Final Project RMD

Marali Benitez, Youcef Djoudi, Kelly Thackery, Samina Issa, Samira Bechi

2025-04-09

##loading in packages and data  
library(wooldridge)  
library(AER)  
library(quantmod)  
library(PerformanceAnalytics)  
library(lmtest)  
  
#ceosal2 data  
ceosalary <- ceosal2  
  
#Murder rates data  
data("MurderRates")  
mr <- MurderRates

#### Data: ceosal2

##### (a) Describe the data. Describe the variables; response, predictors, continuous, categorical variables and missing data.

library(wooldridge)  
data("ceosal2")  
head(ceosal2)

## salary age college grad comten ceoten sales profits mktval lsalary lsales  
## 1 1161 49 1 1 9 2 6200 966 23200 7.057037 8.732305  
## 2 600 43 1 1 10 10 283 48 1100 6.396930 5.645447  
## 3 379 51 1 1 9 3 169 40 1100 5.937536 5.129899  
## 4 651 55 1 0 22 22 1100 -54 1000 6.478509 7.003066  
## 5 497 44 1 1 8 6 351 28 387 6.208590 5.860786  
## 6 1067 64 1 1 7 7 19000 614 3900 6.972606 9.852194  
## lmktval comtensq ceotensq profmarg  
## 1 10.051908 81 4 15.580646  
## 2 7.003066 100 100 16.961130  
## 3 7.003066 81 9 23.668638  
## 4 6.907755 484 484 -4.909091  
## 5 5.958425 64 36 7.977208  
## 6 8.268732 49 49 3.231579

*This dataset (ceosal2) includes information on CEOs and their companies. The analysis focuses on how various CEO and firm characteristics influence company profits*.

##### Variables:

**Response Variable:** - profits: Annual firm profits (in millions).  
- Type: **Continuous**.

| Variable | Description | Type |
| --- | --- | --- |
| salary | CEO salary (in thousands) | Continuous |
| age | CEO age | Continuous |
| college | 1 = undergraduate degree | Categorical (binary) |
| grad | 1 = graduate degree | Categorical (binary) |
| comten | Years with the company | Continuous |
| ceoten | Years as CEO | Continuous |
| sales | Company sales (in millions) | Continuous |
| mktval | Market value (in millions) | Continuous |
| lmktval | Log of market value | Continuous |
| comtensq | Square of company tenure | Continuous |
| ceotensq | Square of CEO tenure | Continuous |
| profmarg | Profit margin = profits/sales | Continuous |

**Excluded Predictors:** - lsalary, lsales: Log-transformed versions of salary and sales; excluded from final model.

#Check for missing values  
colSums(is.na(ceosal2))

## salary age college grad comten ceoten sales profits   
## 0 0 0 0 0 0 0 0   
## mktval lsalary lsales lmktval comtensq ceotensq profmarg   
## 0 0 0 0 0 0 0

*NO missing values in the data set*.

##### (b) Build an optimal model. Print summary: discuss F-test, t-tests, Rsquare, S-square. Take out outliers.

#preliminary model  
model <- lm(profits ~ .-lsalary -lmktval -lsales, data=ceosal2)  
summary(model)

##   
## Call:  
## lm(formula = profits ~ . - lsalary - lmktval - lsales, data = ceosal2)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -595.72 -24.52 7.21 37.55 543.72   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 21.725679 119.334493 0.182 0.85576   
## salary 0.011629 0.021625 0.538 0.59146   
## age -0.067916 1.592252 -0.043 0.96603   
## college -47.570474 68.855153 -0.691 0.49061   
## grad -1.020730 23.409112 -0.044 0.96527   
## comten 1.261179 3.511631 0.359 0.71995   
## ceoten -4.988814 4.417210 -1.129 0.26037   
## sales 0.016682 0.002843 5.867 2.36e-08 \*\*\*  
## mktval 0.044778 0.002732 16.392 < 2e-16 \*\*\*  
## comtensq -0.014689 0.077158 -0.190 0.84925   
## ceotensq 0.133996 0.147531 0.908 0.36507   
## profmarg 1.900953 0.627057 3.032 0.00283 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 146.5 on 165 degrees of freedom  
## Multiple R-squared: 0.877, Adjusted R-squared: 0.8688   
## F-statistic: 107 on 11 and 165 DF, p-value: < 2.2e-16

*In the primary model, all variables were included except for lsalary, lmktval, and lsales. The R² value is 0.877, indicating that approximately 87.7% of the variation in profits is accounted for by the model. The F-test p-value is less than 2.2e-16, confirming that the model is statistically significant and not due to random chance. Based on the summary output, the predictors that show statistical significance are sales, market value (mktval), and profit margin (profmarg). The residual standard error is 146.5, suggesting that, on average, the model’s profit predictions differ from the actual values by about 146.5 units.*

#taking out outliers  
cooksd <- cooks.distance(model)  
influential <- as.numeric(names(cooksd)[(cooksd > (4 / nrow(ceosalary)))])  
influential

## [1] 1 30 34 40 43 44 47 50 52 77 91 101 107 114 135 168

newceosalary <- ceosalary[-influential]  
#creating reduced model with no outliers in model  
newceosalary\_model <- lm(profits~.-lsalary -lmktval -lsales, data = newceosalary)  
summary(newceosalary\_model)

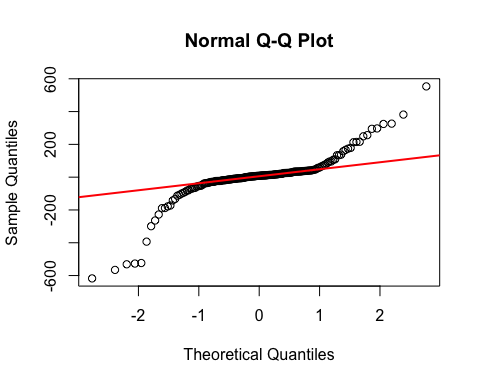
##   
## Call:  
## lm(formula = profits ~ . - lsalary - lmktval - lsales, data = newceosalary)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -594.21 -25.51 6.62 35.76 553.20   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 27.703509 118.560977 0.234 0.81553   
## age -0.028988 1.587197 -0.018 0.98545   
## college -48.581175 68.681976 -0.707 0.48035   
## grad -1.832048 23.310377 -0.079 0.93745   
## comten 1.206672 3.502645 0.345 0.73090   
## ceoten -4.414495 4.276975 -1.032 0.30350   
## sales 0.016885 0.002812 6.005 1.17e-08 \*\*\*  
## mktval 0.045084 0.002666 16.910 < 2e-16 \*\*\*  
## comtensq -0.014633 0.076992 -0.190 0.84950   
## ceotensq 0.118753 0.144472 0.822 0.41227   
## profmarg 1.877254 0.624166 3.008 0.00304 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 146.2 on 166 degrees of freedom  
## Multiple R-squared: 0.8768, Adjusted R-squared: 0.8694   
## F-statistic: 118.1 on 10 and 166 DF, p-value: < 2.2e-16

#creating newer updated model using step function  
updatedceosalmodel <- step(newceosalary\_model, direction = "both", trace = 0)  
summary(updatedceosalmodel)

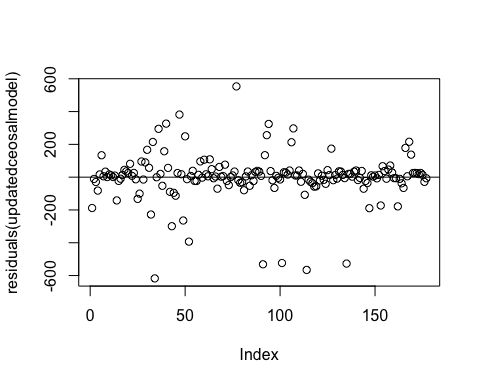
##   
## Call:  
## lm(formula = profits ~ sales + mktval + profmarg, data = newceosalary)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -617.41 -23.28 8.08 34.34 553.52   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -26.818425 13.237245 -2.026 0.04430 \*   
## sales 0.017124 0.002732 6.267 2.83e-09 \*\*\*  
## mktval 0.045084 0.002588 17.422 < 2e-16 \*\*\*  
## profmarg 1.852935 0.612487 3.025 0.00286 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 144 on 173 degrees of freedom  
## Multiple R-squared: 0.8754, Adjusted R-squared: 0.8732   
## F-statistic: 405 on 3 and 173 DF, p-value: < 2.2e-16

##### (c) Briefly discuss residuals.

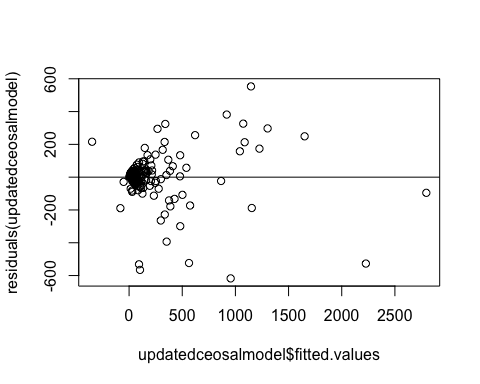
# Q-Q plot: check for normality of residuals  
qqnorm(residuals(updatedceosalmodel))  
qqline(residuals(updatedceosalmodel), col = "red", lwd = 2)



#checking if residuals are randomly distributed  
plot(residuals(updatedceosalmodel))  
abline(a = 0, b = 0)



#residuals vs fitted values plot  
plot(updatedceosalmodel$fitted.values, residuals(updatedceosalmodel))  
abline(a = 0, b = 0)



bptest(updatedceosalmodel)

##   
## studentized Breusch-Pagan test  
##   
## data: updatedceosalmodel  
## BP = 40.915, df = 3, p-value = 6.818e-09

##### (d) Make 2 predictions with CI’s. Interpret predictions if needed.

# Hypothetical CEO 1  
new\_ceo1 <- data.frame(  
 sales = 4500,  
 mktval = 1800,  
 profmarg = 10  
)  
  
# Hypothetical CEO 2  
new\_ceo2 <- data.frame(  
 sales = 7000,  
 mktval = 3000,  
 profmarg = 15  
)  
  
#make predictions with 95% CI  
# Predictions for CEO 1  
predict(updatedceosalmodel, newdata = new\_ceo1, interval = "confidence")

## fit lwr upr  
## 1 149.9177 123.9457 175.8898

# Predictions for CEO 2  
predict(updatedceosalmodel, newdata = new\_ceo2, interval = "confidence")

## fit lwr upr  
## 1 256.0919 223.5525 288.6312

#### Data: MurderRates.

##### (a) Describe the data. Describe the variables; response, predictors, continuous, categorical variables and missing data.

head(mr)

## rate convictions executions time income lfp noncauc southern  
## 1 19.25 0.204 0.035 47 1.10 51.2 0.321 yes  
## 2 7.53 0.327 0.081 58 0.92 48.5 0.224 yes  
## 3 5.66 0.401 0.012 82 1.72 50.8 0.127 no  
## 4 3.21 0.318 0.070 100 2.18 54.4 0.063 no  
## 5 2.80 0.350 0.062 222 1.75 52.4 0.021 no  
## 6 1.41 0.283 0.100 164 2.26 56.7 0.027 no

*The murder rates data set contains data collected in 1950 that reflects 44 states’ murder rates among other variables. The variables for this dataset are as follows:*

* rate: (continuous) the states murder rate per 100,000 according to the FBI estimate.

the convictions (which in this data set reflects the number of convictions divided by number of murders in 1950), the average number of executions from 1946-1950 divided by the number of convictions in 1950, the median time served in months of convicted murderers released in 1951, the median family income for the state in 1949 in thousands of dollars, the labor force participation rate for the state in 1950 (in percent), the proportion of the states population that is non-Caucasian in 1950, and southern, which indicates if the state is in the south or not. The response variable in this data would be the rate, while the other variables would be the predictors. There were no missing values in this data.\*

##### (b) Build an optimal model. Print summary: discuss F-test, t-tests, Rsquare, S-square. Take out outliers\*\*.

#creating preliminary model  
murdermodel <- lm(rate~., data = mr)  
summary(murdermodel)

##   
## Call:  
## lm(formula = rate ~ ., data = mr)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -4.9913 -1.1943 -0.3538 1.2383 6.5574   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.44436 9.96694 0.045 0.9647   
## convictions -4.33938 2.78313 -1.559 0.1277   
## executions 2.85276 6.12313 0.466 0.6441   
## time -0.01547 0.00705 -2.194 0.0348 \*  
## income -2.50013 1.68519 -1.484 0.1466   
## lfp 0.19357 0.20614 0.939 0.3540   
## noncauc 10.39903 5.40610 1.924 0.0623 .  
## southernyes 3.26216 1.32980 2.453 0.0191 \*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 2.459 on 36 degrees of freedom  
## Multiple R-squared: 0.7459, Adjusted R-squared: 0.6965   
## F-statistic: 15.1 on 7 and 36 DF, p-value: 5.105e-09

#calculation for preliminary models' s-squared value  
#mean(murdermodel$residuals^2)

*In the preliminary model, all predictor variables were used to create the model. The R^2 value for this preliminary model is .7459, meaning that 74.59% of the variation in the response can be explained by these predictor variables. The p-value for the f-test in this preliminary model is 5.105e-09, meaning that this model is non trivial. The summary output for the preliminary model also shows that the only statistically significant predictors in this model are time and the southern factor. The S-squared value for this model is about 4.94, meaning that on average, the preliminary model’s predictions deviate by about 4.94 murders from the true murder rate.*

#taking out outliers  
cooks <- cooks.distance(murdermodel)  
inf <- as.numeric(names(cooks)[cooks>(4/44)])  
newmurderdata <- mr[-inf,]  
  
#creating reduced model with no outliers in model  
newmurdermodel <- lm(rate~., data = newmurderdata)  
summary(newmurdermodel)

##   
## Call:  
## lm(formula = rate ~ ., data = newmurderdata)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -3.3145 -1.4250 0.0003 1.2285 4.4268   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 3.217234 8.374625 0.384 0.703398   
## convictions -6.017805 2.196373 -2.740 0.009966 \*\*   
## executions -4.791480 9.061282 -0.529 0.600603   
## time -0.008623 0.005536 -1.558 0.129156   
## income -2.328506 1.347126 -1.728 0.093538 .   
## lfp 0.128752 0.174727 0.737 0.466569   
## noncauc 8.937551 4.390175 2.036 0.050119 .   
## southernyes 4.379154 1.046576 4.184 0.000208 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 1.841 on 32 degrees of freedom  
## Multiple R-squared: 0.8288, Adjusted R-squared: 0.7913   
## F-statistic: 22.13 on 7 and 32 DF, p-value: 1.404e-10

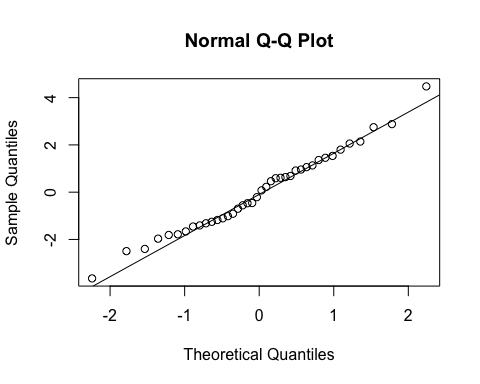
#creating newer updated model using step function  
updatedmurdermodel <- step(newmurdermodel, direction = "both", trace = 0)  
summary(updatedmurdermodel)

##   
## Call:  
## lm(formula = rate ~ convictions + time + income + noncauc + southern,   
## data = newmurderdata)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -3.6485 -1.2694 -0.0675 1.0782 4.4753   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 9.266257 2.479583 3.737 0.000683 \*\*\*  
## convictions -6.139134 2.043407 -3.004 0.004969 \*\*   
## time -0.009475 0.005222 -1.814 0.078439 .   
## income -1.919536 1.068678 -1.796 0.081357 .   
## noncauc 8.934485 4.095912 2.181 0.036171 \*   
## southernyes 4.233961 1.012899 4.180 0.000193 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 1.813 on 34 degrees of freedom  
## Multiple R-squared: 0.8234, Adjusted R-squared: 0.7974   
## F-statistic: 31.7 on 5 and 34 DF, p-value: 7.059e-12

*After taking out the outliers, we can see that the R^2 value has increased to .8288, and the the S^2 value has gone down to 1.841. We then reduced the model using the step function, where we can see that the r^2 and s^2 did not change by much (the new r^2 value is .8234 and the new s^2 value is 1.814). The predictors for this model have been reduced to convictions, time, income, noncaucasian, and southern. More of the predictors are statistically significant in this model, with the intercept, convictions, noncaucasian, and southern all being statistically significant according to the p-value found by the t-tests. The f-test for this reduced model shows that it is nontrivial, as the p-value for the f-test is 7.059e-12.*

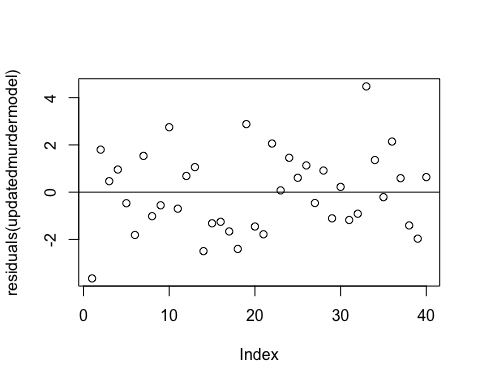
##### (c) Briefly discuss residuals.

##taking a look at residuals  
  
#checking if data is normally distributed  
qqnorm(residuals(updatedmurdermodel))  
qqline(residuals(updatedmurdermodel))



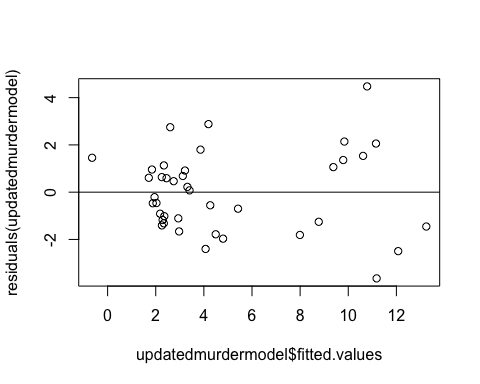
*The normal qqplot for this model suggests that the residuals in this model are normally distributed*.

#checking if residuals are randomly distributed  
plot(residuals(updatedmurdermodel))  
abline(a = 0, b = 0)



*The residuals for this model also seem to be randomly scattered around 0, meaning that a linear model may be a good fit for the data*.

#residuals vs fitted values plot  
plot(updatedmurdermodel$fitted.values, residuals(updatedmurdermodel))  
abline(a = 0, b = 0)



bptest(updatedmurdermodel)

##   
## studentized Breusch-Pagan test  
##   
## data: updatedmurdermodel  
## BP = 11.773, df = 5, p-value = 0.03803

*In the fitted values vs residuals plot, there does seem to somewhat of a pattern, suggesting that there is heteroscedasticity present. This is confirmed by the Breusch-Pagan test (which has a p-value of about .038)*.

##### (d) Make 2 predictions with CI’s. Interpret predictions if needed.

## hypothetical state 1  
state1 <- data.frame(  
 convictions = .3,  
 time = 150,  
 income = 2,  
 noncauc = .015,  
 southern = "yes"  
)  
  
##hypothetical state 2  
state2 <- data.frame(  
 convictions = .15,  
 time =100,  
 income = 3,  
 noncauc = .2,  
 southern = "no"  
)  
  
prediction1 <- predict(updatedmurdermodel, newdata = state1, interval = "confidence")  
prediction2 <- predict(updatedmurdermodel, newdata = state2, interval = "confidence")  
  
prediction1

## fit lwr upr  
## 1 6.532212 4.461027 8.603398

prediction2

## fit lwr upr  
## 1 3.426203 0.4326271 6.419779

*Hypothetical state 1 inputs: convictions = .3, time = 150, income = 2, noncaucasian = .015, southern = “yes”. Hypothetical state 2 inputs: convictions = .15, time = 100, income = 3, noncaucasian = .2, southern = “no”. The 95% confidence interval for hypothetical state 1 is (4.46, 8.6), meaning that for a state with those inputs, we are 95% confident that the true murder rate per 100,000 per the FBI estimates for that state is in between those two values. For the 2nd hypothetical state, we would be 95% that the true murder rate per 100,000 per the FBI estimates for that state is in between (.43, 6.42).*

#### Finance Application

##### Read stock prices for 5 different stocks. Close prices for 3 years, most recent 2024. Criteria Beta > 1, PE > 10, growth > 10%, different industries.

# Read stock prices  
start<- as.Date("2021-01-01")  
end<-as.Date("2024-01-01")  
DIS <- getSymbols("DIS", from=start, to=end, auto.assign=FALSE)  
NVDA <- getSymbols("NVDA", from=start, to=end, auto.assign=FALSE)  
CVNA<-getSymbols("CVNA", from=start, to=end, auto.assign=FALSE)  
SPOT<-getSymbols("SPOT", from=start, to=end, auto.assign=FALSE)  
DASH<-getSymbols("DASH", from=start, to=end, auto.assign=FALSE)  
  
  
## Closed price  
DIS.close <- DIS[ ,4]  
NVDA.close <- NVDA[ ,4]  
CVNA.close <- CVNA[ ,4]  
SPOT.close <- SPOT[ ,4]  
DASH.close <- DASH[ ,4]  
  
## Returns  
DIS.return <- dailyReturn(DIS.close)  
NVDA.return <- dailyReturn(NVDA.close)  
CVNA.return <- dailyReturn(CVNA.close)  
SPOT.return <- dailyReturn(SPOT.close)  
DASH.return <- dailyReturn(DASH.close)

**Disney: beta(1.44), P/E(29.69), Growth Estimate(11.28), Industry: Entertainment**  
**Nvidia: beta(1.96), P/E(38.89), Growth Estimate(27.98%), Industry: Semiconductors**  
**Carvana: beta(3.62), P/E(138.64), Growth Estimate(43.53%), Industry: Auto & Truck Dealerships Spotify: beta(1.75), P/E(90.03), Growth Estimate(27.20%), Industry: Internet Content & Information**  
**DoorDash: beta(1.69), P/E(627.07), Growth Estimate(31.84%), Industry: Communication Services**

##### (a)Plot close prices on three different plots. Notice the general “trend” and “variation” on different spans.

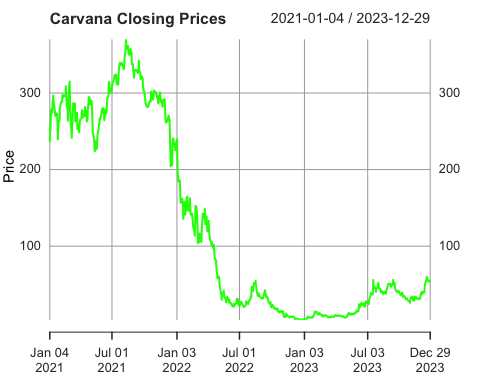
#Plot close stock prices on different plots.  
plot(DIS.close, main = "Disney Closing Prices", type = "l", col = "blue", xlab = "Date", ylab = "Price")



plot(NVDA.close, main = "NVIDIA Closing Prices", type = "l", col = "red", xlab = "Date", ylab = "Price")



plot(CVNA.close, main = "Carvana Closing Prices", type = "l", col = "green", xlab = "Date", ylab = "Price")



plot(SPOT.close, main = "Spotify Closing Prices", type = "l", col = "magenta", xlab = "Date", ylab = "Price")



plot(DASH.close, main = "DoorDash Closing Prices", type = "l", col = "purple", xlab = "Date", ylab = "Price")



**Remark: The closing prices for Disney, Carvana, Spotify and Doordash show a decline in 2022. Carvana doesn’t have much variation through 2023. Spotify shows an increase with slight variation in 2023. Disney and Doordash both show variation in 2023. The closing price for NVIDIA rises until the end of 2021 then falls until it starts to rise again toward the end of 2022.**

##### (b) Calculate annualized average return and annualized risk. Present the correlations matrix.

library(PerformanceAnalytics)  
#Calculate the annualized return and annualized risk of each stock.  
## A function to compute the Annualized Expected return/Risk  
mu.sigma <- function(return){  
 mu.ann <- mean(return) \* 252  
 sigma.ann <- sd(return) \* sqrt(252)  
 return(c(mu.ann, sigma.ann))  
}  
  
## Annualized Expected Return and Annualized Risk  
dis <- mu.sigma(DIS.return)  
nvda <- mu.sigma(NVDA.return)  
cvna <- mu.sigma(CVNA.return)  
spot <- mu.sigma(SPOT.return)  
dash <- mu.sigma(DASH.return)  
  
  
cat('Disney:', dis, '\n')

## Disney: -0.1814154 0.2993281

cat('NVIDIA:', nvda, '\n')

## NVIDIA: 0.5831085 0.5296374

cat('Carvana:', cvna, '\n')

## Carvana: 0.3134923 1.296106

cat('Spoify:', spot, '\n')

## Spoify: -0.03417431 0.5166859

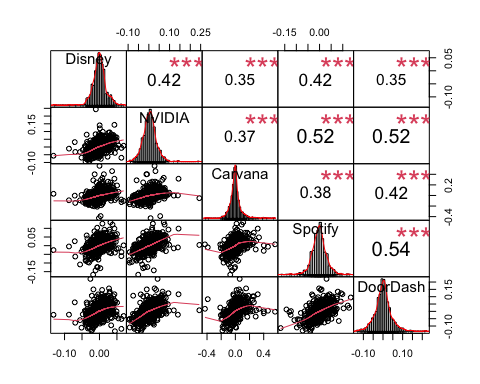
cat('DoorDash:', dash, '\n')

## DoorDash: 0.1089975 0.6705528

## Correlation Matrix  
returns <- cbind(DIS.return,NVDA.return,CVNA.return,SPOT.return,DASH.return)  
colnames(returns) <- c('Disney', 'NVIDIA', 'Carvana', 'Spotify','DoorDash')  
head(returns)

## Disney NVIDIA Carvana Spotify DoorDash  
## 2021-01-04 0.0000000000 0.000000000 0.00000000 0.000000000 0.00000000  
## 2021-01-05 0.0042774077 0.022209954 0.08172852 0.008745985 0.03764631  
## 2021-01-06 0.0038107637 -0.058952971 -0.01858518 0.001657494 -0.03060305  
## 2021-01-07 -0.0030147013 0.057830260 0.07706415 0.054480638 0.09113642  
## 2021-01-08 0.0006159738 -0.005039737 0.03064850 0.065638583 0.02173196  
## 2021-01-11 0.0022384794 0.025966476 -0.01547909 -0.025601026 0.06989560

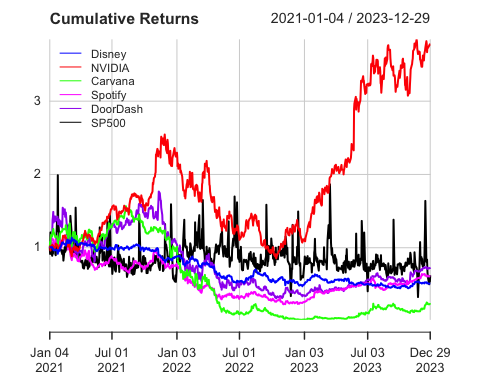
chart.Correlation(returns)



**Remark: Disney has a -18.1% return with a 29.9% risk.**  
**NVIDIA has a 58.3% return with a 53% risk.**  
**Carvana has a 31.3% return with a 129.6% risk.**  
**Spoify has a -3.4% return with a 51.7% risk.**  
**DoorDash has a 10.9% return with a 67.1% risk. From the correlation coefficients we can see these companies generally move in the same direction. The Spotify-Doordash scatter plot shows a tight upward trend, indicating a strong positive correlation between the two.**

##### (c)Plot cumulative returns on one common plot.

## SP500(Benchmark)  
SP500<- getSymbols("^GSPC", from=start, to=end, auto.assign = FALSE)  
SP500.close <- SP500[ , 5]  
SP500.return<- dailyReturn(SP500.close)  
  
## Combine all returns into one data frame  
returns <- cbind(DIS.return, NVDA.return, CVNA.return, SPOT.return, DASH.return, SP500.return)  
colnames(returns) <- c('Disney', 'NVIDIA', 'Carvana', 'Spotify', 'DoorDash', 'SP500')  
  
  
## Plot cumulative returns using chart.CumReturns from PerformanceAnalytics  
library(PerformanceAnalytics)  
chart.CumReturns(returns, wealth.index = TRUE, legend.loc = 'topleft',  
 main = 'Cumulative Returns', colorset=c('blue','red','green','magenta','purple','black'))



**Remark: NVIDIA shows a significant upward trend, indicating strong growth. Disney, Carvana, Spotify, and DoorDash have more moderate growth. You can see the SP500**

##### (d)Estimate alpha, beta, Rsquare.

##Riskfree rates   
rf <- read.csv("F-F\_Research\_Data\_Factors\_daily.CSV", head=T, skip=3) # a data frame  
head(rf)

## X Mkt.RF SMB HML RF  
## 1 19260701 0.10 -0.25 -0.27 0.009  
## 2 19260702 0.45 -0.33 -0.06 0.009  
## 3 19260706 0.17 0.30 -0.39 0.009  
## 4 19260707 0.09 -0.58 0.02 0.009  
## 5 19260708 0.21 -0.38 0.19 0.009  
## 6 19260709 -0.71 0.43 0.57 0.009

rf$dates <- as.Date(rf$X, format="%Y%m%d") # create dates as X in the fama.french  
## sort ff according to dates   
rf.new <- rf[rf$dates >= "2021-01-01" & rf$dates<= "2024-01-01", ]   
head(rf.new)

## X Mkt.RF SMB HML RF dates  
## 24897 20210104 -1.41 0.30 0.48 0 2021-01-04  
## 24898 20210105 0.86 1.24 0.48 0 2021-01-05  
## 24899 20210106 0.79 2.17 3.91 0 2021-01-06  
## 24900 20210107 1.76 0.30 -0.79 0 2021-01-07  
## 24901 20210108 0.51 -0.79 -1.32 0 2021-01-08  
## 24902 20210111 -0.52 0.28 1.24 0 2021-01-11

#dim(rf.new)  
#tail(rf.new)  
rf.new <- rf.new[ -754, ]  
#dim(rf.new)  
  
# Estimate alpha, beta, Rsquare of 3 stocks.  
beta <- function(stock.return, market.return, riskfree.rate) {  
 stock.excess <- stock.return - riskfree.rate  
 market.excess <- market.return - riskfree.rate  
 model <- lm(stock.excess ~ market.excess)  
 coefs <- coef(model)  
 Rsquare <- summary(model)$r.squared  
 results <- data.frame(alpha = coefs[1], beta = coefs[2], Rsquare = Rsquare)  
 return(results)  
}  
  
# Call beta function for each stock  
res\_DIS <- beta(DIS.return, SP500.return, rf.new$RF)  
res\_NVDA <- beta(NVDA.return, SP500.return, rf.new$RF)  
res\_CVNA <- beta(CVNA.return, SP500.return, rf.new$RF)  
res\_SPOT <- beta(SPOT.return, SP500.return, rf.new$RF)  
res\_DASH <- beta(DASH.return, SP500.return, rf.new$RF)  
  
# Combine the individual results into one table  
results\_table <- rbind(Disney = res\_DIS, NVIDIA = res\_NVDA, Carvana = res\_CVNA, SPOT = res\_SPOT, DASH = res\_DASH)  
  
# Display the resulting table  
print(results\_table)

## alpha beta Rsquare  
## Disney -0.008944452 -0.002584245 0.0005040206  
## NVIDIA -0.005883104 -0.007491570 0.0015339546  
## Carvana -0.006947954 -0.008393031 0.0003389432  
## SPOT -0.008395093 0.003643927 0.0003865928  
## DASH -0.007830721 0.004316040 0.0003240453

**Remark: The alphas tell us the stocks underperformed relative to market expectations. Betas are less than the baseline of 1, suggesting less sensitivity to market movements.**