Chapter 8

The Bootstrap

Statistical science is the science of learning from experience. Efron and Tibshirani (1993) say "Most people are not natural-born statisticians. Left to our own devices we are not very good at picking out patterns from a sea of noisy data. To put it another way, we are all too good at picking out non existing patterns that happen to suit our purposes."

Suppose we find ourselves in the following common data-analytic situation: a random sample $\mathbf{x} = (x_1, \ldots, x_n)$ from an unknown probability distribution F has been observed and we wish to estimate a parameter of interest $\theta = t(F)$ on the basis of $\mathbf{x}$. For this purpose, we calculate an estimate $\hat{\theta} = s(\mathbf{x})$ from $\mathbf{x}$.

A common estimate is the *plug-in* estimate $t(\hat{F})$ where $\hat{F}$ is the empirical distri-

bution defined by

$$F(x) = \frac{\text{number of values in } \mathbf{x} \text{ equal to } x}{n}$$

Can you think of a plug-in estimate that is commonly used?

The bootstrap was introduced by Efron (1979) as a computer based method to estimate the standard deviation of $\hat{\theta}$.

What are the advantages:

- It is completely automatic
- Requires no theoretical calculations
- Not based on asymptotic results
- Available no matter how complicated the estimator $\hat{\theta}$ is.

A bootstrap sample is defined to be a random sample of size n drawn from $\hat{F}$, say $\mathbf{x}^* = (x_1^*, \ldots, x_n^*)$.

For each bootstrap sample $\mathbf{x}^*$ there is a bootstrap replicate of $\hat{\theta}$,

$$\hat{\theta}^* = s(\mathbf{x}^*).$$

The bootstrap estimate of $\mathbf{se}_F(\hat{\theta})$ is defined by

$$\mathbf{se}_{\hat{F}}(\hat{\theta}^*). \tag{8.1}$$

This is called the *ideal bootstrap estimate* of the standard error of $s(\mathbf{x})$.

Notice that for the case where θ is the expected value or mean of $\mathbf{x}_1$ we have

$$\mathbf{se}_{\hat{F}}(\bar{x}^*) = \mathbf{se}_{\hat{F}}(x_1^*)/\sqrt{n} = \sqrt{n^{-1}\sum_{i=1}^{n}(x_i - \hat{x})^2}/\sqrt{n}$$

and the ideal bootstrap estimate is the estimate we are used to. However, for any other estimator other than the mean obtaining (8.1) there is no neat formula that enables us to compute a numerical value in practice.

The bootstrap algorithm is a computational way of obtaining a good approximation to the numerical value of (8.1).

8.1 The bootstrap algorithm

The bootstrap algorithm works by drawing many independent bootstrap samples, evaluating the corresponding bootstrap replications, and estimating the standard error of $\hat{\theta}$ by the empirical standard error, denoted by $\hat{\mathbf{se}}_B$, where B is the number of bootstrap samples used.

1. Select B independent bootstrap samples $\mathbf{x}_1^*, \ldots, \mathbf{x}_B^*$, each consisting of n data values drawing with replacement from $\mathbf{x}$.
2. Evaluate the bootstrap replication corresponding to each bootstrap sample
$$\hat{\theta}^*(b) = s(\mathbf{x}_b^*), b = 1, \ldots, B$$

3. Estimate the standard error $\mathbf{se}_F(\hat{\theta})$ by the sample standard error of the B replicates

$$\hat{\mathbf{se}}_B = \left[\frac{1}{B-1} \sum_{b=1}^{B} \{\hat{\theta}^*(b) - \hat{\theta}^*(\cdot)\}^2 \right]$$

with

$$\hat{\theta}^*(\cdot) = B^{-1} \sum_{b=1}^{B} \hat{\theta}^*(b)$$

The limit of $\hat{\mathbf{se}}_B$ as B goes to infinity is the ideal bootstrap estimate of (8.1). But how close is (8.1) to $\mathbf{se}_F(\hat{\theta})$? See Efron and Tibshirani (1993) for more details.

8.2 Example: Curve fitting

In this example we will be estimating regression functions in two ways, by a standard least-squares line and by loess.

A total of 164 mean took part in an experiment to see if the drug cholostyramine lowered blood cholesterol levels. The men were supposed to take six packets of cholostyramine per day, but many of them actually took much less. Figure 8.1 shows compliance plotted against percentage of the intended dose actually taken. We also show a fitted line and a loess fit (using span=2/3). Notice the curves similar from 0 to 60, a little different from 60 to 80 and quite different from 80 to 100.

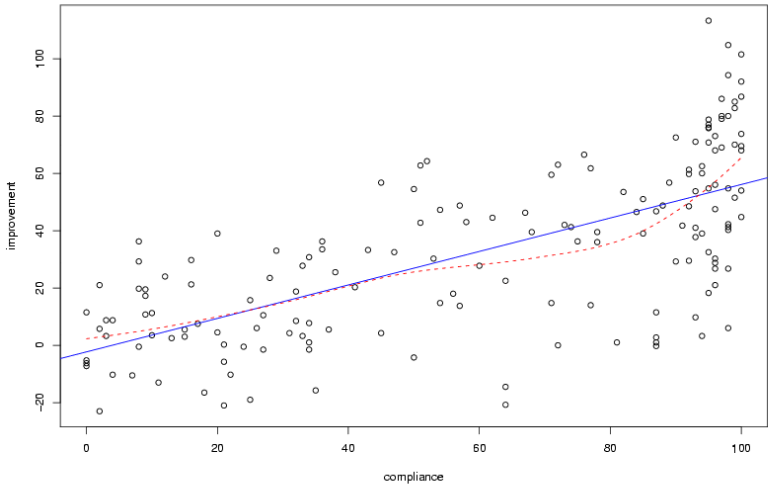


Figure 8.1: Estimated regression curves of Improvement on Compliance.

Assume the points a regression model

$$y_i = f(x_i) + \varepsilon_i, i = 1, \ldots, n$$

with the ε_i IID.

Say we are interested in the difference in rate of change of $f(x)$ in the 60–80 and 80–100 sections. We could define as the parameter to describe this. How can we do this?

Notice that finding a standard error for this estimate is not straight-forward. We can use the bootstrap.

Table 8.1: Estimates and bootstrap standard errors of $f(60), f(80)$, and $f(100)$.

	$\hat{f}_{\text{line}}(60)$	$\hat{f}_{\text{line}}(80)$	$\hat{f}_{\text{line}}(100)$	$\hat{f}_{\text{loess}}(60)$	$\hat{f}_{\text{loess}}(80)$	$\hat{f}_{\text{loess}}(100)$
value:	33	44	56	28	35	66
$\hat{\text{se}}_{50}$:	2	2	3	5	4	4

As seen in Figure 8.2. Even when there is no parameter of interest, the bootstrap estimates of f give us an idea of what a confidence set is for the nonparametric estimates. We will see more of this in Chapter 7 and 8.

8.3 Confidence "intervals" for linear smoothers

It is easy to show that the variance-covariance matrix of the vector of fitted values $\hat{\mathbf{f}} = \mathbf{S}\mathbf{y}$ is

$$\mathbf{cov}(\hat{\mathbf{f}}) = \mathbf{S}\mathbf{S}'\sigma^2$$

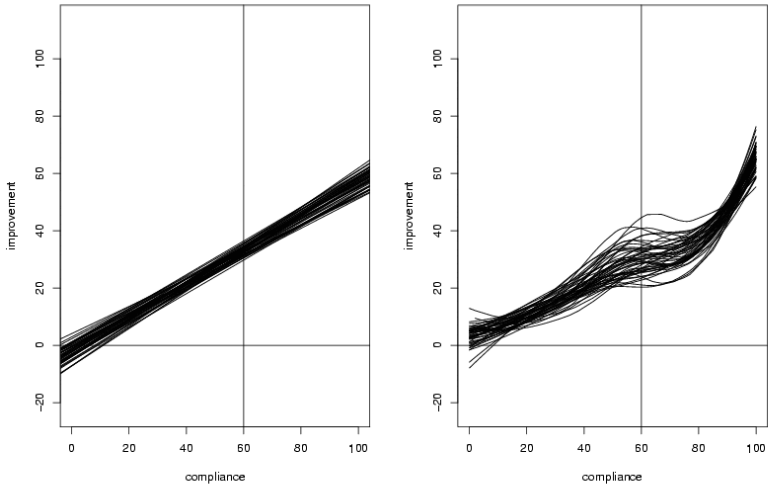


Figure 8.2: 50 bootstrap curves for each estimation technique.

and given an estimate of σ^2 this can be used to give point-wise standard errors, mainly by looking at $\text{diag}(\mathbf{S}\mathbf{S}')\sigma^2$.

Can we construct confidence intervals? What do we need?

First of all we need to know the distribution (at least approximately) of $\hat{\mathbf{f}}$. If the errors are normal we know that $\mathbf{f}$ is normally distributed. Why?

In the normal case, what are the confidence intervals for?

Remember that our estimates are usually biased, $\text{E}(\hat{\mathbf{f}}) = \mathbf{S}\mathbf{f} \neq \mathbf{f}$. If our null hypothesis is $\mathbf{S}\mathbf{f} = \mathbf{f}$ (in the case of splines this is equivalent to assuming $f \in \mathcal{G}$) then our confidence intervals are for $\mathbf{f}$ otherwise it is much more convenient to compute them for $\mathbf{S}\mathbf{f}$. We will start using the notation $\tilde{\mathbf{f}} = \mathbf{S}\mathbf{f}$. We can think of $\tilde{\mathbf{f}}$ as the best possible approximation to "the truth" $\mathbf{f}$ when using the $\mathbf{S}$ as a smoother.

To see how point-wise estimates can be useful, notice that we can get an idea of how variable $\hat{\mathbf{f}}(x_0)$ is. However, it isn't very helpful when we want to see how variable $\hat{\mathbf{f}}$ is as a whole.

What if we want to know if a certain function, say a line, is in our "confidence interval"? Point-wise intervals don't really help us with this.

8.4 Global confidence bands

Remember that $\hat{\mathbf{f}} \in \mathbb{R}^n$. This means that talking about confidence intervals doesn't make much sense. We need to consider confidence sets.

For example if the errors are normal we know that

$$\chi(\tilde{\mathbf{f}}) = (\hat{\mathbf{f}} - \tilde{\mathbf{f}})'(\mathbf{S}\mathbf{S}'\sigma^2)^{-1}(\hat{\mathbf{f}} - \tilde{\mathbf{f}})$$

is χ^2_n distributed. This permits us to construct confidence sets (which you can think of as random n-dimensional balls) for $\tilde{\mathbf{f}}$ of probability α

$$C_\alpha = \{\mathbf{g} \in \mathbb{R}^b; \chi(\mathbf{g}) \leq \chi_{1-\alpha}\} = \{\mathbf{g} \in \mathbb{R}^b; (\hat{\mathbf{f}} - \mathbf{g})'(\mathbf{S}\mathbf{S}'\sigma^2)^{-1}(\hat{\mathbf{f}} - \mathbf{g}) \leq \chi_{1-\alpha}\}.$$

Notice that the probability that the random ball doesn't fall on the approximate truth $\tilde{\mathbf{f}}$ is α:

$$\Pr(\tilde{\mathbf{f}} \notin C_\alpha) = \Pr\left[(\hat{\mathbf{f}} - \tilde{\mathbf{f}})'(\mathbf{S}\mathbf{S}'\sigma^2)^{-1}(\hat{\mathbf{f}} - \tilde{\mathbf{f}}) > \chi_{1-\alpha}\right] = \alpha.$$

This is only the case if we know σ^2.

Usually we construct an estimate

$$\hat{\sigma}^2 = (\mathbf{y} - \hat{\mathbf{f}})'(\mathbf{y} - \hat{\mathbf{f}})/\{n - \text{tr}(2\mathbf{S} - \mathbf{S}\mathbf{S}')\}$$

and define confidence sets

$$C(\tilde{\mathbf{f}}) = \{\mathbf{g} \in \mathbb{R}^b; \nu(\mathbf{g}) \leq G_{1-\alpha}\}$$

based on

$$\nu(\tilde{\mathbf{f}}) = (\hat{\mathbf{f}} - \tilde{\mathbf{f}})'(\mathbf{S}\mathbf{S}'\hat{\sigma}^2)^{-1}(\hat{\mathbf{f}} - \tilde{\mathbf{f}}).$$

Here $G_{1-\alpha}$ is the $(1-\alpha)$th quantile of the distribution of $\nu(\tilde{\mathbf{f}})$.

Do we know G? Not necessarily.

In the case of linear regression, where the Gaussian model is correct and $\mathbf{S}$ is a p-dimensional projection, $\nu(\tilde{\mathbf{f}}) = \nu(\mathbf{f})$ has distribution $(n-p) + pF_{p,n-p}$.

When this is not the case we can argue that the distribution is approximately

$$\{n - \text{tr}(2\mathbf{S} - \mathbf{S}\mathbf{S}')\} + \text{tr}(\mathbf{S}\mathbf{S}')F_{\text{tr}(\mathbf{S}\mathbf{S}'),n-\text{tr}(2\mathbf{S}-\mathbf{S}\mathbf{S}')}$$

If we are not sure of the normality assumption or that $\tilde{\mathbf{f}} \approx \mathbf{f}$ we can use the bootstrap to construct an approximate distribution $\hat{G}$ of G.

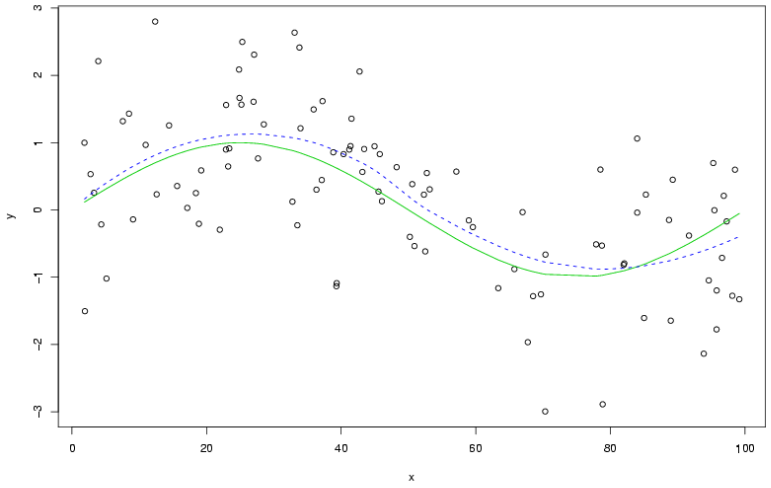
How do we do it?

8.5 Bootstrap estimate of $G_{1-\alpha}$

A bootstrap sample is generated in the following way

- For some data $\mathbf{y}$ use some procedure (a linear smoother for example) to obtain an estimate $\hat{\mathbf{f}}$ of some estimand (in this case the regression function $\mathbf{f}$).
- Obtain residuals $\hat{\boldsymbol{\varepsilon}} = \mathbf{y} - \hat{\mathbf{f}}$.

Figure 8.3: The regression curve and an outcome with $n = 100$ and $\sigma^2 = 1$.



- Take a simple random sample of size B from the residuals $\hat{\varepsilon}_1, \ldots, \hat{\varepsilon}_n$. Notice that this makes them IID just like the εs.
- Construct a "new" data set

$$\mathbf{y}^* = \hat{\mathbf{f}} + \hat{\boldsymbol{\varepsilon}}^*$$

  with $\hat{\boldsymbol{\varepsilon}}^*$ the vector of re-sampled residuals.
- From the new data form a new estimate $\hat{\mathbf{f}}^*$.
- Finally we obtain the value of

$$\nu^* = (\hat{\mathbf{f}}^* - \hat{\mathbf{f}})'(\mathbf{S}\mathbf{S}'\hat{\sigma}^{*2})^{-1}(\hat{\mathbf{f}}^* - \hat{\mathbf{f}})$$

- We repeat this procedure many times and form an approximate distribution $\hat{G}$ with the values of ν^*. We may use the $(1-\alpha)$th quantile of $\hat{G}$ as an estimate of $G_{1-\alpha}$.

Let's consider the model $y_i = f(x_i) + \varepsilon_i, i = 1, \ldots, n$ with ε_i IID normal. In Figure 8.4 we see qqplots of the true G, the bootstrap G and the F-distribution approximation.

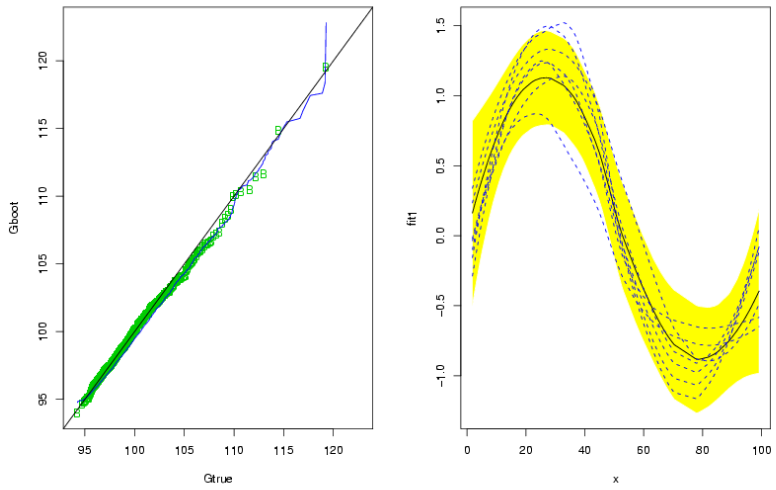
8.6 Displaying the confidence sets

Displaying an $n - dimensional$ ball is not easy.

Global confidence bands usually show the projections of the confidence set onto each of the component sub-spaces. Notice that a function (now I'm using function and n-dimensional vector interchangeably) in this set would actually be in a confidence cube as opposed to a ball! So a vector within the confidence bands isn't necessarily in the confidence ball. However its true that being in the ball implies being within the band.

Another popular approach is selecting a few functions at random from $N(\hat{\mathbf{f}}, \mathbf{SS}'\hat{\sigma}^2)$ and checking to see if they are in the confidence set. If they are, we plot them. This enables us to see what kind of "shape" functions in the confidence set have. Maybe they all have a bump, maybe a large amount of them are close to being constant lines, etc...

Figure 8.4: QQ-plot of bootstrap vs. true G and the F-distribution approximation. We also see point-wise confidence intervals and curves in (blue) and out (green) of the bootstrap confidence set.



8.7 Approximate F-test

Using the F-distribution approximations we may construct F-tests for testing various hypotheses.

The p-value given by the S-Plus function `gam()` is usually testing for linearity and using an F-distribution approximation.

Suppose we wish to compare 2 smoothers $\hat{\mathbf{f}}_1 = \mathbf{S}_1\mathbf{y}$ and $\hat{\mathbf{f}}_2 = \mathbf{S}_2\mathbf{y}$. For example, $\hat{\mathbf{f}}_1$ may be linear regression and $\hat{\mathbf{f}}_2$ may be a "rougher" smoother.

Let RSS_1 and RSS_2 be the residual sum of squares obtained for each smoother. Which one do you expect to be bigger?

and γ_1 and γ_2 be the degrees of freedom of each smoother, tr$(2\mathbf{S}_j - \mathbf{S}_j\mathbf{S}_j')$, $j = 1, 2$. An approximation that may be useful for this comparison is

$$\frac{(RSS_1 - RSS_2)/(\gamma_2 - \gamma_1)}{RSS2/(n - \gamma_2)} \sim F_{\gamma_2-\gamma_1, n-\gamma_2}$$

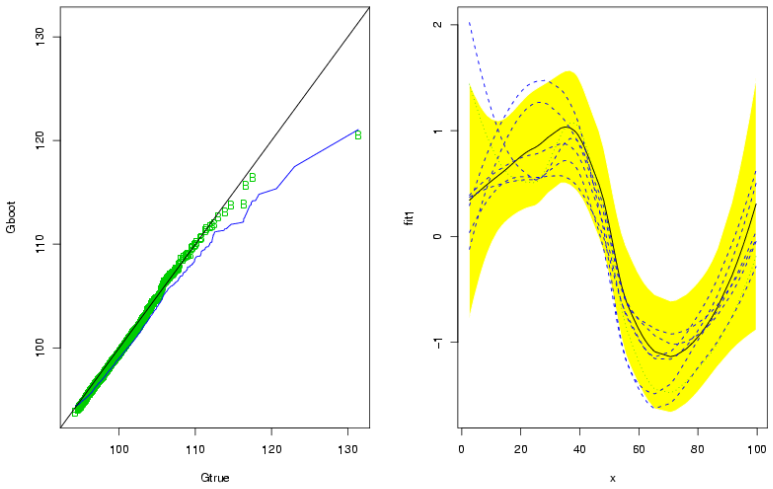
There are moment corrections that can make this a better approximation (see H&T).

8.8 Bootstrap and MLE

The bootstrap is sort of a computer implementation of nonparametric or parametric maximum likelihood. An advantage of the bootstrap is that it permits us to compute maximum likelihood estimates of standard errors and other quantities when no closed form solutions are available.

For example, consider a B-spline problem. If we chose the knots with some automatic procedure and wanted to include the variation introduced by this data-driven procedure, it would be very difficult to obtain closed form solutions for the standard errors of our estimates. Using the bootstrap we can get these.

Figure 8.5: Same as previos figure but with t-distributed errors



Bibliography

[1] Efron B. and Tibshirani, R.J (1993), *An Introduction of the Bootstrap*. Chapman and Hall/CRC: New York.

[2] Efron, B. (1979). Bootstrap Methods: Another Look At the Jackknife, *The Annals of Statistics* 7, 1–26.