**Forecast the CocaCola prices data set. Prepare a document for each model explaining how many dummy variables you have created and RMSE value for each model. Finally which model you will use for Forecasting.**

**Ans:**

> library(forecast)

> library(timeSeries)

> cococola <- readxl::read\_xlsx(file.choose())

> cococola$Quarter

[1] "Q1\_86" "Q2\_86" "Q3\_86" "Q4\_86" "Q1\_87" "Q2\_87" "Q3\_87" "Q4\_87" "Q1\_88" "Q2\_88"

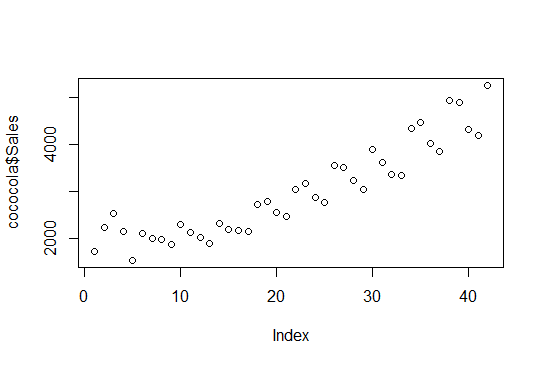
[11] "Q3\_88" "Q4\_88" "Q1\_89" "Q2\_89" "Q3\_89" "Q4\_89" "Q1\_90" "Q2\_90" "Q3\_90" "Q4\_90"

[21] "Q1\_91" "Q2\_91" "Q3\_91" "Q4\_91" "Q1\_92" "Q2\_92" "Q3\_92" "Q4\_92" "Q1\_93" "Q2\_93"

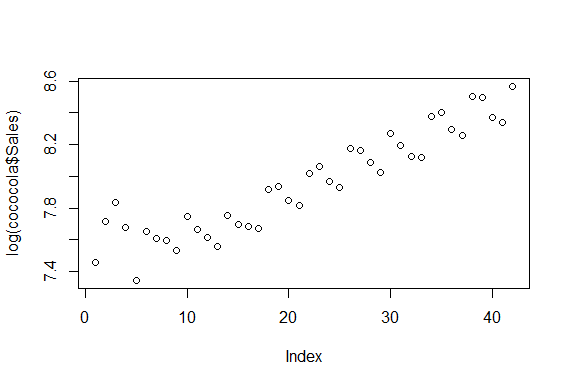
[31] "Q3\_93" "Q4\_93" "Q1\_94" "Q2\_94" "Q3\_94" "Q4\_94" "Q1\_95" "Q2\_95" "Q3\_95" "Q4\_95"

[41] "Q1\_96" "Q2\_96"

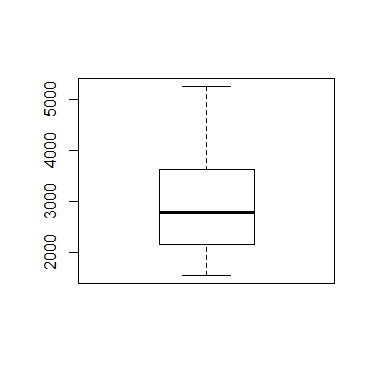
> plot(cococola$Sales)



> plot(log(cococola$Sales))



> boxplot(cococola$Sales)



**From Boxplot, No Outliers in df**

> str(cococola)

Classes ‘tbl\_df’, ‘tbl’ and 'data.frame': 42 obs. of 2 variables:

$ Quarter: chr "Q1\_86" "Q2\_86" "Q3\_86" "Q4\_86" ...

$ Sales : num 1735 2245 2534 2155 1548 ...

**Pre-processing data**

> quaterly <- data.frame(outer(rep(c("Q1","Q2","Q3","Q4"),length= nrow(cococola)),c("Q1","Q2","Q3","Q4"),'==')+0)

> colnames(quaterly) <- c("Q1","Q2","Q3","Q4")

> cococola2 <- cbind(cococola,quaterly)

> cococola2["t"] <- 1:42

> cococola2["log"] <- log(cococola2$Sales)

> cococola2["t\_sq"] <- (cococola2$t)\*(cococola2$t)

> head(cococola2)

Quarter Sales Q1 Q2 Q3 Q4 t log t\_sq

1 Q1\_86 1734.827 1 0 0 0 1 7.458663 1

2 Q2\_86 2244.961 0 1 0 0 2 7.716443 4

3 Q3\_86 2533.805 0 0 1 0 3 7.837477 9

4 Q4\_86 2154.963 0 0 0 1 4 7.675529 16

5 Q1\_87 1547.819 1 0 0 0 5 7.344602 25

6 Q2\_87 2104.412 0 1 0 0 6 7.651791 36

> attach(cococola2)

**Splitting of data**

> trn <- cococola2[1:30,]

> tst <- cococola2[31:42,]

**1. Linear model**

> linear\_mod <- lm(Sales~t, data = trn)

> summary(linear\_mod)

Call:

lm(formula = Sales ~ t, data = trn)

Residuals:

Min 1Q Median 3Q Max

-454.21 -267.67 -52.73 162.30 726.27

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) 1640.354 120.095 13.659 6.62e-14 \*\*\*

t 55.727 6.765 8.238 5.77e-09 \*\*\*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 320.7 on 28 degrees of freedom

Multiple R-squared: 0.7079, Adjusted R-squared: 0.6975

F-statistic: 67.86 on 1 and 28 DF, p-value: 5.766e-09

**Here, R^2 = 0.7079**

> linear\_pred <- data.frame(predict(linear\_mod,interval = 'predict',newdata=tst))

> linear\_rmse <- sqrt(mean((tst$Sales-linear\_pred$fit)^2,na.rm = T))

> linear\_rmse

[1] 714.0144

**RMSE = 714.0144**

**2. exp model**

> exp\_model <- lm(log~t , data = trn)

> summary(exp\_model)

Call:

lm(formula = log ~ t, data = trn)

Residuals:

Min 1Q Median 3Q Max

-0.22826 -0.08760 -0.01667 0.07490 0.30797

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) 7.464474 0.046851 159.325 < 2e-16 \*\*\*

t 0.021677 0.002639 8.214 6.11e-09 \*\*\*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.1251 on 28 degrees of freedom

Multiple R-squared: 0.7067, Adjusted R-squared: 0.6962

F-statistic: 67.47 on 1 and 28 DF, p-value: 6.112e-09

**Here, R^2 = 0.7067**

> exp\_pred <- data.frame(predict(exp\_model,interval = 'predict',newdata = tst))

> exp\_rmse <- sqrt(mean((tst$Sales-exp\_pred$fit)^2,na.rm = T))

> exp\_rmse

[1] 4252.189

**RMSE = 4252.189**

**3. Quadratic model**

> quad\_mod <- lm(Sales~t+t\_sq,data = trn)

> summary(quad\_mod)

Call:

lm(formula = Sales ~ t + t\_sq, data = trn)

Residuals:

Min 1Q Median 3Q Max

-493.8 -182.5 -26.1 214.2 503.9

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) 2092.2906 154.5963 13.534 1.51e-13 \*\*\*

t -29.0108 22.9889 -1.262 0.217760

t\_sq 2.7335 0.7196 3.799 0.000752 \*\*\*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 263.6 on 27 degrees of freedom

Multiple R-squared: 0.8097, Adjusted R-squared: 0.7956

F-statistic: 57.42 on 2 and 27 DF, p-value: 1.879e-10

**Here, R^2 = 0.8097**

> quad\_pred <- data.frame(predict(quad\_mod,interval = "predict",newdata = tst))

> quad\_rmse <- sqrt(mean((tst$Sales-quad\_pred$fit)^2,na.rm = T))

> quad\_rmse

[1] 646.2715

**RMSE = 646.2715**

**4. Additive seasonality**

> add\_seas <- lm(Sales~Q1+Q2+Q3+Q4,data = trn)

> summary(add\_seas)

Call:

lm(formula = Sales ~ Q1 + Q2 + Q3 + Q4, data = trn)

Residuals:

Min 1Q Median 3Q Max

-673.3 -445.1 -173.4 407.8 1121.2

Coefficients: (1 not defined because of singularities)

Estimate Std. Error t value Pr(>|t|)

(Intercept) 2432.3 214.0 11.365 1.39e-11 \*\*\*

Q1 -242.4 293.1 -0.827 0.416

Q2 345.4 293.1 1.179 0.249

Q3 189.9 302.7 0.627 0.536

Q4 NA NA NA NA

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 566.3 on 26 degrees of freedom

Multiple R-squared: 0.1544, Adjusted R-squared: 0.05684

F-statistic: 1.583 on 3 and 26 DF, p-value: 0.2175

**Here, R^2 = 0.1544**

> add\_seas\_pred <- data.frame(predict(add\_seas,interval = "predict",newdata = tst))

Warning message:

In predict.lm(add\_seas, interval = "predict", newdata = tst) :

prediction from a rank-deficient fit may be misleading

> add\_seas\_rmse <- sqrt(mean((tst$Sales-add\_seas\_pred$fit)^2,na.rm = T))

> add\_seas\_rmse

[1] 1778.007

**RMSE = 1778.007**

**5. Additive seasonality with linear**

> add\_seast <- lm(Sales~t+Q1+Q2+Q3+Q4,data = trn)

> summary(add\_seast)

Call:

lm(formula = Sales ~ t + Q1 + Q2 + Q3 + Q4, data = trn)

Residuals:

Min 1Q Median 3Q Max

-415.72 -151.18 -45.05 139.33 573.47

Coefficients: (1 not defined because of singularities)

Estimate Std. Error t value Pr(>|t|)

(Intercept) 1549.763 125.094 12.389 3.62e-12 \*\*\*

t 55.161 5.212 10.583 1.01e-10 \*\*\*

Q1 -187.235 127.780 -1.465 0.1553

Q2 345.410 127.673 2.705 0.0121 \*

Q3 245.094 131.963 1.857 0.0751 .

Q4 NA NA NA NA

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 246.7 on 25 degrees of freedom

Multiple R-squared: 0.8457, Adjusted R-squared: 0.821

F-statistic: 34.25 on 4 and 25 DF, p-value: 8.282e-10

**Here, R^2 = 0.8457**

> add\_seast\_pred <- data.frame(predict(add\_seast,interval = "predict",newdata = tst))

Warning message:

In predict.lm(add\_seast, interval = "predict", newdata = tst) :

prediction from a rank-deficient fit may be misleading

> add\_seast\_rmse <- sqrt(mean((tst$Sales-add\_seast\_pred$fit)^2,na.rm = T))

> add\_seast\_rmse

[1] 637.9405

**RMSE = 637.9405**

**6. additive seasonality with quadratic**

> add\_seasq <- lm(Sales~t+t\_sq+Q1+Q2+Q3+Q4,data = trn)

> summary(add\_seasq)

Call:

lm(formula = Sales ~ t + t\_sq + Q1 + Q2 + Q3 + Q4, data = trn)

Residuals:

Min 1Q Median 3Q Max

-233.63 -96.29 16.93 79.27 311.71

Coefficients: (1 not defined because of singularities)

Estimate Std. Error t value Pr(>|t|)

(Intercept) 2050.5047 97.0953 21.118 < 2e-16 \*\*\*

t -33.0378 12.1860 -2.711 0.012191 \*

t\_sq 2.8451 0.3816 7.455 1.07e-07 \*\*\*

Q1 -244.1372 72.0253 -3.390 0.002419 \*\*

Q2 288.5079 71.9661 4.009 0.000515 \*\*\*

Q3 245.0939 73.9647 3.314 0.002913 \*\*

Q4 NA NA NA NA

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 138.3 on 24 degrees of freedom

Multiple R-squared: 0.9535, Adjusted R-squared: 0.9438

F-statistic: 98.34 on 5 and 24 DF, p-value: 3.503e-15

**Here, R^2 = 0.9535**

> add\_seasq\_pred <- data.frame(predict(add\_seasq,interval = "predict",newdata = tst))

Warning message:

In predict.lm(add\_seasq, interval = "predict", newdata = tst) :

prediction from a rank-deficient fit may be misleading

> add\_seasq\_rmse <- sqrt(mean((tst$Sales-add\_seasq\_pred$fit)^2,na.rm = T))

> add\_seasq\_rmse

[1] 586.0533

**RMSE =586.0533**

**7. Multiplicative seasonality**

> mul\_seas\_model <- lm(log~Q1+Q2+Q3+Q4,data = trn)

> summary(mul\_seas\_model)

Call:

lm(formula = log ~ Q1 + Q2 + Q3 + Q4, data = trn)

Residuals:

Min 1Q Median 3Q Max

-0.32198 -0.16488 -0.05584 0.17563 0.36260

Coefficients: (1 not defined because of singularities)

Estimate Std. Error t value Pr(>|t|)

(Intercept) 7.78096 0.08256 94.250 <2e-16 \*\*\*

Q1 -0.11438 0.11305 -1.012 0.321

Q2 0.12491 0.11305 1.105 0.279

Q3 0.07154 0.11675 0.613 0.545

Q4 NA NA NA NA

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.2184 on 26 degrees of freedom

Multiple R-squared: 0.1699, Adjusted R-squared: 0.07414

F-statistic: 1.774 on 3 and 26 DF, p-value: 0.1768

**Here, R^2 = 0.1699**

> mul\_seas\_pred <- data.frame(predict(mul\_seas\_model,interval = 'predict',newdata = tst))

Warning message:

In predict.lm(mul\_seas\_model, interval = "predict", newdata = tst) :

prediction from a rank-deficient fit may be misleading

> mul\_seas\_rmse <- sqrt(mean((tst$Sales-mul\_seas\_pred$fit)^2,na.rm = T))

> mul\_seas\_rmse

[1] 4252.639

**RMSE =4252.639**

**8. Multiplicative seasonality with linear**

> mul\_seast\_model <- lm(log~t+Q1+Q2+Q3+Q4,data = trn)

> summary(mul\_seast\_model)

Call:

lm(formula = log ~ t + Q1 + Q2 + Q3 + Q4, data = trn)

Residuals:

Min 1Q Median 3Q Max

-0.153321 -0.061340 -0.008758 0.051417 0.242178

Coefficients: (1 not defined because of singularities)

Estimate Std. Error t value Pr(>|t|)

(Intercept) 7.438022 0.046624 159.530 < 2e-16 \*\*\*

t 0.021434 0.001943 11.033 4.25e-11 \*\*\*

Q1 -0.092951 0.047626 -1.952 0.0623 .

Q2 0.124913 0.047586 2.625 0.0146 \*

Q3 0.092976 0.049185 1.890 0.0704 .

Q4 NA NA NA NA

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.09194 on 25 degrees of freedom

Multiple R-squared: 0.8586, Adjusted R-squared: 0.8359

F-statistic: 37.94 on 4 and 25 DF, p-value: 2.826e-10

**Here, R^2 = 0.8586**

> mul\_seast\_pred <- data.frame(predict(mul\_seast\_model,interval = 'predict',newdata = tst))

Warning message:

In predict.lm(mul\_seast\_model, interval = "predict", newdata = tst) :

prediction from a rank-deficient fit may be misleading

> mul\_seast\_rmse <- sqrt(mean((tst$Sales-mul\_seast\_pred$fit)^2,na.rm = T))

> mul\_seast\_rmse

[1] 4252.186

**RMSE = 4252.186**

> table\_formate <- data.frame(c("linear\_rmse","exp\_rmse","quad\_rmse","add\_seas\_rmse","add\_seasq\_rmse","add\_seast\_rmse","mul\_seas\_rmse","mul\_seast\_rmse"),c(linear\_rmse,exp\_rmse,quad\_rmse,add\_seas\_rmse,add\_seasq\_rmse,add\_seast\_rmse,mul\_seas\_rmse,mul\_seast\_rmse))

> colnames(table\_formate) <- c("model","RMSE")

> View(table\_formate)

> table\_formate

model RMSE

1 linear\_rmse 714.0144

2 exp\_rmse 4252.1890

3 quad\_rmse 646.2715

4 add\_seas\_rmse 1778.0065

5 add\_seasq\_rmse 586.0533

6 add\_seast\_rmse 637.9405

7 mul\_seas\_rmse 4252.6387

8 mul\_seast\_rmse 4252.1857

**Final model**

> finalmodel <- lm(Sales~t+t\_sq+Q1+Q2+Q3+Q4,data = cococola2)

> summary(finalmodel)

Call:

lm(formula = Sales ~ t + t\_sq + Q1 + Q2 + Q3 + Q4, data = cococola2)

Residuals:

Min 1Q Median 3Q Max

-331.45 -103.31 23.02 119.30 344.12

Coefficients: (1 not defined because of singularities)

Estimate Std. Error t value Pr(>|t|)

(Intercept) 1849.5889 98.7974 18.721 < 2e-16 \*\*\*

t -2.6187 9.0018 -0.291 0.772791

t\_sq 1.7581 0.2031 8.658 2.52e-10 \*\*\*

Q1 -222.1355 75.3113 -2.950 0.005563 \*\*

Q2 417.0186 75.2795 5.540 2.87e-06 \*\*\*

Q3 332.1306 76.8621 4.321 0.000117 \*\*\*

Q4 NA NA NA NA

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 171.8 on 36 degrees of freedom

Multiple R-squared: 0.9729, Adjusted R-squared: 0.9691

F-statistic: 258.5 on 5 and 36 DF, p-value: < 2.2e-16

**Auto.arima**

> cococola3 <- as.ts(cococola$Sales)

> cococola\_ts <- ts(data = cococola3,start = c(1986,1),end = c(1996,2),frequency = 4)

> class(cococola\_ts)

[1] "ts"

> cycle(cococola\_ts)

Qtr1 Qtr2 Qtr3 Qtr4

1986 1 2 3 4

1987 1 2 3 4

1988 1 2 3 4

1989 1 2 3 4

1990 1 2 3 4

1991 1 2 3 4

1992 1 2 3 4

1993 1 2 3 4

1994 1 2 3 4

1995 1 2 3 4

1996 1 2

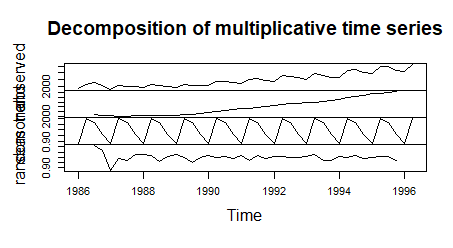
> summary(cococola\_ts)

Min. 1st Qu. Median Mean 3rd Qu. Max.

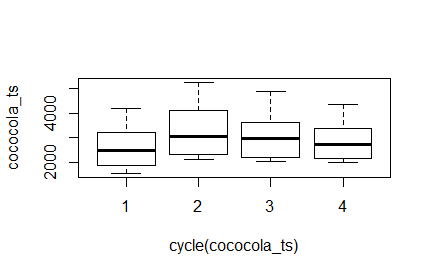
1548 2160 2782 2994 3609 5253

> decomp <- decompose(cococola\_ts,"multiplicative")

> plot(decomp)



> boxplot(cococola\_ts~cycle(cococola\_ts))



**Auto arima model**

> arimamodel <- auto.arima(cococola\_ts,ic="aic",trace = T)

ARIMA(2,1,2)(1,1,1)[4] : Inf

ARIMA(0,1,0)(0,1,0)[4] : 488.6475

ARIMA(1,1,0)(1,1,0)[4] : 491.971

ARIMA(0,1,1)(0,1,1)[4] : 491.8391

ARIMA(0,1,0)(1,1,0)[4] : 490.0841

ARIMA(0,1,0)(0,1,1)[4] : 490.058

ARIMA(0,1,0)(1,1,1)[4] : 491.9662

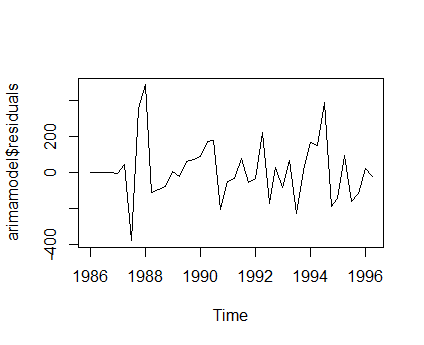
ARIMA(1,1,0)(0,1,0)[4] : 490.5959

ARIMA(0,1,1)(0,1,0)[4] : 490.5453

ARIMA(1,1,1)(0,1,0)[4] : Inf

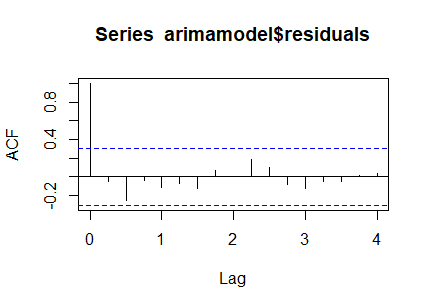
Best model: ARIMA(0,1,0)(0,1,0)[4]

> plot.ts(arimamodel$residuals)

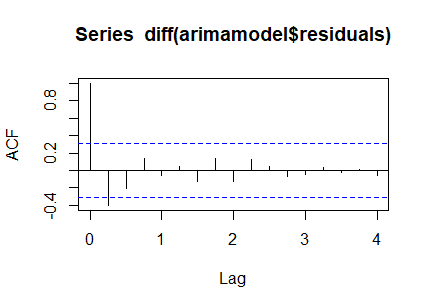


**Verifying p,d,q**

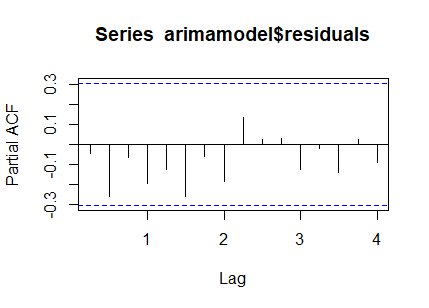
> acf(arimamodel$residuals)#q=0



> acf(diff(arimamodel$residuals))#d=1



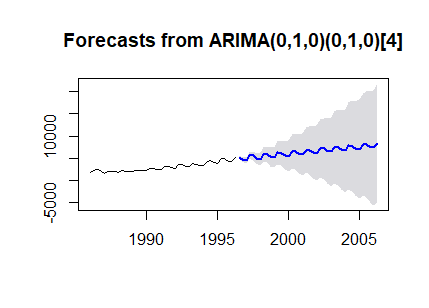
> pacf(arimamodel$residuals)#p=0



**Forecasting**

> forecastmodel <- forecast(arimamodel,level = c(95),h=10\*4)

> plot(forecastmodel)



**Testing**

> Box.test(arimamodel$residuals,lag=5,type = "Ljung-Box")

Box-Ljung test

data: arimamodel$residuals

X-squared = 3.9366, df = 5, p-value = 0.5586

> Box.test(arimamodel$residuals,lag=10,type="Ljung-Box")

Box-Ljung test

data: arimamodel$residuals

X-squared = 7.4886, df = 10, p-value = 0.6787

**P values are smaller.**