

# 1 Alternative Estimation Strategies (Soc 504)

When regression assumptions are violated to the point that they degrade the quality of the OLS estimator, we may use alternative strategies for estimating the model (or use alternative models). I discuss 4 types of alternate estimation strategies here: bootstrapping, robust estimation for M-estimators, Weighted Least Squares (WLS) estimation, and Generalized Least Squares estimation.

## 2 Bootstrapping

Bootstrapping is useful when your sample size is small enough that the asymptotic properties of MLE or OLS estimators is questionable. It is also useful when you know the errors (in a small sample) aren't normally distributed.

The bootstrapping approach for a simple statistic is relatively simple. Given a sample of  $n$  observations, we take  $m$  resamples with replacement of size  $n$  from the original sample. For each of these resamples, we compute the statistic of interest and form the empirical distribution of the statistic from the results.

For example, I took a sample of 10 observations from a  $U(0,1)$  distribution. This size sample is hardly large enough to justify using normal theory for estimating the standard error of the mean. In this example, the sample mean was .525, and the estimated standard error was .1155. After taking 1000 bootstrap samples, the mean of the distribution of bootstrap sample means was .525, and the estimated standard error was .1082. The distribution of means looked like:

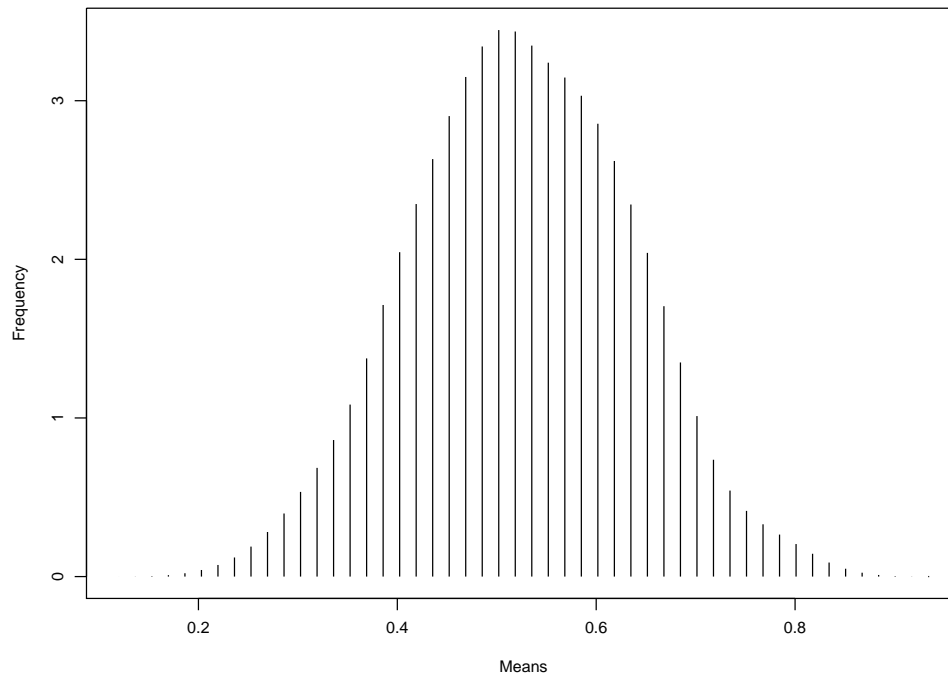


Figure 1. Bootstrap Sample Means.

This distribution is approximately normal, as it should be. The empirical 95% confidence interval for the mean was (.31, .74). This interval can be found by taking the 2.5<sup>th</sup> and 97.5<sup>th</sup> largest values from the empirical bootstrap distribution.

In this case, the bootstrap results did not differ much from the original results. However, we can better trust the bootstrap results, because normal theory really doesn't allow us to be confident in our original estimate of the standard error.

Below is the c program that produces the bootstrap samples for the above example.

```
#include<stdio.h>
#include<math.h>
#include<stdlib.h>

double uniform(void);

main(int argc, char *argv[])
{
int samples,rep,pick;
double mean,threshold;
double replicate,r;

double y[10]={.382,.100681,.596484,.899106,.88461,.958464,.014496,.407422,.863247,.138585};

FILE *fpout;

for(samples=1;samples<=1000;samples++)
{
printf("doing sample %d\n",samples);
mean=0;
for(rep=0;rep<10;rep++)
```

```

    {
        r=uniform();
        threshold=0;
        for(pick=0;pick<10;pick++)
            {
                if(r>threshold){replicate=y[pick];}
                threshold+=.1;
            }
        mean+=replicate;
    }
    mean/=10;

    if ((fpout=fopen(argv[1],"a")==NULL)
        {printf("couldn't open the file\n"); exit(0);}

    fprintf(fpout,"%d %f\n",samples,mean);

    fclose(fpout);

}
return 0;
}

double uniform(void)
{
    double x;
    double deviate;

    x=random();
    deviate=x/(2147483647);
    if (deviate<.00000000000000001){deviate=.00000000000000001;}

    return deviate;
}

```

In a regression setting, there are two ways to conduct bootstrapping. In one approach, we assume the  $X$  variables are random (rather than fixed). Then, we can obtain bootstrap estimates of the sampling distribution of  $\beta$ , by taking samples of size  $n$  from the original sample and forming the distribution of  $(X^{(j)T} X^{(j)})^{-1} X^{(j)T} Y^{(j)}$  (the OLS estimates from each bootstrap sample “ $j$ ”). In the other approach, we treat  $X$  as fixed. If  $X$  is fixed, then we must sample the error term (the only random component of the model). We do this as follows:

1. Compute the OLS estimates for the original sample.
2. Obtain  $e_i = Y_i - X_i \hat{\beta}$ .
3. Bootstrap samples of  $e$ ,
4. Compute  $Y_i^{(j)} = Y_i + e_i^{(j)}$  for each bootstrap sample.
5. Compute the OLS estimates for each bootstrap sample  $Y^{(j)}$ .

### 3 Robust Estimation (with M-Estimators)

Fox discusses M-Estimation as a supplemental method to OLS for examining the effect of outliers. The book introduces the notion of “influence plots” by showing how a single outlying observation influences a sample estimate (e.g., the mean, median, etc.):

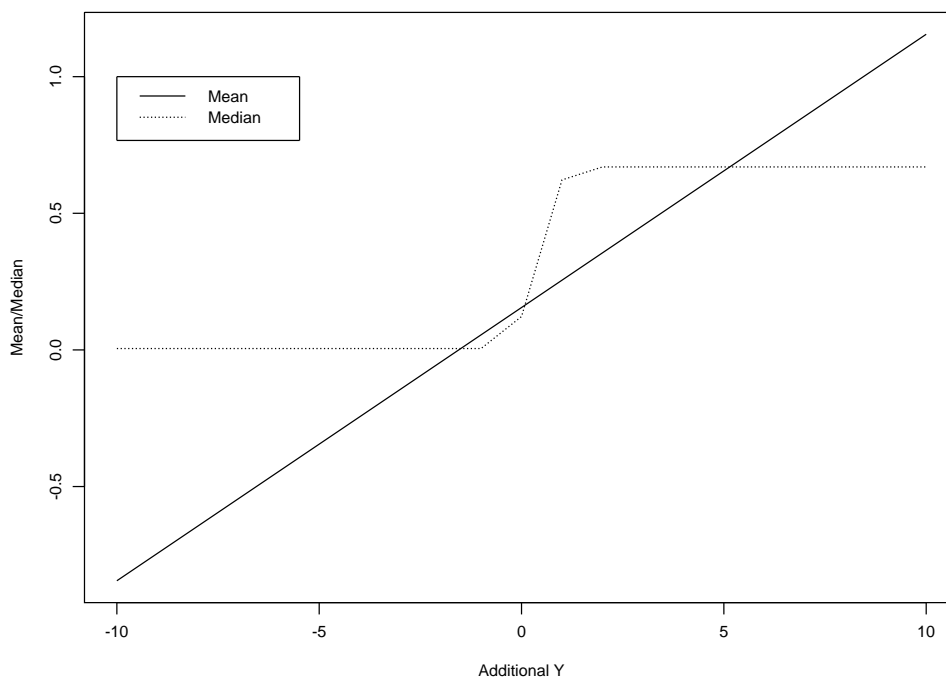


Figure 2. Influence Plot Example.

The influence plot was produced by taking a sample of 9  $N(0, 1)$  observations and adding a 10<sup>th</sup> observation taking values incrementally in the range of  $(-10, 10)$ . The statistics (mean and median) were then computed. The median is clearly more resistant to the outlying observation, as indicated by the plot. The true mean and median of a  $N(0, 1)$  variable are 0, and the sample median remains much closer to this value as the 10<sup>th</sup> observation becomes more extreme.

When outliers are a problem, OLS estimation may not be the best approach to estimating regression coefficients, because the OLS estimator minimizes the mean squared error of the observations. Outliers generate undue influence on the coefficients, then, because the error terms are squared. In order to determine whether our estimates are robust, we can try other criteria rather than OLS. For example, a common alternative is the least absolute value estimator:

$$LAV = \min \sum (| Y_i - \hat{Y} |)$$

Fox shows that this class of estimators can be estimated generally using iteratively reweighted least squares (IRLS). In IRLS, the derivative of the objective function is re-

expressed in terms of weights that apply to each observation. These weights are a function of the error term (for LAV,  $w_i = \frac{1}{E_i}$ ), which, of course is a function of the regression coefficients in a regression model, or they can simply be  $Y - \mu$ . Thus, we can solve for the regression coefficients by using a starting value for them, computing the weights, recomputing the regression coefficients, etc. While an estimate of a sample mean is:

$$\hat{\mu} = \frac{\sum w_i Y_i}{\sum w_i}$$

the estimate of regression coefficients is:

$$\hat{\beta} = (X^T W X)^{-1} (X^T W Y).$$

I illustrate the use of IRLS estimation with the same sample of 9  $N(0, 1)$  observations as used above. In this example, I use IRLS on the LAV objective function to estimate the mean. Notice that the LAV estimate of the mean is bounded. If we think about the median of a sample, the median is the mean of the two centermost observations. In this sample data, when the 10<sup>th</sup> observation is an extreme negative value, the centermost observations are  $-.23$  and  $.24$ . When the 10<sup>th</sup> observation is an extreme positive value, the centermost observations are  $.24$  and  $1.10$ . Thus, the LAV estimate of the mean will be in the range of  $(-.23, 1.10)$  as the figure below illustrates. Once the value becomes extreme enough, the weight for that observation becomes very small in the IRLS routine, so the influence of the observation is minimal. Below is a c program that estimates the LAV function using IRLS.

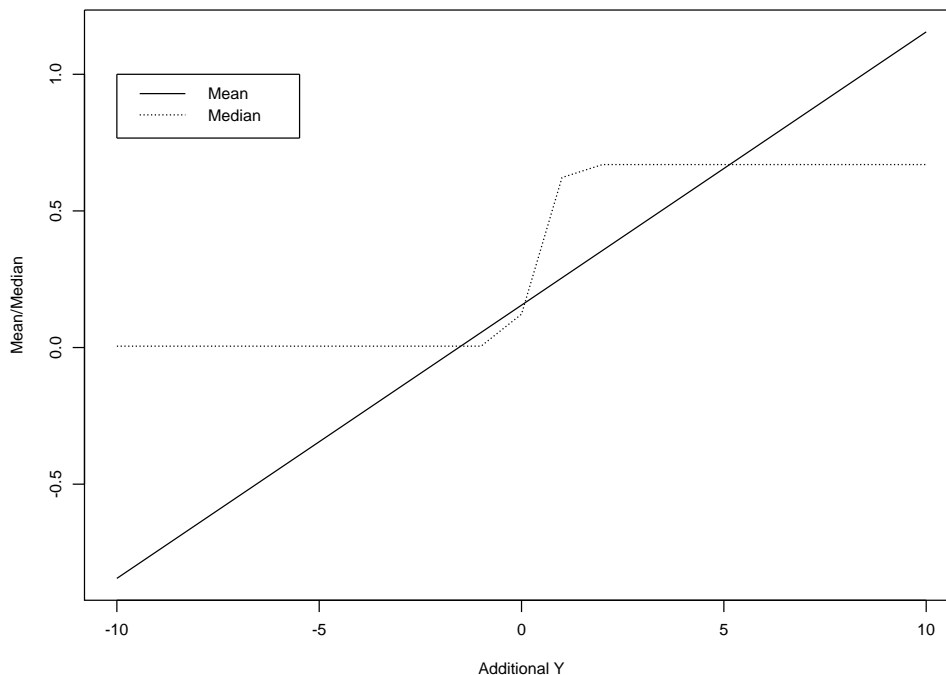


Figure 3. IRLS Results of Using LAV Function to Estimate a Mean.

```

#include<stdio.h>
#include<math.h>
#include<stdlib.h>

main(int argc, char *argv[])
{
int i,j,k,loop;
double mean[100],weight[10],num,denom;

double y[10]={-.30023, -1.27768, .244257, 1.276474, 1.19835,
1.733133,-2.18359,-.23418, 1.095023, 0.0};

FILE *fpout;

for(i=-20;i<=20;i++)
{
y[9]=i*1.0;
mean[0]=0;
mean[1]=0;
for(j=0;j<=9;j++)
{mean[1]+=y[j];}
mean[1]/=10;

loop=1;
while(fabs(mean[loop]-mean[loop-1])>.000001)
{
printf("mean %d=%f\n",loop,mean[loop]);

for(k=0;k<=9;k++){weight[k]=1.0/(fabs(y[k]-mean[loop]));}

num=0; denom=0;
for(k=0;k<=9;k++){denom+=weight[k]; num+=(weight[k]*y[k]);}

loop++;
mean[loop]=num/denom;
}

if ((fpout=fopen(argv[1], "a"))==NULL)
{
printf("couldn't open the file\n"); exit(0);
}

fprintf(fpout,"%f %f %f\n",y[9],mean[1],mean[loop]);
fclose(fpout);
}
return 0;
}

```

Within the while() loop, the previous value of the mean is used to compute the weights of each observation ( $\text{weight}[k]=1/\text{fabs}(y[k]-\text{mean}[\text{loop}])$ ). Then, given the new weights, a new value of the mean is computed ( $\text{mean}[\text{loop}]=\text{num}/\text{denom}$ ). If we wanted to use an alternate objective function (e.g., Huber, bisquare), then we would simply replace the weight calculation. I do not illustrate IRLS for a robust regression, but it is a straightforward modification of this algorithm in which  $\mu$  is replaced by  $X\beta$  in the weight calculation, and the calculation of  $\mu$  is replaced by the weighted OLS estimator shown above.

## 4 Weighted Least Squares and Generalized Least Squares Estimation

When errors are heteroscedastic, the error term is no longer constant across all observations. Thus, the assumption:  $\sigma_i \sim N(0, \sigma_e I)$  is no longer true. Rather,  $\sigma \sim N(0, \Sigma)$ , where  $\Sigma$  is a diagonal matrix (off-diagonal elements are still 0).

In this case, the likelihood function becomes modified to incorporate this altered error variance term:

$$L(\beta | X, Y) = \prod \frac{1}{(2\pi)^{\frac{1}{2}} \sigma_i} \exp \left\{ -\frac{(Y_i - (X'\beta)_i)^2}{2\sigma_i^2} \right\},$$

We can typically assume that the diagonal elements of the matrix are weighted values of a constant error variance, say:

$$\Sigma = \sigma_e \begin{bmatrix} \frac{1}{w_1} & 0 & \dots & 0 \\ 0 & \frac{1}{w_2} & \ddots & \vdots \\ \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \dots & \frac{1}{w_n} \end{bmatrix}$$

which gives us a matrix expression for the likelihood:

$$L(\beta | X, Y) = \frac{1}{(2\pi)^{\frac{n}{2}} |\Sigma|^{\frac{1}{2}}} \exp \left\{ -\frac{1}{2} (Y - X\beta)^T \Sigma (Y - X\beta) \right\}.$$

The estimator for the parameters then becomes  $(X^T W X)^{-1} (X^T W Y)$ , and the variance of the estimator is:  $\sigma_e^2 (X^T W X)^{-1}$ .

The obvious question is: “What do we use for weights?” We may know that the error variance is related to one of the observed variables, in which we could either build this function into the likelihood function above (and estimate it), or we could use the inverse of the variable as the weights. Alternatively, we could simply bypass WLS estimation altogether and simply divide every variable in the model by the offending  $X$  variable and use OLS estimation.

Generalized Least Squares estimation is the generalization of WLS. In WLS, we assume the off-diagonal elements of the  $\Sigma$  matrix are 0. This assumption, if you recall from our notes on multiple regression theory, implies that errors are independent across observations. Often, however, this may be an unreasonable assumption (e.g. in time series, or in clustered data). We can relax this assumption to obtain the GLS estimator:  $\beta = (X^T \Sigma^{-1} X)^{-1} X^T \Sigma^{-1} Y$ . This should look similar to the WLS estimator—in fact, it’s the same (the WLS estimator just has 0’s on the off-diagonal elements). The variance estimator for  $\beta$  is:  $(X^T \Sigma^{-1} X)^{-1}$ .

Obviously, we cannot estimate all the elements of  $\Sigma$ —there are  $\frac{n(n+1)}{2}$  unique elements in the matrix. Thus, we may use functions or other simplifications to reduce the elements to be estimated. This is the essence of basic time series models, as we will discuss in a few weeks.