library(tidyverse)

## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --

## v ggplot2 3.3.3 v purrr 0.3.4  
## v tibble 3.0.5 v dplyr 1.0.3  
## v tidyr 1.1.2 v stringr 1.4.0  
## v readr 1.4.0 v forcats 0.5.0

## -- Conflicts ------------------------------------------ tidyverse\_conflicts() --  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag() masks stats::lag()

library(dplyr)

library(readxl)  
task\_data <- read\_excel("C:/Users/KUNAL/Desktop/data/task\_data.xlsx")  
View(task\_data)

data\_task<-task\_data %>% slice(-c(1))

data\_task

## # A tibble: 55 x 8  
## Year Country Export Import Population `Population\_in ~ `GDP \_(current ~  
## <chr> <chr> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 2008~ Singap~ 8209. 7514. 4839396 4.84 193611986713.  
## 2 2009~ Singap~ 7577. 6455. 4987573 4.99 194152286009.  
## 3 2010~ Singap~ 9818. 7143. 5076732 5.08 239809387605.  
## 4 2011~ Singap~ 16769. 8500. 5183688 5.18 279351168707.  
## 5 2012~ Singap~ 13600. 7492. 5312437 5.31 295087220933.  
## 6 2013~ Singap~ 12511. 6763. 5399162 5.40 307576360585.  
## 7 2014~ Singap~ 9809. 7124. 5469724 5.47 314851156183.  
## 8 2015~ Singap~ 7720. 7308. 5535002 5.54 308004146058.  
## 9 2016~ Singap~ 9565. 7087. 5607283 5.61 318068476294.  
## 10 2017~ Singap~ 10203. 7467. 5612253 5.61 338406474039.  
## # ... with 45 more rows, and 1 more variable: `GDP \_(current  
## # \_USMillion$)` <dbl>

library(ggpubr)

library(rstatix)

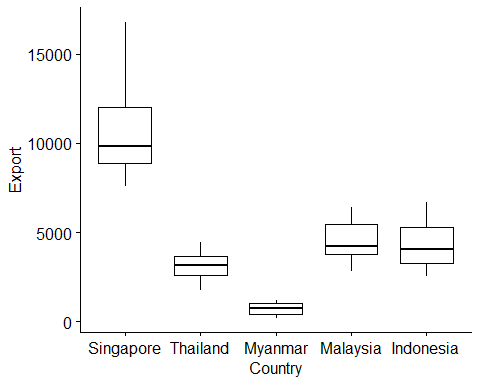
##   
## Attaching package: 'rstatix'

## The following object is masked from 'package:stats':  
##   
## filter

# Summary Statistics  
data\_task %>% group\_by(Country) %>% get\_summary\_stats(Export,type = "mean\_sd")

## # A tibble: 5 x 5  
## Country variable n mean sd  
## <chr> <chr> <dbl> <dbl> <dbl>  
## 1 Indonesia Export 11 4341. 1330.  
## 2 Malaysia Export 11 4515. 1127.  
## 3 Myanmar Export 11 705. 361.  
## 4 Singapore Export 11 10668. 2779.  
## 5 Thailand Export 11 3090. 839.

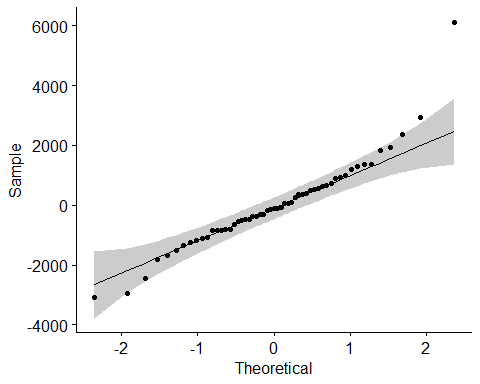
# Visualization  
ggboxplot(data\_task, x="Country", y="Export")



# outliers  
data\_task %>% group\_by(Country) %>% identify\_outliers("Export")

## [1] Country Year   
## [3] Export Import   
## [5] Population Population\_in \_million   
## [7] GDP \_(current \_US$) GDP \_(current \_USMillion$)  
## [9] is.outlier is.extreme   
## <0 rows> (or 0-length row.names)

# Normality assumptions  
model <- lm(Export~Country,data=data\_task)  
ggqqplot(residuals(model))



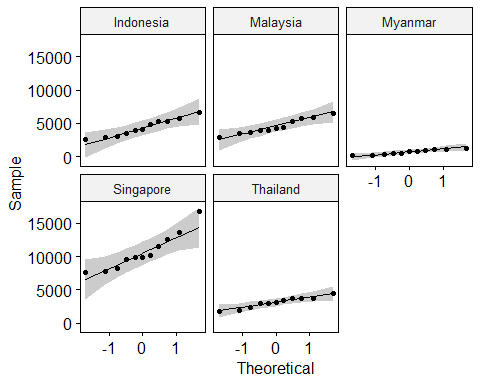
shapiro\_test(residuals(model))

## # A tibble: 1 x 3  
## variable statistic p.value  
## <chr> <dbl> <dbl>  
## 1 residuals(model) 0.922 0.00155

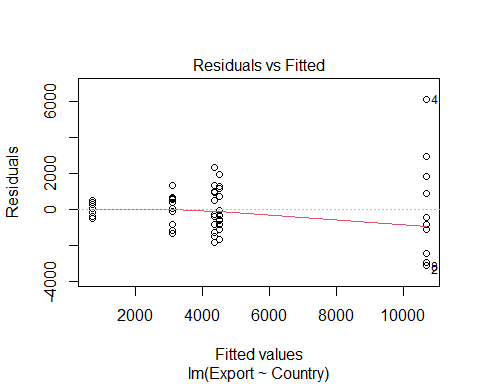
# Check normality assumption by group  
data\_task %>% group\_by(Country) %>% shapiro\_test(Export)

## # A tibble: 5 x 4  
## Country variable statistic p  
## <chr> <chr> <dbl> <dbl>  
## 1 Indonesia Export 0.960 0.772  
## 2 Malaysia Export 0.952 0.664  
## 3 Myanmar Export 0.925 0.366  
## 4 Singapore Export 0.910 0.246  
## 5 Thailand Export 0.946 0.592

ggqqplot(data\_task, "Export",facet.by = "Country")



# Homogeneity of variance assuptions  
# residual vs fit plot  
plot(model,1)



# Levence test for homogeneity of variance  
data\_task %>% levene\_test(Export~ Country)

## Warning in leveneTest.default(y = y, group = group, ...): group coerced to  
## factor.

## # A tibble: 1 x 4  
## df1 df2 statistic p  
## <int> <int> <dbl> <dbl>  
## 1 4 50 3.94 0.00746

res.aov <- data\_task %>% anova\_test(Export~ Country)

## Coefficient covariances computed by hccm()

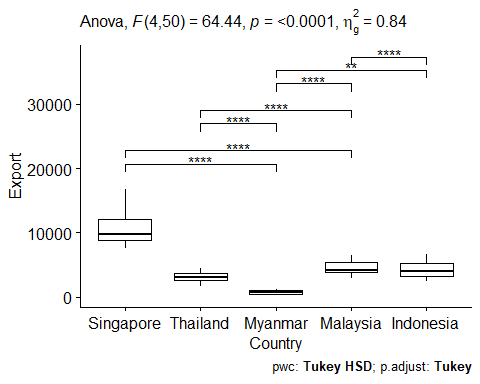
res.aov

## ANOVA Table (type II tests)  
##   
## Effect DFn DFd F p p<.05 ges  
## 1 Country 4 50 64.439 4.08e-19 \* 0.838

# post hoc test for pairwise comparison  
pwc <- data\_task %>% tukey\_hsd(Export~ Country)  
pwc

## # A tibble: 10 x 9  
## term group1 group2 null.value estimate conf.low conf.high p.adj  
## \* <chr> <chr> <chr> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 Coun~ Indon~ Malay~ 0 174. -1664. 2011. 9.99e- 1  
## 2 Coun~ Indon~ Myanm~ 0 -3637. -5474. -1799. 8.77e- 6  
## 3 Coun~ Indon~ Singa~ 0 6327. 4490. 8164. 0.   
## 4 Coun~ Indon~ Thail~ 0 -1251. -3088. 586. 3.17e- 1  
## 5 Coun~ Malay~ Myanm~ 0 -3810. -5648. -1973. 3.41e- 6  
## 6 Coun~ Malay~ Singa~ 0 6153. 4316. 7991. 4.99e-12  
## 7 Coun~ Malay~ Thail~ 0 -1425. -3262. 413. 1.99e- 1  
## 8 Coun~ Myanm~ Singa~ 0 9964. 8126. 11801. 0.   
## 9 Coun~ Myanm~ Thail~ 0 2386. 548. 4223. 5.05e- 3  
## 10 Coun~ Singa~ Thail~ 0 -7578. -9415. -5741. 0.   
## # ... with 1 more variable: p.adj.signif <chr>

# visualisation of box plot with p-values  
pwc <- pwc %>% add\_xy\_position(x= "Country")   
ggboxplot(data\_task, x="Country", y = "Export")+  
stat\_pvalue\_manual(pwc,hide.ns = TRUE) + labs(subtitle = get\_test\_label(res.aov,detailed = TRUE),caption = get\_pwc\_label(pwc))



data\_task\_sing <- filter(data\_task,Country=="Singapore")  
data\_task\_sing

## # A tibble: 11 x 8  
## Year Country Export Import Population `Population\_in ~ `GDP \_(current ~  
## <chr> <chr> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 2008~ Singap~ 8209. 7514. 4839396 4.84 193611986713.  
## 2 2009~ Singap~ 7577. 6455. 4987573 4.99 194152286009.  
## 3 2010~ Singap~ 9818. 7143. 5076732 5.08 239809387605.  
## 4 2011~ Singap~ 16769. 8500. 5183688 5.18 279351168707.  
## 5 2012~ Singap~ 13600. 7492. 5312437 5.31 295087220933.  
## 6 2013~ Singap~ 12511. 6763. 5399162 5.40 307576360585.  
## 7 2014~ Singap~ 9809. 7124. 5469724 5.47 314851156183.  
## 8 2015~ Singap~ 7720. 7308. 5535002 5.54 308004146058.  
## 9 2016~ Singap~ 9565. 7087. 5607283 5.61 318068476294.  
## 10 2017~ Singap~ 10203. 7467. 5612253 5.61 338406474039.  
## 11 2018~ Singap~ 11571. 16282. 5638676 5.64 364156657770.  
## # ... with 1 more variable: `GDP \_(current \_USMillion$)` <dbl>

data\_task\_thai <- filter(data\_task,Country=="Thailand")  
data\_task\_thai

## # A tibble: 11 x 8  
## Year Country Export Import Population `Population\_in ~ `GDP \_(current ~  
## <chr> <chr> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 2008~ Thaila~ 1897. 2686. 66530984 66.5 291383081232.  
## 2 2009~ Thaila~ 1734. 2927. 66866839 66.9 281710095725.  
## 3 2010~ Thaila~ 2271. 4271. 67195028 67.2 341105009515.  
## 4 2011~ Thaila~ 2974. 5381. 67195028 67.2 370818747397.  
## 5 2012~ Thaila~ 3733. 5351. 67835957 67.8 397558094270.  
## 6 2013~ Thaila~ 3703. 5340. 68144501 68.1 420333333333.  
## 7 2014~ Thaila~ 3465. 5866. 68438730 68.4 407339361696.  
## 8 2015~ Thaila~ 2988. 5510. 68714511 68.7 401295970240.  
## 9 2016~ Thaila~ 3133. 5415. 68971331 69.0 412352789520.  
## 10 2017~ Thaila~ 3654. 7134. 69209858 69.2 455275517239.  
## 11 2018~ Thaila~ 4441. 7442. 69428524 69.4 504992757705.  
## # ... with 1 more variable: `GDP \_(current \_USMillion$)` <dbl>

res<- t.test(data\_task\_sing$Population,data\_task\_thai$Population)  
res

##   
## Welch Two Sample t-test  
##   
## data: data\_task\_sing$Population and data\_task\_thai$Population  
## t = -201.34, df = 11.528, p-value < 2.2e-16  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## -63397170 -62033623  
## sample estimates:  
## mean of x mean of y   
## 5332902 68048299

data\_task\_2 <-rbind(data\_task\_sing,data\_task\_thai)  
data\_task\_2

## # A tibble: 22 x 8  
## Year Country Export Import Population `Population\_in ~ `GDP \_(current ~  
## <chr> <chr> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 2008~ Singap~ 8209. 7514. 4839396 4.84 193611986713.  
## 2 2009~ Singap~ 7577. 6455. 4987573 4.99 194152286009.  
## 3 2010~ Singap~ 9818. 7143. 5076732 5.08 239809387605.  
## 4 2011~ Singap~ 16769. 8500. 5183688 5.18 279351168707.  
## 5 2012~ Singap~ 13600. 7492. 5312437 5.31 295087220933.  
## 6 2013~ Singap~ 12511. 6763. 5399162 5.40 307576360585.  
## 7 2014~ Singap~ 9809. 7124. 5469724 5.47 314851156183.  
## 8 2015~ Singap~ 7720. 7308. 5535002 5.54 308004146058.  
## 9 2016~ Singap~ 9565. 7087. 5607283 5.61 318068476294.  
## 10 2017~ Singap~ 10203. 7467. 5612253 5.61 338406474039.  
## # ... with 12 more rows, and 1 more variable: `GDP \_(current  
## # \_USMillion$)` <dbl>

total\_trade <- data\_task$Export-data\_task$Import  
total\_trade

## [1] 694.70 1122.39 2674.55 8269.75 6107.61 5748.01 2684.89  
## [8] 411.43 2478.01 2735.83 -4710.50 -789.00 -1193.19 -2000.25  
## [15] -2406.77 -1617.86 -1636.97 -2401.05 -2522.30 -2281.96 -3480.63  
## [22] -3000.41 -707.33 -1081.83 -697.05 -835.77 -868.03 -608.66  
## [29] -458.30 86.38 40.64 326.55 684.11 -3655.20 -2316.47  
## [36] -2648.81 -5567.91 -5523.83 -5032.14 -5301.19 -5377.00 -3708.73  
## [43] -3310.02 -4383.35 -4168.70 -5565.42 -4216.49 -7858.12 -9551.49  
## [50] -9898.31 -10961.32 -10312.44 -9939.87 -12475.03 -10568.34

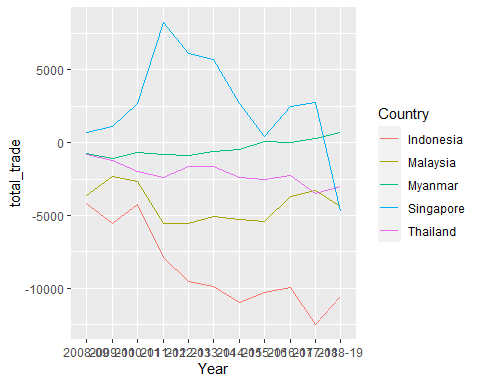
data\_task\_3 <- cbind(data\_task,total\_trade)  
class(data\_task\_3)

## [1] "data.frame"

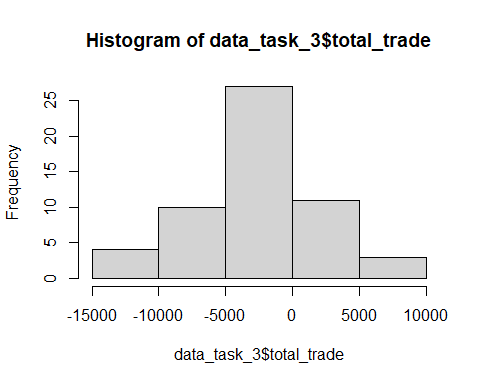
res1<- data\_task\_2 %>% t\_test(Population~Country,var.equal = TRUE,detailed = TRUE) %>% add\_significance()  
res1

## # A tibble: 1 x 16  
## estimate estimate1 estimate2 .y. group1 group2 n1 n2 statistic  
## \* <dbl> <dbl> <dbl> <chr> <chr> <chr> <int> <int> <dbl>  
## 1 -6.27e7 5332902. 68048299. Popu~ Singa~ Thail~ 11 11 -201.  
## # ... with 7 more variables: p <dbl>, df <dbl>, conf.low <dbl>,  
## # conf.high <dbl>, method <chr>, alternative <chr>, p.signif <chr>

ggplot(data\_task\_3,aes(x=Year,y=total\_trade,colour=`Country`,group=`Country`))+geom\_line()



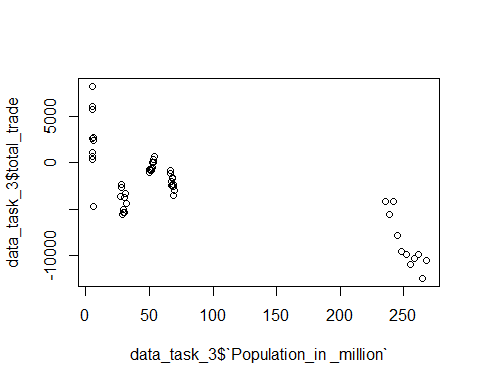
hist(data\_task\_3$total\_trade)



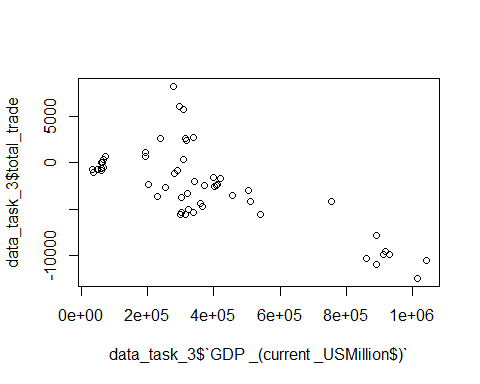
cor(data\_task\_3$`Population\_in \_million`,data\_task\_3$`GDP \_(current \_USMillion$)`)

## [1] 0.846537

plot(data\_task\_3$`Population\_in \_million`,data\_task\_3$total\_trade)



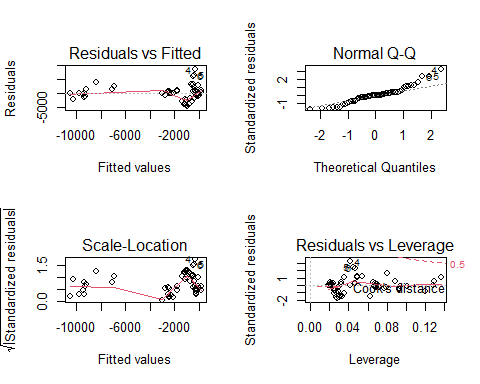
plot(data\_task\_3$`GDP \_(current \_USMillion$)`,data\_task\_3$total\_trade)



model <- lm(total\_trade ~ `Population\_in \_million` + `GDP \_(current \_USMillion$)`,data=data\_task\_3)  
summary(model)

##   
## Call:  
## lm(formula = total\_trade ~ `Population\_in \_million` + `GDP \_(current \_USMillion$)`,   
## data = data\_task\_3)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -4621.2 -1311.5 -92.0 922.6 8573.4   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1.321e+03 6.455e+02 2.047 0.0458 \*   
## `Population\_in \_million` -2.314e+01 7.851e+00 -2.947 0.0048 \*\*  
## `GDP \_(current \_USMillion$)` -5.387e-03 2.534e-03 -2.126 0.0383 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 2725 on 52 degrees of freedom  
## Multiple R-squared: 0.6177, Adjusted R-squared: 0.603   
## F-statistic: 42 on 2 and 52 DF, p-value: 1.391e-11

par(mfrow=c(2,2))  
plot(model)



par(mfrow=c(1,1))

broom::glance(model)

## # A tibble: 1 x 12  
## r.squared adj.r.squared sigma statistic p.value df logLik AIC BIC  
## <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 0.618 0.603 2725. 42.0 1.39e-11 2 -512. 1031. 1039.  
## # ... with 3 more variables: deviance <dbl>, df.residual <int>, nobs <int>

cor(data\_task\_3$`Population\_in \_million`,data\_task\_3$`GDP \_(current \_USMillion$)`)

## [1] 0.846537

cor(data\_task\_3$`Population\_in \_million`,data\_task\_3$total\_trade)

## [1] -0.7644972

cor(data\_task\_3$`GDP \_(current \_USMillion$)`,data\_task\_3$total\_trade)

## [1] -0.7441967