

# Giving your data the bootstrap

Daniel B. Wright first explains how to compare means with a bootstrap, and then Andy P. Field puts the method into use in regression analysis

I'm selling my house at the moment and it has got me thinking about lots of things. The estate agents all wanted about 2 per cent commission and I thought, 'How much money do these people make?' One of things that it depends on is the cost of the houses that they sell. Let's compare two hypothetical estate agents: Sarah Beeny Fan Club (SBFC) and the We Love Sarah Agency (WLSA). Even in these days of negative equity and post-subprime lending they will sell hundreds of houses, so I just took a sample of 20 houses from each of their sold prices (in £1000s):

SBFC: 200, 210, 240, 190, 205, 265, 1660, 1480, 240, 250, 255, 1450, 240, 235, 250, 180, 200, 1720, 210, 320  
 WLSA: 285, 310, 255, 260, 265, 260, 270, 300, 395, 265, 255, 260, 240, 380, 380, 280, 390, 370, 290, 290

The mean for SBFC is £500,000 and for WLSA is £300,000, so this looks like SBFC makes more money. But I took a sample of only 20 houses from each place. If I wanted to test whether in their respective populations (i.e. all the houses they sell) the mean is the same, then I need to do some kind of statistical test. This is where life gets tricky.

One approach would be to run a standard *t* test on these data, but there is a problem with this – the standard deviations are very different in these groups (£556,000 for SBFC and £52,000 for WLSA) and SBFC have several outliers. Some people would rank these and run a Wilcoxon rank sum test. This results in a significant effect ( $W = 112, p = .02$ ), but the direction of the result suggests WLSA have the higher values.

But if SBFC target houses in the rich areas, they will get that moola whether these are outliers or not. Ranking is an extreme transformation. I like transforming data (really), but most transformations will change the units being expressed. If we take the square roots of all the values, the means of the two agencies are nearly equivalent. But

the agencies don't care about the square root of the amount sold, because they get 2 per cent of the actual amount sold. They want to know the mean. Thus, the Wilcoxon and other transformations of the data are also wrong.

This is a case where computational power should be used instead of mathematical wizardry. In *An Introduction to the Bootstrap*, Efron and Tibshirani (1993) describe the bootstrap method as a way to bypass hard mathematics. Pretend you write all 20 prices for SBFC onto ping-pong balls and place them into a bowl. You choose one, write the number down, and return the ball to the bowl. You repeat this until you have 20 numbers written down. You do the same with WLSA's prices. This is your first bootstrap sample. Then you subtract the means (SBFC – WLSA) and store this value. If you repeat this, say, 2000 times, you get 2000 difference scores. This sounds time consuming, but in fact it takes only a couple of lines in the freeware R using Canty and Ripley's boot package (<http://cran.r-project.org/web/packages/boot/>). Efron and Tibshirani describe several ways to construct 95 per cent confidence intervals with bootstrapping. The method they prefer, called the bias corrected and accelerated, or BCa, method produces an interval from 3 to 500 for the estate agent data. Because there is random element the precise boundaries will vary, but this shows the 95 per cent confidence

Table 1: Regression summary statistics when using normal OLS regression, and robust regression procedures

Parameter	b	SE	95% Confidence Interval		p
			Lower	Upper	
<i>OLS (SPSS 16):</i>					
Intercept	2719.67	888.89	544.64	4894.70	.022
Number of Pubs	4.56	1.97	-0.26	9.39	.060
<i>Bootstrap CI:</i>					
Intercept			494.00	5614.00	
Number of Pubs			0.65	15.35	
<i>Theil-Sen estimates with bootstrap CI:</i>					
Intercept	1771.51	1373.33	-251.87	4771.30	.124
Number of Pubs	10.16	4.36	2.38	17.24	< .001

interval just about touches zero (or  $p \approx .05$ ), which depending on your philosophy about *p* values suggests that the mean house prices are higher at SBFC than at WLSA.

To run the bootstrap in R you first enter the data:

```
SBFC <- c(200, 210, 240, 190, 205, 265, 1660, 1480, 240, 250, 255, 1450, 240, 235, 250, 180, 200, 1720, 210, 320)
WLSA <- c(285, 310, 255, 260, 265, 260, 270, 300, 395, 265, 255, 260, 240, 380, 380, 280, 390, 370, 290, 290)
```

Then combine these variables into a data frame (I've called it x):

```
x <- as.data.frame(cbind(SBFC,WLSA))
```

Then load the bootstrap library:

```
set.seed(1)
library(boot)
```

Next, we define a function for the difference between two means:

```
meandiff <- function(x,i)
md <- mean(x$SBFC[i])-mean(x$WLSA[i])
```

Finally, we run the bootstrap:

```
mdboot <- boot(x,meandiff, R=2000)
boot.ci(mdboot, type="bca")
```

and find the confidence interval (from 3 to 500).

Traditional procedures, like the *t* and Wilcoxon tests, and transforming the data, are not appropriate for this example. The bootstrap offers a suitable alternative for this and many other circumstances.

**Daniel B. Wright** is Professor of Psychology at Florida International University [dwright@fiu.edu](mailto:dwright@fiu.edu)

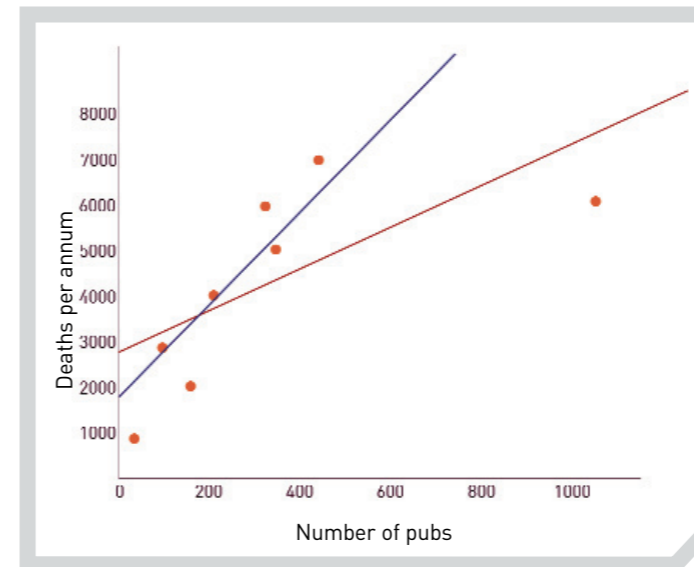


Figure 1: Scatterplot of the number of pubs and mortality rate in eight boroughs in London. The red line shows the regression line using parameters estimated using normal OLS regression, the blue line uses parameters estimated using robust methods.

## Bootstrap for regression analysis

Like many academics I have a perennial interest in drinking alcohol to dull the pain of a stressful life. However, trips to the pub invariably get me thinking about whether or not I am facilitating my premature demise. In my textbook *Discovering Statistics Using SPSS*, I describe some example data that show the relationship between deaths and the number of pubs in various boroughs of London. In the book I use these data as an example of how an extreme case (a district of London with a disproportionate number of pubs) can influence a regression line without being an outlier (a case that the regression model fits badly). In other words, the regression model fits that data point well, but the data point exerts a huge influence on the regression model.

Figure 1 shows an adaptation of these data; the red line shows the regression line that you would get using a normal ordinary least squared (OLS) regression using SPSS 16. It is clear that this line does not represent the general trend very well and the regression coefficient (*b*) for this model is not significant too (see Table 1). This has happened because the model has been affected by the borough in which there were over a thousand pubs. However, this data point would not show up as an outlier because the model fits it fairly well: its standardised residual is only -0.9, which would not give anyone cause for concern.

There are other situations where a regression analysis might be biased, for

often do not correct the problems and bring with them their own unique set of issues (as mentioned above). Using the freeware R, robust regression parameters can be estimated using a variety of functions written by Rand Wilcox and described in his excellent book on robust methods. To do a robust regression in R, we first instruct R to access Wilcox's functions, which are in the files on his web page. We source these two files directly from his web page in R by typing:

```
source("http://tinyurl.com/allfun")
```

Next, we need to enter the data. For this example we create two variables ('mortality' and 'pubs') and for each we specify the values for each of the eight London boroughs:

```
pubs<-c(35, 99, 163, 214, 326, 344, 454, 1054)
mortality<-c(957, 2990, 2017, 4025, 5997, 5007, 7012, 6022)
```

First, we could compute the confidence interval for the regression parameters using a bootstrap (because we do not have the mathematics available to estimate it directly when our data are distributed weirdly). To do this in R type:

```
library(boot)
x<-as.data.frame(cbind(pubs,mortality))
set.seed(1)
regcoef1<-function(x,i) md<-
lm(x$mortality[i]-x$pubs[i])$coef[1]
set.seed(1)
```

example, when the variables are unusually distributed. A common question I get asked by people facing this unpleasant situation is '...but what can I do, there is not a non-parametric version of regression?' The answer is to use robust methods such as the bootstrap.

Of course, you can transform your data and plough ahead in SPSS, but these transformations

```
regcoef2<-function(x,i) md<-
lm(x$mortality[i]-x$pubs[i])$coef[2]
regcoefboot1<-boot(x,regcoef1,R=2000)
regcoefboot2<-boot(x,regcoef2,R=2000)
boot.ci(regcoefboot1, type="bca")
boot.ci(regcoefboot2, type="bca")
```

The resulting 95 per cent confidence intervals are shown in Table 1. Note that the OLS confidence interval for the number of pubs crosses zero (-0.26, 9.39) indicating a non-significant *b*, but the bootstrapped confidence interval does not cross zero (0.65, 15.35) indicating that the relationship between the number of pubs and the number of deaths is significant.

However, these bootstrapped confidence intervals do not give us an estimate of the regression parameters. To get a robust estimate of the regression coefficients (*bs*), we can use Wilcox's function *tsreg*, which estimates the *bs* using the Theil-Sen estimator. This estimator is the value *b* that makes Kendall's statistic, between  $Y_i - bX_i$  and  $X_i$ , approximately zero and it has many desirable properties. To compute this estimate of *b*, type:

```
tsreg(pubs,mortality)
```

The first variable in parenthesis should be your predictor (*x*) and the second one the outcome (*y*). To compute a confidence interval for Theil-Sen estimated *bs* we need to again use a bootstrap because we do not have the mathematics to compute it directly. To do this in R we use Wilcox's function *regci* by typing:

```
regci(pubs,mortality)
```

The results of these analyses are shown in Table 1 along with the normal OLS results from SPSS 16. The dotted line in Figure 1 shows the regression line based on these robust estimates. The main thing to note is that the robust regression model reflects the trend in the data much better than the OLS model. The confidence interval for the number of pubs no longer crosses zero, and the slope is now highly significant (as it should be given the near perfect linear trend between the number of pubs and mortality rates in the first seven London boroughs).

I hope to have shown with this simple example that the bootstrap (and robust methods in general) can be used to overcome problems in data sets that researchers commonly face when needing to conduct regression analyses.

**Andy P. Field** is a Reader in Psychology at the University of Sussex [andyf@sussex.ac.uk](mailto:andyf@sussex.ac.uk)