective
## [1] -30.23411
```

We can check this analytically:

```r
mean(x1)
```

```
## [1] 6.785714
```

The solutions are exactly the same.

For `x2`, using `optimize`:

```r
optimize(l, c(0, 10), x = x2, maximum = T)
```

```
## $maximum
## [1] 3.03
##
## $objective
## [1] -573.0289
```

Again, analytically:

```r
mean(x2)
```

```
## [1] 3.03
```

Exercise 3

Question 1

$$\ell(\alpha, \beta|x_1, x_2, ..., x_n) = \sum_{i=1}^{n} \alpha \ln(\beta) + (\alpha - 1)\ln(x_i) - \beta x_i - ln\big((\alpha - 1)!\big)$$

Question 2

See `?gamma`. `lgamma` returns the natural logarithm of the absolute value of $\Gamma(\alpha)$. Notice that we need to use the negative log-likelihood in order to make `optim` maximize.

```
lg <- function(param, x){
  alpha <- param[1]
  beta <- param[2]
  return(- sum(alpha * log(beta) + (alpha - 1)*log(x) - beta*x - lgamma(alpha)))
}
```

Or we can just use:

```
lg <- function(param, x){
  alpha <- param[1]
  beta <- param[2]
  return(- sum(dgamma(x, alpha, beta, log = T)))
}
```

```
x = iris$Petal.Width
```

Question 3

Nelder-Mead:

```
# Again, further arguments can be passed.
optim(c(1, 1), fn = lg, x = x)
```

```
## $par
## [1] 1.557958 1.299068
##
## $value
## [1] 169.4108
##
## $counts
## function gradient
##       59       NA
##
## $convergence
## [1] 0
##
## $message
## NULL
```

L-BFGS-B:

```
optim(c(1, 1), fn = lg, x = x, method = "L-BFGS-B",
      lower = c(0, 0))
```

```
## $par
## [1] 1.557824 1.298908
##
## $value
## [1] 169.4108
##
## $counts
## function gradient
##       10       10
##
## $convergence
## [1] 0
##
## $message
## [1] "CONVERGENCE: REL_REDUCTION_OF_F <= FACTR*EPSMCH"
```

The result is almost exactly the same in both methods.

Question 4

One way to do this by applying a log transformation to both α and β. So $\alpha = \exp(a)$ and $\beta = \exp(b)$.

```
lg <- function(param, x){
  alpha <- exp(param[1])
  beta <- exp(param[2])
  return(- sum(dgamma(x, alpha, beta, log = T)))
}
```

Then we use `optim` and transform the parameters to the original scale.

```
my_sol <- optim(c(2, 2), fn = lg, x = x)
exp(my_sol$par) # Transform to original scale!
```

```
## [1] 1.557949 1.299017
```

The solution is almost exactly the same as before.

Exercise 4

Question 1

```
x <- split(iris$Sepal.Length, iris$Species)
```

Question 2

We can use `sapply` or `lapply`. The latter is probably more neat.

```
res <- lapply(x, t.test)

res$setosa$conf.int
```

```
## [1] 4.905824 5.106176
## attr(,"conf.level")
```

```
## [1] 0.95
```

```
res$versicolor$conf.int
```

```
## [1] 5.789306 6.082694
## attr(,"conf.level")
## [1] 0.95
```

```
res$virginica$conf.int
```

```
## [1] 6.407285 6.768715
## attr(,"conf.level")
## [1] 0.95
```

Question 3

We just need to change the `mu` value at which we evaluate the mean and increase the confidence interval (`conf.level`) to make $\alpha = 0.01$.

```
t.test(x$setosa, mu = 5, conf.level = 0.99)
```

```
##
##  One Sample t-test
##
## data:  x$setosa
## t = 0.12036, df = 49, p-value = 0.9047
## alternative hypothesis: true mean is not equal to 5
## 99 percent confidence interval:
##  4.872406 5.139594
## sample estimates:
## mean of x
##     5.006
```

We fail to reject the null hypothesis for the "Setosa" type.

Question 4

The default `conf.level` is 0.95 and that suits us because we want $\alpha = 0.05$.

```
t.test(x$versicolor, mu = 5)
```

```
##
##  One Sample t-test
##
## data:  x$versicolor
## t = 12.822, df = 49, p-value < 2.2e-16
## alternative hypothesis: true mean is not equal to 5
## 95 percent confidence interval:
##  5.789306 6.082694
## sample estimates:
## mean of x
##     5.936
```

We reject the null hypothesis.

Question 5

```r
t.test(x$versicolor, x$virginica, conf.level = 0.99)
```

```
##
##  Welch Two Sample t-test
##
## data:  x$versicolor and x$virginica
## t = -5.6292, df = 94.025, p-value = 1.866e-07
## alternative hypothesis: true difference in means is not equal to 0
## 99 percent confidence interval:
##  -0.9565202 -0.3474798
## sample estimates:
## mean of x mean of y
##     5.936     6.588
```

The data provides enough evidence to reject the null.