Project: STAT 525

KRISHNAN RAMAN, 2nd Year PhD. Statistics

12/10/2019

An exploration of trading strategies with Dow Jones components

The Dow Jones Index (DIA) is a weighted average of 30 stocks.
These 30 companies are some of the largest industrial giants in the US.
In this analysis, we explore the following questions:

- Does the log-normal assumption apply to Dow stock returns ?
- Predict the DIA using multiple linear regression, as a function of the 30 Dow components.
- Use Model Selection to systematically select the optimal number of covariates.
- Create a high 90% Rsquare Dow Regression model with fewest possible number of covariate stocks.
- Track the performance of the DIA versus the Dow Regression Model over a year
- 1-way Anova: Compare Diversification(Holding all 30 stocks) vs Holding a Single Stock
- Cell Means Model, Means Only Model, Effects Plot
- 1-way Anova: Compare annual returns across 10 different portfolios with 3 stocks in each portfolio
- 1-way Anova: Compare Buy & Hold Trading Strategy vs Buy Highest Sell Lowest vs Contrarian Strategy

Data

The Daily Holding Period returns for the Dow Jones Index (DIA) and its 30 stock components were sourced from **Wharton Research Data Services**, a subscription-only source of Financial Time Series. For each of the **30 stocks + DIA = 31 tickers**, we obtained data for all of 2018 (**There were 251 trading days in 2018**). That gives us **31x251 = 7781 rows** of daily returns. These are shown below:

```
rm(list=ls())
library(fitdistrplus)
library(ggplot2)
library(gtools)
library(leaps)
library(effects)

###########   CHANGE THIS LOCATION TO POINT TO THE SOURCE OF DATA ######
dowfile = "~/Desktop/525/project/dow.csv"

df = read.csv(dowfile)
head(df)
```

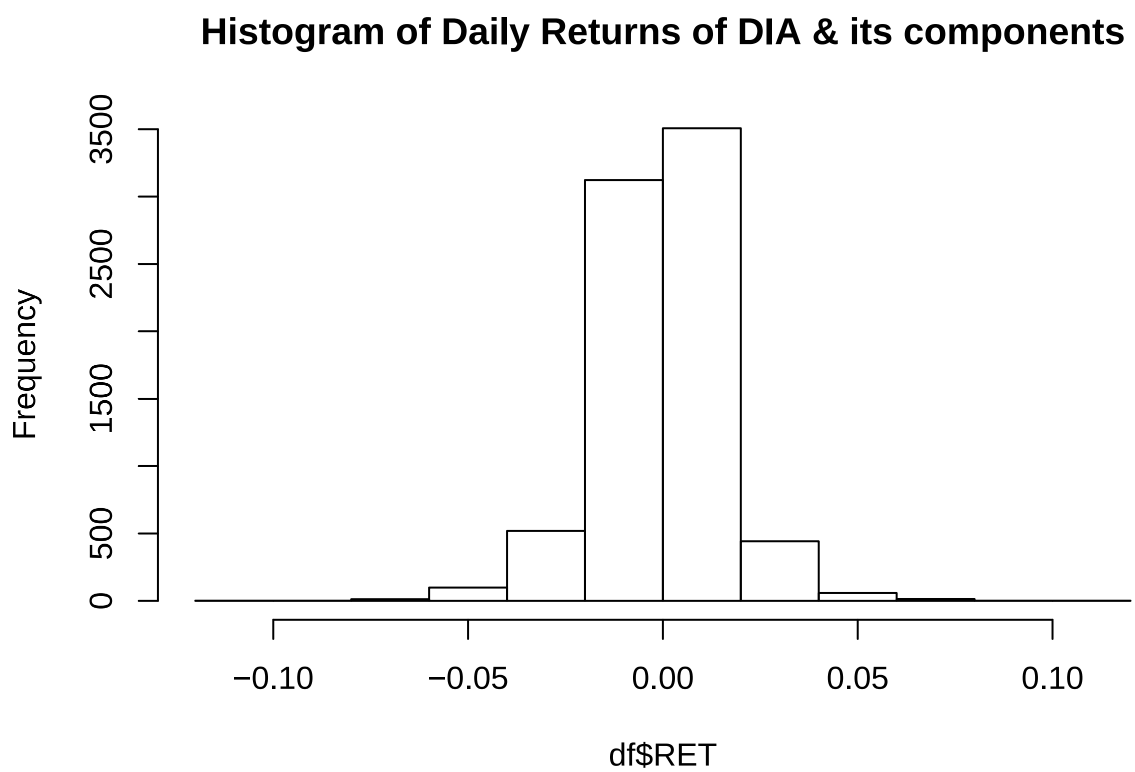
```
##    date TICKER       RET
## 1 20118   MSFT  0.004793
## 2 30118   MSFT  0.004654
## 3 40118   MSFT  0.008801
## 4 50118   MSFT  0.012398
## 5 80118   MSFT  0.001020
## 6 90118   MSFT -0.000680
```

```r
tail(df)
```

```
##        date TICKER       RET
## 7776 211218    DIA -0.018280
## 7777 241218    DIA -0.026730
## 7778 261218    DIA  0.048647
## 7779 271218    DIA  0.011149
## 7780 281218    DIA -0.003373
## 7781 311218    DIA  0.011801
```

The histogram of daily returns is firmly centered at 0. On a given day, the Dow components don't move that much.

```r
hist(df$RET, main="Histogram of Daily Returns of DIA & its components")
```



In fact, the max loss is -10.18 %, and the max gain 11.13 %, on the Dow components. The median daily gain is approx 0%.

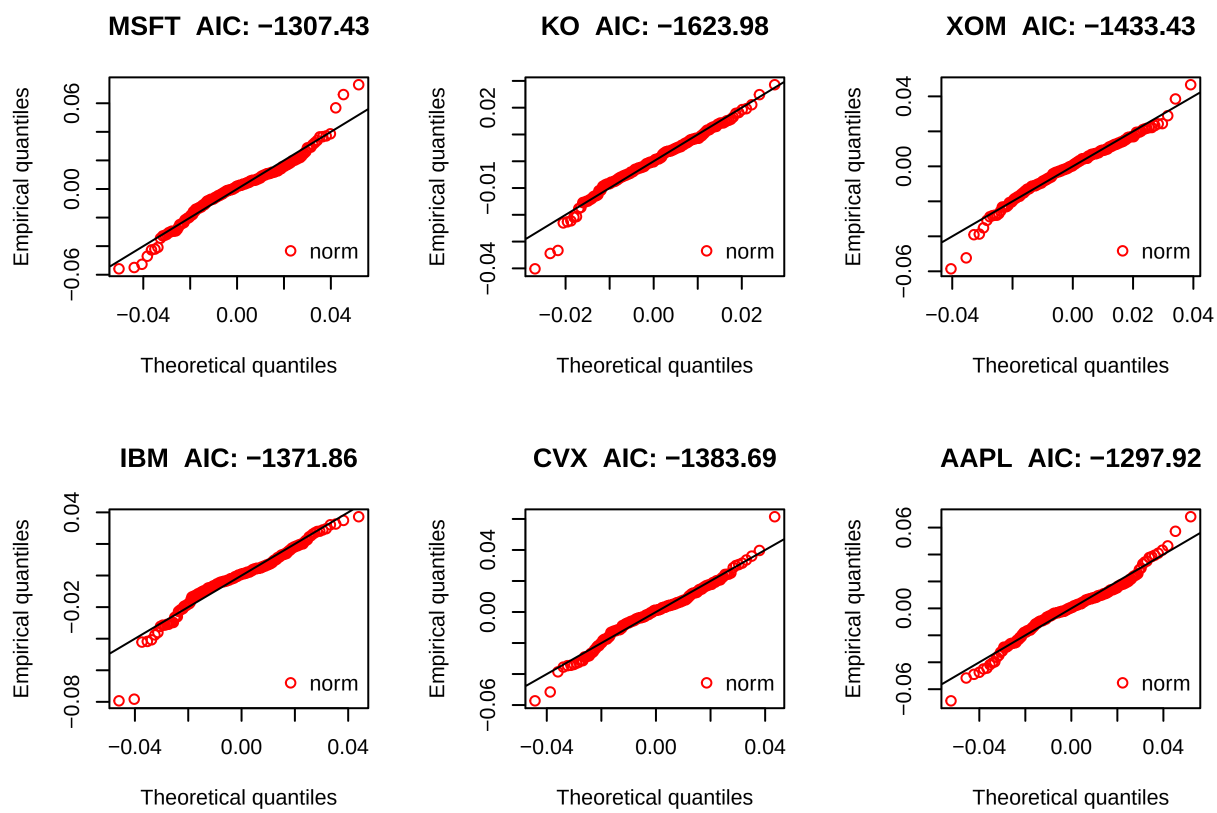
```r
summary(df$RET)
```

```
##       Min.    1st Qu.     Median       Mean    3rd Qu.       Max.
## -1.018e-01 -7.184e-03  5.840e-04  5.655e-05  8.283e-03  1.113e-01
```

LogNormality of Returns

We split the data by ticker. Financial returns are postulated to be log-normal ie. the log of returns is gaussian. We first fit a Normal distribution upon the log returns. We test the goodness of fit by visualizing the qq plot of six of the fitted distributions below.

```
tickerplot = function(ticker) {
  ret = log(1.0 + tickersplit[[ticker]]$RET )
  fitnorm<-fitdist(ret,"norm") # fit a normal distribution
  qqcomp(fitnorm, main=paste(ticker," AIC:",round(fitnorm$aic, 2)))
}

tickersplit = split(df, df$TICKER)
par(mfrow=c(2,3))
tickers = as.character(unique(df$TICKER))
tickers = tickers[tickers != "DIA"]
sapply(tickers[1:6], tickerplot)
```




```
## $MSFT
## NULL
##
## $KO
## NULL
##
## $XOM
```

```
## NULL
##
## $IBM
## NULL
##
## $CVX
## NULL
##
## $AAPL
## NULL
```

```
par(mfrow=c(1,1))
```

From the QQ plots above, we note that the lognormal assumption is violated in the tails.

Model Selection

In order to build a multiple linear regression model with Dow Jones (DIA) as the Response, and each of the 30 Dow stocks as the predictors, it will be convenient to construct a simple matrix with 31 columns & 251 rows. The first column DIA is the Response, the remaining 30 columns are the predictor stocks, and the 251 rows are the daily returns over the 2018 trading year. This process is shown below:

```
dow = matrix(0,nrow=251,ncol=31)
dow[,1]=tickersplit[["DIA"]]$RET
for(i in 2:31) {
  myticker = tickers[i-1]
  dow[,i] = tickersplit[[myticker]]$RET
}
colnames(dow) <- c("DIA", tickers)
dowdf = data.frame(dow)
dow[1:5,1:7]
```

```
##            DIA     MSFT        KO       XOM      IBM       CVX      AAPL
## [1,]  0.002587 0.004793 -0.007411  0.016619 0.005410  0.019091  0.017905
## [2,]  0.003750 0.004654 -0.002196  0.019640 0.027488  0.007289 -0.000174
## [3,]  0.006628 0.008801  0.014085  0.001384 0.020254 -0.003113  0.004645
## [4,]  0.008460 0.012398 -0.000217 -0.000806 0.004886 -0.001639  0.011385
## [5,] -0.000514 0.001020 -0.001519  0.004496 0.006031  0.004926 -0.003714
```

Now we are ready to perform Forward Stepwise Model Selection.

```
x = dow[,2:31]
y = dow[,1]
forward_varsec = summary(regsubsets(x=x,y=y,method="forward", nbest=1,nvmax=30, all.best=FALSE))
forward_varsec$outmat
```

```
##           MSFT KO  XOM IBM CVX AAPL UTX PG  CAT WBA BA  PFE JNJ MMM MRK
## 1  ( 1 )  " "  " " " " " " " " " "  " " " " " " " " " " " " " " "*" " "
## 2  ( 1 )  " "  " " " " " " " " " "  " " " " " " " " " " " " " " "*" " "
## 3  ( 1 )  " "  " " " " " " " " " "  " " " " " " " " "*" " " " " "*" " "
## 4  ( 1 )  " "  " " " " " " " " " "  " " " " " " " " "*" " " " " "*" " "
```

```
## 5  ( 1 )  " "  " " " " " " " " " "  " " " " " " " " "*" " " " " "*" " "
## 6  ( 1 )  " "  " " " " " " " " " "  " " " " " " " " "*" " " " " "*" " "
## 7  ( 1 )  " "  " " "*" " " " " " "  " " " " " " " " "*" " " " " "*" " "
## 8  ( 1 )  " "  " " "*" " " " " "*"  " " " " " " " " "*" " " " " "*" " "
## 9  ( 1 )  " "  " " "*" " " " " "*"  " " " " " " " " "*" " " " " "*" " "
## 10  ( 1 ) " "  " " "*" " " " " "*"  " " " " " " " " "*" " " " " "*" " "
## 11  ( 1 ) " "  " " "*" " " " " "*"  "*" " " " " " " "*" " " " " "*" " "
## 12  ( 1 ) " "  " " "*" " " " " "*"  "*" " " " " " " "*" " " " " "*" " "
## 13  ( 1 ) " "  " " "*" " " " " "*"  "*" " " " " " " "*" " " " " "*" " "
## 14  ( 1 ) " "  " " "*" " " " " "*"  "*" " " " " " " "*" " " " " "*" " "
## 15  ( 1 ) " "  " " "*" "*" " " "*"  "*" " " " " " " "*" " " " " "*" " "
## 16  ( 1 ) " "  " " "*" "*" " " "*"  "*" " " " " " " "*" " " "*" "*" " "
## 17  ( 1 ) " "  " " "*" "*" " " "*"  "*" " " "*" " " "*" " " "*" "*" " "
## 18  ( 1 ) "*"  " " "*" "*" " " "*"  "*" " " "*" " " "*" " " "*" "*" " "
## 19  ( 1 ) "*"  " " "*" "*" " " "*"  "*" "*" "*" " " "*" " " "*" "*" " "
## 20  ( 1 ) "*"  " " "*" "*" " " "*"  "*" "*" "*" " " "*" " " "*" "*" " "
## 21  ( 1 ) "*"  " " "*" "*" "*" "*"  "*" "*" "*" " " "*" " " "*" "*" " "
## 22  ( 1 ) "*"  " " "*" "*" "*" "*"  "*" "*" "*" " " "*" " " "*" "*" " "
## 23  ( 1 ) "*"  " " "*" "*" "*" "*"  "*" "*" "*" " " "*" " " "*" "*" " "
## 24  ( 1 ) "*"  " " "*" "*" "*" "*"  "*" "*" "*" " " "*" " " "*" "*" " "
## 25  ( 1 ) "*"  " " "*" "*" "*" "*"  "*" "*" "*" " " "*" " " "*" "*" "*"
## 26  ( 1 ) "*"  " " "*" "*" "*" "*"  "*" "*" "*" " " "*" " " "*" "*" "*"
## 27  ( 1 ) "*"  "*" "*" "*" "*" "*"  "*" "*" "*" " " "*" " " "*" "*" "*"
## 28  ( 1 ) "*"  "*" "*" "*" "*" "*"  "*" "*" "*" " " "*" "*" "*" "*" "*"
## 29  ( 1 ) "*"  "*" "*" "*" "*" "*"  "*" "*" "*" " " "*" "*" "*" "*" "*"
## 30  ( 1 ) "*"  "*" "*" "*" "*" "*"  "*" "*" "*" "*" "*" "*" "*" "*" "*"
##           DIS MCD JPM WMT NKE AXP INTC TRV VZ  HD  C   CSCO GS  V   UNH
## 1  ( 1 )  " " " " " " " " " " " " " "  " " " " " " " " " "  " " " " " "
## 2  ( 1 )  " " " " " " " " " " " " " "  " " " " " " " " " "  " " "*" " "
## 3  ( 1 )  " " " " " " " " " " " " " "  " " " " " " " " " "  " " "*" " "
## 4  ( 1 )  " " " " "*" " " " " " " " "  " " " " " " " " " "  " " "*" " "
## 5  ( 1 )  " " " " "*" " " " " " " " "  " " " " " " " " " "  " " "*" "*"
## 6  ( 1 )  " " " " "*" " " " " " " " "  " " " " "*" " " " "  " " "*" "*"
## 7  ( 1 )  " " " " "*" " " " " " " " "  " " " " "*" " " " "  " " "*" "*"
## 8  ( 1 )  " " " " "*" " " " " " " " "  " " " " "*" " " " "  " " "*" "*"
## 9  ( 1 )  " " " " "*" " " " " " " " "  " " "*" "*" " " " "  " " "*" "*"
## 10  ( 1 ) " " " " "*" "*" " " " " " "  " " "*" "*" " " " "  " " "*" "*"
## 11  ( 1 ) " " " " "*" "*" " " " " " "  " " "*" "*" " " " "  " " "*" "*"
## 12  ( 1 ) " " "*" "*" "*" " " " " " "  " " "*" "*" " " " "  " " "*" "*"
## 13  ( 1 ) " " "*" "*" "*" " " " " " "  " " "*" "*" " " " "  "*" "*" "*"
## 14  ( 1 ) " " "*" "*" "*" "*" " " " "  " " "*" "*" " " " "  "*" "*" "*"
## 15  ( 1 ) " " "*" "*" "*" "*" " " " "  " " "*" "*" " " " "  "*" "*" "*"
## 16  ( 1 ) " " "*" "*" "*" "*" " " " "  " " "*" "*" " " " "  "*" "*" "*"
## 17  ( 1 ) " " "*" "*" "*" "*" " " " "  " " "*" "*" " " " "  "*" "*" "*"
## 18  ( 1 ) " " "*" "*" "*" "*" " " " "  " " "*" "*" " " " "  "*" "*" "*"
## 19  ( 1 ) " " "*" "*" "*" "*" " " " "  " " "*" "*" " " " "  "*" "*" "*"
## 20  ( 1 ) " " "*" "*" "*" "*" " " " "  "*" "*" "*" " " " "  "*" "*" "*"
## 21  ( 1 ) " " "*" "*" "*" "*" " " " "  "*" "*" "*" " " " "  "*" "*" "*"
## 22  ( 1 ) "*" "*" "*" "*" "*" " " " "  "*" "*" "*" " " " "  "*" "*" "*"
## 23  ( 1 ) "*" "*" "*" "*" "*" "*" " "  "*" "*" "*" " " " "  "*" "*" "*"
## 24  ( 1 ) "*" "*" "*" "*" "*" "*" "*"  "*" "*" "*" " " " "  "*" "*" "*"
## 25  ( 1 ) "*" "*" "*" "*" "*" "*" "*"  "*" "*" "*" " " " "  "*" "*" "*"
## 26  ( 1 ) "*" "*" "*" "*" "*" "*" "*"  "*" "*" "*" " " "*"  "*" "*" "*"
## 27  ( 1 ) "*" "*" "*" "*" "*" "*" "*"  "*" "*" "*" " " "*"  "*" "*" "*"
```

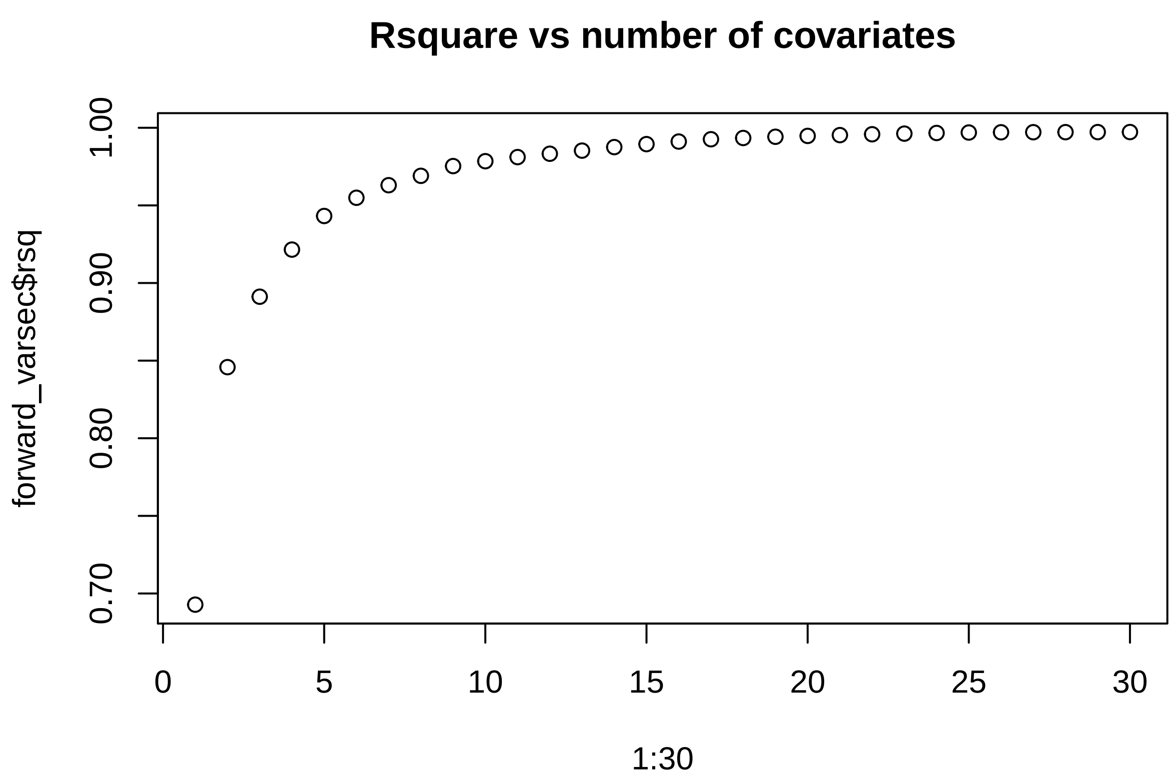
```
## 28  ( 1 ) "*" "*" "*" "*" "*" "*" "*"  "*" "*" "*" " " "*"  "*" "*" "*"
## 29  ( 1 ) "*" "*" "*" "*" "*" "*" "*"  "*" "*" "*" "*" "*"  "*" "*" "*"
## 30  ( 1 ) "*" "*" "*" "*" "*" "*" "*"  "*" "*" "*" "*" "*"  "*" "*" "*"
```

It is clear that a single company 3M (MMM) alone is a good proxy for the Dow.
From row 5 above, **the most variation in the Dow Jones Index DIA is explained by the top 5 companies: 3M, Boeing(BA), JP Morgan (JPM), Visa (V) & United Healthcare (UNH)** From row 29,30 above, the company that explains the least variation in the Dow is Walgreens Boots Alliance (WBA).
Similar conclusions may be drawn from Backward Selection.

```r
#backward_varsec = summary(regsubsets(x=x,y=y,method="backward", nbest=1,nvmax=30,all.best=FALSE))
#backward_varsec$outmat
```

To visualize the explanatory power of individual covariates, let us plot Rsquare as more & more covariates are added to the model.

```r
plot(1:30, forward_varsec$rsq, main="Rsquare vs number of covariates")
```



Create a high 90% Rsquare Dow Regression model with fewest possible number of covariate stocks.

From the above plot, we obtain about 90% Rsq from just the top 3 stocks. Lets now build a multiple linear regression model to predict the Dow Jones Returns (DIA), using just the top 3 stocks: 3M (MMM), Boeing (BA) & Visa (V)

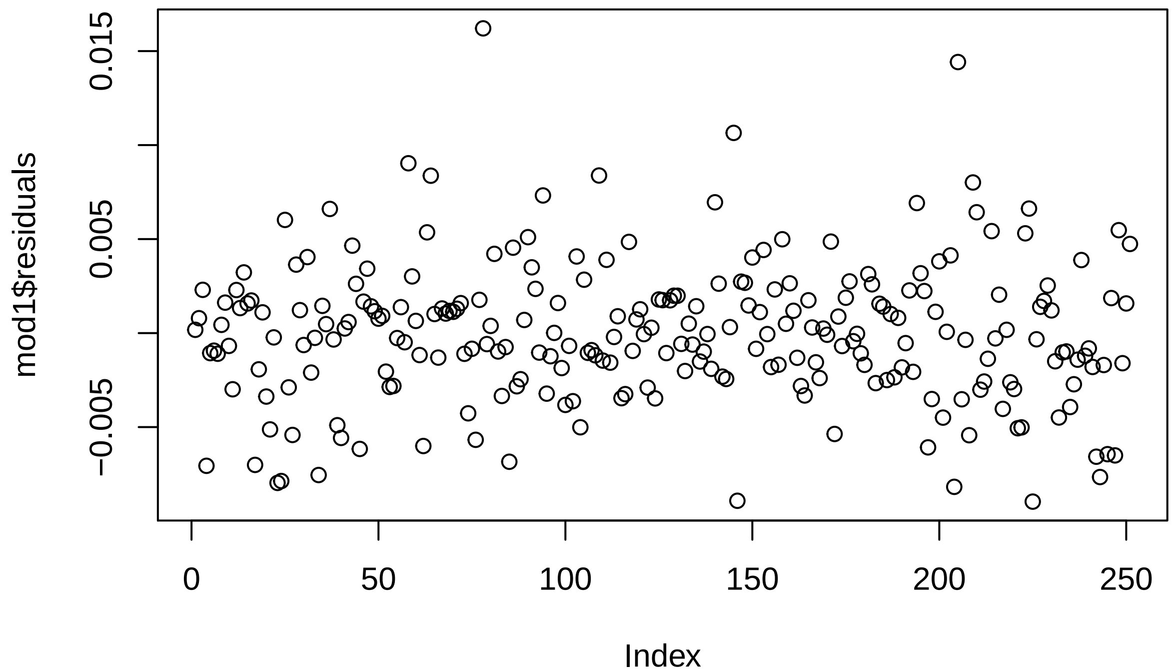
```r
# from the 3 best (forward selection) predictors
mod1 = lm(dow[,"DIA"]~dow[,"MMM"]+dow[,"BA"]+dow[,"V"])
summary(mod1)
```

```
##
## Call:
## lm(formula = dow[, "DIA"] ~ dow[, "MMM"] + dow[, "BA"] + dow[,
##     "V"])
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -0.0089642 -0.0020556 -0.0000545  0.0017604  0.0162088
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -0.0001877  0.0002379  -0.789    0.431
## dow[, "MMM"]  0.3198295  0.0213530  14.978   <2e-16 ***
## dow[, "BA"]   0.1689630  0.0166476  10.149   <2e-16 ***
## dow[, "V"]    0.2624178  0.0206945  12.681   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.003748 on 247 degrees of freedom
## Multiple R-squared:  0.8912, Adjusted R-squared:  0.8899
## F-statistic: 674.3 on 3 and 247 DF,  p-value: < 2.2e-16
```

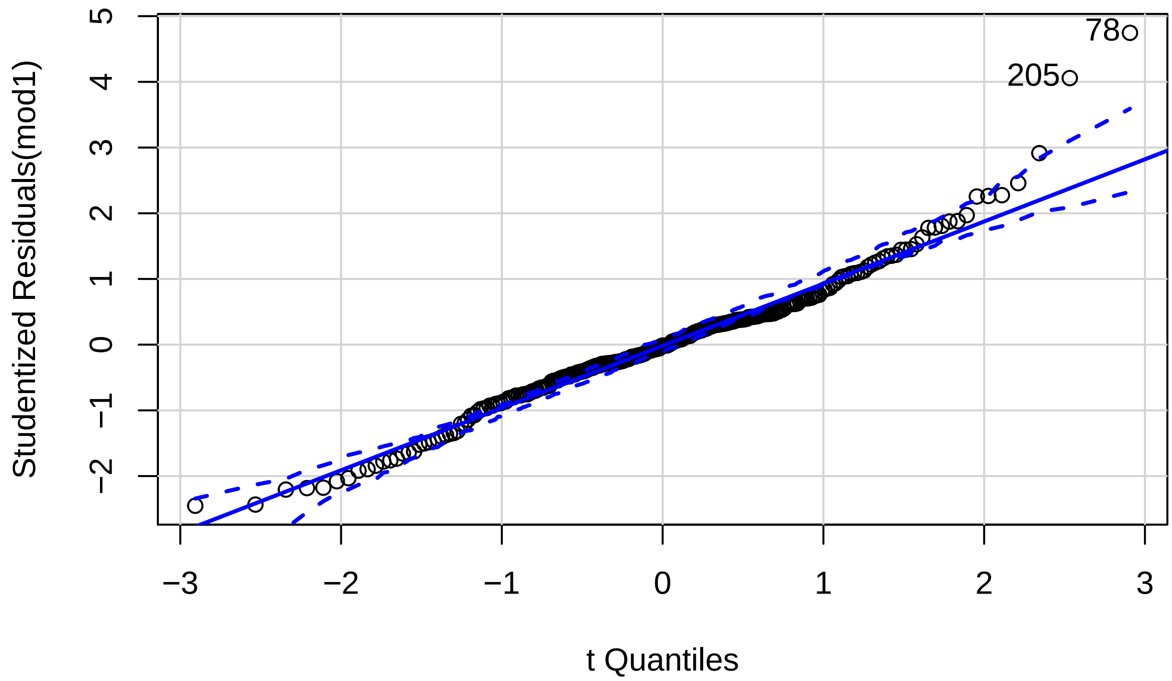
```r
anova(mod1)
```

```
## Analysis of Variance Table
##
## Response: dow[, "DIA"]
##               Df    Sum Sq   Mean Sq F value    Pr(>F)
## dow[, "MMM"]   1 0.0220972 0.0220972 1572.67 < 2.2e-16 ***
## dow[, "BA"]    1 0.0040676 0.0040676  289.49 < 2.2e-16 ***
## dow[, "V"]     1 0.0022593 0.0022593  160.80 < 2.2e-16 ***
## Residuals    247 0.0034705 0.0000141
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
#residual plot & diagnostics ( lecture notes Chp 11)
plot(mod1$residuals)
```




```r
library(car)
qqPlot(mod1)
```



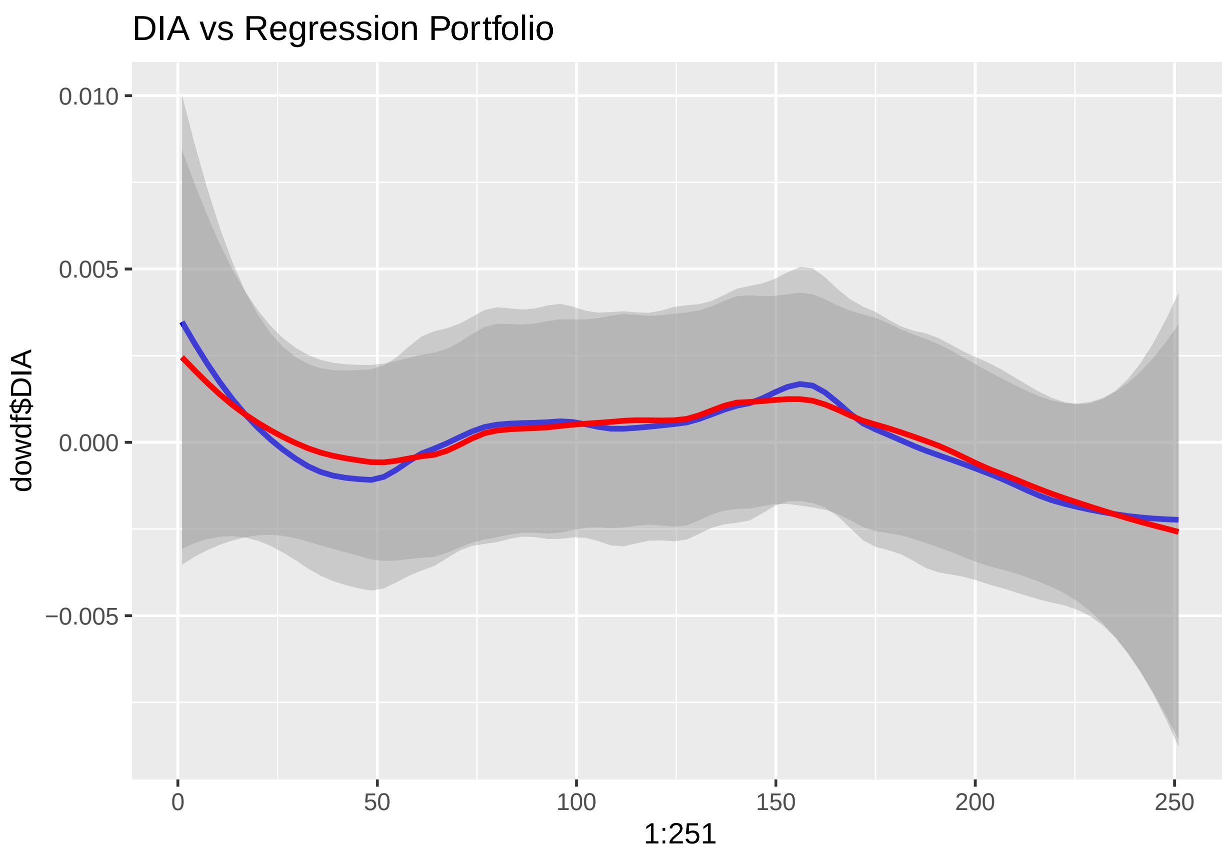

```
## [1]  78 205
```

As expected, we obtain 89% Rsquare from a linear model with just 3 covariates. The linear model's F statistic is highly significant so the model is a good fit. Also, each of the 3 covariates have a highly significant t statistic.Finally, the residual plot does not show any pattern or overdispersion. While the QQ plot does show 2 outliers (78,205) that may have outsize leverage, the normality assummption of the residual errors holds.

Track the performance of the DIA versus the Dow Regression Model over a year

We construct a portfolio with similar returns as the Dow, using the betas (coefficients) from the regression above. We track the performance of this portfolio versus the Dow over 1 year.

```
myportfolio = 0.32*dow[, "MMM"] + 0.17*dow[, "BA"]  + 0.26*dow[, "V"]
dowdf$myportfolio = myportfolio
ggplot(dowdf, aes(x=1:251, y=dowdf$DIA)) + geom_smooth(method="loess", span = 0.5, color="blue") + geom_
```



So we see the Dow in blue tracks the returns of myportfolio in red very closely. We are able to visualize them apart only because the 2 loess curves have different spans (In fact, if we match the span of the 2 loess curves, we cannot distinguish beween the 2 curves!)

1-way Anova: Compare Diversification(Holding all 30 stocks) vs Holding a Single Stock

Assume the first investor buys just 1 stock, MSFT. Whereas the second investor diversifies i.e. distributes money among all 30 Dow stocks ie. buys the DIA index. They both hold the instrument for 6 months. Lets compare their returns to see if there is a statistically significant difference.

```
#Nondiversification vs Diversification
n = 251-180+1 #6 months = 180 days. We have 251 trading days in 2018, so there are n such "6 month peri
returns = matrix(0, nrow=2*n, ncol=2)
colnames(returns) <- c("Returns", "Portfolio")

for(i in 1:n) {
  returns[i,1] = sum(dow[, "MSFT"][i:(180+i)])
  returns[i,2] = 1 # factor #1
}
for(i in (n+1):(2*n)) {
  y = i-n
  returns[i,1] = sum(dow[, "DIA"][y:(180+y)])
  returns[i,2] = 2 #factor #2
}

returns = data.frame(returns)
```

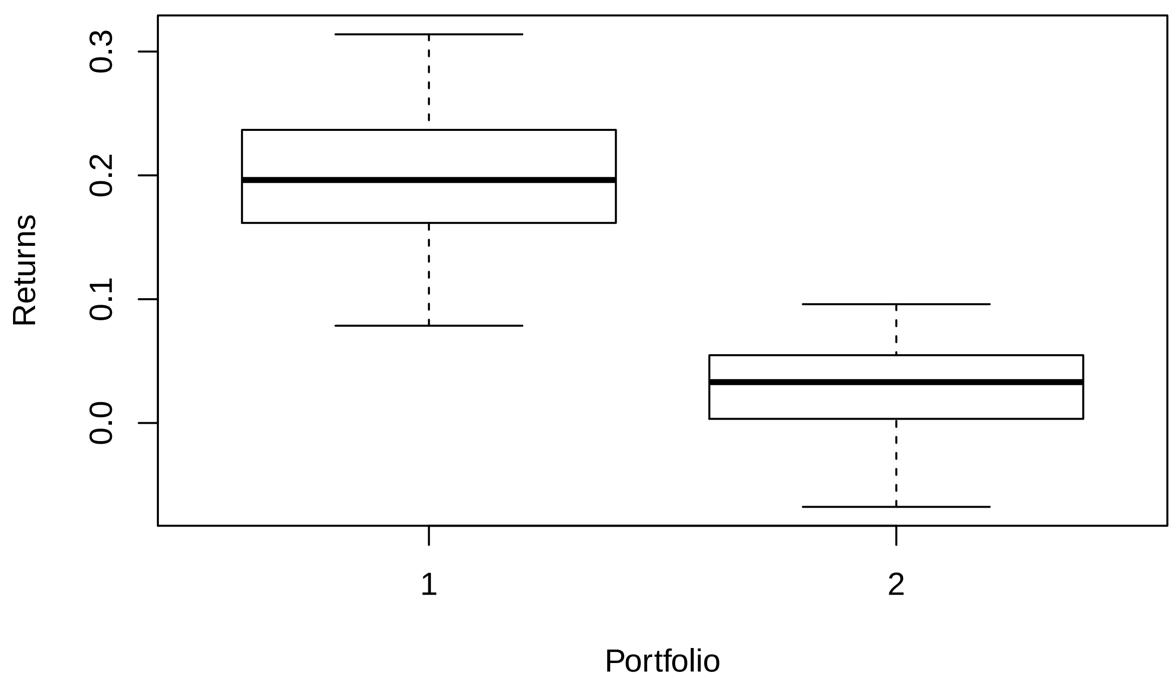
```r
returns$Returns = as.double(as.character( returns$Returns ))
returns$Portfolio = factor( returns$Portfolio )
res.aov <- aov(Returns~Portfolio, data = returns)
summary(res.aov)
```

```
##              Df Sum Sq Mean Sq F value Pr(>F)
## Portfolio     1 1.0673  1.0673   464.9 <2e-16 ***
## Residuals   140 0.3214  0.0023
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 2 observations deleted due to missingness
```

```r
TukeyHSD(res.aov)
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = Returns ~ Portfolio, data = returns)
##
## $Portfolio
##           diff        lwr        upr p adj
## 2-1 -0.1733915 -0.1892907 -0.1574922     0
```

```r
boxplot(Returns~Portfolio, data = returns)
```



```r
# Tukey vs Welch's t test since homogenaity of variance is not met
```

From the p-value on the Tukey Test, we conclude there is a significant difference in the returns of the Dow versus holding just an individual Dow component. The Dow offers 17.3% lower returns on average than MSFT, over a 6 month period in 2018. But it is clear from the Boxplots that the homogenaiety of variance is not met. So we can perform a Welch's T test to confirm the Tukey Results.

```r
t.test(returns$Returns[returns$Portfolio=="1"],returns$Returns[returns$Portfolio=="2"])
```

```
##
##  Welch Two Sample t-test
##
## data:  returns$Returns[returns$Portfolio == "1"] and returns$Returns[returns$Portfolio == "2"]
## t = 21.561, df = 116.94, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.1574648 0.1893181
## sample estimates:
##  mean of x  mean of y
## 0.20129204 0.02790056
```

Once again, the T test p-value (slightly higher p-value than Tukey, since Tukey uses a common variance for the 2 groups) shows significant difference in the mean returns (Equivalently, the confidence interval of mean differences does NOT include zero)

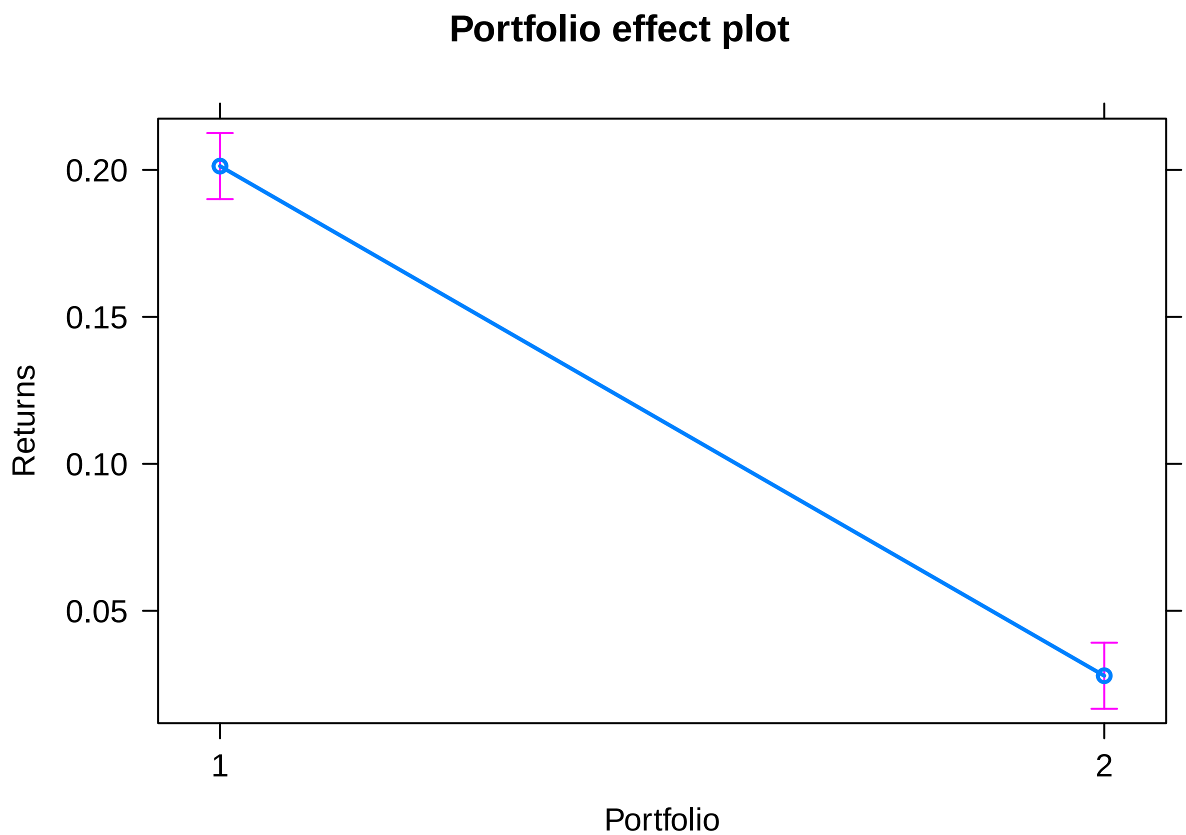
Cell Means Model, Means Only Model, Effects Plot

We obtain the Cell Means model. We also obtain the regular linear model with intercept & compare it versus the mean-only model. Further, we examine an effects plot for the 2 strategies.

```r
lm1 = lm(returns$Returns~returns$Portfolio - 1) # Cell means model
summary(lm1)
```

```
##
## Call:
## lm(formula = returns$Returns ~ returns$Portfolio - 1)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.122717 -0.030990 -0.000133  0.029740  0.112617
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## returns$Portfolio1 0.201292   0.005686  35.398  < 2e-16 ***
## returns$Portfolio2 0.027901   0.005686   4.906 2.54e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.04792 on 140 degrees of freedom
##   (2 observations deleted due to missingness)
## Multiple R-squared:  0.9012, Adjusted R-squared:  0.8998
## F-statistic: 638.6 on 2 and 140 DF,  p-value: < 2.2e-16
```

```r
lm2 = lm(Returns~Portfolio,data=returns) # Regular linear model
plot(allEffects(lm2))
```




```r
lm3 <- lm(Returns~1,data=returns) # mean only model
summary(lm3)
```

```
##
## Call:
## lm(formula = Returns ~ 1, data = returns)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.18234 -0.08153 -0.01935  0.08111  0.19931
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 0.114596   0.008328   13.76   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.09924 on 141 degrees of freedom
##   (2 observations deleted due to missingness)
```

```
anova(lm2,lm3)
```

```
## Analysis of Variance Table
##
## Model 1: Returns ~ Portfolio
## Model 2: Returns ~ 1
##   Res.Df     RSS Df Sum of Sq      F    Pr(>F)
## 1    140 0.32142
## 2    141 1.38871 -1   -1.0673 464.88 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

1-way Anova: Compare annual returns across 10 different portfolios with 3 stocks in each portfolio

Lets construct 10 combinations of 3 Dow stocks, and see if the mean returns are statistically different over a 6 month duration. Each portfolio is a linear combination of 3 different Dow stocks.

```
# 1-way anova lect notes 12
n = 251-180+1 #6 months = 180 days. We have 251 trading days in 2018, so there are n such "6 month peri
nc3 = 10 # 5 choose 3 = 10
threecomb = combinations(5,3,tickers)
returns = matrix(0, nrow=(n*nc3), ncol=2)
colnames(returns) <- c("Returns", "Portfolio")

s = 1
for(t in 1:nc3) {
  threetickers = threecomb[t,]
  ticker1 = threetickers[1]
  ticker2 = threetickers[2]
  ticker3 = threetickers[3]
  for(i in 1:n) {
    # portfolio is simple linear combination of 3 tickers
    portfolio = sum(dow[, ticker1][i:(180+i)]) + sum(dow[, ticker2][i:(180+i)]) + sum(dow[, ticker3][i:

    returns[s,1] = portfolio/3 #one third of each stock
    returns[s,2] = t # t is the factor of the t-th portfolio
    s = s + 1
  }
}

returns = data.frame(returns)
returns$Returns = as.double(as.character( returns$Returns ))
returns$Portfolio = factor( returns$Portfolio )
res.aov <- aov(Returns~Portfolio, data = returns)
summary(res.aov)
```

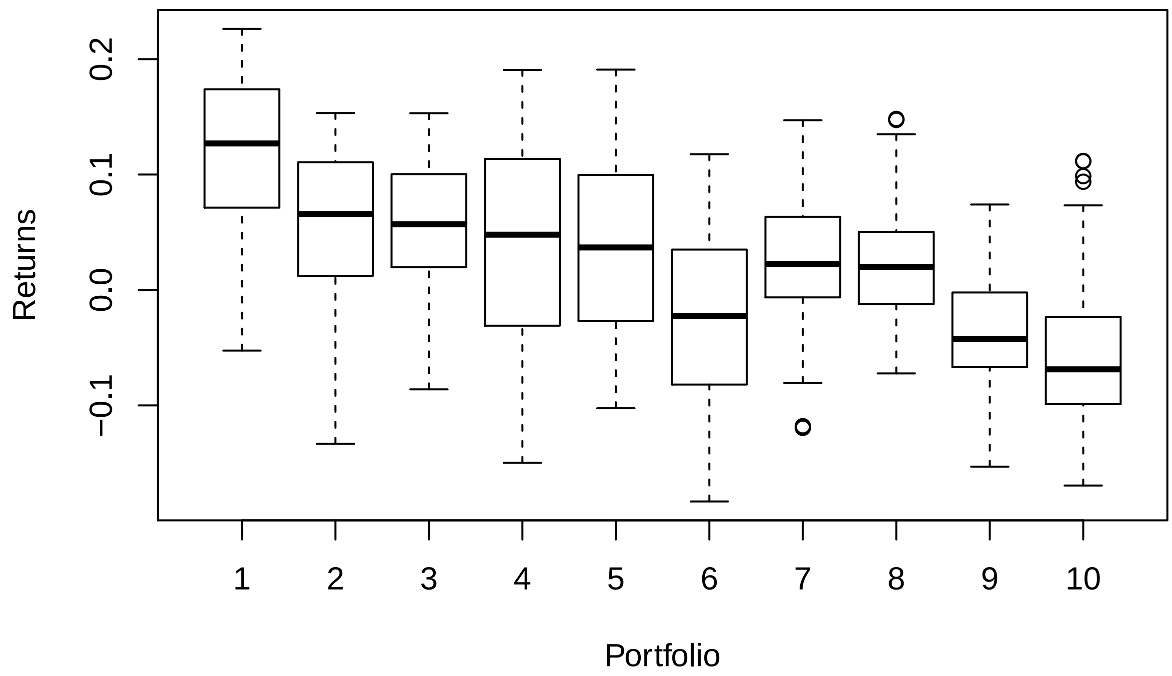
```
##              Df Sum Sq Mean Sq F value Pr(>F)
## Portfolio     9  1.632 0.18131   39.75 <2e-16 ***
## Residuals   700  3.193 0.00456
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 10 observations deleted due to missingness
```

```r
TukeyHSD(res.aov)
```

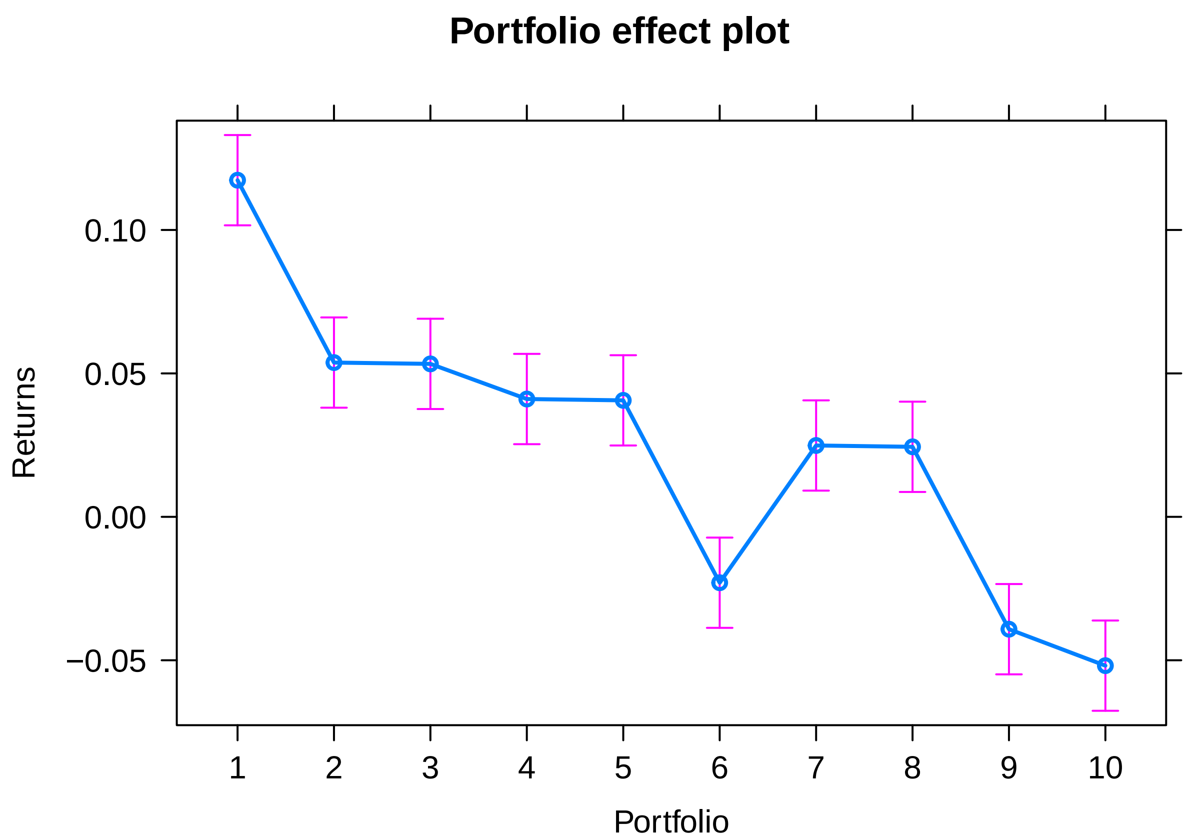
```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = Returns ~ Portfolio, data = returns)
##
## $Portfolio
##                diff         lwr          upr     p adj
## 2-1  -0.0635758169 -0.09955162 -0.027600016 0.0000013
## 3-1  -0.0640349390 -0.10001074 -0.028059139 0.0000010
## 4-1  -0.0762889296 -0.11226473 -0.040313129 0.0000000
## 5-1  -0.0767480516 -0.11272385 -0.040772251 0.0000000
## 6-1  -0.1403238685 -0.17629967 -0.104348068 0.0000000
## 7-1  -0.0924847465 -0.12846055 -0.056508946 0.0000000
## 8-1  -0.0929438685 -0.12891967 -0.056968068 0.0000000
## 9-1  -0.1565196854 -0.19249549 -0.120543885 0.0000000
## 10-1 -0.1692327981 -0.20520860 -0.133256998 0.0000000
## 3-2  -0.0004591221 -0.03643492  0.035516678 1.0000000
## 4-2  -0.0127131127 -0.04868891  0.023262688 0.9824575
## 5-2  -0.0131722347 -0.04914804  0.022803566 0.9776485
## 6-2  -0.0767480516 -0.11272385 -0.040772251 0.0000000
## 7-2  -0.0289089296 -0.06488473  0.007066871 0.2437194
## 8-2  -0.0293680516 -0.06534385  0.006607749 0.2237579
## 9-2  -0.0929438685 -0.12891967 -0.056968068 0.0000000
## 10-2 -0.1056569812 -0.14163278 -0.069681181 0.0000000
## 4-3  -0.0122539906 -0.04822979  0.023721810 0.9864232
## 5-3  -0.0127131127 -0.04868891  0.023262688 0.9824575
## 6-3  -0.0762889296 -0.11226473 -0.040313129 0.0000000
## 7-3  -0.0284498075 -0.06442561  0.007525993 0.2647823
## 8-3  -0.0289089296 -0.06488473  0.007066871 0.2437194
## 9-3  -0.0924847465 -0.12846055 -0.056508946 0.0000000
## 10-3 -0.1051978592 -0.14117366 -0.069222059 0.0000000
## 5-4  -0.0004591221 -0.03643492  0.035516678 1.0000000
## 6-4  -0.0640349390 -0.10001074 -0.028059139 0.0000010
## 7-4  -0.0161958169 -0.05217162  0.019779984 0.9179805
## 8-4  -0.0166549390 -0.05263074  0.019320861 0.9038212
## 9-4  -0.0802307559 -0.11620656 -0.044254955 0.0000000
## 10-4 -0.0929438685 -0.12891967 -0.056968068 0.0000000
## 6-5  -0.0635758169 -0.09955162 -0.027600016 0.0000013
## 7-5  -0.0157366948 -0.05171250  0.020239106 0.9306861
## 8-5  -0.0161958169 -0.05217162  0.019779984 0.9179805
## 9-5  -0.0797716338 -0.11574743 -0.043795833 0.0000000
## 10-5 -0.0924847465 -0.12846055 -0.056508946 0.0000000
## 7-6   0.0478391221  0.01186332  0.083814922 0.0011381
## 8-6   0.0473800000  0.01140420  0.083355800 0.0013480
## 9-6  -0.0161958169 -0.05217162  0.019779984 0.9179805
## 10-6 -0.0289089296 -0.06488473  0.007066871 0.2437194
## 8-7  -0.0004591221 -0.03643492  0.035516678 1.0000000
## 9-7  -0.0640349390 -0.10001074 -0.028059139 0.0000010
## 10-7 -0.0767480516 -0.11272385 -0.040772251 0.0000000
## 9-8  -0.0635758169 -0.09955162 -0.027600016 0.0000013
## 10-8 -0.0762889296 -0.11226473 -0.040313129 0.0000000
```

```
## 10-9 -0.0127131127 -0.04868891  0.023262688 0.9824575
```

```r
boxplot(Returns~Portfolio, data = returns)
```




```r
plot(allEffects(lm(Returns~Portfolio,data=returns)))
```




```
threecomb
```

```
##       [,1]   [,2]  [,3]
##  [1,] "AAPL" "AXP" "BA"
##  [2,] "AAPL" "AXP" "C"
##  [3,] "AAPL" "AXP" "CAT"
##  [4,] "AAPL" "BA"  "C"
##  [5,] "AAPL" "BA"  "CAT"
##  [6,] "AAPL" "C"   "CAT"
##  [7,] "AXP"  "BA"  "C"
##  [8,] "AXP"  "BA"  "CAT"
##  [9,] "AXP"  "C"   "CAT"
## [10,] "BA"   "C"   "CAT"
```

Since we have 10 portfolios, we can compare any 2 of these in 10C2 = 45 different ways. Of these 45, the p-values tell us that atleast 25 of these comparisons are significantly different over the 6 month duration. This tells us that it is not too difficult to find an effective Buy and Hold Strategy with linear combination of Dow stocks that outperforms the Dow Jones Index with statistical significance.

Looking carefully at the 3-tuple combinations & the effects plot, it is clear that the holding Apple-Amex-Boeing over a 6 month period offers significantly better return (+ 12%) than holding Boeing-Citi-Caterpillar (- 5%)

While performance is obvious in hindsight, how much the optimal tuple be discovered ? Below, we compare long-only trading strategies that attempt to beat the Dow by active trading, instead of Buy & Hold.

1-way Anova on LONG-ONLY

1. Buy yesterday's WORST performing stock
2. Buy yesterday's BEST performing stock.
3. Just buy the index (DIA).

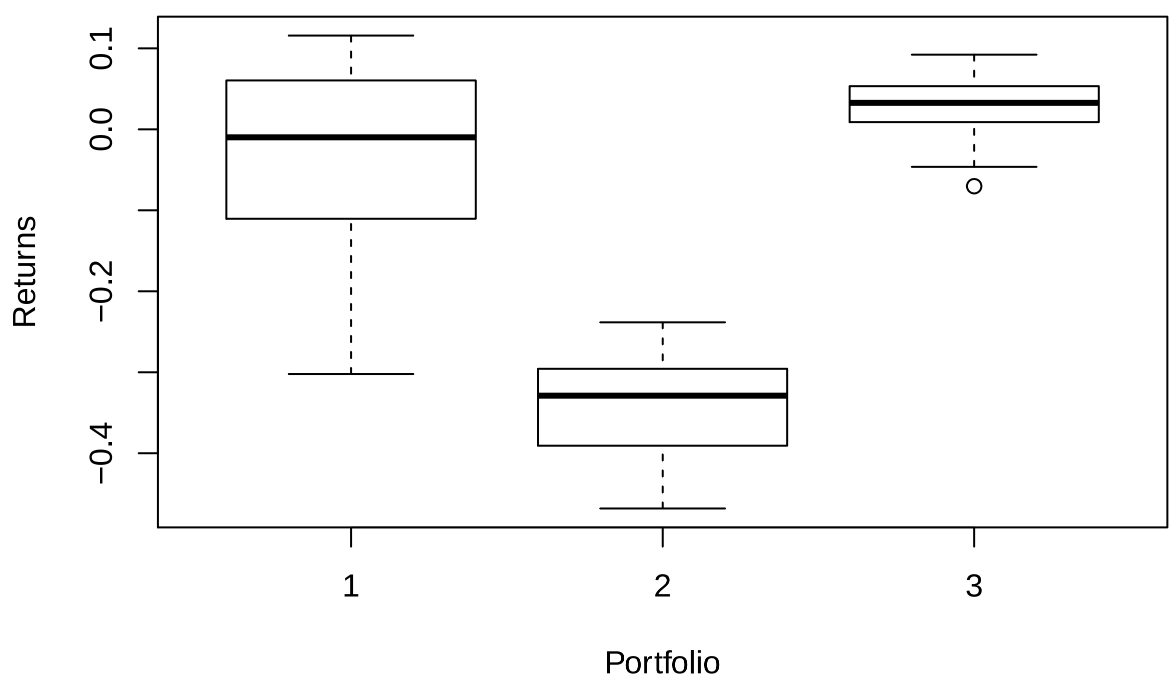
```
n = 251-180 #6 months = 180 days. We have 251 trading days in 2018, so there are n such "6 month period
components = dow[,2:31]
tradingreturns = matrix(0, nrow=3*n, ncol=2)

# i from 1 to n=72
# for i = 1, look at days 2 to 181
# for i = 2, look at days 3 to 182
# etc.
for(i in 1:n) {
  trading = matrix(0,nrow=180,ncol=3)

  for(s in (1+i):(180+i)) {
    yesterday = components[s-1,]
    today = components[s,]
    res = sort(yesterday, decreasing=TRUE, index.return=TRUE)
    best = res$ix[1]
    worst = res$ix[30]
    trading[(s-i),1] = today[worst]
    trading[(s-i),2] = today[best]
    trading[(s-i),3] = dow[s,1]
  }
  # arithmetic prog
  # start = a + (n-1)*d
  j = 1 + (i-1)*3
  tradingreturns[j,1] = sum(trading[,1])
  tradingreturns[j,2] = 1 # first trading strategy is factor 1

  tradingreturns[(j+1),1] = sum(trading[,2])
  tradingreturns[(j+1),2] = 2 # second trading strategy is factor 2

  tradingreturns[(j+2),1] = sum(trading[,3])
  tradingreturns[(j+2),2] = 3 # dow strategy is factor 3
}

colnames(tradingreturns) <- c("Returns", "Portfolio")
tradingreturns = data.frame(tradingreturns)
tradingreturns$Returns = as.double(as.character( tradingreturns$Returns ))
tradingreturns$Portfolio = factor( tradingreturns$Portfolio )
res.aov <- aov(Returns~Portfolio, data = tradingreturns)
summary(res.aov)
```

```
##              Df Sum Sq Mean Sq F value Pr(>F)
## Portfolio     2  5.476  2.7381   471.5 <2e-16 ***
## Residuals   210  1.219  0.0058
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
TukeyHSD(res.aov)
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = Returns ~ Portfolio, data = tradingreturns)
##
## $Portfolio
##            diff         lwr         upr p adj
## 2-1 -0.30081438 -0.33100402 -0.27062474 0e+00
## 3-1  0.06828417  0.03809453  0.09847381 7e-07
## 3-2  0.36909855  0.33890891  0.39928819 0e+00
```

```r
boxplot(Returns~Portfolio, data = tradingreturns)
```



Both these active trading strategies fare poorly versus just holding the Dow. Simply buying the Index beats buying yesterday's worst by 6.8%, and beats buying yesterday's best by a whopping 36.9%. Further, both these trading strategies had a negative median return and much higher variance than the index.

1-way Anova on LONG-SHORT:

1. Buy yesterday's WORST performing stock, SELL yesterday's BEST performing stock
2. Buy yesterday's BEST performing stock, SELL yesterday's WORST performing stock
3. Just buy the index (DIA).

```r
n = 251-180 #6 months = 180 days. We have 251 trading days in 2018, so there are n such "6 month period
components = dow[,2:31]
tradingreturns = matrix(0, nrow=3*n, ncol=2)

# i from 1 to n=72
# for i = 1, look at days 2 to 181
# for i = 2, look at days 3 to 182
# etc.
for(i in 1:n) {
  trading = matrix(0,nrow=180,ncol=3)

  for(s in (1+i):(180+i)) {
    yesterday = components[s-1,]
    today = components[s,]
    res = sort(yesterday, decreasing=TRUE, index.return=TRUE)
    best = res$ix[1]
    worst = res$ix[30]
    trading[(s-i),1] = today[worst] - today[best]
    trading[(s-i),2] = today[best] - today[worst]
    trading[(s-i),3] = dow[s,1]
  }
  # arithmetic prog
  # start = a + (n-1)*d
  j = 1 + (i-1)*3
  tradingreturns[j,1] = sum(trading[,1])
  tradingreturns[j,2] = 1 # first trading strategy is factor 1

  tradingreturns[(j+1),1] = sum(trading[,2])
  tradingreturns[(j+1),2] = 2 # second trading strategy is factor 2

  tradingreturns[(j+2),1] = sum(trading[,3])
  tradingreturns[(j+2),2] = 3 # dow strategy is factor 3
}

colnames(tradingreturns) <- c("Returns", "Portfolio")
tradingreturns = data.frame(tradingreturns)
tradingreturns$Returns = as.double(as.character( tradingreturns$Returns ))
tradingreturns$Portfolio = factor( tradingreturns$Portfolio )
res.aov <- aov(Returns~Portfolio, data = tradingreturns)
summary(res.aov)
```

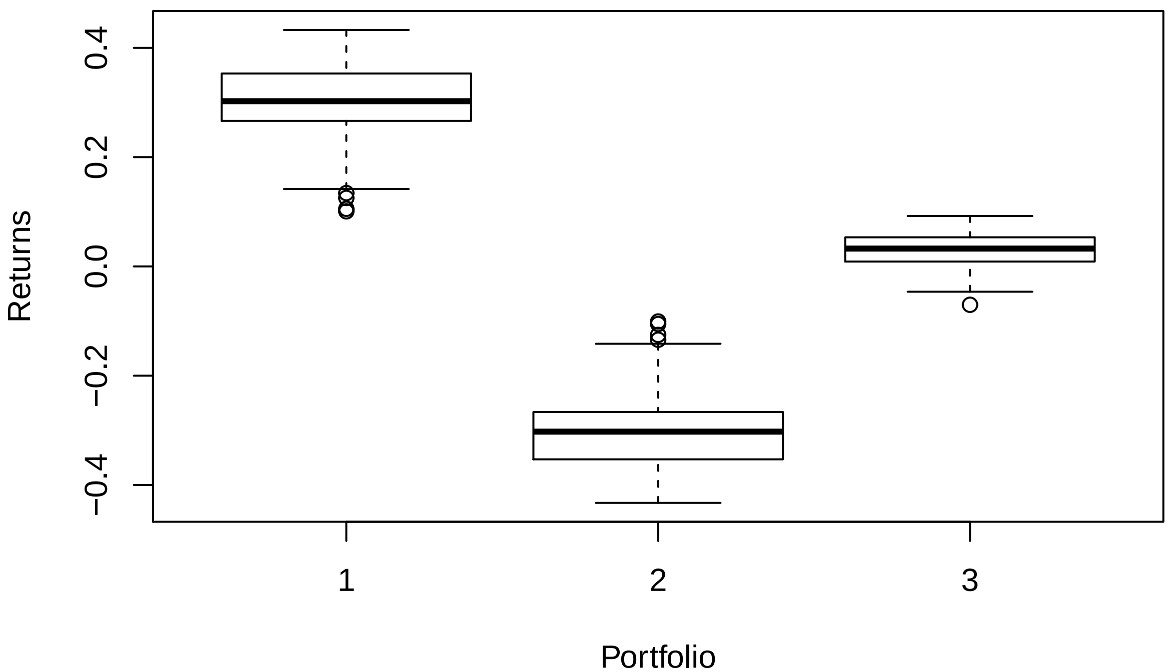
```
##              Df Sum Sq Mean Sq F value Pr(>F)
## Portfolio     2 12.886   6.443    1423 <2e-16 ***
## Residuals   210  0.951   0.005
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
TukeyHSD(res.aov)
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = Returns ~ Portfolio, data = tradingreturns)
```

```
##
## $Portfolio
##           diff        lwr        upr p adj
## 2-1 -0.6016288 -0.6282894 -0.5749681     0
## 3-1 -0.2728592 -0.2995199 -0.2461985     0
## 3-2  0.3287696  0.3021089  0.3554302     0
```

```r
boxplot(Returns~Portfolio, data = tradingreturns)
```



Finally we have an active trading strategy that handily beats owning the index! The Tukey 1-3 comparison tells us that our "Buy Yesterday's Worst, Sell Yesterday's Best" strategy beats holding the Dow by 27%, over the average 6 month holding period in 2018. Lets attempt to scale up strategy 1 by buying the worst k & selling the worst k.

1-way Anova on scaled-up LONG-SHORT strategies:

1. Buy yesterday's WORST performing stock, SELL yesterday's BEST performing stock
2. Buy yesterday's 2 WORST performing stock, SELL yesterday's 2 BEST performing stock
3. Buy yesterday's 3 WORST performing stock, SELL yesterday's 3 BEST performing stock
4. Buy yesterday's 4 WORST performing stock, SELL yesterday's 4 BEST performing stock
5. Buy yesterday's 5 WORST performing stock, SELL yesterday's 5 BEST performing stock
6. Just buy the index (DIA).

```r
n = 251-180 #6 months = 180 days. We have 251 trading days in 2018, so there are n such "6 month period
components = dow[,2:31]
tradingreturns = matrix(0, nrow=6*n, ncol=2)

# i from 1 to n=72
# for i = 1, look at days 2 to 181
# for i = 2, look at days 3 to 182
# etc.
for(i in 1:n) {
  trading = matrix(0,nrow=180,ncol=6)

  for(s in (1+i):(180+i)) {
    yesterday = components[s-1,]
    today = components[s,]
    res = sort(yesterday, decreasing=TRUE, index.return=TRUE)
    trading[(s-i),1] = sum(today[res$ix[30:30]]) - sum(today[res$ix[1:1]])
    trading[(s-i),2] = sum(today[res$ix[29:30]]) - sum(today[res$ix[1:2]])
    trading[(s-i),3] = sum(today[res$ix[28:30]]) - sum(today[res$ix[1:3]])
    trading[(s-i),4] = sum(today[res$ix[27:30]]) - sum(today[res$ix[1:4]])
    trading[(s-i),5] = sum(today[res$ix[26:30]]) - sum(today[res$ix[1:5]])
    trading[(s-i),6] = dow[s,1]
  }
  # arithmetic prog
  # start = a + (n-1)*d
  j = 1 + (i-1)*6
  tradingreturns[j,1] = sum(trading[,1])
  tradingreturns[j,2] = 1 # first trading strategy is factor 1

  tradingreturns[(j+1),1] = sum(trading[,2])
  tradingreturns[(j+1),2] = 2 # second trading strategy is factor 2

  tradingreturns[(j+2),1] = sum(trading[,3])
  tradingreturns[(j+2),2] = 3 # dow strategy is factor 3

  tradingreturns[(j+3),1] = sum(trading[,4])
  tradingreturns[(j+3),2] = 4 # dow strategy is factor 3

  tradingreturns[(j+4),1] = sum(trading[,5])
  tradingreturns[(j+4),2] = 5 # dow strategy is factor 3

  tradingreturns[(j+5),1] = sum(trading[,6])
  tradingreturns[(j+5),2] = 6 # dow strategy is factor 3

}

colnames(tradingreturns) <- c("Returns", "Portfolio")
tradingreturns = data.frame(tradingreturns)
tradingreturns$Returns = as.double(as.character( tradingreturns$Returns ))
tradingreturns$Portfolio = factor( tradingreturns$Portfolio )
res.aov <- aov(Returns~Portfolio, data = tradingreturns)
summary(res.aov)
```

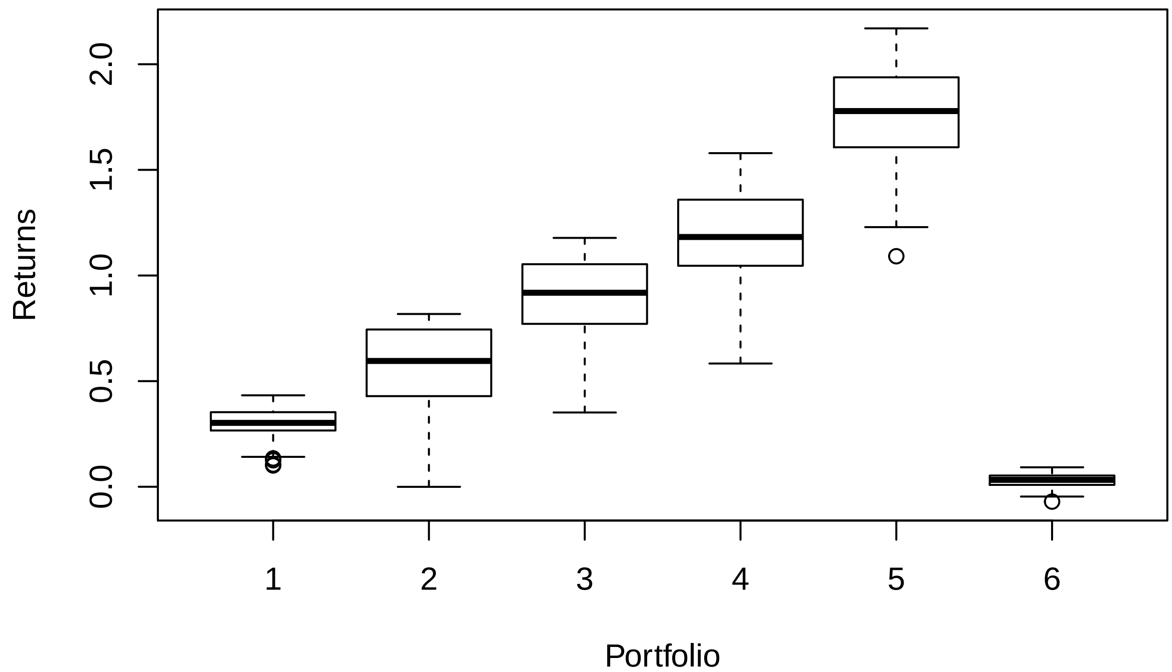
```
##              Df Sum Sq Mean Sq F value Pr(>F)
```

```
## Portfolio     5  140.2  28.043   795.8 <2e-16 ***
## Residuals   420   14.8   0.035
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

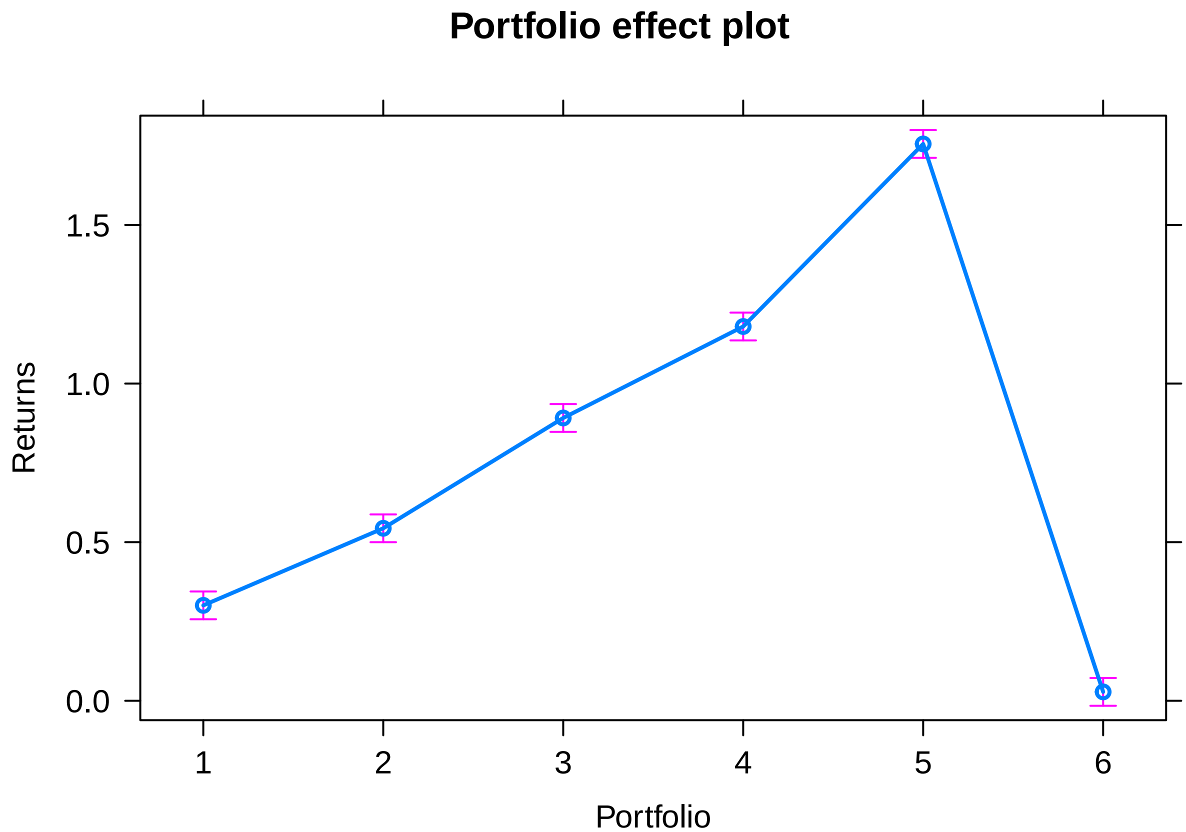
```r
TukeyHSD(res.aov)
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = Returns ~ Portfolio, data = tradingreturns)
##
## $Portfolio
##           diff        lwr        upr p adj
## 2-1  0.2429238  0.1527246  0.3331230     0
## 3-1  0.5907545  0.5005553  0.6809537     0
## 4-1  0.8790660  0.7888668  0.9692652     0
## 5-1  1.4546479  1.3644487  1.5448471     0
## 6-1 -0.2728592 -0.3630584 -0.1826600     0
## 3-2  0.3478307  0.2576315  0.4380299     0
## 4-2  0.6361422  0.5459430  0.7263414     0
## 5-2  1.2117241  1.1215249  1.3019233     0
## 6-2 -0.5157830 -0.6059822 -0.4255838     0
## 4-3  0.2883115  0.1981123  0.3785107     0
## 5-3  0.8638934  0.7736942  0.9540926     0
## 6-3 -0.8636137 -0.9538129 -0.7734145     0
## 5-4  0.5755819  0.4853827  0.6657811     0
## 6-4 -1.1519252 -1.2421244 -1.0617260     0
## 6-5 -1.7275071 -1.8177063 -1.6373079     0
```

```r
boxplot(Returns~Portfolio, data = tradingreturns)
```




```r
plot(allEffects(lm(Returns~Portfolio,data=tradingreturns)))
```



We note that by buying yesterday's 5 worst & selling yesterday's 5 best performing Dow components, we can beat the Dow by 173% (see Tukey 6-5) over the average 6 month duration in 2018.

Conclusion: There exist several long-short strategies for alpha generation in the Dow family.