

# Financial Econometrics - Individual coursework

Parham Allboje

04 March, 2021

```
library(readxl)
library(dynlm)
library(sandwich)
library(lmtest)
library(tseries)
library(dplyr)
library(ggplot2)
library(gridExtra)
```

## Question 1

```
sp500 <- read_excel("data/SP500WeekDays.xlsx")

sp500_lm <- lm(sp500$sp ~ sp500$T + sp500$W + sp500$R + sp500$F)
summary(sp500_lm)

##
## Call:
## lm(formula = sp500$sp ~ sp500$T + sp500$W + sp500$R + sp500$F)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.204627 -0.005214  0.000142  0.005379  0.115842
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -4.214e-05  2.994e-04  -0.141  0.8881
## sp500$T       7.825e-04  4.158e-04   1.882  0.0599 .
## sp500$W       6.819e-04  4.157e-04   1.640  0.1010
## sp500$R       1.065e-04  4.177e-04   0.255  0.7988
## sp500$F       3.711e-04  4.184e-04   0.887  0.3752
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.01117 on 7314 degrees of freedom
## Multiple R-squared:  0.0007543, Adjusted R-squared:  0.0002078
## F-statistic:  1.38 on 4 and 7314 DF, p-value: 0.238
```

For the intercept in this linear model I chose the value for Monday to avoid the Dummy Variable trap. The Dummy Variable trap is a scenario in which the independent variables are multicollinear, meaning adding a fifth weekday does not add any information to the regression. Therefore, we regress over the other weekdays.

Accordingly, it is observable that none of the weekday effects are significant at the 5% level. Only tuesday with a p-value of 0.599 and a t-statistic of 1.882 can be considered to be significant at 10% level.

```
coeftest(sp500_lm, vcov = NeweyWest,)
```

```
##
## t test of coefficients:
##
##           Estimate Std. Error t value Pr(>|t|)
## (Intercept) -4.2137e-05 3.4526e-04 -0.1220 0.9029
## sp500$T      7.8249e-04 4.7787e-04 1.6375 0.1016
## sp500$W      6.8187e-04 4.7748e-04 1.4281 0.1533
## sp500$R      1.0646e-04 4.3806e-04 0.2430 0.8080
## sp500$F      3.7109e-04 4.2856e-04 0.8659 0.3866
```

Using the HAC estimator of the covariance matrix shows that now none of the weekdays are statistically significant neither at the 5% level nor at the 10% level. The HAC estimator changes the significance of the Tuesday regressor, which is not significant at 10% level anymore.

## Question 2

i.

```
usMacro <- read_excel("data/USMacro_Quarterly.xls")

usMacro<- usMacro %>% filter(Date >= '1955:01')
usMacro$logYt = log(usMacro$RealGDP)
usMacro$deltaYt = usMacro$logYt - lag(usMacro$logYt, 1)
deltaYt =na.omit(usMacro$deltaYt)
meanDeltaYt = mean(deltaYt)
meanDeltaYt_annualPct = meanDeltaYt*400
sigmaDeltaYt = sd(deltaYt)
sigmaDeltaYt_annualPct = sigmaDeltaYt*400
```

a)

```
## Mean of delta Yt is: 0.008.
```

b)

```
## Mean growth rate in percentage points at an annual rate is: 3.263.
```

c)

```
## Volatility of delta Yt in percentage points at an annual rate is: 3.647.
```

d)

```
acf(deltaYt, lag.max = 4, plot = FALSE)
```

```
##
## Autocorrelations of series 'deltaYt', by lag
##
```

```
##      0      1      2      3      4
## 1.000 0.287 0.169 0.030 -0.009
```

This pattern reveals that  $\Delta Y_t$  follows an autoregressive process, as the sample autocorrelation function decays quickly for the first few lags and probably revolves around zero for higher lag orders. Units are quarterly growth rates. Units are quarterly rates of growth as  $\Delta Y_t$  is the change in quarterly rate of growth.

ii.

```
usMacro_AR1 <- dynlm(ts(usMacro$deltaYt) ~ L(ts(usMacro$deltaYt, 1)))
coeftest(usMacro_AR1, vcoc.= sandwich, type = "NeweyWest")
```

```
##
## t test of coefficients:
##
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)      0.00578015 0.00083507   6.9218 6.202e-11 ***
## L(ts(usMacro$deltaYt, 1)) 0.28665557 0.06828590   4.1979 4.084e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## Estimated Coefficient: 0.287.
## Coefficient is significantly different from zero, check p-value: 4.08439441087776e-05.
Confidence Interval
```

```
print("Confidence Interval:")
```

```
## [1] "Confidence Interval:"
```

```
confint(usMacro_AR1, level = 0.95)
```

```
##              2.5 %      97.5 %
## (Intercept)      0.004133277 0.007427019
## L(ts(usMacro$deltaYt, 1)) 0.151986138 0.421325005
```

a)

```
usMacro_AR2 <- dynlm(ts(usMacro$deltaYt) ~ L(ts(usMacro$deltaYt, 1)) + L(ts(usMacro$deltaYt, 2)))
coeftest(usMacro_AR2, vcoc.= sandwich, type = "NeweyWest")
```

```
##
## t test of coefficients:
##
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)      0.00522454 0.00093246   5.6029 7.15e-08 ***
## L(ts(usMacro$deltaYt, 1)) 0.25789440 0.07145154   3.6094 0.0003906 ***
## L(ts(usMacro$deltaYt, 2)) 0.09512436 0.07131466   1.3339 0.1838109
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## Estimated Coefficient for AR(2): 0.095.
## Coefficient is not significantly different from zero, check p-value: 0.183810894155643.
## R^2-AR(1)=0.082; R^2-AR(2)=0.090.
```

Even though based on  $R^2$  the AR(2) model seems to explain the results better but only slightly. As AR2 adds one variable to the equation this makes sense. Let us look at the adjusted  $R^2$ :

```
##  $R^2$ -AR(1)=0.078;  $R^2$ -AR(2)0.080.
```

The better explanation decreases, but Adjusted  $R^2$  is still higher. However, this model appears not to be preferable to AR(1) still as the probability of AR(2) coefficient not being zero is not significant at 5%, 10% or 15% level.

b)

```
usMacro_AR3 <- dynlm(ts(usMacro$deltaYt) ~ L(ts(usMacro$deltaYt, 1)) +L(ts(usMacro$deltaYt, 2))+L(ts(usMacro$deltaYt, 3))
coeftest(usMacro_AR3, vcoc.= sandwich, type = "NeweyWest")
```

```
##
## t test of coefficients:
##
##              Estimate Std. Error t value  Pr(>|t|)
## (Intercept)      0.0054654   0.0010091   5.4162 1.807e-07 ***
## L(ts(usMacro$deltaYt, 1)) 0.2631944   0.0720366   3.6536 0.0003335 ***
## L(ts(usMacro$deltaYt, 2)) 0.1092100   0.0740621   1.4746 0.1419650
## L(ts(usMacro$deltaYt, 3)) -0.0457993   0.0718800  -0.6372 0.5247772
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
usMacro_AR4 <- dynlm(ts(usMacro$deltaYt) ~ L(ts(usMacro$deltaYt, 1)) +L(ts(usMacro$deltaYt, 2))+L(ts(usMacro$deltaYt, 3))+L(ts(usMacro$deltaYt, 4))
coeftest(usMacro_AR4, vcoc.= sandwich, type = "NeweyWest")
```

```
##
## t test of coefficients:
##
##              Estimate Std. Error t value  Pr(>|t|)
## (Intercept)      0.0056326   0.0010835   5.1986 5.162e-07 ***
## L(ts(usMacro$deltaYt, 1)) 0.2579150   0.0721520   3.5746 0.0004447 ***
## L(ts(usMacro$deltaYt, 2)) 0.1154370   0.0744821   1.5499 0.1228385
## L(ts(usMacro$deltaYt, 3)) -0.0329387   0.0744642  -0.4423 0.6587445
## L(ts(usMacro$deltaYt, 4)) -0.0265813   0.0719545  -0.3694 0.7122275
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
BIC <- function(model) {
  ssr <- sum(model$residuals^2)
  t <- length(model$residuals)
  npar <- length(model$coef)
  return(round((log(ssr/t) + npar * log(t)/t), 4))
}
BICVector <- c("AR1"=BIC(usMacro_AR1), "AR2"=BIC(usMacro_AR2), "AR3"= BIC(usMacro_AR3), "AR4"= BIC(usMacro_AR4))
BICVector
```

```
##      AR1      AR2      AR3      AR4
## -9.4319 -9.4095 -9.3809 -9.3593
```

```
message(paste("The Bayes information criterion (BIC) chooses: ", which.min(BICVector), " lag(s).", sep = " "))
```

```
## The Bayes information criterion (BIC) chooses: 1 lag(s).
```

```
AIC <- function(model) {
  ssr <- sum(model$residuals^2)
```

```

t <- length(model$residuals)
npar <- length(model$coef)
return(round((log(ssr/t) + npar * 2/t), 4))
}
AICVector <- c("AR1"=AIC(usMacro_AR1), "AR2"=AIC(usMacro_AR2), "AR3"= AIC(usMacro_AR3), "AR4"= AIC(usMacro_AR4))
AICVector

##      AR1      AR2      AR3      AR4
## -9.4651 -9.4595 -9.4478 -9.4432

message(paste("Akaike information criterion (AIC) choses: ", which.min(AICVector), " lag(s).", sep = ""))

## Akaike information criterion (AIC) choses: 1 lag(s).

```

iii.

```

adf.test(deltaYt)

## Warning in adf.test(deltaYt): p-value smaller than printed p-value
##
## Augmented Dickey-Fuller Test
##
## data:  deltaYt
## Dickey-Fuller = -5.6627, Lag order = 5, p-value = 0.01
## alternative hypothesis: stationary

```

The Augmented Dickey Fuller test yields a statistic of -5.6627 (<-3.43) and a p-value of 0.01 which is significant at the 1% level. Therefore we reject  $H_0$  that  $Y_t$  is an autoregressive root. In contrast,  $Y_t$  appears to be stationary around a deterministic trend.

### Question 3

a)

Using the Augmented Dickey Fuller critical values for intercept included regressions we cannot reject  $H_0$ :  $\beta=0$  for both the Canadian/US inflation rates. Therefore these variables are not stationary. Looking at the deltas of inflation rates it is observable that t statistics for US/Canada are both significant at the 1% level ( $|-5.24|, |-4.31| > 2.58$ ) and so  $H_0$ :  $\beta=0$  (unit root) is rejected in favour of  $H_1$ :  $\beta=1$ . Thus, both inflation rates are stationary.

b)

A Engle-Granger Augmented Dickey Fuller statistic of -7.34 being lower than -5.07 (1% significance for 4 variables) means that it is highly significant.  $H_0$ :  $\beta=0$  for the residuals can be rejected in favor of the alternate of cointegration and stationarity.

c)

Knowing that a cointegration coefficient exists by which calculated residuals are stationary, I would use a Dickey Fuller test with a cointegration coefficient of 1. If the Dickey Fuller test is significant we can reject  $H_0$ :  $\beta=0$  in favour of the residuals being stationary. Having this one can say that the cointegrating coefficient is 1 thus the US/Canada inflation rates are cointegrated.

d)

No a rejection of  $H_0$  of a unit cointegrating coefficient is not sufficient. It says in Stock & Watson that “cointegration tests can be misleading (they can improperly reject the null hypothesis of no cointegration more frequently than they should, and frequently they improperly fail to reject the null hypothesis”. Therefore main emphasis when estimating/using cointegrating relationships should be laid on economic theory, domain knowledge, and common sense.

## Question 4

Functions

```
MonteCarloSimulation <- function(T) {
  x <- w <- rnorm(n = T, mean = 0, sd = 1)
  for (t in 2:T) {
    x[t] <- x[t - 1] + w[t]
  }
  return (x)
}

MonteCarloRegression <- function(T) {
  MC1 <- MonteCarloSimulation(T)
  MC2 <- MonteCarloSimulation(T)
  MC_LM <- lm(MC1 ~ MC2)
  return (MC_LM)
}

MultipleMC <- function(T,N) {
  R2s<- vector()
  tvals <- vector()

  for (i in 1:N) {
    MC <- MonteCarloRegression(T)
    R2s[[i]]<-summary(MC)$r.squared
    tvals[[i]]<-summary(MC)$coefficients[2,3]
  }
  return(list(R2s,tvals))
}
```

a)

```
MC_regression <- MonteCarloRegression(1000)
summary(MC_regression)

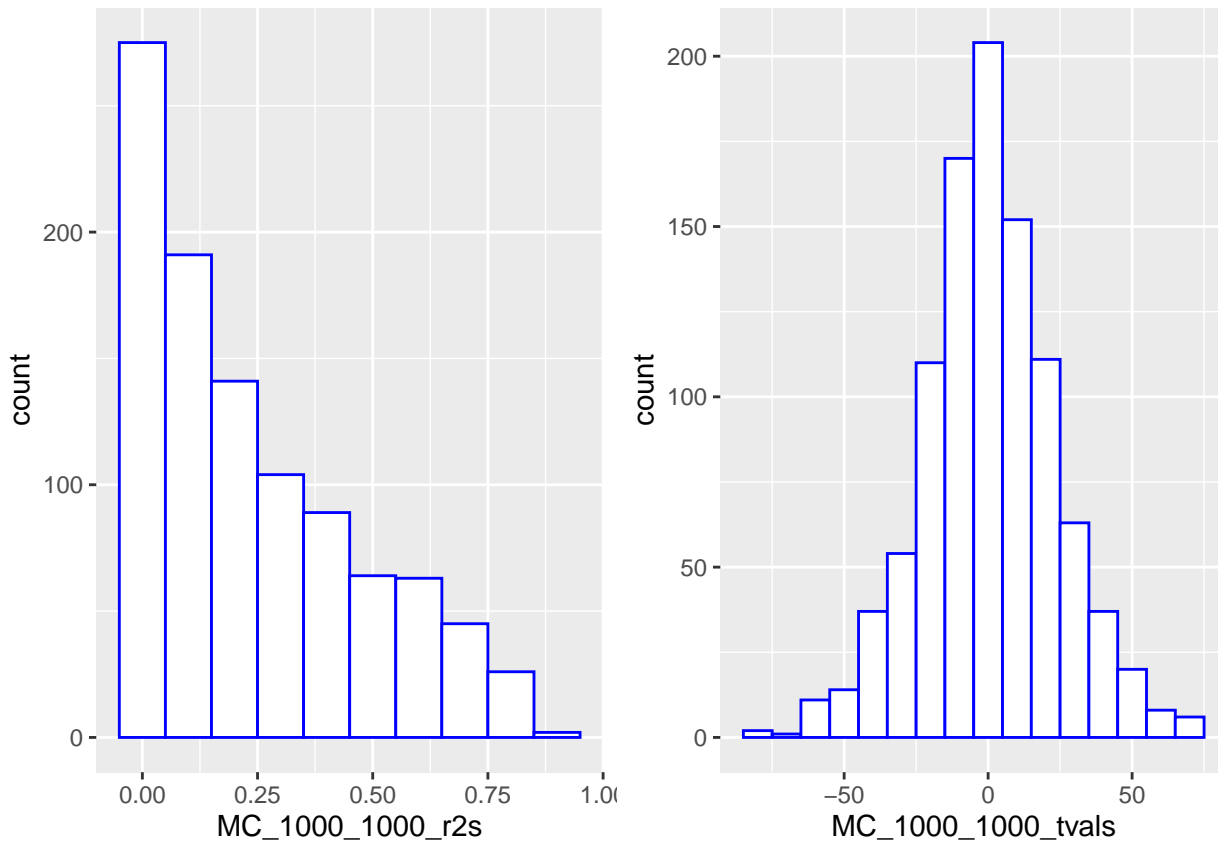
##
## Call:
## lm(formula = MC1 ~ MC2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -19.027  -8.027  -1.690   8.645  20.686
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept) -12.53261    0.31755 -39.466    <2e-16 ***
## MC2          -0.05768    0.03855  -1.496     0.135
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.947 on 998 degrees of freedom
## Multiple R-squared:  0.002238,    Adjusted R-squared:  0.001238
## F-statistic: 2.238 on 1 and 998 DF,  p-value: 0.1349

## The t-statistic for H0: b1=0 using 5% critical value of 1.96 is: -1.49615032740823.
## The Regression Rsquared is: 0.00223793213192266.
```

b)

```
MC_1000_1000 <- MultipleMC(T=1000,N=1000)
df <- data.frame(MC_1000_1000[1], MC_1000_1000[2])
names(df) <- c('MC_1000_1000_r2s', 'MC_1000_1000_tvals')
plot_ggplot_MC_1000_1000_r2s <- ggplot(df, aes(x=MC_1000_1000_r2s)) +
  geom_histogram(binwidth=.1, colour="blue", fill="white")
plot_ggplot_MC_1000_1000_tvals <- ggplot(df, aes(x=MC_1000_1000_tvals)) +
  geom_histogram(binwidth=10, colour="blue", fill="white")
grid.arrange(plot_ggplot_MC_1000_1000_r2s, plot_ggplot_MC_1000_1000_tvals, ncol=2)
```



Percentiles:  $R^2$ :

```
quantile(unlist(MC_1000_1000[1]), probs = c(0.05, 0.5, 0.95))
```

```
##           5%           50%           95%
```

```
## 0.0009626048 0.1693993988 0.6863663692
```

T-value:

```
quantile(unlist( MC_1000_1000[2]), probs = c(0.05, 0.5, 0.95))
```

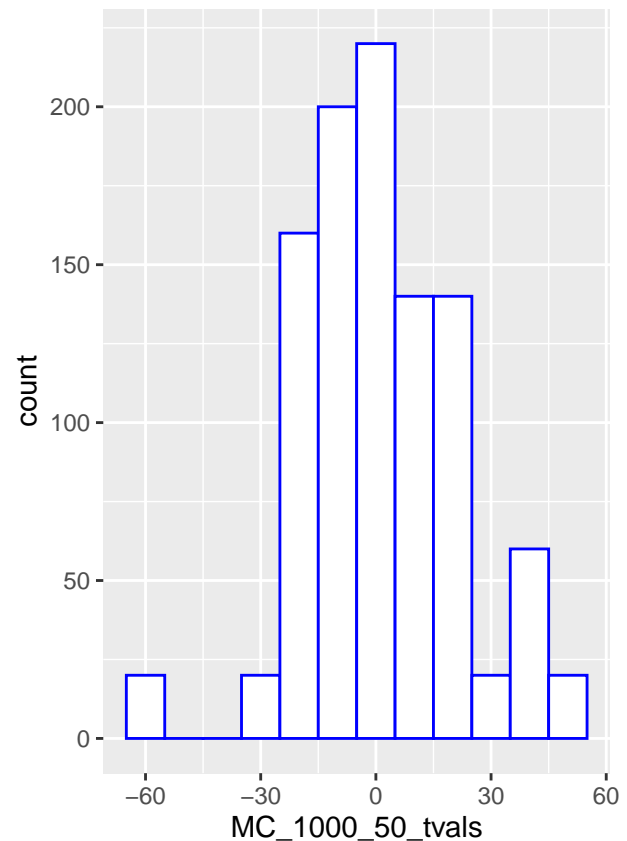
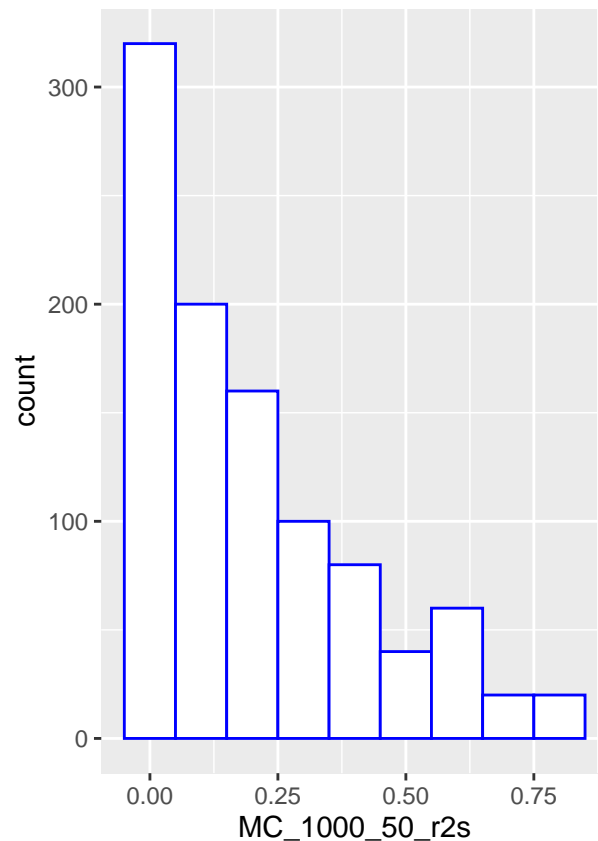
```
##          5%          50%          95%  
## -37.9525765 -0.1629394  39.0763067
```

```
message(paste("Fraction of t-statistic exceeding 1.96 for N=1000 is: ", (sum(abs(unlist( MC_1000_1000[2]
```

```
## Fraction of t-statistic exceeding 1.96 for N=1000 is: 0.911.
```

c)

```
MC_1000_50 <- MultipleMC(T=1000,N=50)  
df['MC_1000_50_r2s'] <- MC_1000_50[1]  
df['MC_1000_50_tvals'] <- MC_1000_50[2]  
#ggplot(df, aes(x=MC_1000_50_r2s)) + geom_histogram(binwidth=.1, colour="blue", fill="white")  
plot_ggplot_MC_1000_50_r2s <- ggplot(df, aes(x=MC_1000_50_r2s)) +  
  geom_histogram(binwidth=.1, colour="blue", fill="white")  
plot_ggplot_MC_1000_50_tvals <- ggplot(df, aes(x=MC_1000_50_tvals)) +  
  geom_histogram(binwidth=10, colour="blue", fill="white")  
grid.arrange(plot_ggplot_MC_1000_50_r2s, plot_ggplot_MC_1000_50_tvals, ncol=2)
```



Percentiles:  $R^2$ :

```
quantile(unlist(MC_1000_50[1]), probs = c(0.05, 0.5, 0.95))
```

```
##          5%          50%          95%
```



```
## 0.005214487 0.133336374 0.617831966
```

T-value:

```
quantile(unlist( MC_1000_50[2]), probs = c(0.05, 0.5, 0.95))
```

```
##          5%          50%          95%  
## -23.780136 -1.638225  37.266595
```

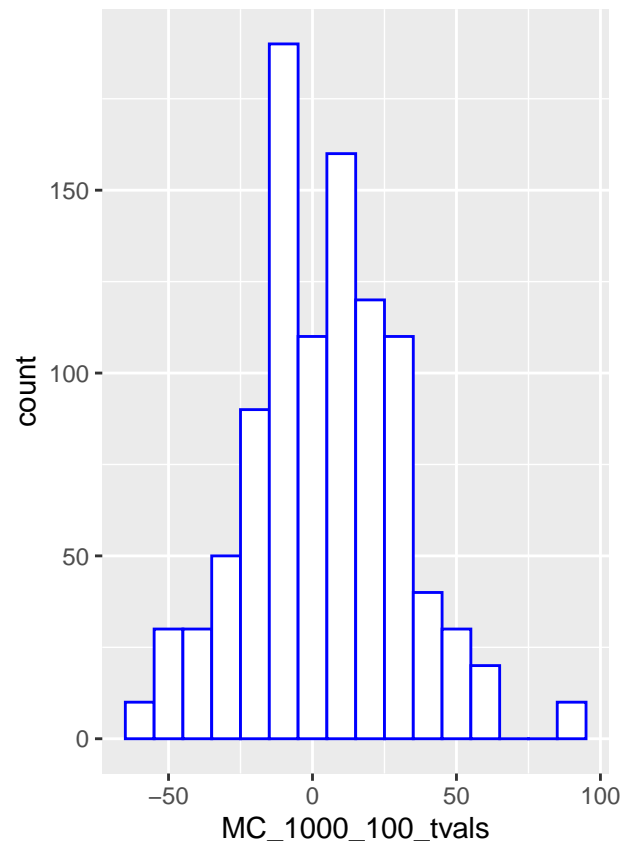
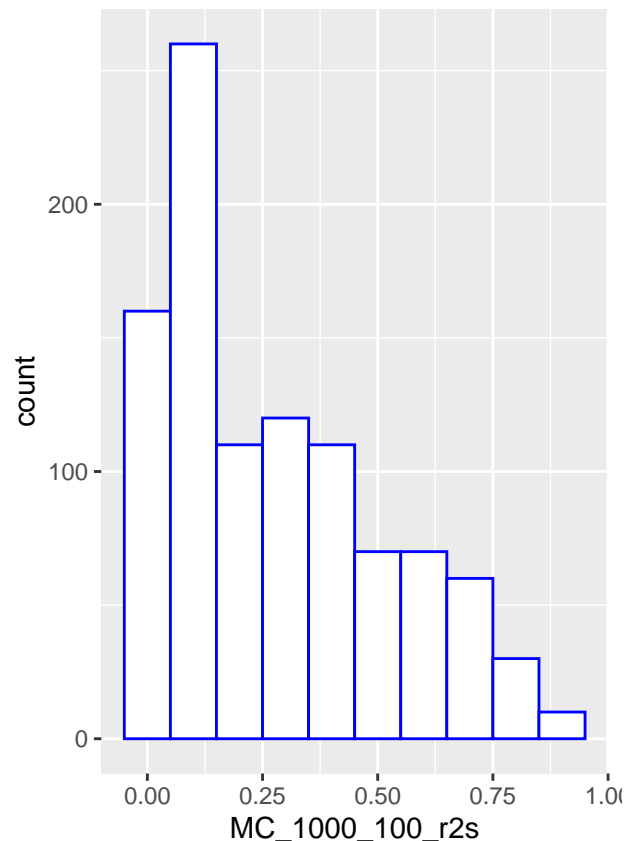
```
message(paste("Fraction of t-statistic exceeding 1.96 for N=50 is: ", (sum(abs(unlist( MC_1000_50[2])) > 1.96))/1000))
```

```
## Fraction of t-statistic exceeding 1.96 for N=50 is: 0.047.
```

```
message(paste("Fraction of t-statistic exceeding 1.96: ", (sum(abs(unlist( MC_1000_50[2])) > 1.96))/1000))
```

```
## Fraction of t-statistic exceeding 1.96: 0.047.
```

```
MC_1000_100 <- MultipleMC(T=1000,N=100)  
df['MC_1000_100_r2s'] <- MC_1000_100[1]  
df['MC_1000_100_tvals'] <- MC_1000_100[2]  
plot_ggplot_MC_1000_100_r2s <- ggplot(df, aes(x=MC_1000_100_r2s)) +  
  geom_histogram(binwidth=.1, colour="blue", fill="white")  
plot_ggplot_MC_1000_100_tvals <- ggplot(df, aes(x=MC_1000_100_tvals)) +  
  geom_histogram(binwidth=10, colour="blue", fill="white")  
grid.arrange(plot_ggplot_MC_1000_100_r2s, plot_ggplot_MC_1000_100_tvals, ncol=2)
```



Percentiles: R<sup>2</sup>:

```
quantile(unlist(MC_1000_100[1]), probs = c(0.05, 0.5, 0.95))
```

```
##          5%          50%          95%
```

```
## 0.01031976 0.20498429 0.69979274
```

T-value:

```
quantile(unlist( MC_1000_100[2]), probs = c(0.05, 0.5, 0.95))
```

```
##          5%          50%          95%  
## -39.542989   3.566781  47.615211
```

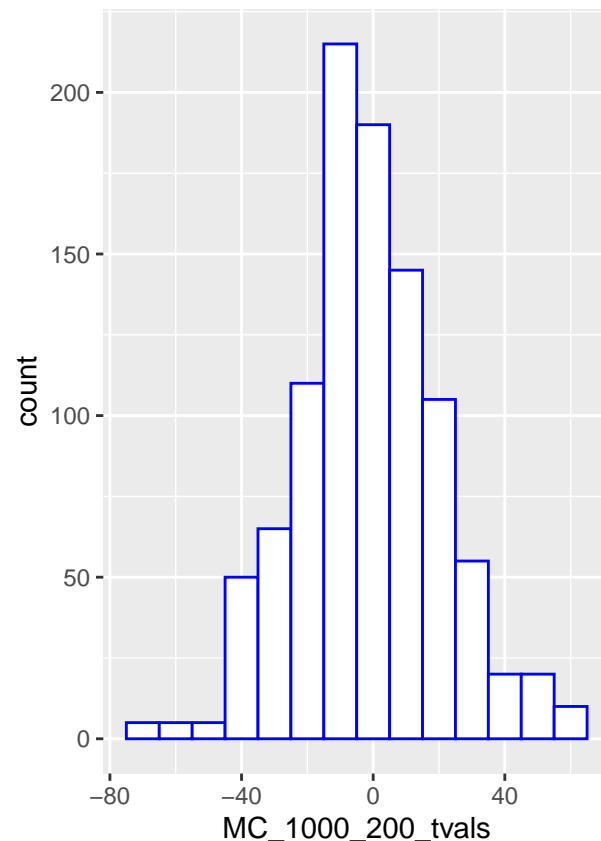
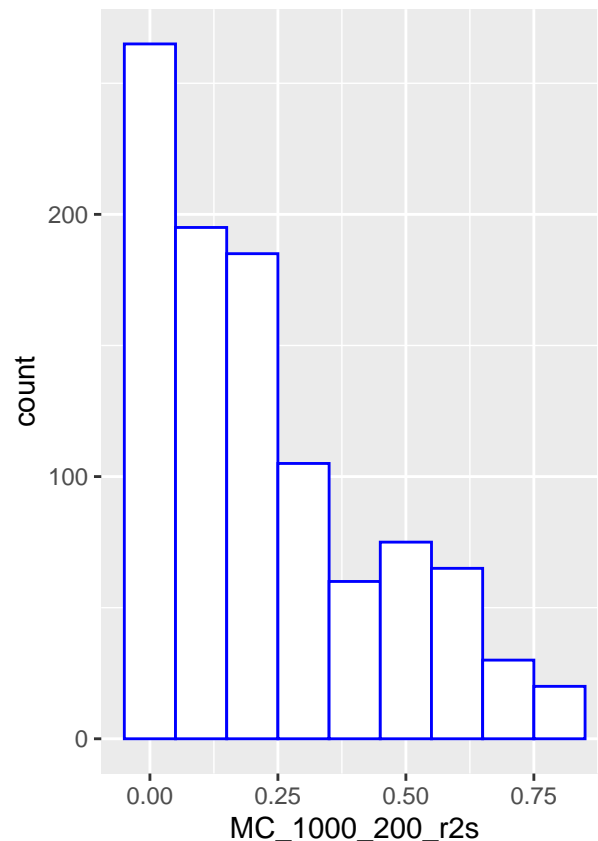
```
message(paste("Fraction of t-statistic exceeding 1.96 for N=100 is: ", (sum(abs(unlist( MC_1000_100[2]))
```

```
## Fraction of t-statistic exceeding 1.96 for N=100 is: 0.097.
```

```
message(paste("Fraction of t-statistic exceeding 1.96: ", (sum(abs(unlist( MC_1000_100[2])) > 1.96))/100
```

```
## Fraction of t-statistic exceeding 1.96: 0.097.
```

```
MC_1000_200 <- MultipleMC(T=1000,N=200)  
df['MC_1000_200_r2s'] <- MC_1000_200[1]  
df['MC_1000_200_tvals'] <- MC_1000_200[2]  
plot_ggplot_MC_1000_200_r2s <- ggplot(df, aes(x=MC_1000_200_r2s)) +  
  geom_histogram(binwidth=.1, colour="blue", fill="white")  
plot_ggplot_MC_1000_200_tvals <- ggplot(df, aes(x=MC_1000_200_tvals)) +  
  geom_histogram(binwidth=10, colour="blue", fill="white")  
grid.arrange(plot_ggplot_MC_1000_200_r2s, plot_ggplot_MC_1000_200_tvals, ncol=2)
```



Percentiles: R<sup>2</sup>:

```
quantile(unlist(MC_1000_200[1]), probs = c(0.05, 0.5, 0.95))
```

```
##          5%          50%          95%
```

```
## 0.00142579 0.16601148 0.64492004
```

T-value:

```
quantile(unlist( MC_1000_200[2]), probs = c(0.05, 0.5, 0.95))
```

```
##          5%          50%          95%  
## -36.772676 -1.319672  34.498351
```

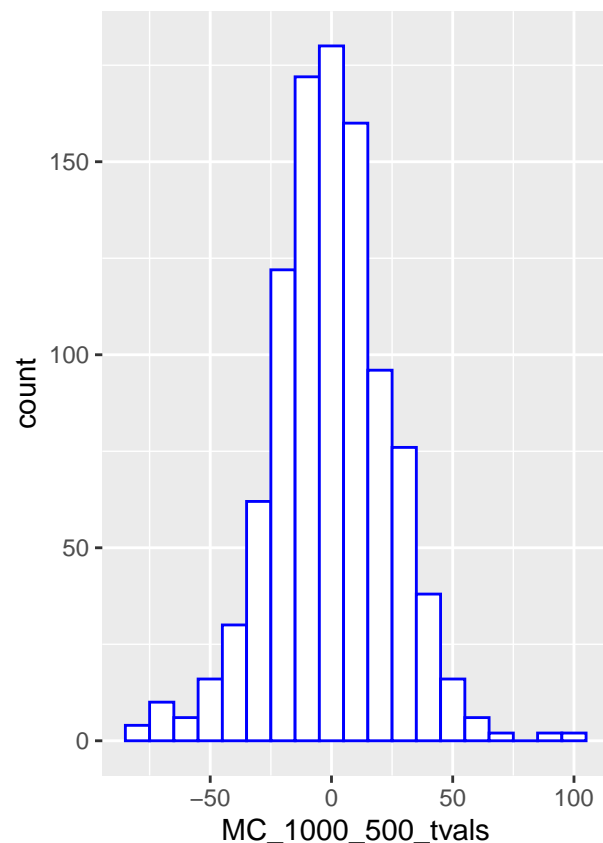
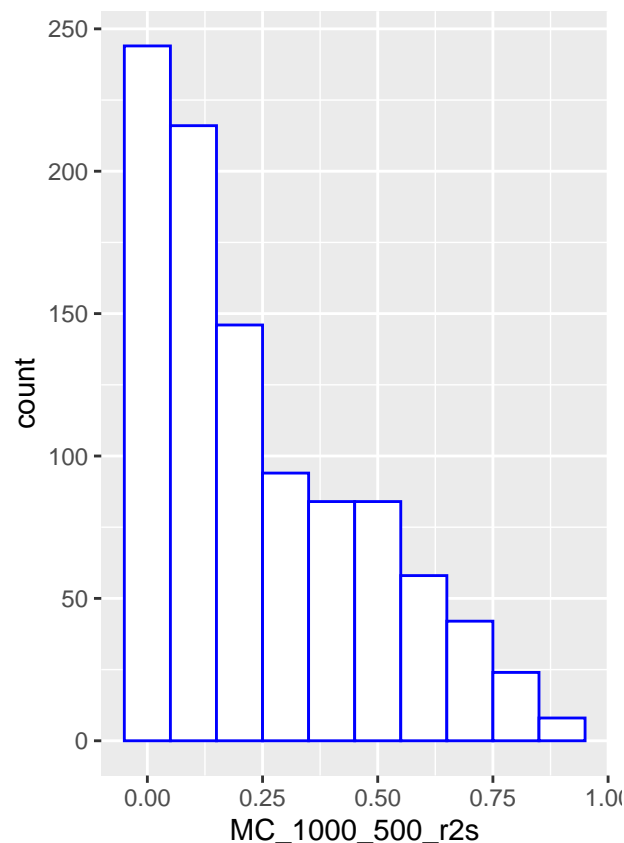
```
message(paste("Fraction of t-statistic exceeding 1.96 for N=200 is: ", (sum(abs(unlist( MC_1000_200[2])) > 1.96))/1000))
```

```
## Fraction of t-statistic exceeding 1.96 for N=200 is: 0.178.
```

```
message(paste("Fraction of t-statistic exceeding 1.96: ", (sum(abs(unlist( MC_1000_200[2])) > 1.96))/1000))
```

```
## Fraction of t-statistic exceeding 1.96: 0.178.
```

```
MC_1000_500 <- MultipleMC(T=1000,N=500)  
df['MC_1000_500_r2s'] <- MC_1000_500[1]  
df['MC_1000_500_tvals'] <- MC_1000_500[2]  
plot_ggplot_MC_1000_500_r2s <- ggplot(df, aes(x=MC_1000_500_r2s)) +  
  geom_histogram(binwidth=.1, colour="blue", fill="white")  
plot_ggplot_MC_1000_500_tvals <- ggplot(df, aes(x=MC_1000_500_tvals)) +  
  geom_histogram(binwidth=10, colour="blue", fill="white")  
grid.arrange(plot_ggplot_MC_1000_500_r2s, plot_ggplot_MC_1000_500_tvals, ncol=2)
```



Percentiles: R<sup>2</sup>:

```
quantile(unlist(MC_1000_500[1]), probs = c(0.05, 0.5, 0.95))
```

```
##          5%          50%          95%
```

```
## 0.001480676 0.174887532 0.701180250
```

T-value:

```
quantile(unlist( MC_1000_500[2]), probs = c(0.05, 0.5, 0.95))
```

```
##           5%           50%           95%  
## -40.4708911 -0.7684695  37.3546491
```

```
message(paste("Fraction of t-statistic exceeding 1.96 for N=500 is: ", (sum(abs(unlist( MC_1000_500[2]))
```

```
## Fraction of t-statistic exceeding 1.96 for N=500 is: 0.462.
```

```
message(paste("Fraction of t-statistic exceeding 1.96: ", (sum(abs(unlist( MC_1000_500[2])) > 1.96))/1000
```

```
## Fraction of t-statistic exceeding 1.96: 0.462.
```

With increasing sample size we reject the Null Hypothesis pretty much approaches 5%, when looking at the fraction being roughly around 50 of 1000. The t-statistic also looks more and more like a normal distribution the higher the T. Essentially this will continue for larger T, proven by the Central Limit Theorem, which says that when independent random variables are added, their properly normalized sum tends toward a normal distribution.