Exercises (with Solutions) · Chapter 4

1. **Problem**

Let $(X_1, \dots, X_n)$ be a random sample with $X_i \sim N(\mu, \sigma^2)$ for all $i = 1, \dots, n$, where $\mu = 3$ and $\sigma = 1.5$. Consider different sample sizes of $n = \{2, 4, 6, \dots, 30\}$.

(a) Conduct the following Monte Carlo simulation:

1. Simulate a realization from the random sample $(X_1, \dots, X_n)$.
2. Compute the corresponding realizations of the biased maximum likelihood estimator $s^2_{ml,n}$ and the unbiased estimator $s^2_{ml,n}$ for the variance $\sigma^2 = 1.5^2$:

$$s^2_{ml,n} = \frac{1}{n} \sum_{i=1}^{n} (X_i - \bar{X}_n)^2$$
$$s^2_{ub,n} = \frac{1}{n-1} \sum_{i=1}^{n} (X_i - \bar{X}_n)^2$$

where $\bar{X}_n = n^{-1} \sum_{i=1}^{n} X_i$.

Repeat Steps 1-2 $B = 10,000$ times and save all estimates $s^2_{ml,n,(1)}, \dots, s^2_{ml,n,(B)}$ and $s^2_{ub,n,(1)}, \dots, s^2_{ub,n,(B)}$. Then approximate the Mean Squared Error (MSE), the squared bias (Bias^2), and the variance (Var) of $s_{ml,n}$ by

$$\mathrm{MSE}(s^2_{ml,n}) \approx \frac{1}{B} \sum_{b=1}^{R} (s^2_{ml,n,(b)} - \sigma)^2$$
$$\mathrm{Bias}^2(s^2_{ml,n}) \approx \left(\left(\frac{1}{B} \sum_{r=1}^{B} s^2_{ml,n,(b)} \right) - \sigma \right)^2$$
$$\mathrm{Var}(s^2_{ml,n}) \approx \frac{1}{B} \sum_{b=1}^{B} \left(s^2_{ml,n,(b)} - \left(\frac{1}{B} \sum_{b=1}^{B} s^2_{ml,n,(b)} \right) \right)^2$$

where $E(s^2_{ml,n}) \approx B^{-1} \sum_{b=1}^{B} s^2_{ml,n,(b)}$, and of $s_{ub,n}$ by

$$\mathrm{MSE}(s^2_{ub,n}) \approx \frac{1}{B} \sum_{b=1}^{R} (s^2_{ub,n,(b)} - \sigma)^2$$
$$\mathrm{Bias}^2(s^2_{ub,n}) \approx \left(\left(\frac{1}{B} \sum_{r=1}^{B} s^2_{ub,n,(b)} \right) - \sigma \right)^2$$
$$\mathrm{Var}(s^2_{ub,n}) \approx \frac{1}{B} \sum_{b=1}^{B} \left(s^2_{ub,n,(b)} - \left(\frac{1}{B} \sum_{b=1}^{B} s^2_{ub,n,(b)} \right) \right)^2$$

where $E(s_{ub,n}) \approx B^{-1} \sum_{b=1}^{B} s^2_{ub,n,(b)}$.

Note: The *Law of Large Numbers* implies that the approximations become arbitrarily precise as $B \to \infty$.

(b) Plot your results (y-axis: MSE, Bias^2, and Var; x-axis: n) for both estimators $s_{ml,n}$ and $s_{ub,n}$.

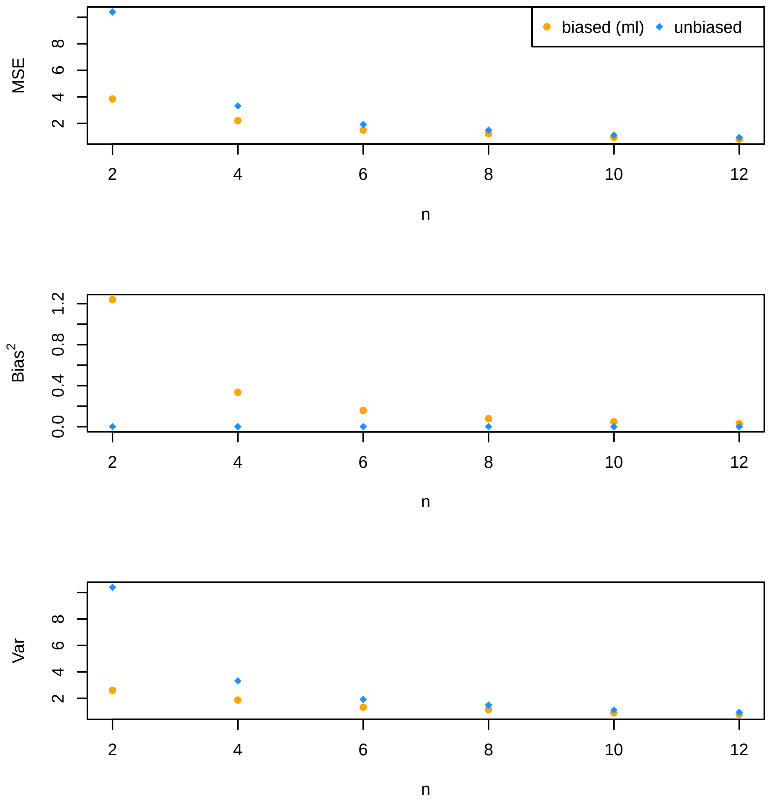
Solution

(a) The following code implements the Monte Carlo Simulation:

```
> set.seed(007)
> mu    <- 3
> sigma <- 1.5
> n_seq <- seq(from = 2, to = 12, by = 2)
> B     <- 10000
> ## Container for the biased maximum likelihood estimator s_ml:
> s2_ml_matrix   <- matrix(nrow = B, ncol = length(n_seq))
> ## Container for the unbiased estimator s_ub:
> s2_ub_matrix   <- s2_ml_matrix
> for (b in 1:B) {
+
+   for (i in 1:length(n_seq)) {
+
+   Xsample           <- rnorm(n_seq[i], mu, sigma)
+
+   s2_ml_matrix[b, i] <- sum(( Xsample - mean(Xsample))^2 )/(n_seq[i]    )
+   s2_ub_matrix[b, i] <- sum(( Xsample - mean(Xsample))^2 )/(n_seq[i] - 1)
+   }
+ }
> mse_fun               <- function(x) {  mean( (x - sigma^2)^2 ) }
> bias_squared_fun      <- function(x) { (mean(x) - sigma^2 )^2}
> variance_fun          <- function(x) {  var(x) }
> mse_biased            <- apply(s2_ml_matrix, 2, mse_fun)
> bias_squared_biased   <- apply(s2_ml_matrix, 2, bias_squared_fun)
> variance_biased       <- apply(s2_ml_matrix, 2, variance_fun)
> mse_unbiased          <- apply(s2_ub_matrix, 2, mse_fun)
> bias_squared_unbiased <- apply(s2_ub_matrix, 2, bias_squared_fun)
> variance_unbiased     <- apply(s2_ub_matrix, 2, variance_fun)
```

(b) The following codes plot the results:

```
> par(mfrow = c(3, 1), mai = c(0.75, 0.75, 0.25, 0.75), oma = c(0, 0, 2, 0))
> ## MSE
> plot(x = n_seq, y = mse_biased, col = "orange", pch = 16,  xaxt = "n",
+      xlab = "n", ylab = "MSE",
+      ylim = range(c(mse_biased, mse_unbiased)))
> axis(1, at = n_seq)
> points(x = n_seq, y = mse_unbiased, col = "dodgerblue", pch = 18)
> legend("topright", c("biased (ml)", "unbiased"),
+        col = c("orange", "dodgerblue"), pch = c(16, 18), horiz = TRUE)
> ## Bias^2
> plot(x = n_seq, y = bias_squared_biased, col = "orange", pch = 16,
+      xaxt = "n", xlab = "n", ylab = expression(Bias^2),
+      ylim = range(c(bias_squared_biased, bias_squared_unbiased)))
> axis(1, at = n_seq)
> points(x = n_seq, y = bias_squared_unbiased, col = "dodgerblue", pch = 18)
> ## Var
> plot(x = n_seq, y = variance_biased, col = "orange", pch = 16,
+      xaxt = "n", xlab = "n", ylab = "Var",
+      ylim = range(c(variance_biased, variance_unbiased)))
> axis(1, at = n_seq)
> points(x = n_seq, y = variance_unbiased, col = "dodgerblue", pch = 18)
```


biased (ml)
unbiased
MSE
Bias^2
Var
n