The Random Walk Hypothesis, Bootstrapped

Matthew Pollard

11th November 2007

Abstract

Efron's (1979) bootstrap method is used to test for serial correlation in stock returns. The procedure used does not make distributional assumptions for returns and can be applied to any test statistic. The studied test statistic is the popular Box-Pierce statistic, or the sum the first K squared sample autocorrelations. We use the bootstrap to generate the distribution of this statistic under the null of zero-correlation and calculate p-values from this distribution. The size and power of the test is calculated by Monte Carlo simulation. Power is calculated against six realistic alternative models, with an AR(1) correlation structure and both student-t and GARCH innovations.

1 Introduction

Whether stock returns are predictable is an important question in finance. Under weak form market-efficiency, future returns are not predictable from historical returns and follow a "random walk". That is, there is no information contained in past returns about the *level*[1] of future returns. A testable consequence is that serial returns are uncorrelated and this is referred to as the "random walk hypothesis".

Constructing tests based sample autocorrelations is non-trivial due to the complex distribution of the test statistics. Tests often resort to overly strong distributional assumptions for returns in order to obtain test-statistic distributions, particularly that returns are identically distributed and normal; see, for instance, Campbell and Mankiw (1987), Cochrane (1987). Tests based on these assumptions perform sub-optimally when the data are heteroscedastic and fat-tailed, which is a consistent empirical finding in daily to monthly returns (Cont, 2003).

Random walk hypothesis formally states that return r_t is uncorrelated from previous returns r_{t-k} for all t and lags k:

$$E[(r_t - \mu)(r_{t-k} - \mu)] = 0, \ \ \forall t, k.$$

If the time-series is stationary, or $E[(r_t - \mu)(r_{t-k} - \mu)] = E[(r_s - \mu)(r_{s-k} - \mu)]$ for all t and s, this condition testable. One test is whether the first K autocorrelations, ρ, are equal to zero:

$$H_0 : \{\rho(1) = \rho(2) = ...\rho(K) = 0\} \tag{1}$$

[1] The future *variation* in returns, or volatility, is predictable from past returns.

The Box-Pierce statistic is a popular test statistic for (1), the sum of the first K squared sample autocorrelations:

$$\hat{S} = T \sum_{t=1}^{K} \hat{\rho}(k)^2.$$

Under the null hypothesis that returns are independent and identically distributed, $\hat{S}$ has an asymptotic χ^2_K distribution. Several papers use this asymptotic distribution to construct critical cut-off values for $\hat{S}$, see for example, Ortega (1975) and Berges (1984), who both reject the null. These papers are implcitly testing a much stronger hypothesis: i.id. returns instead of uncorrelated returns. Stock returns are not *identically* distributed; in particular, they not homoscedastic (Cont, 2001). As Lo and MacKinlay (1988) argue, tests based on the homoscedastic assumption are likely to reject the random walk hypothesis when returns are uncorrelated. In addition, the small-sample properties of $\hat{S}$ are complicated and depend highly on the distribution of returns. Runde (1977) shows that tests based on χ^2_K cutoffs can produce seriously misleading inferences for homoscedastic but non-normal distributions, such as the student-t.

A solution is to avoid asymptotic arguments altogether, and focus on approximating the small-sample properties of $\hat{S}$. The Efron's (1979) bootstrap method is well suited for this. . Given *any* test statistic, T, the bootstrap allows us to draw samples from its sample distribution; or precisely, the *"bootstrap"* distribution. These samples can then be used to calculate standard errors, form a confidence interval, or calculate p-values under a null hypothesis. In our case,

$$\Pr(S > \hat{S} | H_0). \tag{2}$$

We calculate (2) using daily returns from the S&P 500 index and S&P 500 futures contracts, 1998 to 2007. Spurious positive autocorrelation may be present in daily S&P 500 index returns due to the "stale price" effect of Keim and Stambaugh (1986). Futures contracts are continuously traded and do not suffer from this effect.

Conditional volatility of returns is not constant in the sample. This complicates the distribution of $\hat{S}$ and affects the bootstrap resamples. To study the effect, we implement the test using both raw returns and rescaled returns defined by $\widetilde{r}_t = \frac{r_t}{\hat{\sigma}_t}$, where $\hat{\sigma}_t$ is the estimated conditional volatility. The size and power of the test is calculated by Monte Carlo simulation. Power is calculated against six realistic alternative models, with an AR(1) correlation structure and both student-t and GARCH innovations.

The paper is organised as follows. Section 2 introduces the bootstrap principle and its application to time-series data. Section 3 presents the resampling method and section 4 presents the results. Section 5 calculates the size and power of the test, and section 6 discusses the findings.

2 Bootstrap Principle

Monte Carlo bootstrap methods depend on the notion of a *bootstrap resample*. Call the original sample $\chi = (r_1, ..., r_T)$. The bootstrap resample $\chi^* = (r_1^*, ..., r_T^*)$ is a random sample of size T where each r_t is drawn independently with replacement and has an equal

$\frac{1}{T}$ probability of appearing in χ^*. Corresponding to a bootstrap resample χ^* is a bootstrap replication of $\widehat{S} = S[\chi]$,

$$\hat{S}^* = S[\chi^*]$$

The statistic $\widehat{S}^*$ is repeatedly calculated B times by randomly drawing new resamples. Statistical properties of $\widehat{S}$ are then estimated from the set $(\hat{S}_1^*, ..., \hat{S}_T^*)$. For example, the estimated variance of $\hat{S}$ is

$$V\hat{a}r(\hat{S}) = \frac{1}{B-1}\sum_{b=1}^{B}(\widehat{S}_b^* - \overline{S})^2.$$

2.1 Bootstrap for Time Series

Independent resampling from $(r_1, ..., r_T)$ produces resamples that are independent. The resamples χ^* inherit the same distributional properties as χ, however they are now consistent with the null of independence, and allow for calculation of $\Pr(S > \hat{S}|H_0\{\text{random walk}\})$. After forming B resamples of S^*, The Monte Carlo estimator is used:

$$\begin{aligned} \Pr(S > \hat{S}|H_0) \quad &\Longleftrightarrow \quad \Pr(\hat{S} > \hat{S}^*|H_0) \\ \text{Actual world} \quad & \quad\quad\quad \text{Bootstrap world} \\ &\approx \quad \frac{1}{B}\sum_{b=1}^{B} I(\hat{S} > \hat{S}_b^*) \end{aligned} \tag{3}$$

The distribution of $\hat{S}$ *not* conditioned on the null is also important. Whatever dependency structure is present in $(r_1, ..., r_T)$ must be preserved in order for the bootstrap distribution of $\hat{S}$ to map onto the actual distribution. The solution is called the *block bootstrap.* Block bootstrapping samples contiguous "blocks" of ordered returns and constructs resamples by gluing these blocks together. The idea is that if the blocks are long enough (i.e. there is sufficient lag between most of the observations *between* blocks) then much of the original dependency structure will remain in the resamples. Choice of block length is still an open problem; see Pollard (2005) for an overview.

3 Method

We simulate $B = 4000$ resamples using the independent bootstrap and block bootstrap. The block bootstrap uses non-overlapping blocks of length $l = 30$; this length is picked to ensure that the Monte Carlo draws of $\hat{S}^*$ are approximately centered around $\hat{S}$. P values are estimated by (3).

The distribution of S conditioned on volatility uses scaled returns, $\widetilde{r}_t = \frac{r_t}{\hat{\sigma}_t}$, where $\hat{\sigma}_t$ is the estimated volatility from the GARCH(1,1) model:

$$\hat{\sigma}_t^2 = \alpha_0 + \alpha_1(r_t - \overline{r})^2 + \beta\hat{\sigma}_{t-1}^2$$

The model is estimated by maximum likelihood and the estimated coefficients are $\alpha_0 =$ 7.09e-07, $\alpha_1 =$ 8.08e-02, $\beta = 0.915$.

Data	p-value
Index, Unconditional	0.024
Futures, Unconditional	0.010
Index, Conditional	0.26
Futures, Conditional	0.25

Table 1: p-values for the random walk hypothesis, H_0. At α =5%, we reject H_0 for both the unconditional index and futures returns and accept H_0 for the conditional index and futures returns.

4 Results

Figure 1 shows the bootstrap distributions of $S|H_0$ (blue) and S (red). The dashed line is point estimate $\hat{S}$. The p-value of H_0 is measured as the area under $S|H_0$, right of the point estimate. The p-values are in Table 1.

5 Size and Power of Test

5.1 Size

The bootstrap distribution of $\hat{S}$ is an approximation to the true distribution. Our test is based on quantile cut-offs points from this distribution, hence the true size (or type-I error) $\widetilde{\alpha}$ is not exactly equal to α. We calculate $\widetilde{\alpha}$ by simulating returns consistent H_0 by randomly reordering S&P 500 index returns, and also use independent student-t generated returns. The algorithm using the S&P 500 index is:

1. For b in 1 to B
   (a) randomly reorder conditional returns to yield new sample, χ_b.
   (b) Calculate $\hat{S}_b = \hat{S}[\chi_b]$.
   (c) Find bootstrap distribution of $\hat{S}_b$, calculate p-value $\hat{p}_b$.
2. True coverage is estimated by $\hat{\alpha} = \frac{1}{B}\sum I(\hat{p}_b < \alpha)$

The same routine is used for generating student-t returns, except step (a) draws new, independent returns instead of re-ordering returns.

This is run $B = 10,000$ times with a size of $\alpha = 0.05$. The true coverage probability for the conditional data is $\hat{\alpha} = 0.0503$ and student-t is $\hat{\alpha} = 0.0497$; both are very close to 0.05. The standard error the estimates is $s.e.(\hat{\alpha}) = \frac{p(\hat{p}_b<\alpha)p(\hat{p}_b\geq\alpha)}{\sqrt{B}} \doteq 0.00475$.

5.2 Power

In calculating the power of the test, we use an AR(1) alternative where

$$r_t = \alpha_1 r_{t-1} + \varepsilon_t.$$

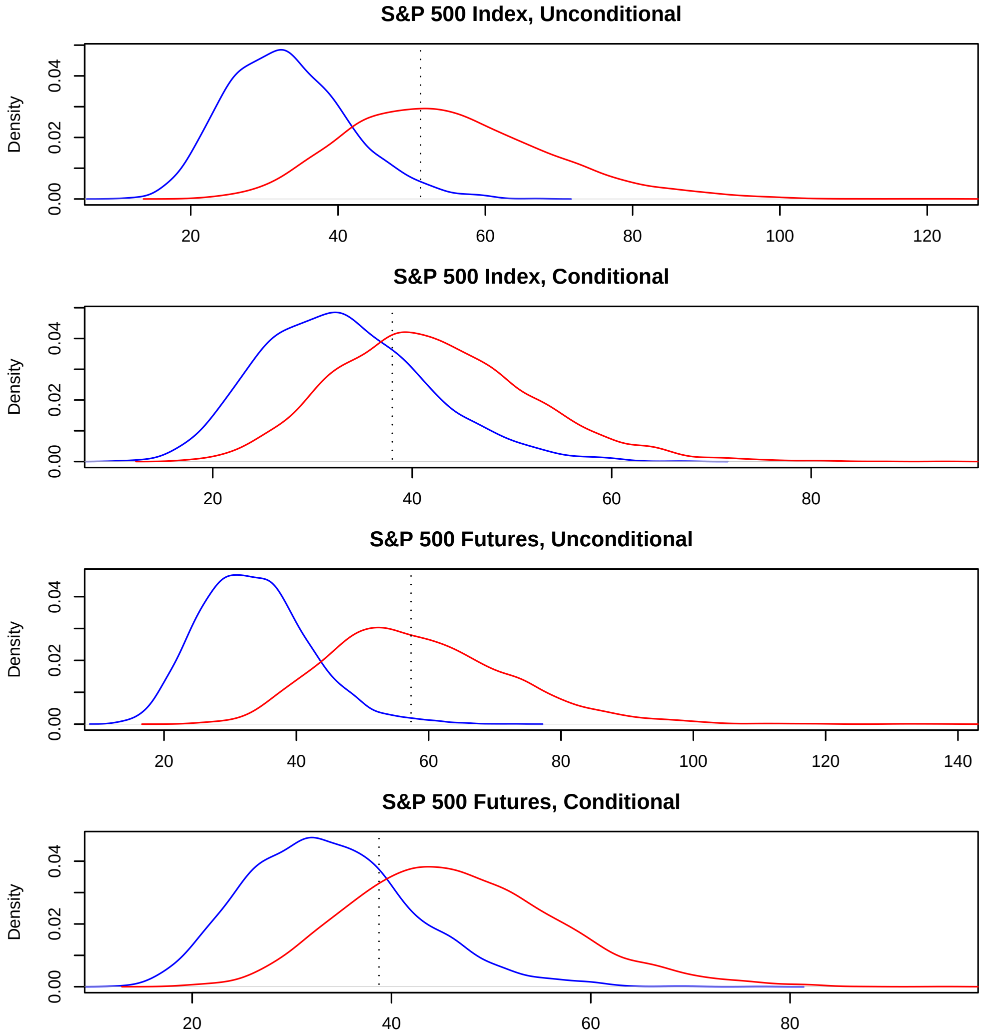

Figure 1: Bootstrap distributions of $S|H_0$ (blue) and S (red) using the independent and block bootstrap respectively. The dashed line is point estimate $\hat{S}$. The p-value of H_0 is measured as the area under $S|H_0$, right of the point estimate. We reject H_0 for both unconditional series at 5%, and accept H_0 for the conditional series.

Model	$\hat{\beta}(\mathcal{M})$
$(0.2, t)$	1.00
$(0.2, \text{GARCH})$	0.99
$(0.1, t)$	0.94
$(0.1, \text{GARCH})$	0.85
$(0.05, t)$	0.55
$(0.05, \text{GARCH})$	0.52

Table 2: Estimated power for the four alternative models.

and ε_t are (1) independently drawn from a t-distribution with 5 degrees of freedom, or (2) follow the GARCH(1,1) process:

$$\varepsilon_t \sim \mathcal{N}(0, V_t), \;\; V_t = 0.1 + 0.9V_{t-1} + 0.05\varepsilon_{t-1}^2$$

Th GARCH parameters above are set equal to the estimates for S&P returns. The six models (α, ε_t) considered are:

$$, \begin{array}{rcccrcc} \mathcal{M}_1 & = & (0.2, t) & & \mathcal{M}_4 & = & (0.1, \text{GARCH}) \\ \mathcal{M}_2 & = & (0.2, \text{GARCH}) & & \mathcal{M}_5 & = & (0.05, t) \\ \mathcal{M}_3 & = & (0.1, t) & & \mathcal{M}_6 & = & (0.05, \text{GARCH}) \end{array} ,$$

The models are chosen to best mimic S&P 500 index returns. Small, positive lag-1 autocorrelation are the most visible feature in sample autocorrelation plots of S&P 500 returns.

The algorithm for estimating the power, β, is:

1. Given $\mathcal{M}$, for b in 1 to B,
   (a) generate 2500 realizations from model $\mathcal{M}$,
   (b) Scale returns by estimated volatility,
   (c) Apply the independent-sampling bootstrap, calculate p value.
2. Set $\hat{\beta}(\mathcal{M}) = 1 - \frac{1}{N}\sum I(p_b > \alpha)$

We use $B = 1000$ simulations. Table 2 contains the power estimates $\hat{\beta}(\mathcal{M})$.

6 Discussion

The random walk model is rejected at 5% for the unconditional index and futures returns and accepted for both conditional index and futures returns. Two interesting results are:

1. Returns that are rescaled by volatility display considerably lower point estimates for $\hat{S}$. The bootstrap distributions of $\hat{S}$ under the null are almost identical, the rescaled p-values are lower. There is far less separation between the block-bootstrap and independent-bootstrap using rescaled returns.

2. There is almost no difference whether index and futures returns are used. The test p-values are almost equal and display similar distributions for S.

The first result can be explained as follows. Applying the independent bootstrap to unconditional returns, we are implicitly testing a stronger joint hypothesis:

$$H_0(strong): \ \{r_t \sim \text{i.i.d } \forall t\},$$

The independent bootstrap independently samples observations. This eliminates correlation in both the level of returns and the volatility structure. The block bootstrap resamples preserve both. Returns are not $i.i.d.$ due to volatility clustering, and we are seeing that reflected in the p-values and degree of separation between the two distributions. The correct test of H_0 uses returns conditional on volatility.

The second result indicates there is very little difference in correlation structure between futures and index data. We hypothesised that non-synchronous trading effects would introduce correlation artifacts in S&P 50 index returns, but not in futures data which are continuously traded. The results suggest that non-synchronous effects are not significant for S&P 500 index returns.

The size of the test is almost exactly the intended coverage of 5%. The power of the test to reject the alternatives is fairly high, although a benchmark is needed for comparison.

In conclusion, we accept the random walk hypothesis for both index and futures returns, with p-values equal to 0.26 for the index data and 0.25 for the futures data. Tests for autocorrelation should use returns conditioned on volatility to avoid testing the stronger hypothesis of uncorrelated and identically distributed returns.

References

[1] Cont, R., "Empirical Properties of Asset Returns: Stylized Facts and Statistical Issues," Quantitative Finance, 1, 2, 223-236, 2001.

[2] Efron, B., "Bootstrap methods: another look at the jackknife." *Annals of Statistics,* 7, 1-26, 1979.

[3] Keim, D. & Stambaugh, R., "Predicting Returns in the Stock and Bond Markets," *Journal of Financial Economics*, 17, 357-90, 1986.

[4] Lo, A., & MacKinlay, A., "Stock market prices do not follow random walks: Evidence from a Simple Specification Test," *Review of Financial Studies*, 41-66, 1988.

[5] Runde, R., "The asymptotic null distribution of the Box-Pierce Q-statistic for random variables with infinite variance an application to German stock returns," *Journal of Econometrics*, 78, 2, 205-216, 1997.

[6] Scholes, M., & Williams, J., "Estimating Betas from Non-Synchronous Data," *Journal of Financial Economics*, 5, 309-328, 1977.

A Bootstrap Functions

```
boot<-function(x,theta=mean,t=mean,R=600,spit=F){
    data<-matrix(sample(x,size=length(x)*R,replace=T),nrow=R)
    thetastar<-apply(data,1,theta)
    se<-(var(thetastar))^(1/2)
    bias<-t(x)-mean(thetastar)
    original<-theta(x)
    result<-cbind(original,bias,se)
    if (spit!=F)
    result<-thetastar
    return(result)
    }

tsbooty<-function(x,stat,R,b,diag=0){
    n<-length(x)
    k<-round(n/b)
    resample<-1:(k*b)
    data<-matrix(ncol=k*b,nrow=R)
    random<-matrix(ncol=k,nrow=R)
    for(i in 1:R){
    random[i,]<-sample(1:(n-b),k,replace=T)}
    for(p in 1:R){
    for(i in 1:k){
    resample[(b*(i-1)+1):(i*b)]<-x[random[p,i]:(random[p,i]+b-1)]}
    data[p,]<-t(resample)
    }
    statstar<-apply(data,1,stat)
    se<-(var(statstar))^(1/2)
    bias<-mean(statstar)-stat(x)
    original<-stat(x)
    result<-cbind(original,bias,se)
    if (diag==0)
    return(result)
    if (diag==1){
    plot(ts(data[1,]))
    return(list(result,"stat"=statstar))
    }
    }

myacf<-function(x){length(x)*sum((acf(x,plot=F)$acf[-1])^2)}

riskm<-function(r){
    lam<-0.94
    length(r)->n
    h<-rep(0,n)
    h[1]<-(r[1]-mean(r))^2
    for (i in 2:n)
    h[i]<-lam*h[i-1]+(1-lam)*(r[i-1]-mean(r[1:(i-1)]))^2
    return(sqrt(h))}
```

B R Code

This code implements the tests and calculates the size and power.

```
result<-boot(gsr,myacf,R=4000,spit=T)
block<-tsbooty(gsr,myacf,4000,30,diag=T)
#Prewhiten
riskm(gsr)->v
resultw<-boot(gsr/v,myacf,4000,spit=T)
blockw<-tsbooty(gsr/v,myacf,4000,30,diag=T)
##Futures Data
read.csv("c:/futures.csv")->futures
ret(na.omit(futures[,2]))->f

resultf<-boot(f,myacf,R=4000,spit=T)
blockf<-tsbooty(f,myacf,4000,30,diag=T)
###Prewhiten
v2<-riskm(f)
resultfw<-boot(f/v2,myacf,R=4000,spit=T)
blockfw<-tsbooty(f/v2,myacf,4000,30,diag=T)
###

plot(density(result),xlim=c(min(c(block$stat,result)),max(c(block$stat,result))),
    col="blue",main="S&P 500 Index, Unconditional")
    lines(density(block$stat),col="red")
    abline(v=myacf(gsr),lty=3)
    mean(myacf(gsr)<result)

plot(density(result),xlim=c(min(c(blockw$stat,resultw)),max(c(blockw$stat,resultw))),
    col="blue",main="S&P 500 Index, Conditional")
    lines(density(blockw$stat),col="red")
    abline(v=myacf(gsr/v),lty=3)
    mean(myacf(gsr/v)<resultw)

plot(density(resultf),xlim=c(min(c(blockf$stat,resultf)),max(c(blockf$stat,resultf))),
    col="blue",main="S&P 500 Futures, Unconditional")
    lines(density(blockf$stat),col="red")
    abline(v=myacf(f),lty=3)
    mean(myacf(f)<resultf)

plot(density(resultfw),xlim=c(min(c(blockfw$stat,resultfw)),max(c(blockfw$stat,resultfw))),
    col="blue",main="S&P 500 Futures, Conditional")
    lines(density(blockfw$stat),col="red")
    abline(v=myacf(f/v2),lty=3)
    mean(myacf(f/v2)<resultfw)

### coverage
    indic<-rep(0,2500)
    for (i in 1:2500){
    dat<-rt(2500,7)
    S<-myacf(dat)
    indic[i]<-mean(boot(dat,1000,theta=myacf,spit=T)>S)}
```

```
    v<-riskm(gsr)
    for (i in 1:2500){
    N<-sample(1:length(gsr),replace=F,size=length(gsr))
    dat<-gsr[N]/v[N]
    S<-myacf(dat)
    indic[i]<-mean(boot(dat,1000,theta=myacf,spit=T)>S)}
    indic<-rep(0,2500)
    for (i in 1:2500){
    dat<-garch.sim(2500,p=c(0.1,0.3),q=(0.65))
    S<-myacf(dat)
    indic[i]<-mean(boot(dat,1000,theta=myacf,spit=T)>S)}
    indic<-rep(0,2500)
    for (i in 1:2500){
    dat<-garch.sim(2500,p=c(0.1,0.3),q=(0.65))
    v<-riskm(dat)
    dat<-dat/v
    S<-myacf(dat)
    indic[i]<-mean(boot(dat,1000,theta=myacf,spit=T)>S)}

#Power
    alpha=0.05
    indic<-rep(0,2500)
    for (i in 1:2500){
    dat<-arima.sim(list(ar=c(0.2)),n=2500,innov=rt(2500,5))
    S<-myacf(dat)
    indic[i]<-mean(boot(dat,2500,theta=myacf,spit=T)>S)}
    power=mean(alpha>indic[1:i])

    indic<-rep(0,2500)
    for (i in 1:2500){
    dat<-arima.sim(list(ar=c(0.1)),n=2500,innov=rt(2500,5))
    S<-myacf(dat)
    indic[i]<-mean(boot(dat,2500,theta=myacf,spit=T)>S)}
    power=mean(alpha>indic[1:i])

    indic<-rep(0,2500)
    for (i in 1:2500){
    innov<-garch.sim(2500,p=c(0.1,0.1),q=c(0.8))
    dat<-arima.sim(list(ar=c(0.2)),n=2500,innov=innov)
    dat<-dat/riskm(dat)
    S<-myacf(dat)
    indic[i]<-mean(boot(dat,2500,theta=myacf,spit=T)>S)}
    power=mean(alpha>indic[1:i])

    indic<-rep(0,2500)
    for (i in 1:2500){
    innov<-garch.sim(2500,p=c(0.1,0.1),q=c(0.8))
    dat<-arima.sim(list(ar=c(0.1)),n=2500,innov=innov)
    dat<-dat/riskm(dat)
    S<-myacf(dat)
    indic[i]<-mean(boot(dat,2500,theta=myacf,spit=T)>S)}
    power=mean(alpha>indic[1:i])
```