DADS Experiment No: 4

Name: Abhishek S Waghchaure

PRN: **1032221714**

Dept: M Tech DSA

Aim:

By Using suitable real world Time Series Data, determining the data trend.

Objective:

To understand how time series works, what factors are affecting a certain variable(s) at different points of time.

Time series analysis will provide the consequences and insights of features of the given dataset that changes over time.

Supporting to derive the predicting the future values of the time series variable.

Objective of our practical is to provide a reasonable forecast for future sales.

Dataset used:

A Company ABC selling tractors has to forecast its sales for the next 24 months, It has 12 years of past sales data on monthly basis. The data may contain trend, seasonality, or both. Objective is to provide a reasonable forecast for future sales.

Our dataset contains the number of tractors sold per month by year. We have dataset of tractors sold from Jan 2003 to December 2014.

Attributes:

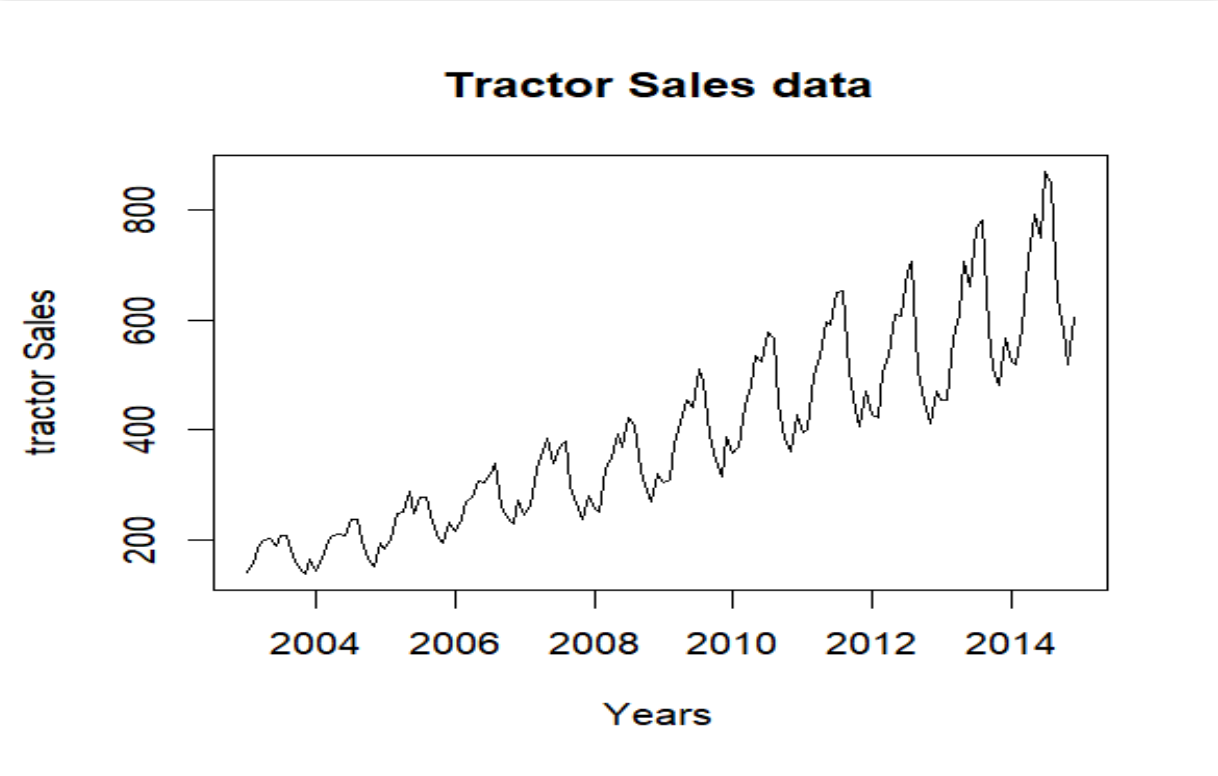
- Month.year
- Number.of.tractor.sold

Sample data:

Month-Yea	Number of Tractor Sol
Jan-03	141
Feb-03	157
Mar-03	185
Apr-03	199
May-03	203
Jun-03	189
Jul-03	207
Aug-03	207
Sep-03	171
Oct-03	150
Nov-03	138
Dec-03	165
Jan-04	145
Feb-04	168
Mar-04	197
Apr-04	208
May-04	210
Jun-04	209
Jul-04	238
Aug-04	238
Sep-04	199
Oct-04	168
Nov-04	152
Dec-04	196
Jan-05	183
Feb-05	200
Mar-05	249
Apr-05	251
May-05	289
Jun-05	249

Code & Visualization techniques for the dataset:

```
> library(ggplot2)

> library(fpp2) # examine seasonal graphically

> library(forecast)

> library(stats)

> library(tseries)

> Tractor_Sales <- read.csv("Tractor-Sales.csv")

> View(Tractor_Sales)

> #convert data into timeseries format

> TractorSalesTS <- ts(Tractor_Sales[,2], start = c(2003,1),frequency = 12)

> #plot the timeseries data

> plot(TractorSalesTS, xlab = 'Years', ylab= "tractor Sales", main = "Tractor Sales data")
```

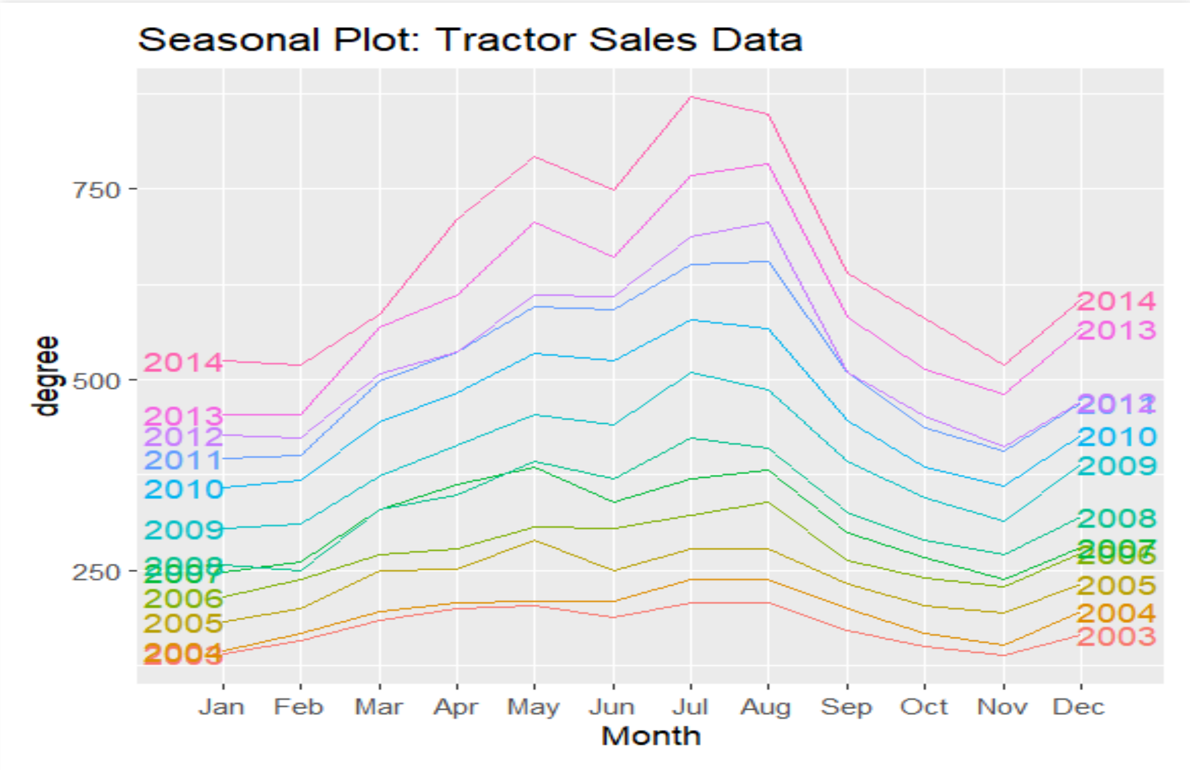


> TractorSalesTS

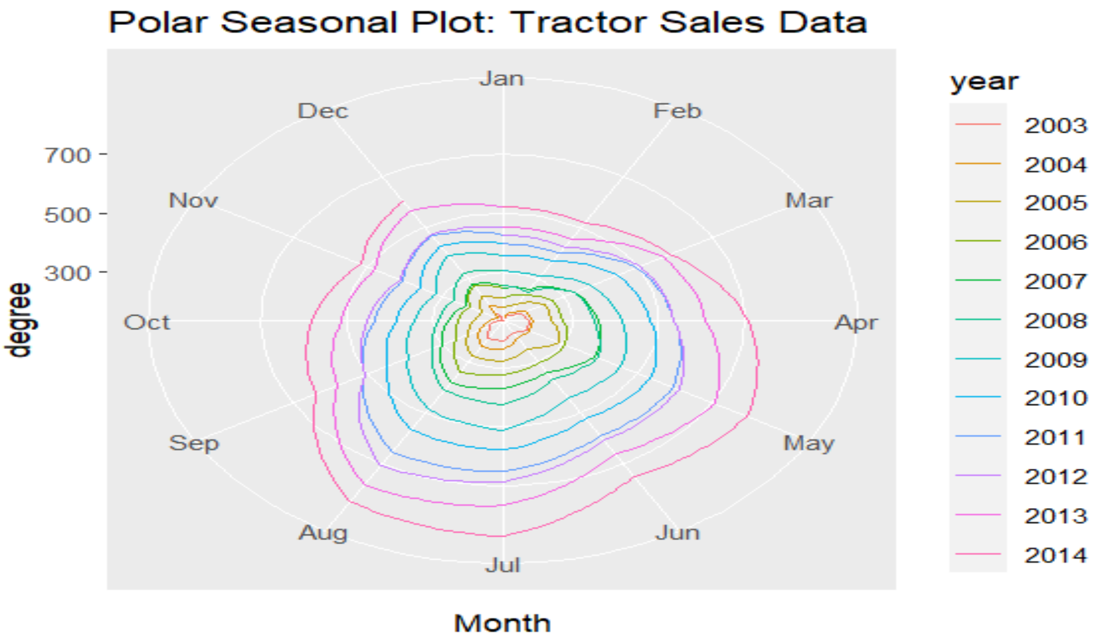
	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
2003	141	157	185	199	203	189	207	207	171	150	138	165
2004	145	168	197	208	210	209	238	238	199	168	152	196
2005	183	200	249	251	289	249	279	279	232	204	194	232
2006	215	239	270	279	307	305	322	339	263	241	229	272
2007	247	261	330	362	385	340	370	381	299	266	239	281
2008	257	250	329	350	393	370	423	410	326	289	270	321
2009	305	310	374	414	454	441	510	486	393	345	315	389
2010	358	368	444	482	534	524	578	567	447	386	360	428
2011	397	400	498	536	596	591	651	654	509	437	406	470
2012	428	423	507	536	610	609	687	707	509	452	412	472
2013	454	455	568	610	706	661	767	783	583	513	481	567
2014	525	520	587	710	793	749	871	848	640	581	519	605

> #Seasonal Plot Year-wise

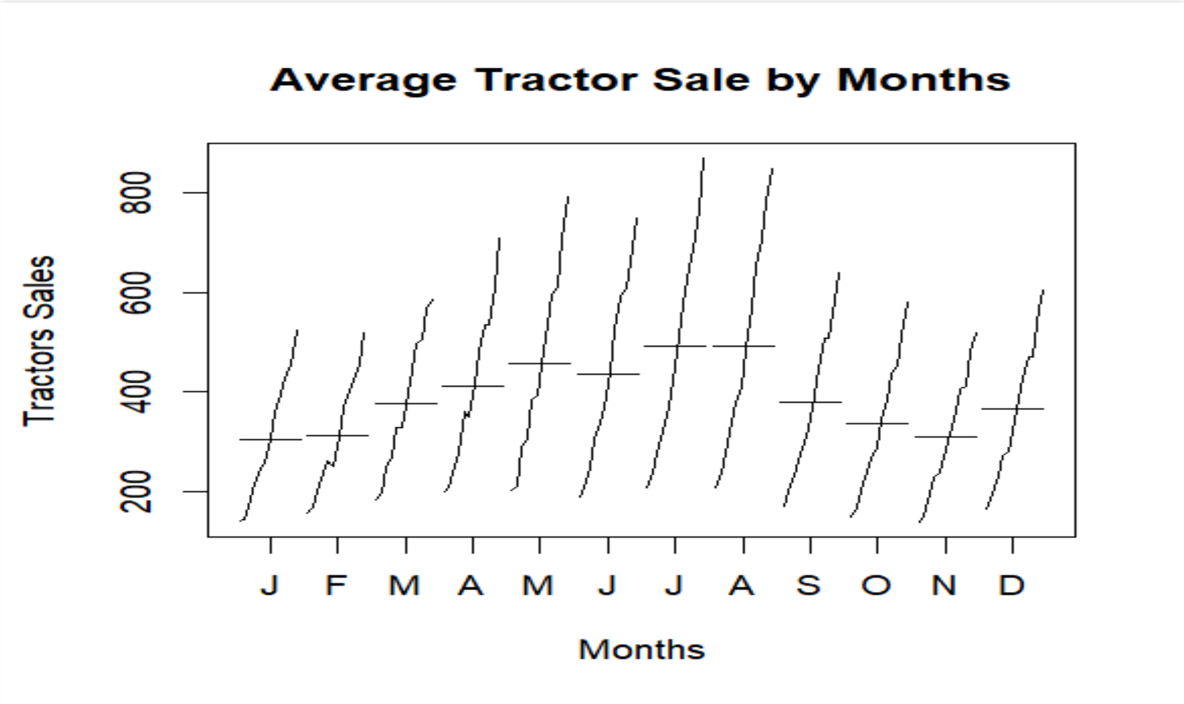
> ggseasonplot(TractorSalesTS, year.labels = TRUE, year.labels.left = TRUE) + ylab("degree") + ggtitle("Seasonal Plot: Tractor Sales Data")



> ggseasonplot(TractorSalesTS,polar = TRUE) + ylab("degree") + ggtitle("Polar Seasonal Plot: Tractor Sales Data")



> monthplot(TractorSalesTS, xlab = 'Months', ylab = 'Tractors Sales', main = "Average Tractor Sale by Months")



> TSDecompose <- decompose(TractorSalesTS, type = "multiplicative")

> TSDecompose

$x

Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec

2003 141 157 185 199 203 189 207 207 171 150 138 165

2004 145 168 197 208 210 209 238 238 199 168 152 196

2005 183 200 249 251 289 249 279 279 232 204 194 232

2006 215 239 270 279 307 305 322 339 263 241 229 272

2007 247 261 330 362 385 340 370 381 299 266 239 281

2008 257 250 329 350 393 370 423 410 326 289 270 321

2009 305 310 374 414 454 441 510 486 393 345 315 389

2010 358 368 444 482 534 524 578 567 447 386 360 428

2011 397 400 498 536 596 591 651 654 509 437 406 470

2012 428 423 507 536 610 609 687 707 509 452 412 472

2013 454 455 568 610 706 661 767 783 583 513 481 567

2014 525 520 587 710 793 749 871 848 640 581 519 605

$seasonal

Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov

2003 0.8233333 0.8438594 1.0123702 1.0805564 1.1857930 1.1209126 1.2360238 1.2301349 0.9619639 0.8364099 0.7655329

2004 0.8233333 0.8438594 1.0123702 1.0805564 1.1857930 1.1209126 1.2360238 1.2301349 0.9619639 0.8364099 0.7655329

2005 0.8233333 0.8438594 1.0123702 1.0805564 1.1857930 1.1209126 1.2360238 1.2301349 0.9619639 0.8364099 0.7655329

2006 0.8233333 0.8438594 1.0123702 1.0805564 1.1857930 1.1209126 1.2360238 1.2301349 0.9619639 0.8364099 0.7655329

2007 0.8233333 0.8438594 1.0123702 1.0805564 1.1857930 1.1209126 1.2360238 1.2301349 0.9619639 0.8364099 0.7655329

2008 0.8233333 0.8438594 1.0123702 1.0805564 1.1857930 1.1209126 1.2360238 1.2301349 0.9619639 0.8364099 0.7655329

2009 0.8233333 0.8438594 1.0123702 1.0805564 1.1857930 1.1209126 1.2360238 1.2301349 0.9619639 0.8364099 0.7655329

2010 0.8233333 0.8438594 1.0123702 1.0805564 1.1857930 1.1209126 1.2360238 1.2301349 0.9619639 0.8364099 0.7655329

2011 0.8233333 0.8438594 1.0123702 1.0805564 1.1857930 1.1209126 1.2360238 1.2301349 0.9619639 0.8364099 0.7655329

2012 0.8233333 0.8438594 1.0123702 1.0805564 1.1857930 1.1209126 1.2360238 1.2301349 0.9619639 0.8364099 0.7655329

2013 0.8233333 0.8438594 1.0123702 1.0805564 1.1857930 1.1209126 1.2360238 1.2301349 0.9619639 0.8364099 0.7655329

2014 0.8233333 0.8438594 1.0123702 1.0805564 1.1857930 1.1209126 1.2360238 1.2301349 0.9619639 0.8364099 0.7655329

Dec

2003 0.9031095

2004 0.9031095

2005 0.9031095

2006 0.9031095

2007 0.9031095

2008 0.9031095

2009 0.9031095

2010 0.9031095

2011 0.9031095

2012 0.9031095

2013 0.9031095

2014 0.9031095

$trend

Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec

2003 NA NA NA NA NA NA 176.1667 176.7917 177.7500 178.6250 179.2917 180.4167

2004 182.5417 185.1250 187.5833 189.5000 190.8333 192.7083 195.5833 198.5000 202.0000 205.9583 211.0417 216.0000

2005 219.3750 222.7917 225.8750 228.7500 232.0000 235.2500 238.0833 241.0417 243.5417 245.5833 247.5000 250.5833

2006 254.7083 259.0000 262.7917 265.6250 268.6250 271.7500 274.7500 277.0000 280.4167 286.3750 293.0833 297.7917

2007 301.2500 305.0000 308.2500 310.7917 312.2500 313.0417 313.8333 313.7917 313.2917 312.7500 312.5833 314.1667

2008 317.6250 321.0417 323.3750 325.4583 327.7083 330.6667 334.3333 338.8333 343.2083 347.7500 352.9583 358.4583

2009 365.0417 371.8333 377.7917 382.9167 387.1250 391.8333 396.8750 401.5000 406.8333 412.5833 418.7500 425.5417

2010 431.8333 438.0417 443.6667 447.6250 451.2083 454.7083 457.9583 460.9167 464.5000 469.0000 473.8333 479.2083

2011 485.0417 491.7083 497.9167 502.6250 506.6667 510.3333 513.3750 515.6250 516.9583 517.3333 517.9167 519.2500

2012 521.5000 525.2083 527.4167 528.0417 528.9167 529.2500 530.4167 532.8333 536.7083 542.3333 549.4167 555.5833

2013 561.0833 567.5833 573.8333 579.4583 584.8750 591.7083 598.6250 604.2917 607.7917 612.7500 620.5417 627.8333

2014 635.8333 642.8750 647.9583 653.1667 657.5833 660.7500 NA NA NA NA NA NA

$random

Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov

2003 NA NA NA NA NA NA 0.9506481 0.9518221 1.0000639 1.0039910 1.0054376

2004 0.9647844 1.0754101 1.0373675 1.0157964 0.9280175 0.9675514 0.9845058 0.9746837 1.0241013 0.9752383 0.9408307

2005 1.0131838 1.0638023 1.0889096 1.0154655 1.0505119 0.9442738 0.9480874 0.9409344 0.9902753 0.9931437 1.0239120

2006 1.0252260 1.0935233 1.0148756 0.9720482 0.9637914 1.0012869 0.9481812 0.9948719 0.9749743 1.0061501 1.0206586

2007 0.9958506 1.0140761 1.0574784 1.0779330 1.0397990 0.9689581 0.9538407 0.9870311 0.9921186 1.0168693 0.9987762

2008 0.9827493 0.9228019 1.0049631 0.9952340 1.0113376 0.9982505 1.0236085 0.9836599 0.9874179 0.9935999 0.9992553

2009 1.0148029 0.9879689 0.9778672 1.0005726 0.9889987 1.0040734 1.0396559 0.9840065 1.0041931 0.9997427 0.9826341

2010 1.0069112 0.9955482 0.9885231 0.9965182 0.9980568 1.0280793 1.0211159 1.0000182 1.0003755 0.9840004 0.9924600

2011 0.9941130 0.9640117 0.9879463 0.9869002 0.9920077 1.0331462 1.0259341 1.0310768 1.0235368 1.0099312 1.0240054

2012 0.9968131 0.9544180 0.9495433 0.9393970 0.9725987 1.0265608 1.0478829 1.0786369 0.9858723 0.9964441 0.9795610

2013 0.9827720 0.9499738 0.9777396 0.9742270 1.0179648 0.9966026 1.0366059 1.0533250 0.9971375 1.0009557 1.0125355

2014 1.0028600 0.9585322 0.8948529 1.0059743 1.0169825 1.0112834 NA NA NA NA NA

Dec

2003 1.0126675

2004 1.0047590

2005 1.0251688

2006 1.0113837

2007 0.9903890

2008 0.9915759

2009 1.0122018

2010 0.9889606

2011 1.0022613

2012 0.9407027

2013 0.9999961

2014 NA

$figure

[1] 0.8233333 0.8438594 1.0123702 1.0805564 1.1857930 1.1209126 1.2360238 1.2301349 0.9619639 0.8364099 0.7655329

[12] 0.9031095

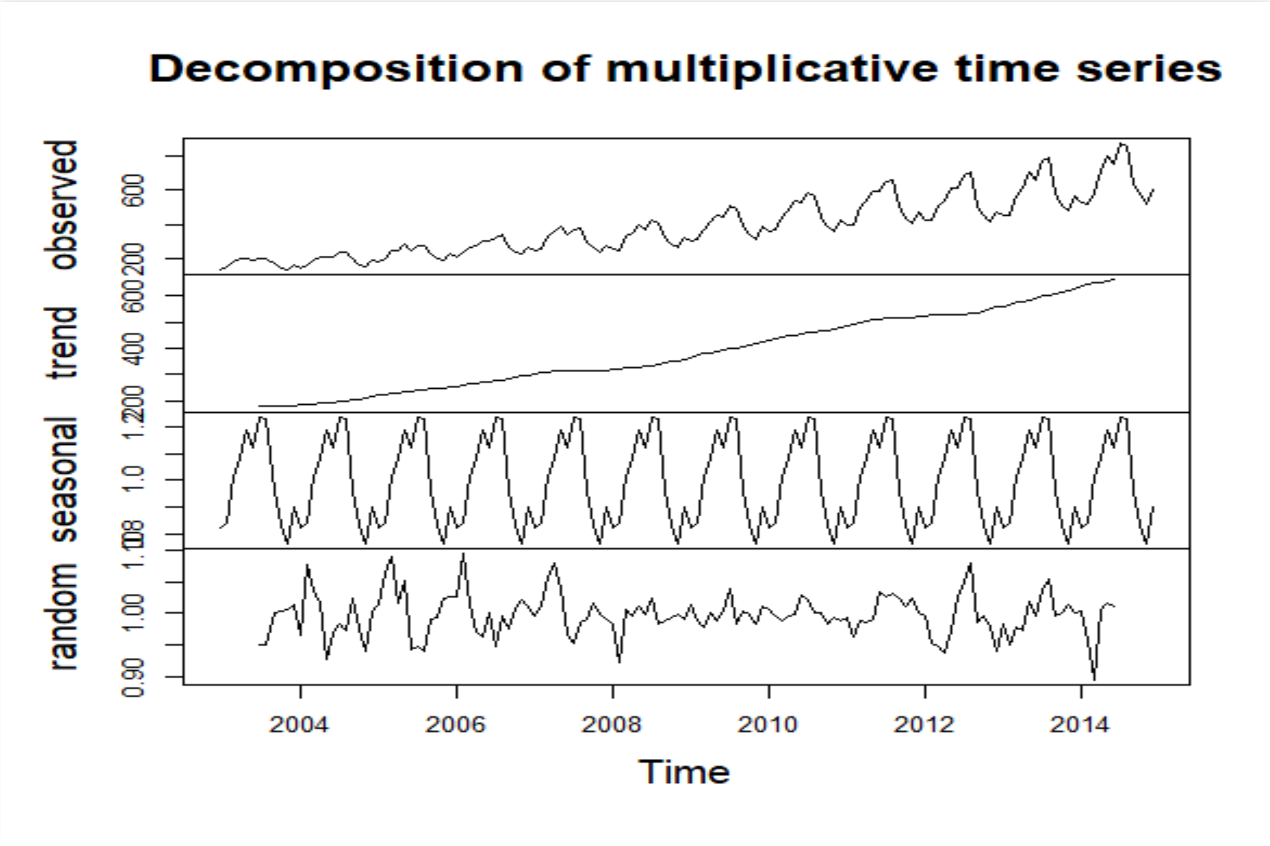
$type

[1] "multiplicative"

attr(,"class")

[1] "decomposed.ts"

> plot(TSDecompose)

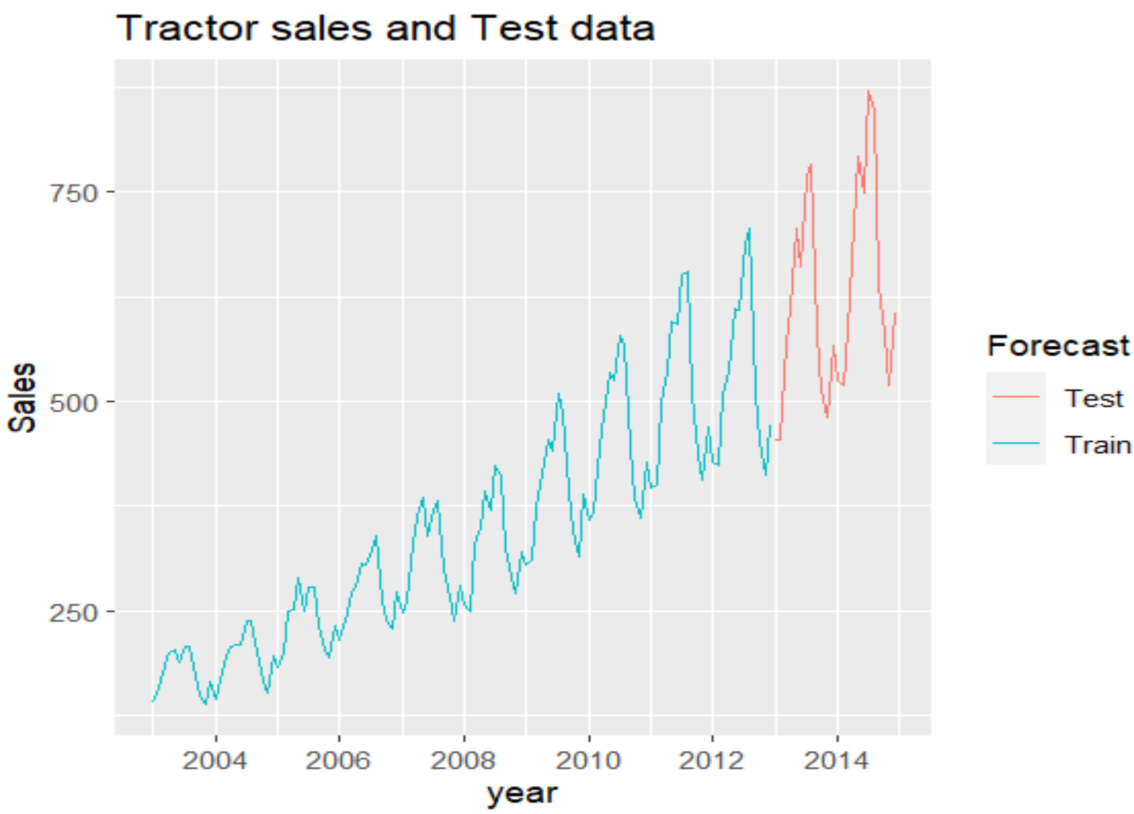



```
> #splitting data into train and test sets
> Ts_Train <- window(TractorSalesTS, start = c(2003,1), end = c(2012,12),freq = 12)
> Ts_Test <- window(TractorSalesTS, start = c(2013,1), freq = 12)
> Ts_Train
     Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
2003 141 157 185 199 203 189 207 207 171 150 138 165
2004 145 168 197 208 210 209 238 238 199 168 152 196
2005 183 200 249 251 289 249 279 279 232 204 194 232
2006 215 239 270 279 307 305 322 339 263 241 229 272
2007 247 261 330 362 385 340 370 381 299 266 239 281
2008 257 250 329 350 393 370 423 410 326 289 270 321
2009 305 310 374 414 454 441 510 486 393 345 315 389
2010 358 368 444 482 534 524 578 567 447 386 360 428
2011 397 400 498 536 596 591 651 654 509 437 406 470
2012 428 423 507 536 610 609 687 707 509 452 412 472
```

> Ts_Test

	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
2013	454	455	568	610	706	661	767	783	583	513	481	567
2014	525	520	587	710	793	749	871	848	640	581	519	605

> autoplot(Ts_Train, series = "Train") + autolayer(Ts_Test, series="Test") + ggtitle("Tractor sales and Test data") + xlab("year") + ylab("Sales") + guides(colour = guide_legend(title="Forecast"))

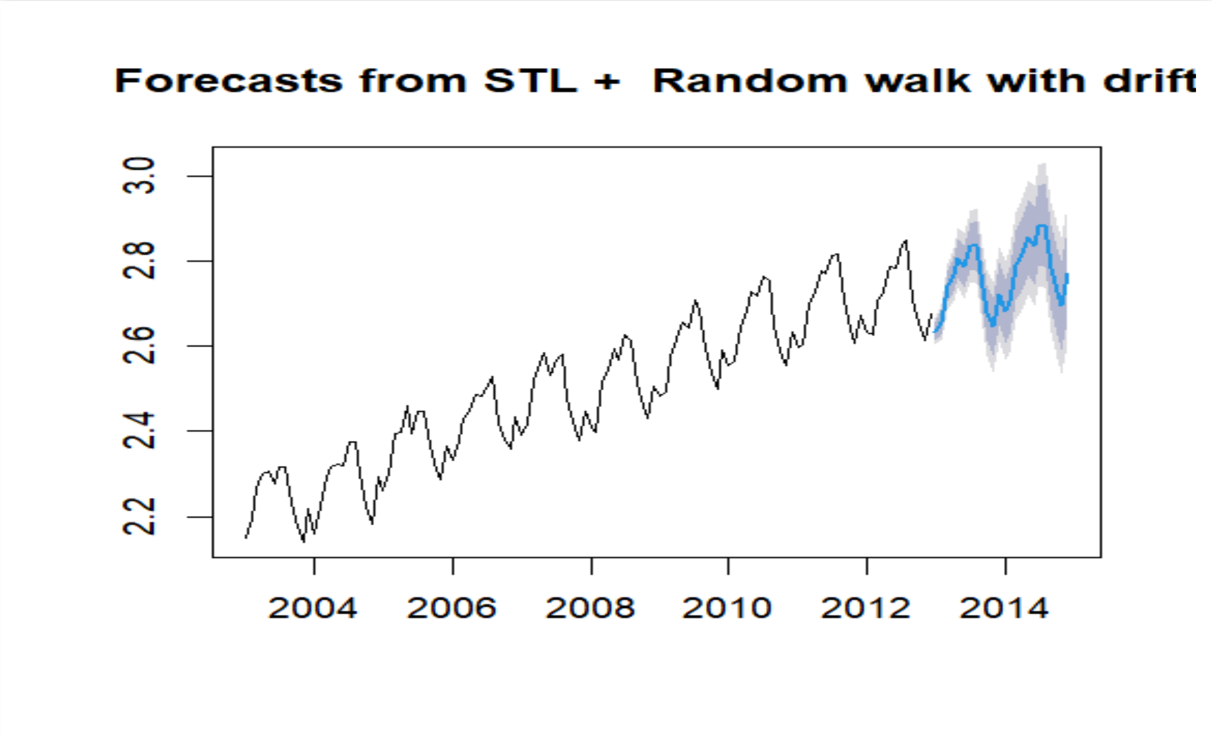


> autoplot(Ts_Train, series = "Train") + autolayer(Ts_Test, series="Test") + ggtitle("Tractor sales and Test data") + xlab("year") + ylab("Sales") + guides(colour = guide_legend(title="Forecast"))

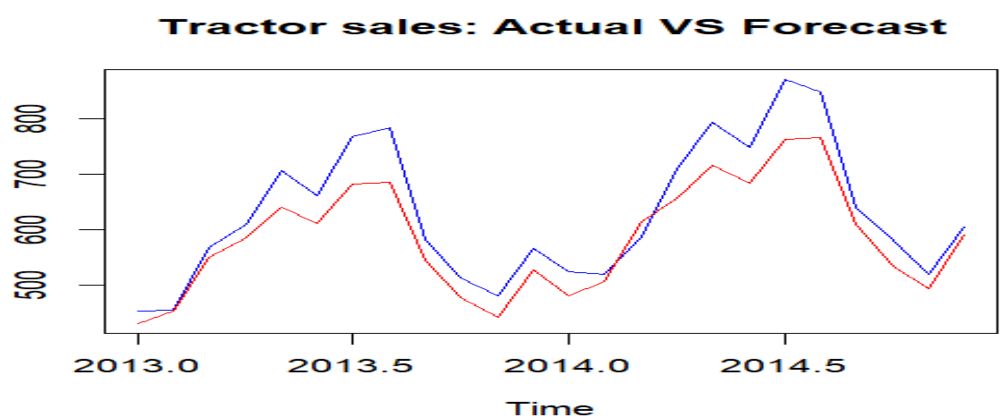
> TSDecompose_train_Log <- stl(log10(Ts_Train), s.window='p')

> TS_train_stl <- forecast(TSDecompose_train_Log, method = 'rwdrift', h =24)

> plot(TS_train_stl)

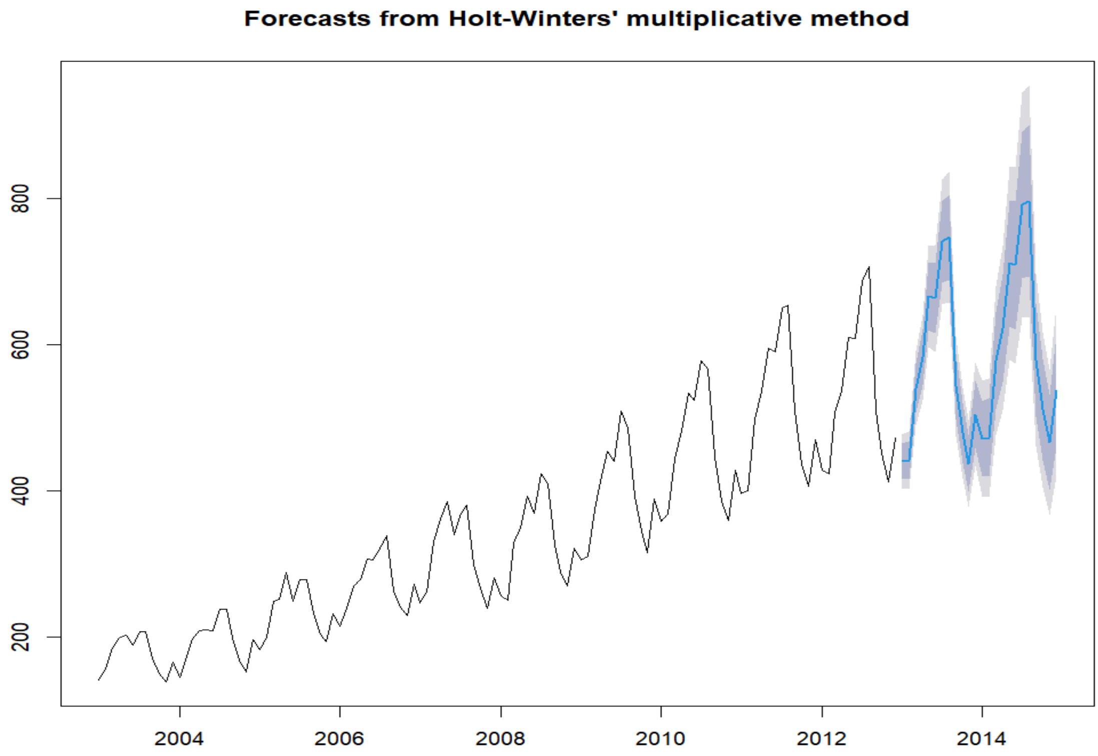



```
> Vec2 <- 10^(cbind(log10(Ts_Test),as.data.frame(forecast(TSDecompose_train_Log, method = 'rwdrift',h=24))[,1]))

> ts.plot(Vec2,col=c('blue','red'), main = "Tractor sales: Actual VS Forecast")
```




```
> RMSE2 <- round(sqrt(sum(((Vec2[,1]-Vec2[,2])^2)/length(Vec2[,1]))),4)

> MAPE2 <- round(mean(abs(Vec2[,1]-Vec2[,2])/Vec2[,1]),4)

> paste("Accuracy Measures: RMSE: ",RMSE2, 'and MAPE: ', MAPE2)

[1] "Accuracy Measures: RMSE:  53.5697 and MAPE:  0.0687"

> Ts_train_HW <- hw(Ts_Train, h=24, seasonal = 'multiplicative')
```

```
> holt(Ts_Train, h=24)
         Point Forecast    Lo 80    Hi 80     Lo 95     Hi 95
Jan 2013       475.2876 413.8077 536.7674 381.26220  569.3129
Feb 2013       478.5814 391.6360 565.5269 345.60982  611.5530
Mar 2013       481.8753 375.3856 588.3650 319.01336  644.7372
Apr 2013       485.1691 362.2001 608.1382 297.10414  673.2342
May 2013       488.4630 350.9728 625.9532 278.18993  698.7361
Jun 2013       491.7569 341.1365 642.3773 261.40286  722.1109
Jul 2013       495.0507 332.3539 657.7476 246.22739  743.8741
Aug 2013       498.3446 324.4055 672.2838 232.32764  764.3616
Sep 2013       501.6385 317.1386 686.1383 219.47030  783.8066
Oct 2013       504.9323 310.4423 699.4224 207.48552  802.3792
Nov 2013       508.2262 304.2327 712.2197 196.24507  820.2073
Dec 2013       511.5201 298.4447 724.5955 185.64935  837.3908
Jan 2014       514.8139 293.0264 736.6015 175.61914  854.0087
Feb 2014       518.1078 287.9359 748.2797 166.09023  870.1254
Mar 2014       521.4017 283.1386 759.6648 157.00968  885.7937
Apr 2014       524.6955 278.6055 770.7856 148.33327  901.0578
May 2014       527.9894 274.3122 781.6666 140.02359  915.9552
Jun 2014       531.2833 270.2378 792.3287 132.04866  930.5179
Jul 2014       534.5771 266.3642 802.7900 124.38089  944.7734
Aug 2014       537.8710 262.6758 813.0662 116.99628  958.7457
Sep 2014       541.1649 259.1588 823.1709 109.87380  972.4559
Oct 2014       544.4587 255.8010 833.1164 102.99489  985.9226
Nov 2014       547.7526 252.5918 842.9134  96.34309  999.1621
Dec 2014       551.0465 249.5214 852.5715  89.90369 1012.1892
> plot(Ts_train_HW)
```



Conclusion:

Time series analysis is a must for every company to understand seasonality, cyclicality, trend and randomness in the sales and other attributes.

Some of the benefits of Time Series are:

- Offers on non seasonal products.
- Seasonal performance of products.
- Better management of warehouse.
- Predict upcoming trends in fashion.
- Seasonal demand for flights can be forecasted, and acted accordingly with better offers and services.
- Based on the historical data, you see major accidents happen in particular route, in a particular season; you can avoid it.
- Mark down and mark up with changing seasons and coming festivals.

References:

- https://www.analyticsvidhya.com/blog/2021/10/a-comprehensive-guide-to-time-series-analysis/
- https://online.stat.psu.edu/stat510/book/export/html/661
- https://www.simplilearn.com/tutorials/python-tutorial/time-series-analysis-in-python#:~:text=Time%20Series%20Analysis%20in%20Python%20considers%20data%20collected%20over%20time,to%20extract%20its%20valuable%20characteristics.&text=Consider%20the%20running%20of%20a,to%20bake%20at%20what%20time
- https://www.kaggle.com/datasets/chirag19/air-passengers
- https://www.kaggle.com/code/prashant111/complete-guide-on-time-series-analysis-in-python/data