Advanced Quantitative Methods
Homework 2: Statistical estimators
answers

Johan A. Elkink
jos.elkink@ucd.ie

Due February 2, 2011, 9am

Please submit by email in PDF format. Add R code in a separate R file.

1. This exercise is meant as a revision of regression in R. You will need the *states.csv* dataset, which you can open with:

```
> states <- read.csv("states.csv")
> states <- within(states, {
+     turnout <- turnout/(population * 1000) * 100
+     young <- young/(population * 1000) * 100
+     farmers <- farmers/(population * 1000) * 100
+ })
```

The second line converts the some variables into percentages instead of just numbers. The variables we will use are:

turnout	Michael McDonald's estimated level of turnout (updated Nov 15)
education	direct expenditures for education (million dollars) in 2002
young	population under 18 years (thousands) in 2004
farmers	number of farm operators (thousands) in 2002

Perform the following regression: $turnout_i = \beta_0 + \beta_1 young_i + \beta_2 education_i + \beta_3 farmers_i + \varepsilon_i$

(a) (15%) Present a regression table properly formatted as for a publication.

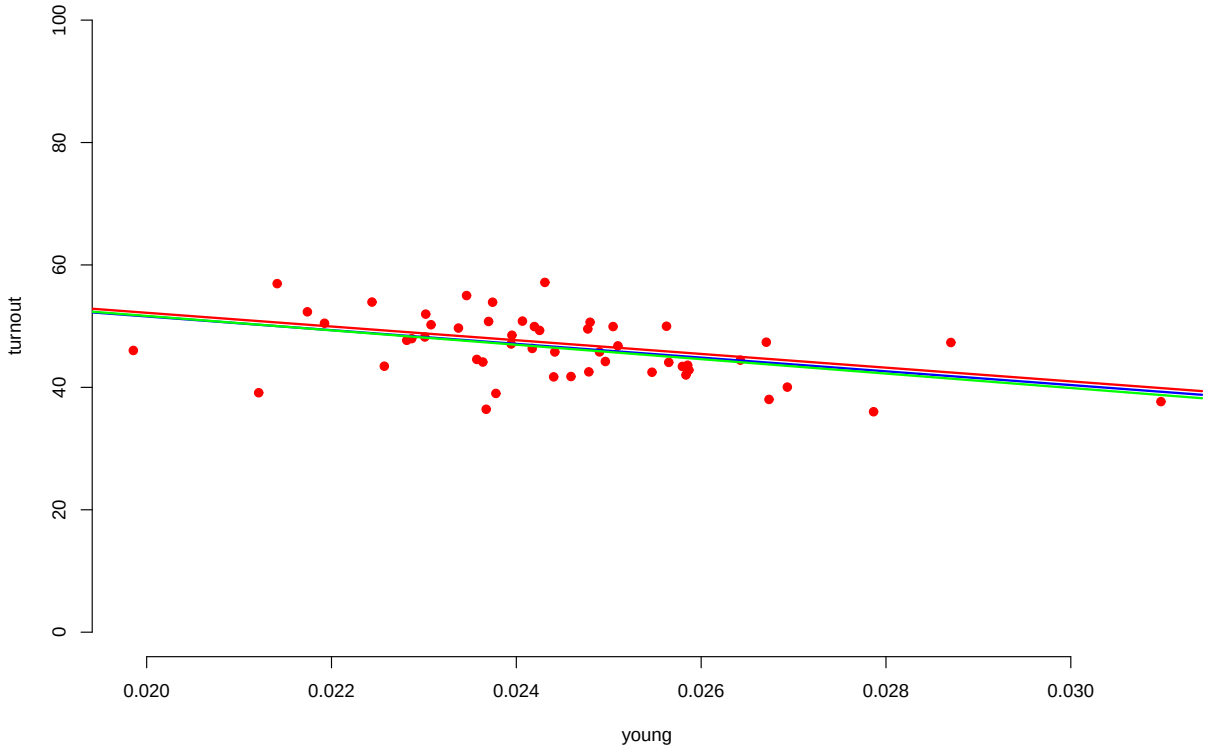
```
> m <- lm(turnout ~ young + education + farmers, data = states)
> library(xtable)
> print(xtable(m), floating = FALSE)
```

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	74.6017	9.0774	8.22	0.0000
young	-1120.8074	367.8741	-3.05	0.0038
education	-0.0004	0.0002	-1.58	0.1206
farmers	317.6016	453.2179	0.70	0.4870

(b) (20%) Interpret the regression coefficients and standard errors - what does it tell you about turnout in U.S. states?

(c) (20%) Plot, based on the previous regression, *turnout* as a function of *young*, including the estimated regression line.

```
> postscript("turnout.eps")
> plot(turnout ~ young, pch = 19, col = "red", bty = "n", ylim = c(0,
+     100), data = states)
> b <- coef(m)
> abline(b["(Intercept)"] + b["education"] * median(states$education,
+     na.rm = TRUE) + b["farmers"] * median(states$farmers, na.rm = TRUE),
+     b["young"], lwd = 2, col = "blue")
> abline(m, lwd = 2, col = "red")
> abline(lm(turnout ~ young, data = states), lwd = 2, col = "green")
> dev.off()
null device
          1
```



2. Write a small Monte Carlo application (25%) and for sample sizes $n = \{5, 10, 50\}$:

   (a) (10%) present a table with the bias and standard error for each value of n of $\hat{\sigma}^2 = \frac{1}{n}\sum_i^n (x_i - \hat{x})^2$ as an estimator of the population variance;

(b) (10%) present a table with the bias and standard error for each value of n of $\hat{\sigma} = \sqrt{\frac{1}{n-1}\sum_i^n (x_i - \hat{x})^2}$ as an estimator of the population standard deviation.

You will need a number of iterations (say, 1000) for each of the three values of n and you can then calculate the mean of the bias across the 1000 iterations and present this in the table. So fill in:

n	$bias_{\hat{\sigma}^2}$	$se_{\hat{\sigma}^2}$	$bias_{\hat{\sigma}}$	$se_{\hat{\sigma}}$
5	?	?	?	?
10	?	?	?	?
50	?	?	?	?

Note that the var(x) function in R returns the unbiased estimator of the variance, so it returns $\frac{1}{n-1}\sum_i^n (x_i - \hat{x})^2$. If you want the biased estimator, that is, the sample variance, you can use:

```
n <- length(x)
sample.var <- ((n-1)/n)*var(x)
```

```
> sigma <- 3
> M <- 1000
> for (n in c(5, 10, 50)) {
+     s2.hat <- rep(NA, M)
+     s.hat <- rep(NA, M)
+     for (i in 1:M) {
+         x <- rnorm(n, 0, sigma)
+         s2.hat[i] <- 1/n * sum((x - mean(x))^2)
+         s.hat[i] <- sqrt(1/(n - 1) * sum((x - mean(x))^2))
+     }
+     cat(sprintf("N = %2d|bias s2 = %4.2f|se s2 = %4.2f|bias s = %4.2f|se s =
+         n, mean(s2.hat - sigma^2), sd(s2.hat - sigma^2), mean(s.hat -
+             sigma), sd(s.hat - sigma)))
+ }
```

```
N =  5|bias s2 = -2.02|se s2 = 4.86|bias s = -0.23|se s = 1.02
N = 10|bias s2 = -0.84|se s2 = 3.98|bias s = -0.08|se s = 0.72
N = 50|bias s2 = -0.13|se s2 = 1.78|bias s = -0.01|se s = 0.30
```