OPEX\_Forecast.Rolling.Stock.Focus

Jason Marshall

August 20, 2018

## Warning: Missing column names filled in: 'X1' [1]

## Parsed with column specification:  
## cols(  
## X1 = col\_integer(),  
## GL.Level = col\_character(),  
## Summary.Account = col\_character(),  
## Parent.Account = col\_character(),  
## Account = col\_character(),  
## Geography = col\_character(),  
## Region = col\_character(),  
## Division = col\_character(),  
## Period = col\_character(),  
## Acct.Per = col\_integer(),  
## Year = col\_integer(),  
## Currency = col\_character(),  
## Amount = col\_integer(),  
## day.holder = col\_integer(),  
## date.holder = col\_date(format = ""),  
## Date = col\_date(format = "")  
## )

## Warning in rbind(names(probs), probs\_f): number of columns of result is not  
## a multiple of vector length (arg 1)

## Warning: 32 parsing failures.  
## row # A tibble: 5 x 5 col row col expected actual file expected <int> <chr> <chr> <chr> <chr> actual 1 1005 Amount no trailing characters e+05 'Y:/Sharepoint Projects/Jaso~ file 2 15347 Amount no trailing characters e+05 'Y:/Sharepoint Projects/Jaso~ row 3 17619 Amount no trailing characters e+05 'Y:/Sharepoint Projects/Jaso~ col 4 30257 Amount no trailing characters e+05 'Y:/Sharepoint Projects/Jaso~ expected 5 35440 Amount no trailing characters e+05 'Y:/Sharepoint Projects/Jaso~  
## ... ................. ... .......................................................................... ........ .......................................................................... ...... .......................................................................... .... .......................................................................... ... .......................................................................... ... .......................................................................... ........ ..........................................................................  
## See problems(...) for more details.

## [1] "Alabama" "Central Cornbelt"   
## [3] "Central Indiana" "Central Iowa"   
## [5] "Central Valley" "Coastal"   
## [7] "Coastal Plains" "Colonial"   
## [9] "Eastern" "Eastern Cornbelt"   
## [11] "Eastern Illinois" "Eastern Kentucky"   
## [13] "Florida" "Georgia"   
## [15] "Michigan" "Mid-South"   
## [17] "Mid N Carolina - Virginia" "Minnesota - South Dakota"   
## [19] "Mississippi" "Nebraska"   
## [21] "North Dakota - NE MT - NW MN" "North Valley"   
## [23] "Northeast" "Northeast Iowa"   
## [25] "Northern High Plains" "Northern Illinois - Wisconsin"   
## [27] "Northern Ohio" "Northwest"   
## [29] "S Carolina - West N Carolina" "Snake River"   
## [31] "South" "South Delta"   
## [33] "South Plains" "Southeast Iowa"   
## [35] "Southern High Plains" "Southern Illinois"   
## [37] "Southern Ohio" "Southwest"   
## [39] "Southwest IN - Northern KY" "Tennessee - Western Kentucky"   
## [41] "United States" "West"   
## [43] "West Central Illinois" "West Iowa - Northeast Nebraska"  
## [45] "Western Cornbelt" "Western Illinois"

## Date Account   
## Min. :2013-01-31 520000 - Hourly Wages : 66   
## 1st Qu.:2014-05-31 520002 - Hourly Overtime : 66   
## Median :2015-10-15 520102 - Hourly Fringe Severance & Stay: 66   
## Mean :2015-10-15 520104 - Hourly Fringe Other : 66   
## 3rd Qu.:2017-02-28 520202 - Hourly Retirement 401k Savings: 66   
## Max. :2018-06-30 520301 - Hourly Ins LT Disability : 66   
## (Other) :4290   
## Geography Expense.Sum   
## United States:4686 Min. :-32513684   
## 1st Qu.: 0   
## Median : 168252   
## Mean : 4575309   
## 3rd Qu.: 4798552   
## Max. :100651144   
##

## [1] TRUE

## Warning in tk\_xts\_.data.frame(data = data, select = select, date\_var =  
## date\_var, : Non-numeric columns being dropped: Date

## [1] "2013-01-31"

## [1] "2018-06-30"

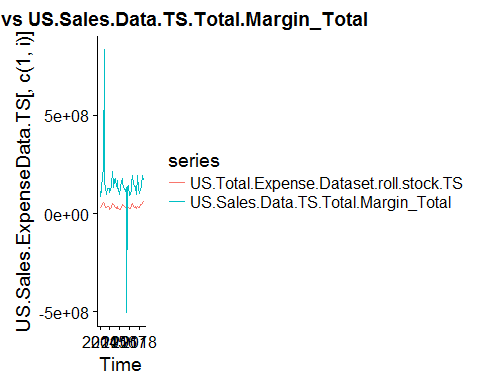
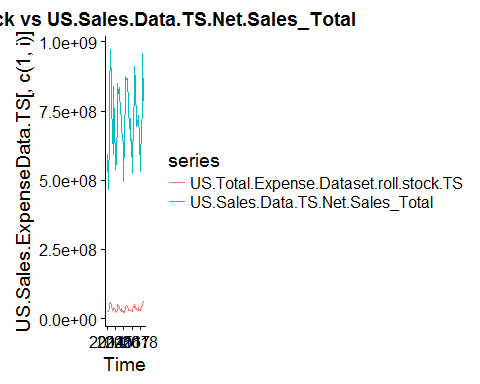
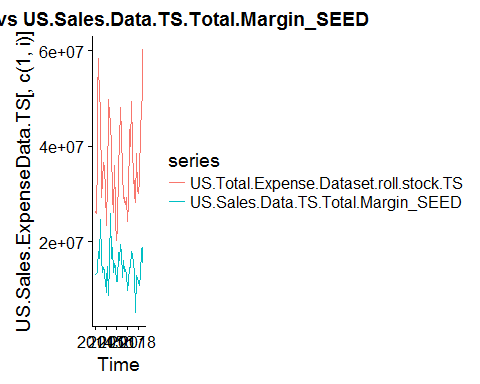
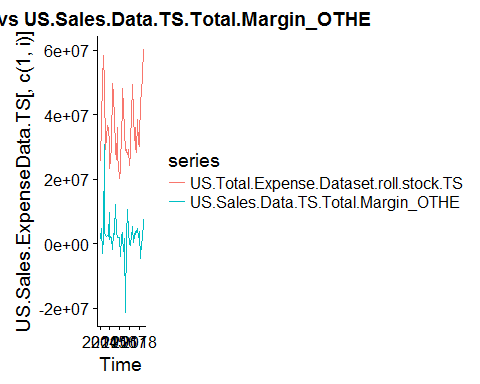
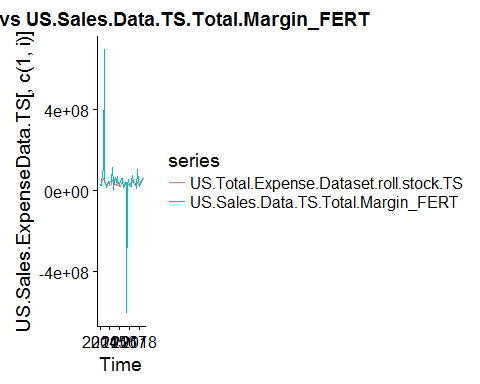
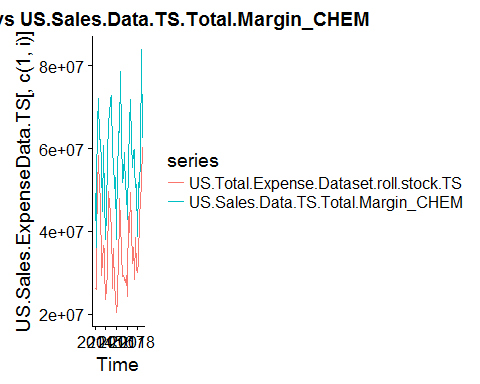
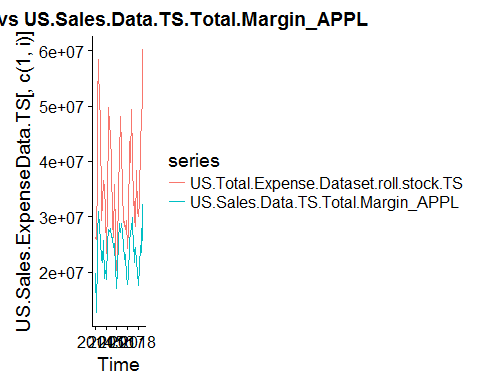
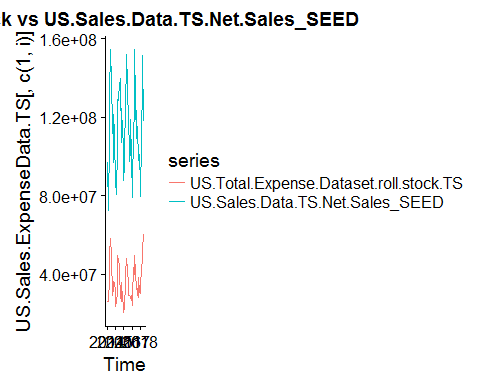
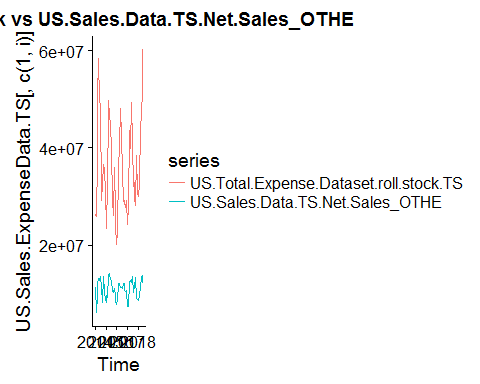
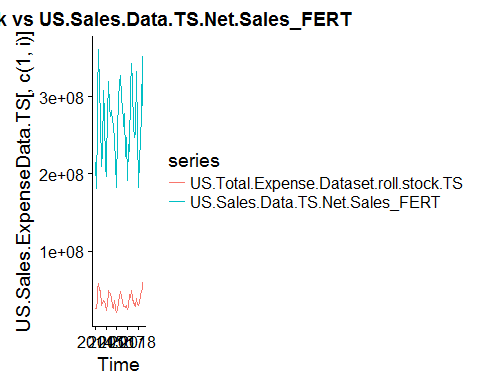
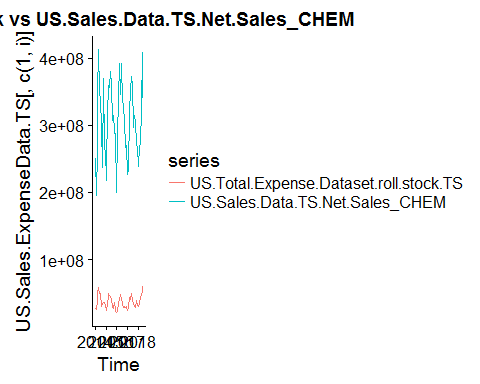
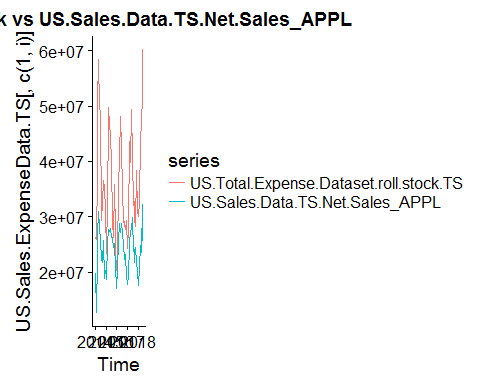
## Parsed with column specification:  
## cols(  
## `Invoice Date` = col\_character(),  
## Region = col\_character(),  
## `Region Number` = col\_integer(),  
## Division = col\_character(),  
## `Divsion Number` = col\_integer(),  
## `GL Shelf` = col\_character(),  
## `Net Sales` = col\_number(),  
## `Total Margin` = col\_number()  
## )

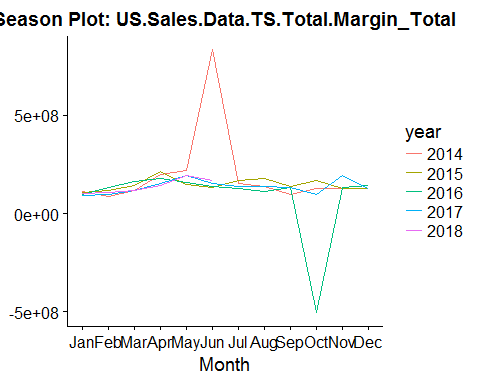
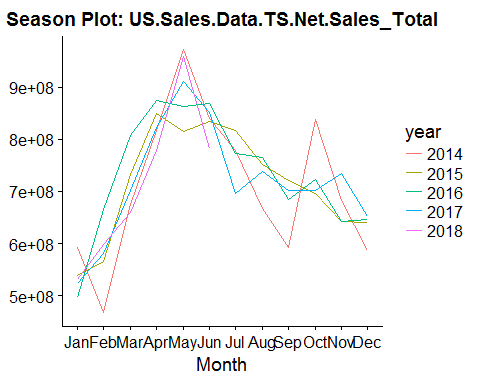
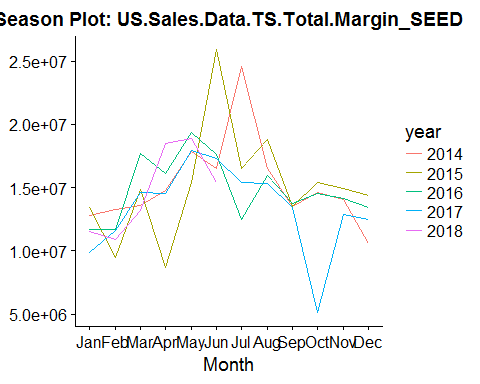
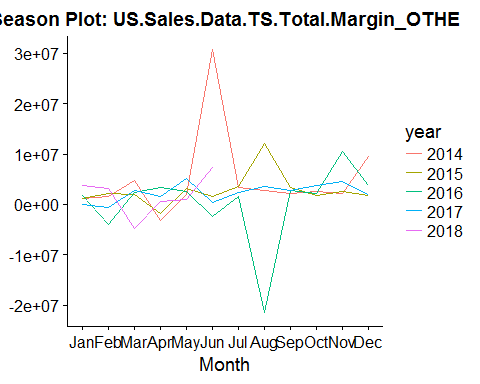
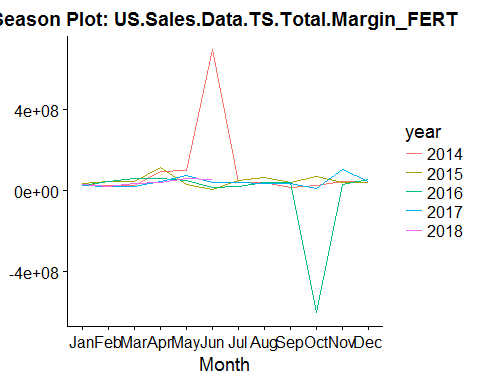
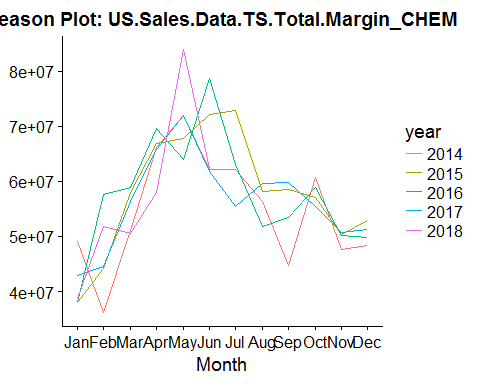
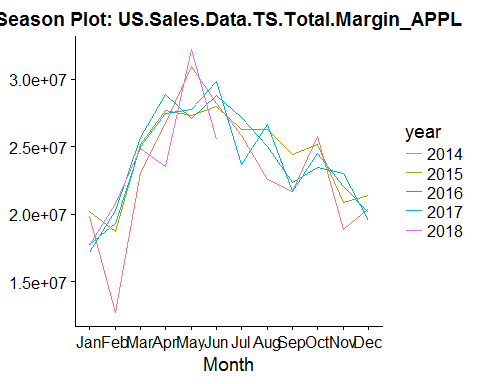
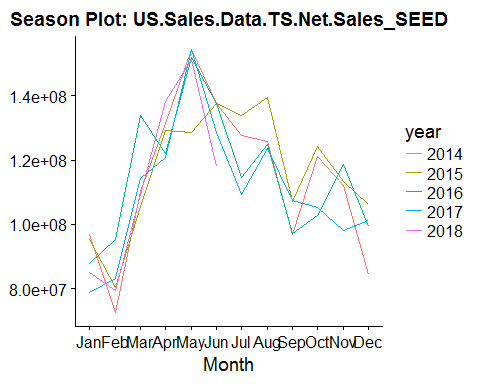
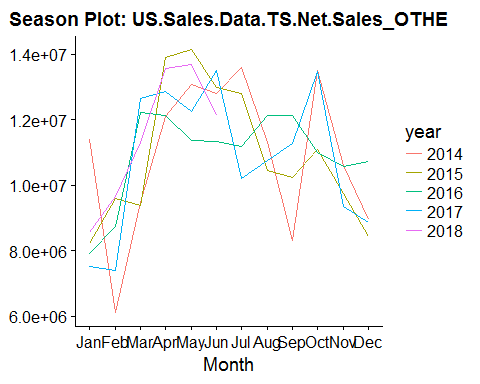
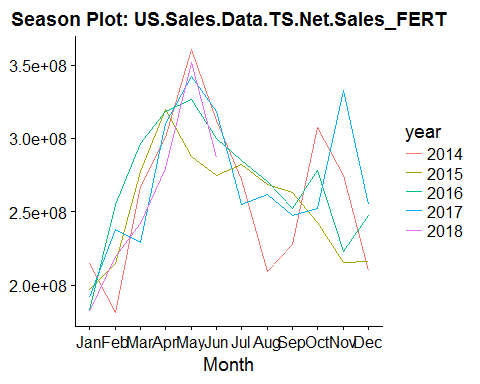
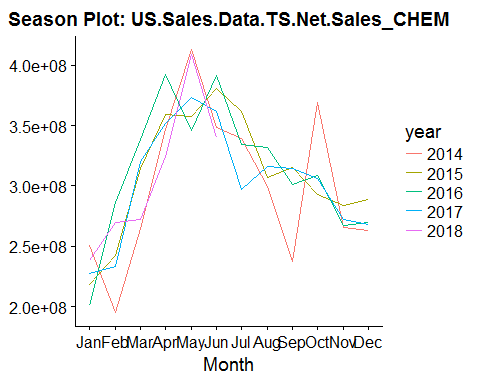
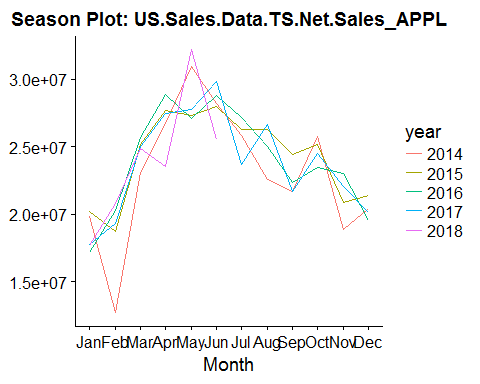
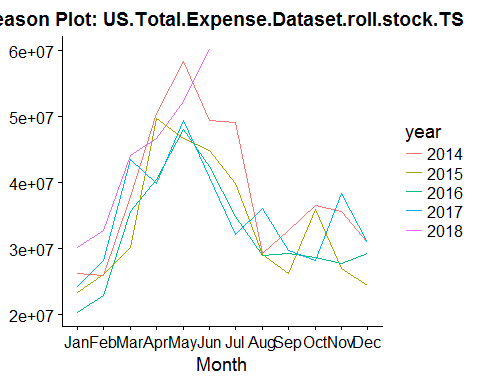
## [1] TRUE

## [1] TRUE

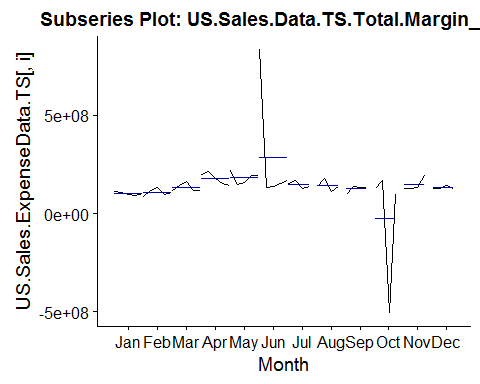
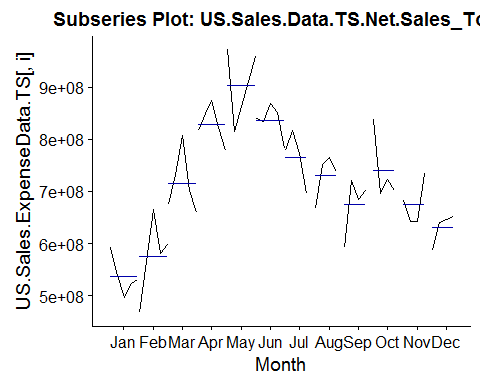
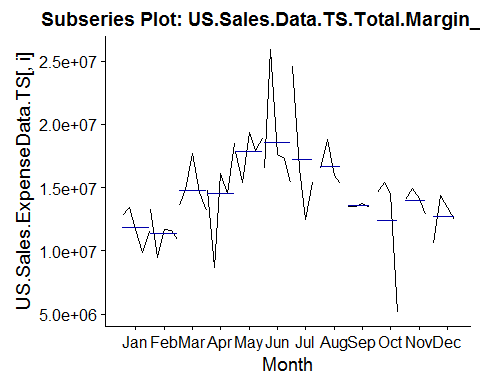
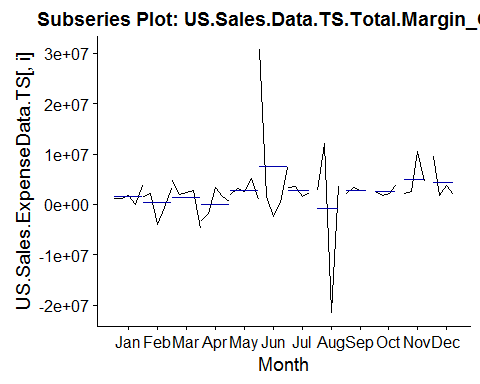
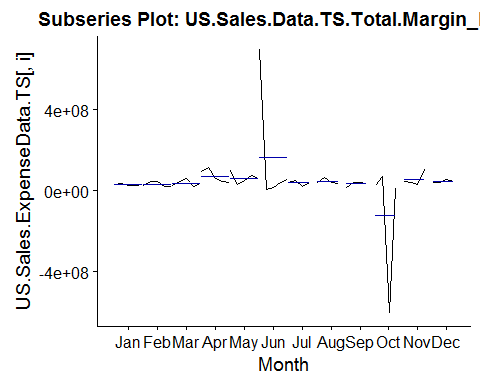
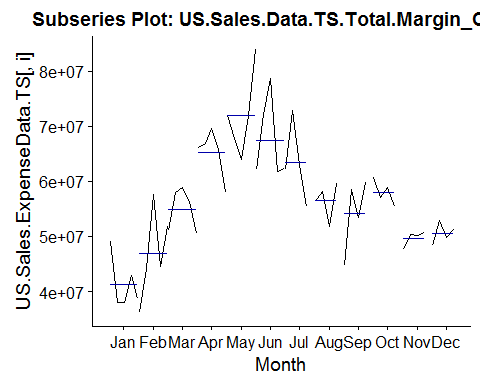
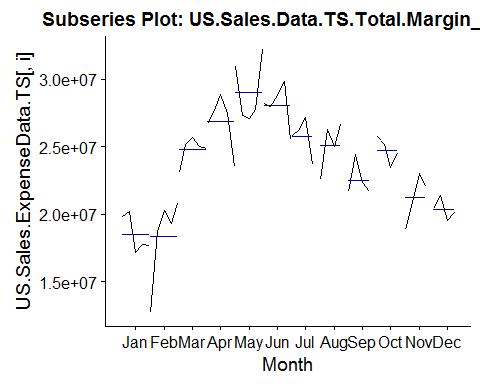
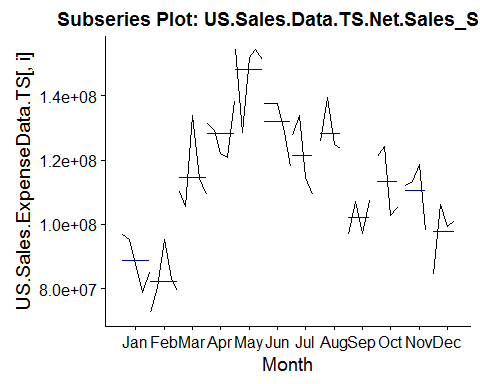
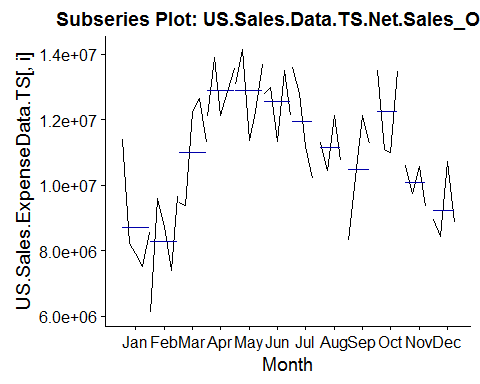
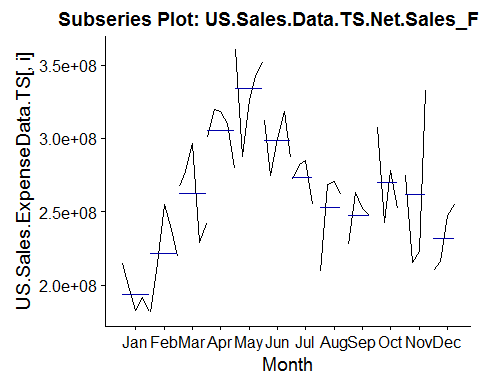
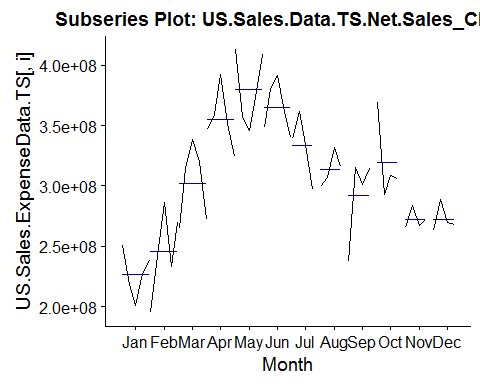
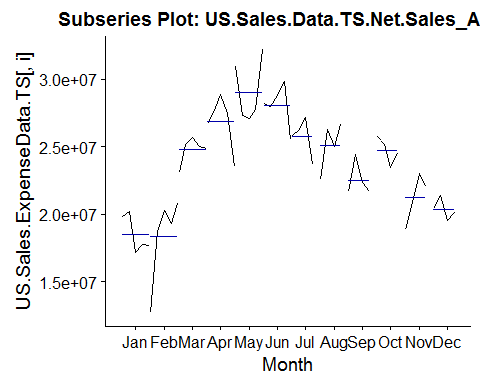
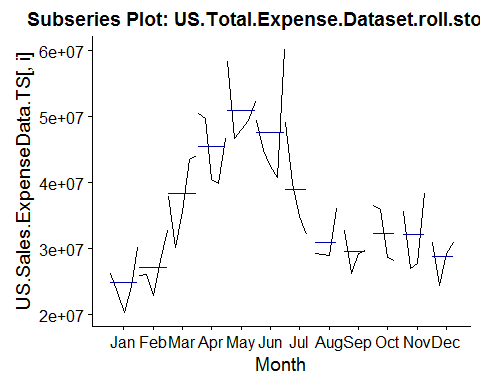
## Warning in tk\_xts\_.data.frame(ret, select = select, silent = silent): Non-  
## numeric columns being dropped: Invoice.Date

## Time Series EDA





cols <- colnames(US.Sales.ExpenseData.TS)  
  
for (i in cols){  
 data.plot <- ggsubseriesplot(US.Sales.ExpenseData.TS[,i]) +  
 ggtitle(paste("Subseries Plot:", i, sep = " "))  
   
 print(data.plot)  
}



## $stats  
## [,1] [,2] [,3] [,4] [,5] [,6] [,7]  
## [1,] 20258345 22925947 30232038 39815753 46724072 40815133 32134337  
## [2,] 23424810 25886334 35637024 40529494 48027629 42466094 33536754  
## [3,] 24207995 26160425 37873874 46700260 49416960 44890574 37404955  
## [4,] 26281759 28149735 43578642 49734186 52295935 49492064 44500413  
## [5,] 30119374 28149735 44141507 50445595 58355947 49492064 49130087  
## [,8] [,9] [,10] [,11] [,12]  
## [1,] 29004916 26196995 28251690 27019411 24347685  
## [2,] 29090479 27714351 28470611 27362763 26806568  
## [3,] 29228159 29513868 32309924 31639803 30103431  
## [4,] 32689980 31290811 36226725 36984861 30967414  
## [5,] 36099683 32785593 36523133 38396230 30993416  
##   
## $n  
## [1] 5 5 5 5 5 5 4 4 4 4 4 4  
##   
## $conf  
## [,1] [,2] [,3] [,4] [,5] [,6] [,7]  
## [1,] 22189282 24561112 32262347 40196248 46400986 39926041 28743663  
## [2,] 26226708 27759738 43485401 53204272 52432934 49855107 46066246  
## [,8] [,9] [,10] [,11] [,12]  
## [1,] 26384553 26688464 26182594 24038345 26816362  
## [2,] 32071765 32339271 38437254 39241261 33390499  
##   
## $out  
## [1] 32716475 60158084  
##   
## $group  
## [1] 2 6  
##   
## $names  
## [1] "1" "2" "3" "4" "5" "6" "7" "8" "9" "10" "11" "12"

## $stats  
## [,1] [,2] [,3] [,4] [,5] [,6] [,7]  
## [1,] 17138230 18707906 24899605 26710511 27097022 27981152 23703594  
## [2,] 17665524 18707906 24899605 26710511 27344242 27981152 24753183  
## [3,] 17755403 19338114 24988391 27485883 27734464 28219728 26038328  
## [4,] 19799958 20281755 25180603 27665451 30943959 28805267 26737186  
## [5,] 20233074 20801551 25180603 28860610 32201169 29825205 27200488  
## [,8] [,9] [,10] [,11] [,12]  
## [1,] 22534312 21690143 23492512 18887051 19512210  
## [2,] 23780898 21691405 24002311 19879976 19806726  
## [3,] 25638108 22023944 24856628 21446991 20251355  
## [4,] 26445518 23373710 25478769 22535709 20875396  
## [5,] 26642303 24392199 25756392 23050337 21349326  
##   
## $n  
## [1] 5 5 5 5 5 5 4 4 4 4 4 4  
##   
## $conf  
## [,1] [,2] [,3] [,4] [,5] [,6] [,7]  
## [1,] 16247217 18226037 24789838 26811125 25190913 27637411 24470966  
## [2,] 19263589 20450191 25186943 28160642 30278015 28802046 27605690  
## [,8] [,9] [,10] [,11] [,12]  
## [1,] 23533058 20694922 23690226 19348962 19407106  
## [2,] 27743158 23352965 26023030 23545020 21095604  
##   
## $out  
## [1] 12665670 23080649 25668608 23508365 25511032  
##   
## $group  
## [1] 2 3 3 4 6  
##   
## $names  
## [1] "1" "2" "3" "4" "5" "6" "7" "8" "9" "10" "11" "12"

## $stats  
## [,1] [,2] [,3] [,4] [,5] [,6] [,7]  
## [1,] 200603840 194764980 264827651 346789005 346398280 340542050 297416240  
## [2,] 218235242 233017793 271914351 346789005 357357009 348817330 316094023  
## [3,] 227102147 242615503 314849221 352119992 373541053 361708287 337065739  
## [4,] 238450958 269455558 320101611 359060573 408802536 380577660 350411581  
## [5,] 250713401 286612732 338924225 359060573 412926393 391955708 361463491  
## [,8] [,9] [,10] [,11] [,12]  
## [1,] 299404127 237028548 292663558 265629054 263155712  
## [2,] 303191040 269253759 299551859 266592600 265423326  
## [3,] 311640253 308085221 307397948 269869638 268587742  
## [4,] 324257786 315109305 338907610 277842349 279083681  
## [5,] 332213018 315527139 369459485 283501569 288682816  
##   
## $n  
## [1] 5 5 5 5 5 5 4 4 4 4 4 4  
##   
## $conf  
## [,1] [,2] [,3] [,4] [,5] [,6] [,7]  
## [1,] 212817773 216868671 280800225 343448933 337189773 339266518 309954868  
## [2,] 241386521 268362336 348898218 360791051 409892333 384150057 364176610  
## [,8] [,9] [,10] [,11] [,12]  
## [1,] 294997523 271859339 276306905 260982336 257796062  
## [2,] 328282982 344311102 338488991 278756940 279379422  
##   
## $out  
## [1] 392617476 324751857  
##   
## $group  
## [1] 4 4  
##   
## $names  
## [1] "1" "2" "3" "4" "5" "6" "7" "8" "9" "10" "11" "12"

## $stats  
## [,1] [,2] [,3] [,4] [,5] [,6] [,7]  
## [1,] 181886962 180993786 228935408 279429981 326852298 274642283 255257636  
## [2,] 182836378 215020056 242205008 300825366 326852298 287159214 263614805  
## [3,] 191566333 219698046 267224350 310005055 342469848 299979407 277071608  
## [4,] 196947473 237830232 277274422 318547155 351827848 312222807 283412750  
## [5,] 214552860 254760450 296387339 319751144 360536442 318588958 284654258  
## [,8] [,9] [,10] [,11] [,12]  
## [1,] 208981682 227678535 242981652 215632244 209911982  
## [2,] 235422646 237470721 247589223 219351413 212927284  
## [3,] 265220000 249818126 265036345 248989768 231712422  
## [4,] 269777405 257775358 292824328 303655978 251213976  
## [5,] 270978419 263177371 307772760 332403003 254945696  
##   
## $n  
## [1] 5 5 5 5 5 5 4 4 4 4 4 4  
##   
## $conf  
## [,1] [,2] [,3] [,4] [,5] [,6] [,7]  
## [1,] 181595469 203580433 242444390 297482883 324822187 282269535 261431232  
## [2,] 201537197 235815658 292004309 322527226 360117509 317689279 292711985  
## [,8] [,9] [,10] [,11] [,12]  
## [1,] 238079741 233777462 229300612 182389162 201465935  
## [2,] 292360260 265858789 300772078 315590374 261958909  
##   
## $out  
## [1] 287460892  
##   
## $group  
## [1] 5  
##   
## $names  
## [1] "1" "2" "3" "4" "5" "6" "7" "8" "9" "10" "11" "12"

## $stats  
## [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8]  
## [1,] 7514416 6100310 9376697 12091282 11359125 11341627 10213004 10439118  
## [2,] 7925796 7398848 9482072 12129644 12263376 12118684 10695079 10603379  
## [3,] 8229774 8752699 11304257 12851288 13084594 12793839 11994944 11040578  
## [4,] 8567237 9588872 12211897 13558591 13697921 12995838 13201771 11723696  
## [5,] 8567237 9642577 12642361 13887002 14137988 13505236 13590807 12133875  
## [,9] [,10] [,11] [,12]  
## [1,] 8301164 10986666 9361899 8443817  
## [2,] 9262183 11044026 9561575 8646290  
## [3,] 10750929 12284446 10168148 8894333  
## [4,] 11697902 13480385 10590109 9839782  
## [5,] 12117149 13493263 10605174 10739661  
##   
## $n  
## [1] 5 5 5 5 5 5 4 4 4 4 4 4  
##   
## $conf  
## [,1] [,2] [,3] [,4] [,5] [,6] [,7]  
## [1,] 7776534 7205233 9375370 11841597 12070947 12174043 10014657  
## [2,] 8683014 10300164 13233145 13860979 14098240 13413634 13975232  
## [,8] [,9] [,10] [,11] [,12]  
## [1,] 10155527 8826711 10359722 9355606 7951474  
## [2,] 11925629 12675147 14209170 10980689 9837191  
##   
## $out  
## [1] 11384686  
##   
## $group  
## [1] 1  
##   
## $names  
## [1] "1" "2" "3" "4" "5" "6" "7" "8" "9" "10" "11" "12"

## $stats  
## [,1] [,2] [,3] [,4] [,5] [,6] [,7]  
## [1,] 78930083 79354053 105685139 120900536 151407634 117930797 109217275  
## [2,] 84958728 79354053 109257357 121936994 151407634 128596479 111948604  
## [3,] 87897323 80487155 110268511 129055751 151810421 137432253 121092381  
## [4,] 95229392 83059812 114415207 131368298 154266949 137625040 130618515  
## [5,] 96752771 83059812 114415207 138266999 154294224 137734689 133732202  
## [,8] [,9] [,10] [,11] [,12]  
## [1,] 123440846 96870934 102819346 98003748 84392690  
## [2,] 124145436 97045507 103962617 105016575 91909255  
## [3,] 125340976 102128241 113171358 112647468 100315056  
## [4,] 132655658 107210695 122751046 115967090 103655387  
## [5,] 139479390 107384987 124265267 118668646 106106482  
##   
## $n  
## [1] 5 5 5 5 5 5 4 4 4 4 4 4  
##   
## $conf  
## [,1] [,2] [,3] [,4] [,5] [,6] [,7]  
## [1,] 80640097 77868675 106623988 122391615 149790036 131052694 106343152  
## [2,] 95154549 83105635 113913035 135719886 153830806 143811812 135841611  
## [,8] [,9] [,10] [,11] [,12]  
## [1,] 118617901 94097742 98328499 103996561 91035611  
## [2,] 132064052 110158739 128014216 121298376 109594500  
##   
## $out  
## [1] 72467976 95179419 133772249 128557083  
##   
## $group  
## [1] 2 2 3 5  
##   
## $names  
## [1] "1" "2" "3" "4" "5" "6" "7" "8" "9" "10" "11" "12"

## $stats  
## [,1] [,2] [,3] [,4] [,5] [,6] [,7]  
## [1,] 17138230 18707906 24899605 26710511 27097022 27981152 23703594  
## [2,] 17665524 18707906 24899605 26710511 27344242 27981152 24753183  
## [3,] 17755403 19338114 24988391 27485883 27734464 28219728 26038328  
## [4,] 19799958 20281755 25180603 27665451 30943959 28805267 26737186  
## [5,] 20233074 20801551 25180603 28860610 32201169 29825205 27200488  
## [,8] [,9] [,10] [,11] [,12]  
## [1,] 22534312 21690143 23492512 18887051 19512210  
## [2,] 23780898 21691405 24002311 19879976 19806726  
## [3,] 25638108 22023944 24856628 21446991 20251355  
## [4,] 26445518 23373710 25478769 22535709 20875396  
## [5,] 26642303 24392199 25756392 23050337 21349326  
##   
## $n  
## [1] 5 5 5 5 5 5 4 4 4 4 4 4  
##   
## $conf  
## [,1] [,2] [,3] [,4] [,5] [,6] [,7]  
## [1,] 16247217 18226037 24789838 26811125 25190913 27637411 24470966  
## [2,] 19263589 20450191 25186943 28160642 30278015 28802046 27605690  
## [,8] [,9] [,10] [,11] [,12]  
## [1,] 23533058 20694922 23690226 19348962 19407106  
## [2,] 27743158 23352965 26023030 23545020 21095604  
##   
## $out  
## [1] 12665670 23080649 25668608 23508365 25511032  
##   
## $group  
## [1] 2 3 3 4 6  
##   
## $names  
## [1] "1" "2" "3" "4" "5" "6" "7" "8" "9" "10" "11" "12"

## $stats  
## [,1] [,2] [,3] [,4] [,5] [,6] [,7]  
## [1,] 37967468 36113504 50572883 65771867 63930358 61769904 55438369  
## [2,] 37999070 44247194 51074777 65771867 67856945 62178236 58809443  
## [3,] 38767337 44536876 56323225 66107208 71918617 62325292 62633115  
## [4,] 42915125 51829057 57915548 66824802 72003190 72187405 68014644  
## [5,] 49084113 57608695 58820690 66824802 72003190 78697772 72943576  
## [,8] [,9] [,10] [,11] [,12]  
## [1,] 51893492 44802906 55481893 47564959 48338671  
## [2,] 54083074 49162954 56271133 48838123 49070244  
## [3,] 57253391 55996653 57999745 50257344 50530910  
## [4,] 58919343 59128603 59798164 50552915 52110071  
## [5,] 59604560 59786902 60657211 50702428 52960139  
##   
## $n  
## [1] 5 5 5 5 5 5 4 4 4 4 4 4  
##   
## $conf  
## [,1] [,2] [,3] [,4] [,5] [,6] [,7]  
## [1,] 35293665 39179551 51489554 65363207 68988891 55252839 55361006  
## [2,] 42241008 49894201 61156897 66851209 74848344 69397745 69905223  
## [,8] [,9] [,10] [,11] [,12]  
## [1,] 53432739 48123790 55213391 48902659 48129446  
## [2,] 61074044 63869515 60786099 51612030 52932374  
##   
## $out  
## [1] 69506981 58006748 83890157  
##   
## $group  
## [1] 4 4 5  
##   
## $names  
## [1] "1" "2" "3" "4" "5" "6" "7" "8" "9" "10" "11" "12"

## $stats  
## [,1] [,2] [,3] [,4] [,5] [,6] [,7]  
## [1,] 24075877 20798417 19341505 40533680 32885237 5821625 21689030  
## [2,] 27877242 20994812 24347911 44877336 48133569 16404160 30121650  
## [3,] 28696004 24528137 34368741 62172817 57692962 42088758 40283868  
## [4,] 32397335 43703864 42948614 92717904 73629036 56639957 46233792  
## [5,] 34536135 45849038 57890327 113380271 98040454 56639957 50454120  
## [,8] [,9] [,10] [,11] [,12]  
## [1,] 35305967 17555067 -604403833 32844190 38137159  
## [2,] 37406634 26274137 -296954046 36781911 39827055  
## [3,] 39539224 37927013 18354708 43876043 42502412  
## [4,] 51854056 41078139 47623046 75158990 49535884  
## [5,] 64136965 41295459 69032418 103285526 55583895  
##   
## $n  
## [1] 5 5 5 5 5 5 4 4 4 4 4 4  
##   
## $conf  
## [,1] [,2] [,3] [,4] [,5] [,6] [,7]  
## [1,] 25502117 8481979 21225531 28368793 39677930 13658245 27555276  
## [2,] 31889891 40574296 47511951 95976842 75707995 70519271 53012460  
## [,8] [,9] [,10] [,11] [,12]  
## [1,] 28125760 26231851 -253861195 13558151 34832438  
## [2,] 50952688 49622174 290570611 74193935 50172386  
##   
## $out  
## [1] 696350144  
##   
## $group  
## [1] 6  
##   
## $names  
## [1] "1" "2" "3" "4" "5" "6" "7" "8" "9" "10" "11" "12"

## $stats  
## [,1] [,2] [,3] [,4] [,5] [,6] [,7]  
## [1,] 1247885 -3915128.4 1962717 -3212178.5 1070392 -2479242.2 1501465  
## [2,] 1247885 -594223.2 1962717 -1831632.0 1895289 313078.9 1970255  
## [3,] 1258580 1526052.0 2468218 603921.4 2506236 1500393.3 2945446  
## [4,] 1844299 2203628.8 2729115 1487770.2 3209145 7396504.9 3535719  
## [5,] 1844299 3108634.8 2729115 3315093.6 3209145 7396504.9 3619592  
## [,8] [,9] [,10] [,11] [,12]  
## [1,] -21522616 2149780 1817758 2234865 1767696  
## [2,] -9344126 2442139 2012450 2432361 1855064  
## [3,] 3241783 2751718 2368583 3630899 2887273  
## [4,] 7863879 3107770 3172388 7542153 6698313  
## [5,] 12078556 3446602 3814750 10452365 9564511  
##   
## $n  
## [1] 5 5 5 5 5 5 4 4 4 4 4 4  
##   
## $conf  
## [,1] [,2] [,3] [,4] [,5] [,6] [,7]  
## [1,] 837155.5 -450903.2 1926683 -1741560 1577868 -3504738 1708729  
## [2,] 1680004.9 3503007.2 3009753 2949403 3434604 6505524 4182163  
## [,8] [,9] [,10] [,11] [,12]  
## [1,] -10352540 2225870 1452233 -405835.8 -938893.5  
## [2,] 16836107 3277566 3284934 7667634.5 6713440.2  
##   
## $out  
## [1] 66099.08 3765279.40 4785611.81 -4715873.00 5236252.02 30693301.09  
##   
## $group  
## [1] 1 1 3 3 5 6  
##   
## $names  
## [1] "1" "2" "3" "4" "5" "6" "7" "8" "9" "10" "11" "12"

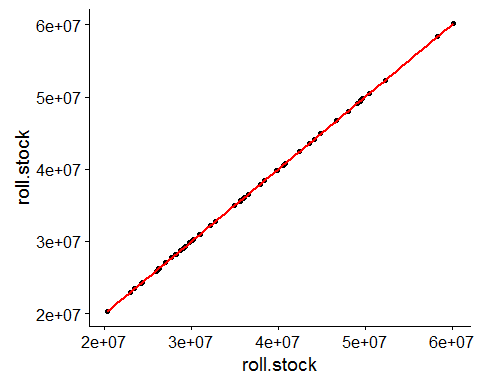
## $stats  
## [,1] [,2] [,3] [,4] [,5] [,6] [,7]  
## [1,] 9849934 10900464 13172401 14517127 17858406 15380228 12516305  
## [2,] 11532908 10900464 13608180 14517127 17858406 16539507 13976110  
## [3,] 11712462 11644720 14613203 14781169 17933059 17301973 15972873  
## [4,] 12840831 11732078 14844454 16087720 18853031 17658935 20540317  
## [5,] 13404000 11732078 14844454 16087720 19382656 17658935 24570803  
## [,8] [,9] [,10] [,11] [,12]  
## [1,] 15337523 13471295 5095532 12863430 10583773  
## [2,] 15666853 13478501 9802420 13452890 11525640  
## [3,] 16244642 13491872 14553827 14080712 12942647  
## [4,] 17658999 13642024 15009801 14512364 13884824  
## [5,] 18824896 13786012 15421255 14905656 14351861  
##   
## $n  
## [1] 5 5 5 5 5 5 4 4 4 4 4 4  
##   
## $conf  
## [,1] [,2] [,3] [,4] [,5] [,6] [,7]  
## [1,] 10788287 11057103 13739655 13671393 17230259 16510988 10787149  
## [2,] 12636636 12232337 15486751 15890946 18635859 18092958 21158596  
## [,8] [,9] [,10] [,11] [,12]  
## [1,] 14670846 13362689 10439996 13243727 11078892  
## [2,] 17818438 13621054 18667658 14917696 14806403  
##   
## $out  
## [1] 13245503 9463905 17715111 8724422 18466184 15400549 25906763  
##   
## $group  
## [1] 2 2 3 4 4 5 6  
##   
## $names  
## [1] "1" "2" "3" "4" "5" "6" "7" "8" "9" "10" "11" "12"

## $stats  
## [,1] [,2] [,3] [,4] [,5] [,6] [,7]  
## [1,] 522868382 566419492 659580579 779515793 814857213 833821974 695807749  
## [2,] 522868382 566419492 674883233 817784462 863517145 833821974 734145694  
## [3,] 531529408 580644798 701082978 823362755 910275690 839788393 775356847  
## [4,] 538874955 598951785 732366083 849419922 957937108 852224165 797341804  
## [5,] 538874955 598951785 806964318 874091880 971785612 869514262 816453554  
## [,8] [,9] [,10] [,11] [,12]  
## [1,] 668065565 591571847 696213008 642920755 586801753  
## [2,] 703541258 638558306 698967735 642977128 613663390  
## [3,] 745369267 693926464 712626308 662546567 643584760  
## [4,] 758462202 711332239 780624440 708016247 649717713  
## [5,] 765202821 720356314 837718726 733972861 652790934  
##   
## $n  
## [1] 5 5 5 5 5 5 4 4 4 4 4 4  
##   
## $conf  
## [,1] [,2] [,3] [,4] [,5] [,6] [,7]  
## [1,] 520219204 557657562 660465740 801009218 843558782 826785451 725431920  
## [2,] 542839612 603632034 741700215 845716291 976992598 852791335 825281773  
## [,8] [,9] [,10] [,11] [,12]  
## [1,] 701981721 636435057 648117511 611165663 615101845  
## [2,] 788756813 751417871 777135105 713927471 672067675  
##   
## $out  
## [1] 593203677 496401568 466992723 665587055 783261776  
##   
## $group  
## [1] 1 1 2 2 6  
##   
## $names  
## [1] "1" "2" "3" "4" "5" "6" "7" "8" "9" "10" "11" "12"

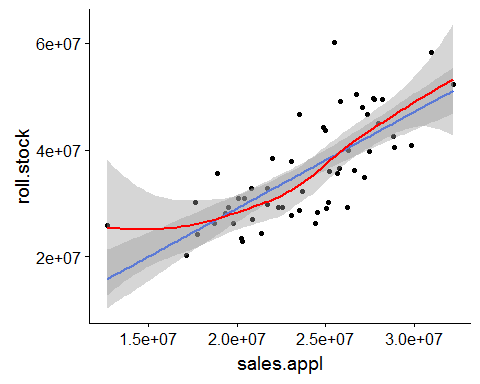
## $stats  
## [,1] [,2] [,3] [,4] [,5] [,6] [,7]  
## [1,] 94662438 88078867 116897129 141118897 146696117 133397339 125993000  
## [2,] 96539701 95723904 117995439 154139984 161049841 139086891 132511693  
## [3,] 104128383 107634519 118297758 179943221 193707712 151298920 146795299  
## [4,] 107420164 118326499 142851936 197104613 196451428 167253013 162180606  
## [5,] 111679486 131556437 162562953 214763314 220741298 167253013 169801003  
## [,8] [,9] [,10] [,11] [,12]  
## [1,] 110965690 99686127 -505255755 129531449 128566180  
## [2,] 124303713 116198306 -202927865 129646564 128912619  
## [3,] 139090645 133202338 114577837 130169465 129832216  
## [4,] 160031415 137181076 149144299 162040829 136276598  
## [5,] 179523276 140667959 168532949 193504405 142147823  
##   
## $n  
## [1] 5 5 5 5 5 5 4 4 4 4 4 4  
##   
## $conf  
## [,1] [,2] [,3] [,4] [,5] [,6] [,7]  
## [1,] 96440275 91663583 100734220 149584522 168693040 131396809 123356857  
## [2,] 111816491 123605455 135861296 210301920 218722385 171201031 170233740  
## [,8] [,9] [,10] [,11] [,12]  
## [1,] 110865760 116625950 -163559172 104577996 124014672  
## [2,] 167315531 149778726 392714847 155760934 135649759  
##   
## $out  
## [1] 833980917  
##   
## $group  
## [1] 6  
##   
## $names  
## [1] "1" "2" "3" "4" "5" "6" "7" "8" "9" "10" "11" "12"

# decompose data (season/trend/cycle)

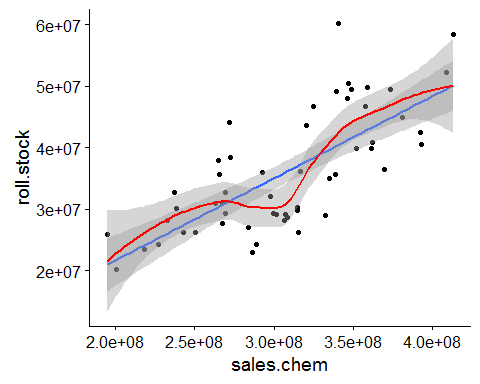
## Don't know how to automatically pick scale for object of type tbl\_df/tbl/data.frame. Defaulting to continuous.  
## Don't know how to automatically pick scale for object of type tbl\_df/tbl/data.frame. Defaulting to continuous.



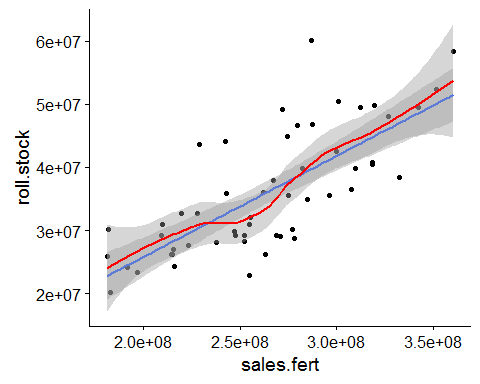
## Don't know how to automatically pick scale for object of type tbl\_df/tbl/data.frame. Defaulting to continuous.  
## Don't know how to automatically pick scale for object of type tbl\_df/tbl/data.frame. Defaulting to continuous.



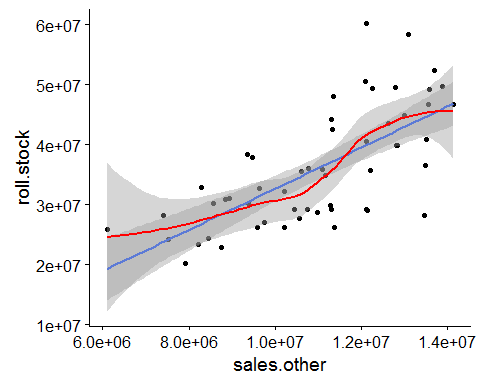
## Don't know how to automatically pick scale for object of type tbl\_df/tbl/data.frame. Defaulting to continuous.  
## Don't know how to automatically pick scale for object of type tbl\_df/tbl/data.frame. Defaulting to continuous.



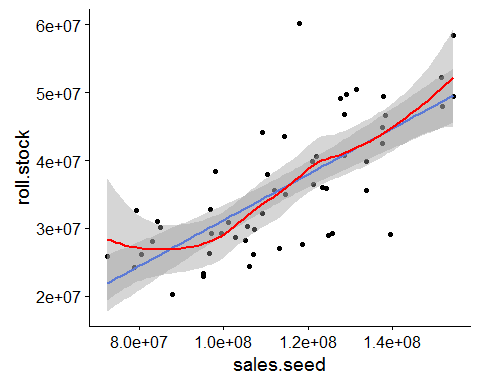
## Don't know how to automatically pick scale for object of type tbl\_df/tbl/data.frame. Defaulting to continuous.  
## Don't know how to automatically pick scale for object of type tbl\_df/tbl/data.frame. Defaulting to continuous.



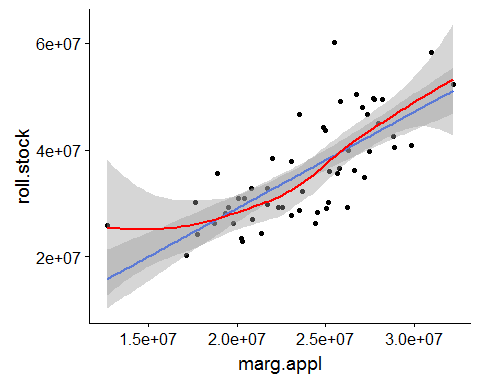
## Don't know how to automatically pick scale for object of type tbl\_df/tbl/data.frame. Defaulting to continuous.  
## Don't know how to automatically pick scale for object of type tbl\_df/tbl/data.frame. Defaulting to continuous.



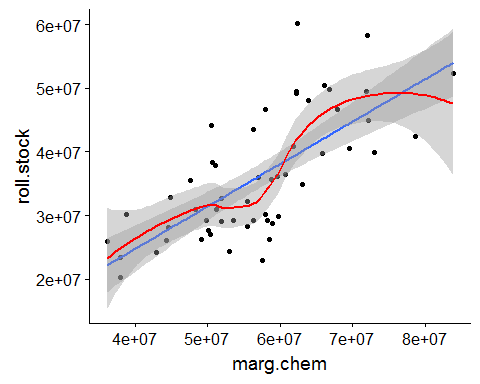
## Don't know how to automatically pick scale for object of type tbl\_df/tbl/data.frame. Defaulting to continuous.  
## Don't know how to automatically pick scale for object of type tbl\_df/tbl/data.frame. Defaulting to continuous.



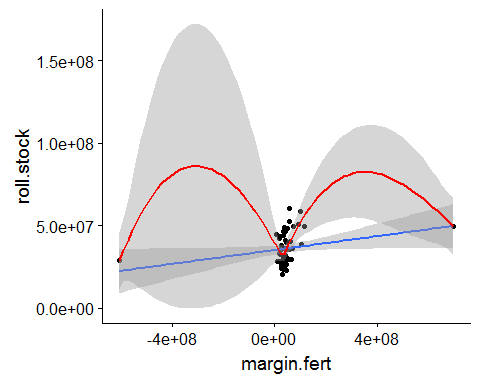
## Don't know how to automatically pick scale for object of type tbl\_df/tbl/data.frame. Defaulting to continuous.  
## Don't know how to automatically pick scale for object of type tbl\_df/tbl/data.frame. Defaulting to continuous.



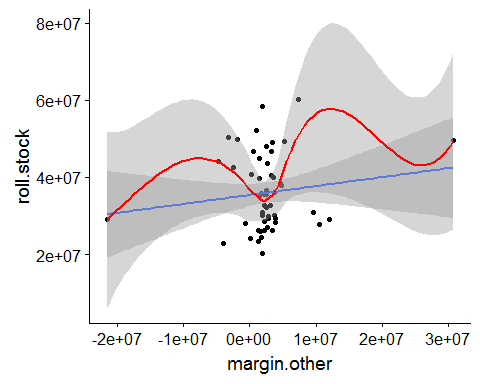
## Don't know how to automatically pick scale for object of type tbl\_df/tbl/data.frame. Defaulting to continuous.  
## Don't know how to automatically pick scale for object of type tbl\_df/tbl/data.frame. Defaulting to continuous.



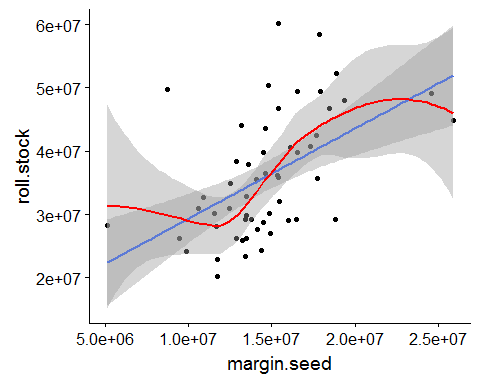
## Don't know how to automatically pick scale for object of type tbl\_df/tbl/data.frame. Defaulting to continuous.  
## Don't know how to automatically pick scale for object of type tbl\_df/tbl/data.frame. Defaulting to continuous.



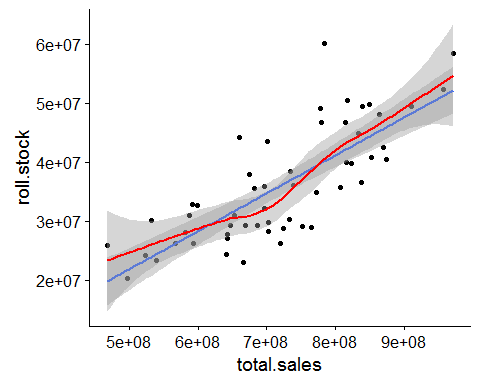
## Don't know how to automatically pick scale for object of type tbl\_df/tbl/data.frame. Defaulting to continuous.  
## Don't know how to automatically pick scale for object of type tbl\_df/tbl/data.frame. Defaulting to continuous.



## Don't know how to automatically pick scale for object of type tbl\_df/tbl/data.frame. Defaulting to continuous.  
## Don't know how to automatically pick scale for object of type tbl\_df/tbl/data.frame. Defaulting to continuous.

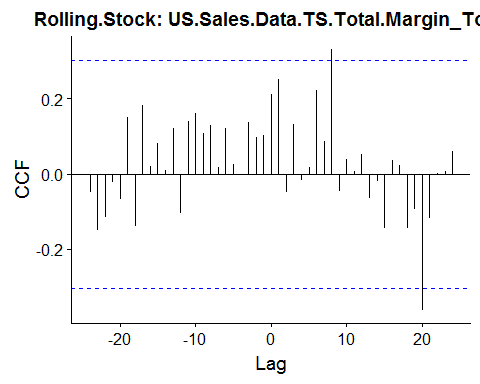
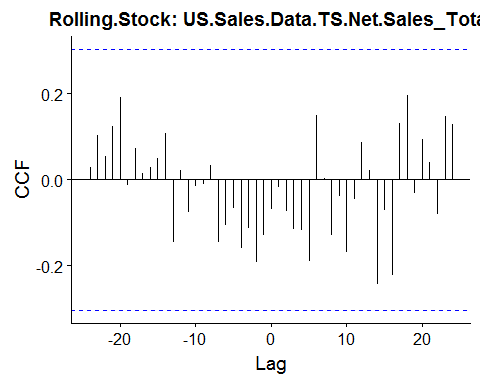
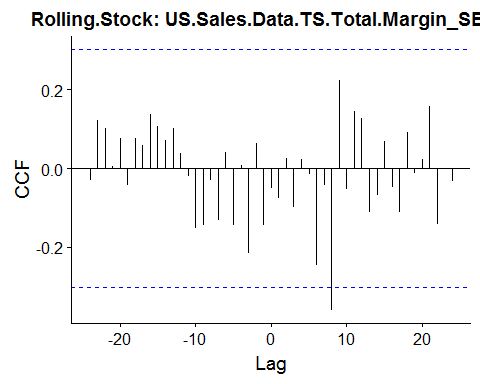
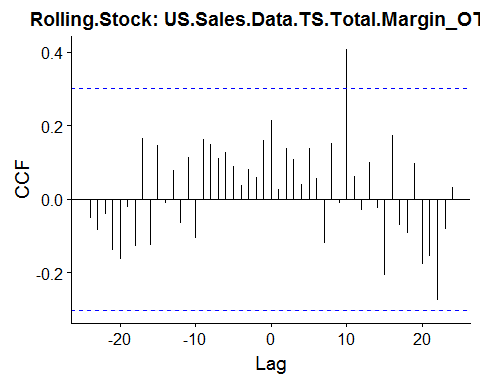
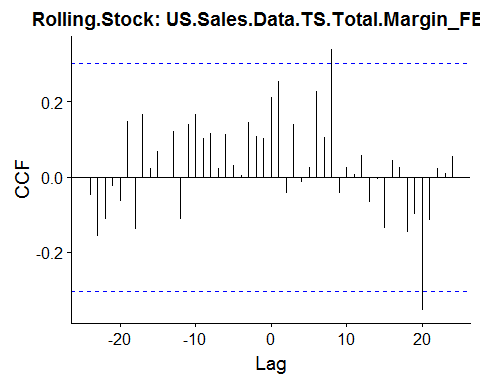
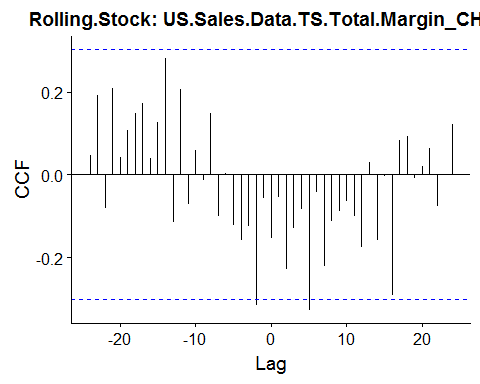
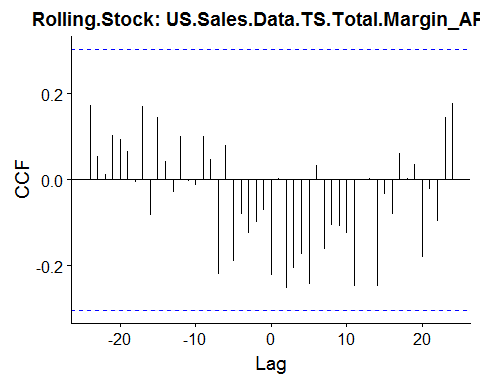
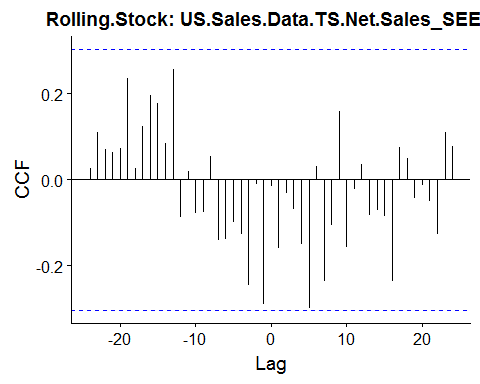
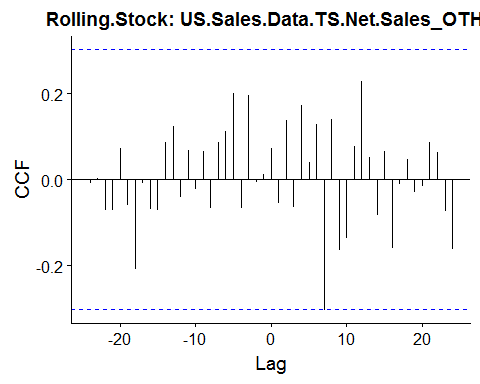
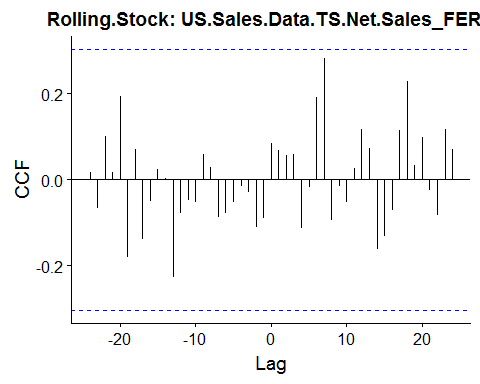
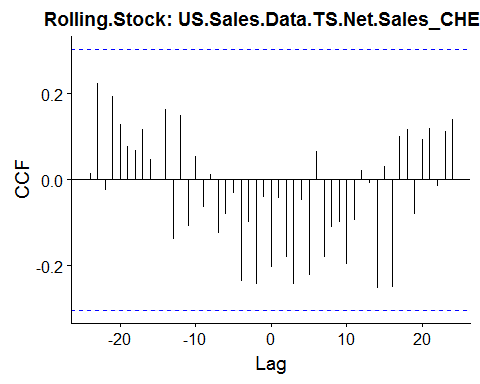
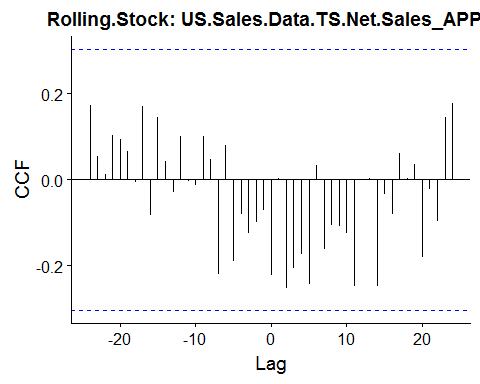


## Don't know how to automatically pick scale for object of type tbl\_df/tbl/data.frame. Defaulting to continuous.  
## Don't know how to automatically pick scale for object of type tbl\_df/tbl/data.frame. Defaulting to continuous.



## # A tibble: 6 x 7  
## rowname roll.stock sales.appl sales.chem sales.fert sales.other  
## <chr> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 roll.stock NA 0.733 0.732 0.745 0.692  
## 2 sales.appl 0.733 NA 0.924 0.829 0.797  
## 3 sales.chem 0.732 0.924 NA 0.847 0.835  
## 4 sales.fert 0.745 0.829 0.847 NA 0.683  
## 5 sales.other 0.692 0.797 0.835 0.683 NA   
## 6 sales.seed 0.726 0.860 0.850 0.769 0.758  
## # ... with 1 more variable: sales.seed <dbl>

# cross correlation plots



## Linear Regression Model Roll Stock to Net Sales

## [1] "US.Total.Expense.Dataset.roll.stock.TS"  
## [2] "US.Sales.Data.TS.Net.Sales\_APPL"   
## [3] "US.Sales.Data.TS.Net.Sales\_CHEM"   
## [4] "US.Sales.Data.TS.Net.Sales\_FERT"   
## [5] "US.Sales.Data.TS.Net.Sales\_OTHE"   
## [6] "US.Sales.Data.TS.Net.Sales\_SEED"   
## [7] "US.Sales.Data.TS.Total.Margin\_APPL"   
## [8] "US.Sales.Data.TS.Total.Margin\_CHEM"   
## [9] "US.Sales.Data.TS.Total.Margin\_FERT"   
## [10] "US.Sales.Data.TS.Total.Margin\_OTHE"   
## [11] "US.Sales.Data.TS.Total.Margin\_SEED"   
## [12] "US.Sales.Data.TS.Net.Sales\_Total"   
## [13] "US.Sales.Data.TS.Total.Margin\_Total"

##   
## Call:  
## lm(formula = US.Total.Expense.Dataset.roll.stock.TS ~ saleslag.1 +   
## saleslag.2 + saleslag.3, data = data)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -12733877 -4970431 -1859295 4960772 13607998   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 2.984e+07 8.049e+06 3.708 0.000551 \*\*\*  
## saleslag.1 5.834e-02 1.121e-02 5.203 4.23e-06 \*\*\*  
## saleslag.2 -1.441e-02 1.396e-02 -1.032 0.307319   
## saleslag.3 -3.584e-02 1.121e-02 -3.197 0.002482 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 6644000 on 47 degrees of freedom  
## (3 observations deleted due to missingness)  
## Multiple R-squared: 0.565, Adjusted R-squared: 0.5372   
## F-statistic: 20.35 on 3 and 47 DF, p-value: 1.358e-08

##   
## Call:  
## tslm(formula = US.Sales.ExpenseData.TS[, 1] ~ US.Sales.ExpenseData.TS[,   
## 12], data = US.Sales.ExpenseData.TS)  
##   
## Coefficients:  
## (Intercept) US.Sales.ExpenseData.TS[, 12]   
## -1.014e+07 6.410e-02

##   
## Call:  
## tslm(formula = US.Sales.ExpenseData.TS[, 1] ~ US.Sales.ExpenseData.TS[,   
## 12], data = US.Sales.ExpenseData.TS)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -9902191 -4340224 -867964 4592089 20093459   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -1.014e+07 5.211e+06 -1.946 0.0571 .   
## US.Sales.ExpenseData.TS[, 12] 6.410e-02 7.149e-03 8.966 3.91e-12 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 6130000 on 52 degrees of freedom  
## Multiple R-squared: 0.6072, Adjusted R-squared: 0.5997   
## F-statistic: 80.39 on 1 and 52 DF, p-value: 3.913e-12

## Linear Regression Model Roll Stock to APPL Sales

##   
## Call:  
## lm(formula = US.Total.Expense.Dataset.roll.stock.TS ~ saleslag.1 +   
## saleslag.2 + saleslag.3, data = data)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -10175910 -4451422 156420 4695028 13628048   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 2.920e+07 7.170e+06 4.073 0.000177 \*\*\*  
## saleslag.1 1.924e+00 2.835e-01 6.786 1.73e-08 \*\*\*  
## saleslag.2 -3.073e-01 3.230e-01 -0.951 0.346233   
## saleslag.3 -1.351e+00 2.767e-01 -4.882 1.25e-05 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 6136000 on 47 degrees of freedom  
## (3 observations deleted due to missingness)  
## Multiple R-squared: 0.6289, Adjusted R-squared: 0.6052   
## F-statistic: 26.55 on 3 and 47 DF, p-value: 3.414e-10

##   
## Call:  
## tslm(formula = US.Sales.ExpenseData.TS[, 1] ~ US.Sales.ExpenseData.TS[,   
## 2], data = US.Sales.ExpenseData.TS)  
##   
## Coefficients:  
## (Intercept) US.Sales.ExpenseData.TS[, 2]   
## -7.175e+06 1.811e+00

##   
## Call:  
## tslm(formula = US.Sales.ExpenseData.TS[, 1] ~ US.Sales.ExpenseData.TS[,   
## 2], data = US.Sales.ExpenseData.TS)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -11180614 -4505169 -654893 5454546 21137278   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -7.175e+06 5.634e+06 -1.274 0.208   
## US.Sales.ExpenseData.TS[, 2] 1.811e+00 2.333e-01 7.761 3.04e-10 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 6658000 on 52 degrees of freedom  
## Multiple R-squared: 0.5367, Adjusted R-squared: 0.5278   
## F-statistic: 60.23 on 1 and 52 DF, p-value: 3.045e-10

## Linear Regression Model Roll Stock to Chem Sales

##   
## Call:  
## lm(formula = US.Total.Expense.Dataset.roll.stock.TS ~ saleslag.1 +   
## saleslag.2 + saleslag.3, data = data)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -15067571 -4219137 -1301999 4213487 14033343   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 3.479e+07 7.477e+06 4.653 2.69e-05 \*\*\*  
## saleslag.1 1.165e-01 2.210e-02 5.270 3.35e-06 \*\*\*  
## saleslag.2 -1.446e-02 2.575e-02 -0.561 0.577   
## saleslag.3 -9.905e-02 2.202e-02 -4.498 4.49e-05 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 6738000 on 47 degrees of freedom  
## (3 observations deleted due to missingness)  
## Multiple R-squared: 0.5526, Adjusted R-squared: 0.524   
## F-statistic: 19.35 on 3 and 47 DF, p-value: 2.603e-08

##   
## Call:  
## tslm(formula = US.Sales.ExpenseData.TS[, 1] ~ US.Sales.ExpenseData.TS[,   
## 3], data = US.Sales.ExpenseData.TS)  
##   
## Coefficients:  
## (Intercept) US.Sales.ExpenseData.TS[, 3]   
## -5.030e+06 1.335e-01

##   
## Call:  
## tslm(formula = US.Sales.ExpenseData.TS[, 1] ~ US.Sales.ExpenseData.TS[,   
## 3], data = US.Sales.ExpenseData.TS)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -10909636 -5466693 -1100267 5074511 19710869   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -5.030e+06 5.372e+06 -0.936 0.353   
## US.Sales.ExpenseData.TS[, 3] 1.335e-01 1.724e-02 7.744 3.24e-10 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 6665000 on 52 degrees of freedom  
## Multiple R-squared: 0.5356, Adjusted R-squared: 0.5267   
## F-statistic: 59.97 on 1 and 52 DF, p-value: 3.237e-10

## Linear Regression Model Roll Stock to Fert Sales

##   
## Call:  
## lm(formula = US.Total.Expense.Dataset.roll.stock.TS ~ saleslag.1 +   
## saleslag.2 + saleslag.3, data = data)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -15707784 -5655393 -690334 6542010 12640149   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 2.352e+07 9.145e+06 2.572 0.0133 \*   
## saleslag.1 1.376e-01 2.996e-02 4.593 3.28e-05 \*\*\*  
## saleslag.2 -3.207e-02 3.508e-02 -0.914 0.3653   
## saleslag.3 -5.876e-02 3.024e-02 -1.944 0.0579 .   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 7497000 on 47 degrees of freedom  
## (3 observations deleted due to missingness)  
## Multiple R-squared: 0.4461, Adjusted R-squared: 0.4108   
## F-statistic: 12.62 on 3 and 47 DF, p-value: 3.552e-06

##   
## Call:  
## tslm(formula = US.Sales.ExpenseData.TS[, 1] ~ US.Sales.ExpenseData.TS[,   
## 4], data = US.Sales.ExpenseData.TS)  
##   
## Coefficients:  
## (Intercept) US.Sales.ExpenseData.TS[, 4]   
## -6.122e+06 1.598e-01

##   
## Call:  
## tslm(formula = US.Sales.ExpenseData.TS[, 1] ~ US.Sales.ExpenseData.TS[,   
## 4], data = US.Sales.ExpenseData.TS)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -11655978 -4123244 -1561716 3490420 20399675   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -6.122e+06 5.309e+06 -1.153 0.254   
## US.Sales.ExpenseData.TS[, 4] 1.598e-01 1.987e-02 8.043 1.09e-10 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 6529000 on 52 degrees of freedom  
## Multiple R-squared: 0.5544, Adjusted R-squared: 0.5458   
## F-statistic: 64.69 on 1 and 52 DF, p-value: 1.089e-10

## Linear Regression Model Roll Stock to Seed Sales

##   
## Call:  
## lm(formula = US.Total.Expense.Dataset.roll.stock.TS ~ saleslag.1 +   
## saleslag.2 + saleslag.3, data = data)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -12315322 -5466841 -112189 4402768 14688057   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 3.798e+07 7.799e+06 4.870 1.30e-05 \*\*\*  
## saleslag.1 2.623e-01 6.154e-02 4.262 9.68e-05 \*\*\*  
## saleslag.2 -4.835e-02 7.068e-02 -0.684 0.497330   
## saleslag.3 -2.337e-01 6.096e-02 -3.833 0.000375 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 7378000 on 47 degrees of freedom  
## (3 observations deleted due to missingness)  
## Multiple R-squared: 0.4635, Adjusted R-squared: 0.4292   
## F-statistic: 13.53 on 3 and 47 DF, p-value: 1.709e-06

##   
## Call:  
## tslm(formula = US.Sales.ExpenseData.TS[, 1] ~ US.Sales.ExpenseData.TS[,   
## 6], data = US.Sales.ExpenseData.TS)  
##   
## Coefficients:  
## (Intercept) US.Sales.ExpenseData.TS[, 6]   
## -2.644e+06 3.384e-01

##   
## Call:  
## tslm(formula = US.Sales.ExpenseData.TS[, 1] ~ US.Sales.ExpenseData.TS[,   
## 6], data = US.Sales.ExpenseData.TS)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -15381408 -3732043 -404049 3930150 22892905   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -2.644e+06 5.156e+06 -0.513 0.61   
## US.Sales.ExpenseData.TS[, 6] 3.384e-01 4.446e-02 7.611 5.26e-10 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 6727000 on 52 degrees of freedom  
## Multiple R-squared: 0.527, Adjusted R-squared: 0.5179   
## F-statistic: 57.93 on 1 and 52 DF, p-value: 5.257e-10

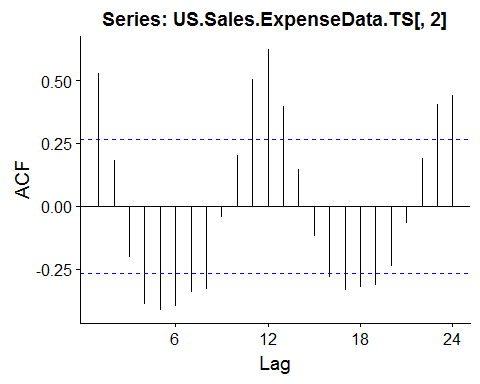
## Forecast external variables

# APPL

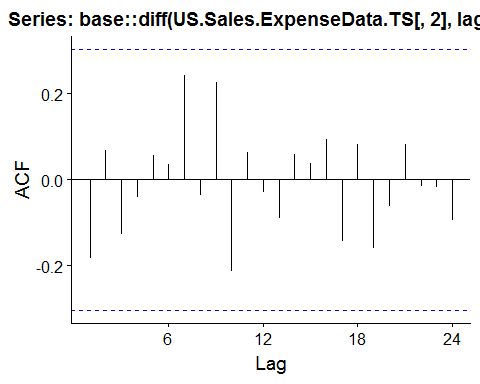
colnames(US.Sales.ExpenseData.TS)[2]

## [1] "US.Sales.Data.TS.Net.Sales\_APPL"

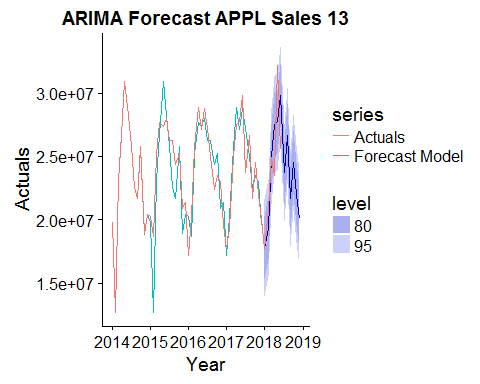
#chem  
ggAcf(US.Sales.ExpenseData.TS[,2])



#seasonal lags  
ggAcf(base::diff(US.Sales.ExpenseData.TS[,2], lag = 12))



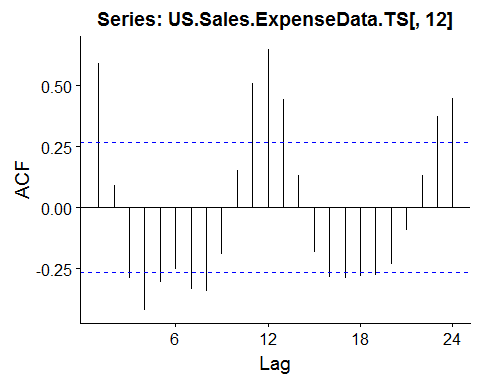
#use 1 seasonal diff  
model.appl <- auto.arima(window(US.Sales.ExpenseData.TS[,2], end = c(2017,12)), D = 1,   
 approximation = FALSE,   
 stepwise = FALSE, seasonal = TRUE)  
  
fcast.appl <- forecast::forecast(model.appl, h = 12)  
  
plot <- autoplot(fcast.appl, series="Forecast") +  
 autolayer(model.appl$fitted, series="Forecast Model") +  
 autolayer(US.Sales.ExpenseData.TS[,2], series = "Actuals") +   
 xlab("Year") +   
 ylab("Actuals") +  
 ggtitle(paste("ARIMA Forecast APPL Sales", i, sep = " "))  
   
print(plot)



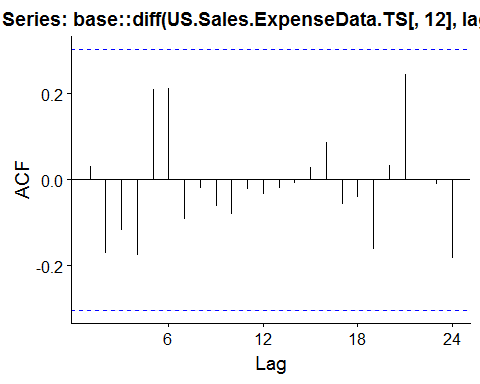
## ARIMA Model Roll Stock APPL as External Variable

# Net Sales

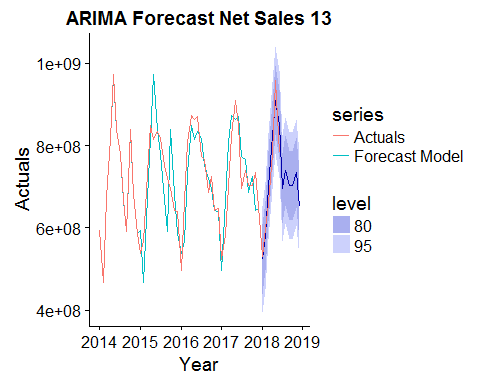
#chem  
ggAcf(US.Sales.ExpenseData.TS[,12])



#seasonal lags  
ggAcf(base::diff(US.Sales.ExpenseData.TS[,12], lag = 12))



#use 1 seasonal diff  
model.sales <- auto.arima(window(US.Sales.ExpenseData.TS[,12], end = c(2017,12)), D = 1,   
 approximation = FALSE,   
 stepwise = FALSE, seasonal = TRUE)  
  
fcast.sales <- forecast::forecast(model.sales, h = 12)  
  
plot <- autoplot(fcast.sales, series="Forecast") +  
 autolayer(model.sales$fitted, series="Forecast Model") +  
 autolayer(US.Sales.ExpenseData.TS[,12], series = "Actuals") +   
 xlab("Year") +   
 ylab("Actuals") +  
 ggtitle(paste("ARIMA Forecast Net Sales", i, sep = " "))  
   
print(plot)

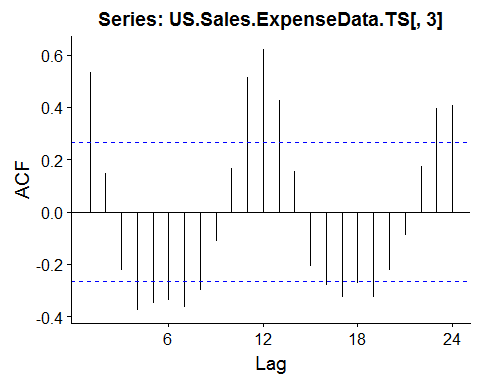


## ARIMA Model CHEM

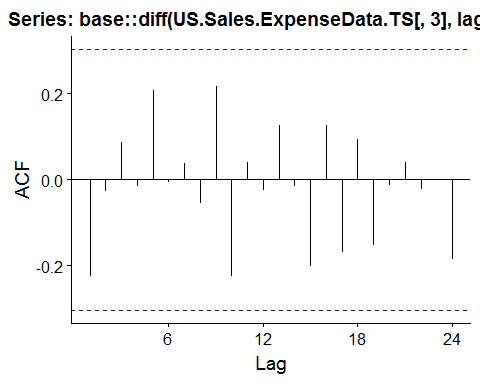
colnames(US.Sales.ExpenseData.TS)[3]

## [1] "US.Sales.Data.TS.Net.Sales\_CHEM"

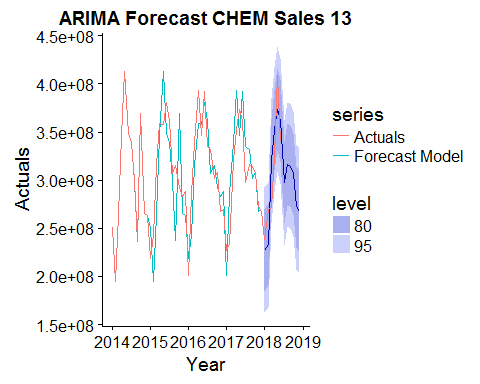
#chem  
ggAcf(US.Sales.ExpenseData.TS[,3])



#seasonal lags  
ggAcf(base::diff(US.Sales.ExpenseData.TS[,3], lag = 12))



#use 1 seasonal diff  
model.chem <- auto.arima(window(US.Sales.ExpenseData.TS[,3], end = c(2017,12)), D = 1,   
 approximation = FALSE,   
 stepwise = FALSE, seasonal = TRUE)  
  
fcast.chem <- forecast::forecast(model.chem, h = 12)  
  
plot <- autoplot(fcast.chem, series="Forecast") +  
 autolayer(model.chem$fitted, series="Forecast Model") +  
 autolayer(US.Sales.ExpenseData.TS[,3], series = "Actuals") +   
 xlab("Year") +   
 ylab("Actuals") +  
 ggtitle(paste("ARIMA Forecast CHEM Sales", i, sep = " "))  
   
print(plot)

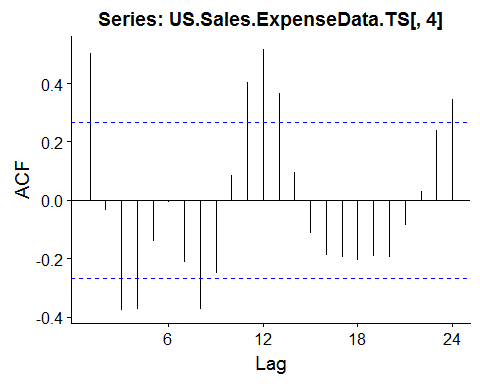


## ARIMA Model FERT

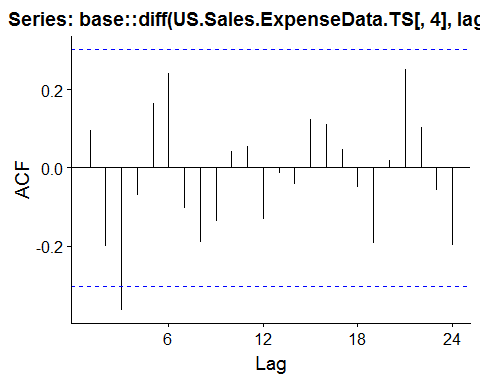
colnames(US.Sales.ExpenseData.TS)[4]

## [1] "US.Sales.Data.TS.Net.Sales\_FERT"

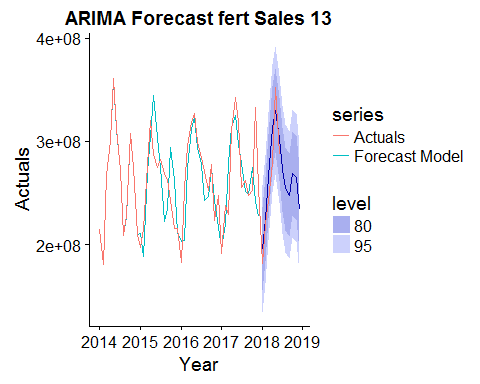
#fert  
ggAcf(US.Sales.ExpenseData.TS[,4])



#seasonal lags  
ggAcf(base::diff(US.Sales.ExpenseData.TS[,4], lag = 12))



#use 1 seasonal diff  
model.fert <- auto.arima(window(US.Sales.ExpenseData.TS[,4], end = c(2017,12)), D = 1,   
 approximation = FALSE,   
 stepwise = FALSE, seasonal = TRUE)  
  
fcast.fert <- forecast::forecast(model.fert, h = 12)  
  
plot <- autoplot(fcast.fert, series="Forecast") +  
 autolayer(model.fert$fitted, series="Forecast Model") +  
 autolayer(US.Sales.ExpenseData.TS[,4], series = "Actuals") +   
 xlab("Year") +   
 ylab("Actuals") +  
 ggtitle(paste("ARIMA Forecast fert Sales", i, sep = " "))  
   
print(plot)

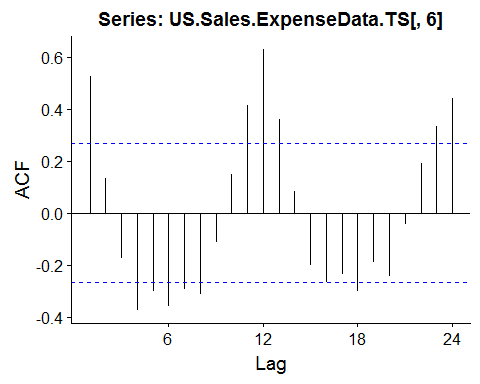


## ARIMA Model SEED

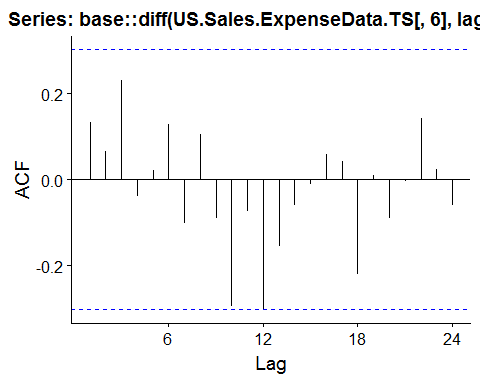
colnames(US.Sales.ExpenseData.TS)[6]

## [1] "US.Sales.Data.TS.Net.Sales\_SEED"

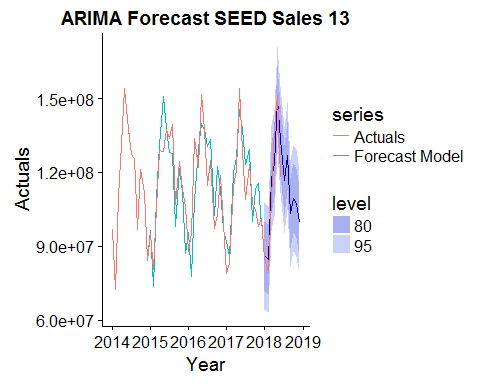
#chem  
ggAcf(US.Sales.ExpenseData.TS[,6])



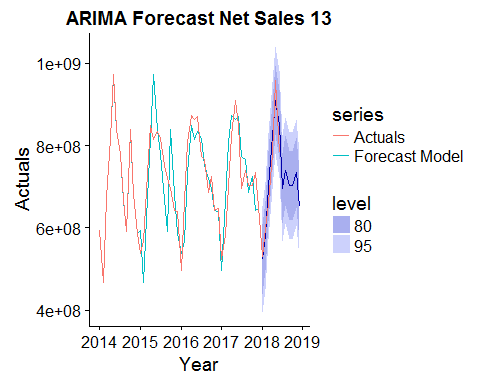
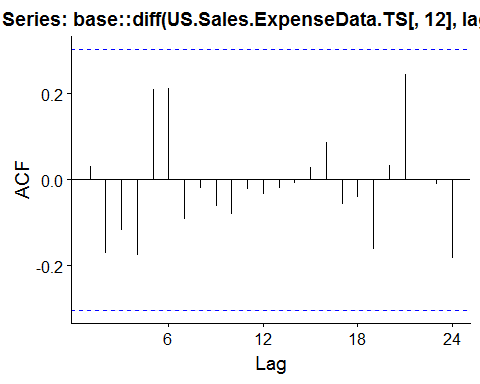
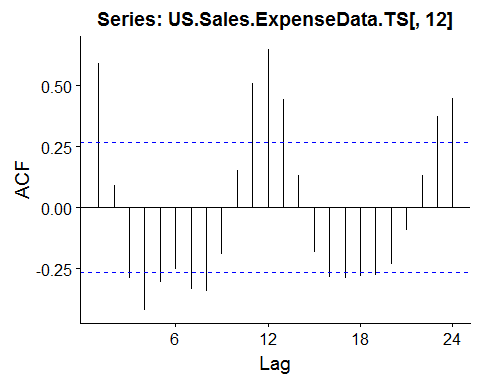
#seasonal lags  
ggAcf(base::diff(US.Sales.ExpenseData.TS[,6], lag = 12))



#use 1 seasonal diff  
model.seed <- auto.arima(window(US.Sales.ExpenseData.TS[,6], end = c(2017,12)), D = 1,   
 approximation = FALSE,   
 stepwise = FALSE, seasonal = TRUE)  
  
fcast.seed <- forecast::forecast(model.seed, h = 12)  
  
plot <- autoplot(fcast.seed, series="Forecast") +  
 autolayer(model.seed$fitted, series="Forecast Model") +  
 autolayer(US.Sales.ExpenseData.TS[,6], series = "Actuals") +   
 xlab("Year") +   
 ylab("Actuals") +  
 ggtitle(paste("ARIMA Forecast SEED Sales", i, sep = " "))  
   
print(plot)



## ARIMA Model NET SALES

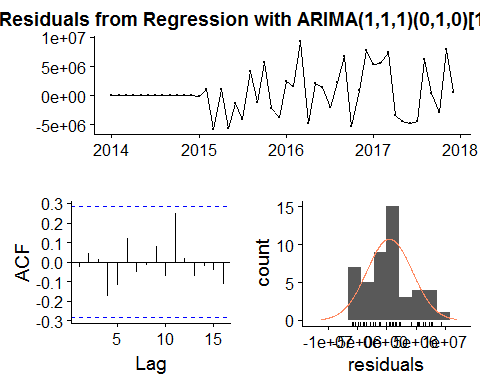


# Forecast models with and without external variables

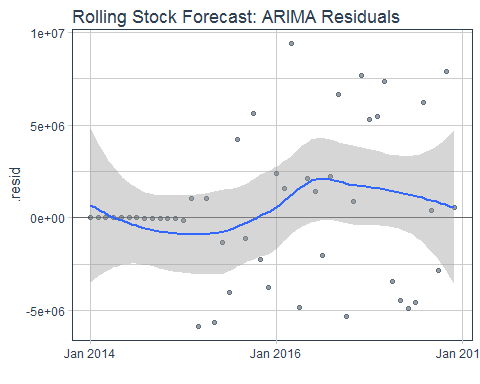
## # A tibble: 3 x 2  
## term estimate  
## <chr> <dbl>  
## 1 ar1 -0.355   
## 2 ma1 -0.651   
## 3 xreg 0.0105

## sigma logLik AIC BIC  
## 1 4742645 -586.6687 1181.337 1187.559

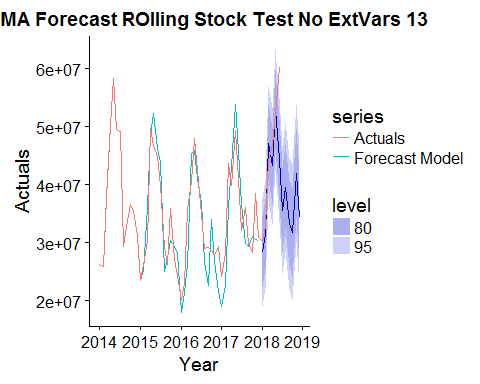
## # A tibble: 48 x 4  
## index .actual .fitted .resid  
## <S3: yearmon> <dbl> <dbl> <dbl>  
## 1 Jan 2014 26281759 26270195 11564  
## 2 Feb 2014 25886334 25880438 5896  
## 3 Mar 2014 37873874 37861735 12139  
## 4 Apr 2014 50445595 50426788 18807  
## 5 May 2014 58355947 58335257 20690  
## 6 Jun 2014 49492064 49481723 10341  
## 7 Jul 2014 49130087 49120964 9123  
## 8 Aug 2014 29280277 29289855 - 9578  
## 9 Sep 2014 32785593 32790025 - 4432  
## 10 Oct 2014 36523133 36526033 - 2900  
## # ... with 38 more rows



##   
## Ljung-Box test  
##   
## data: Residuals from Regression with ARIMA(1,1,1)(0,1,0)[12] errors  
## Q\* = 15.503, df = 21, p-value = 0.7969  
##   
## Model df: 3. Total lags used: 24



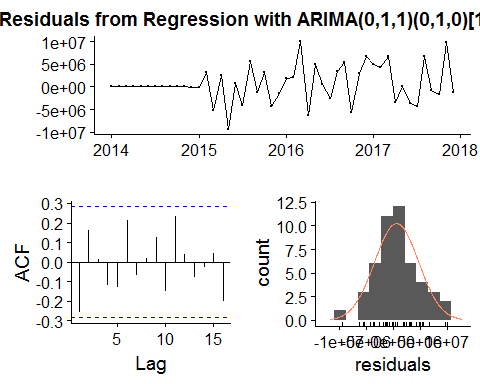
## # A tibble: 108 x 7  
## index key value lo.80 lo.95 hi.80 hi.95  
## <S3: yearmon> <chr> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 Jan 2014 actual 26281759 NA NA NA NA  
## 2 Feb 2014 actual 25886334 NA NA NA NA  
## 3 Mar 2014 actual 37873874 NA NA NA NA  
## 4 Apr 2014 actual 50445595 NA NA NA NA  
## 5 May 2014 actual 58355947 NA NA NA NA  
## 6 Jun 2014 actual 49492064 NA NA NA NA  
## 7 Jul 2014 actual 49130087 NA NA NA NA  
## 8 Aug 2014 actual 29280277 NA NA NA NA  
## 9 Sep 2014 actual 32785593 NA NA NA NA  
## 10 Oct 2014 actual 36523133 NA NA NA NA  
## # ... with 98 more rows



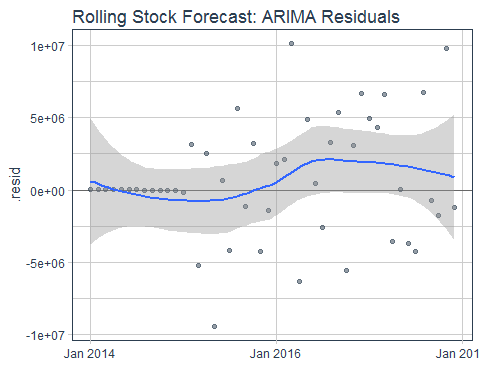
## # A tibble: 2 x 2  
## term estimate  
## <chr> <dbl>  
## 1 ma1 -0.808   
## 2 xreg -0.0119

## sigma logLik AIC BIC  
## 1 4915964 -588.4421 1182.884 1187.55

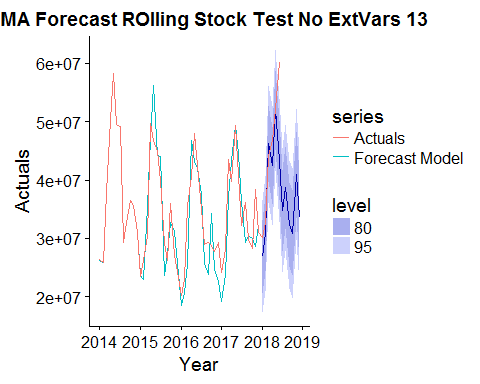
## # A tibble: 48 x 4  
## index .actual .fitted .resid  
## <S3: yearmon> <dbl> <dbl> <dbl>  
## 1 Jan 2014 26281759 26264859 16900  
## 2 Feb 2014 25886334 25879599 6735  
## 3 Mar 2014 37873874 37858627 15247  
## 4 Apr 2014 50445595 50422282 23313  
## 5 May 2014 58355947 58329481 26466  
## 6 Jun 2014 49492064 49478900 13164  
## 7 Jul 2014 49130087 49119256 10831  
## 8 Aug 2014 29280277 29289899 - 9622  
## 9 Sep 2014 32785593 32791530 - 5937  
## 10 Oct 2014 36523133 36523417 - 284  
## # ... with 38 more rows



##   
## Ljung-Box test  
##   
## data: Residuals from Regression with ARIMA(0,1,1)(0,1,0)[12] errors  
## Q\* = 22.576, df = 22, p-value = 0.426  
##   
## Model df: 2. Total lags used: 24



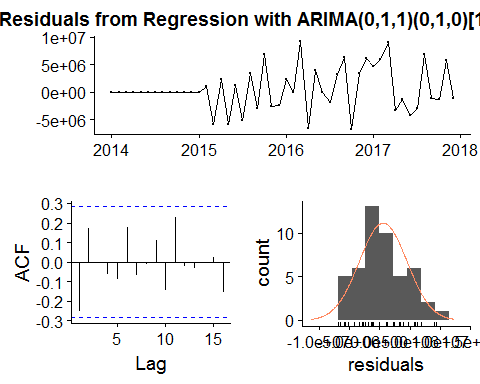
## # A tibble: 108 x 7  
## index key value lo.80 lo.95 hi.80 hi.95  
## <S3: yearmon> <chr> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 Jan 2014 actual 26281759 NA NA NA NA  
## 2 Feb 2014 actual 25886334 NA NA NA NA  
## 3 Mar 2014 actual 37873874 NA NA NA NA  
## 4 Apr 2014 actual 50445595 NA NA NA NA  
## 5 May 2014 actual 58355947 NA NA NA NA  
## 6 Jun 2014 actual 49492064 NA NA NA NA  
## 7 Jul 2014 actual 49130087 NA NA NA NA  
## 8 Aug 2014 actual 29280277 NA NA NA NA  
## 9 Sep 2014 actual 32785593 NA NA NA NA  
## 10 Oct 2014 actual 36523133 NA NA NA NA  
## # ... with 98 more rows



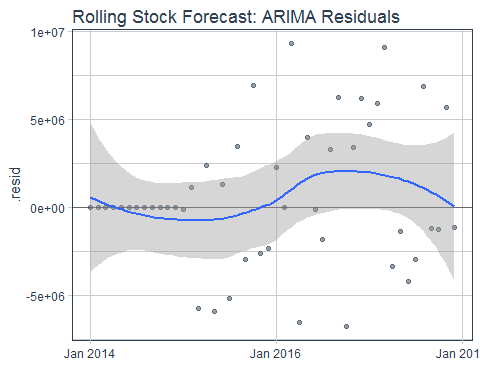
## # A tibble: 2 x 2  
## term estimate  
## <chr> <dbl>  
## 1 ma1 -0.799   
## 2 xreg 0.0328

## sigma logLik AIC BIC  
## 1 4753522 -587.2448 1180.49 1185.156

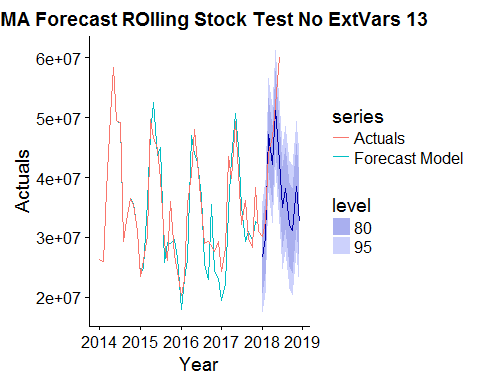
## # A tibble: 48 x 4  
## index .actual .fitted .resid  
## <S3: yearmon> <dbl> <dbl> <dbl>  
## 1 Jan 2014 26281759 26270654 11105  
## 2 Feb 2014 25886334 25880820 5514  
## 3 Mar 2014 37873874 37862527 11347  
## 4 Apr 2014 50445595 50427023 18572  
## 5 May 2014 58355947 58335751 20196  
## 6 Jun 2014 49492064 49481954 10110  
## 7 Jul 2014 49130087 49120535 9552  
## 8 Aug 2014 29280277 29288626 - 8349  
## 9 Sep 2014 32785593 32790277 - 4684  
## 10 Oct 2014 36523133 36526294 - 3161  
## # ... with 38 more rows



##   
## Ljung-Box test  
##   
## data: Residuals from Regression with ARIMA(0,1,1)(0,1,0)[12] errors  
## Q\* = 21.204, df = 22, p-value = 0.5081  
##   
## Model df: 2. Total lags used: 24



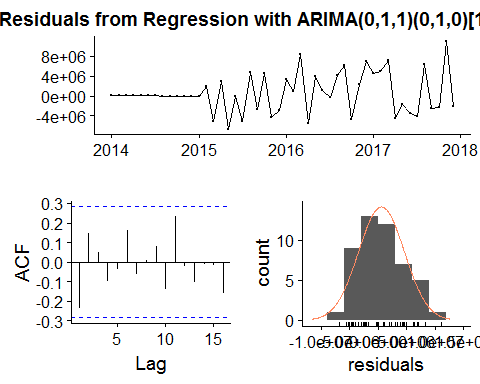
## # A tibble: 108 x 7  
## index key value lo.80 lo.95 hi.80 hi.95  
## <S3: yearmon> <chr> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 Jan 2014 actual 26281759 NA NA NA NA  
## 2 Feb 2014 actual 25886334 NA NA NA NA  
## 3 Mar 2014 actual 37873874 NA NA NA NA  
## 4 Apr 2014 actual 50445595 NA NA NA NA  
## 5 May 2014 actual 58355947 NA NA NA NA  
## 6 Jun 2014 actual 49492064 NA NA NA NA  
## 7 Jul 2014 actual 49130087 NA NA NA NA  
## 8 Aug 2014 actual 29280277 NA NA NA NA  
## 9 Sep 2014 actual 32785593 NA NA NA NA  
## 10 Oct 2014 actual 36523133 NA NA NA NA  
## # ... with 98 more rows



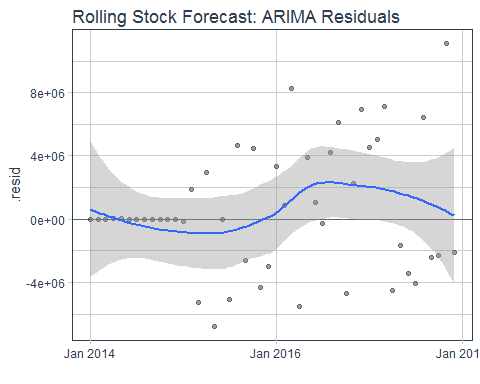
## # A tibble: 2 x 2  
## term estimate  
## <chr> <dbl>  
## 1 ma1 -0.773   
## 2 xreg 0.0757

## sigma logLik AIC BIC  
## 1 4849915 -587.8946 1181.789 1186.455

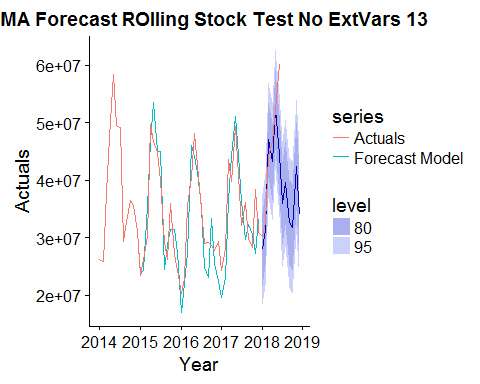
## # A tibble: 48 x 4  
## index .actual .fitted .resid  
## <S3: yearmon> <dbl> <dbl> <dbl>  
## 1 Jan 2014 26281759 26270812 10947  
## 2 Feb 2014 25886334 25880321 6013  
## 3 Mar 2014 37873874 37862224 11650  
## 4 Apr 2014 50445595 50427232 18363  
## 5 May 2014 58355947 58335712 20235  
## 6 Jun 2014 49492064 49482229 9835  
## 7 Jul 2014 49130087 49121281 8806  
## 8 Aug 2014 29280277 29291103 -10826  
## 9 Sep 2014 32785593 32789824 - 4231  
## 10 Oct 2014 36523133 36525138 - 2005  
## # ... with 38 more rows



##   
## Ljung-Box test  
##   
## data: Residuals from Regression with ARIMA(0,1,1)(0,1,0)[12] errors  
## Q\* = 18.497, df = 22, p-value = 0.6761  
##   
## Model df: 2. Total lags used: 24



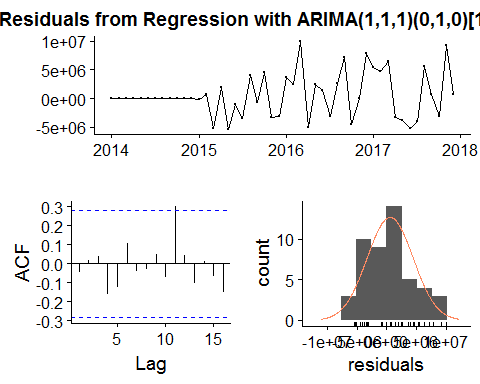
## # A tibble: 108 x 7  
## index key value lo.80 lo.95 hi.80 hi.95  
## <S3: yearmon> <chr> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 Jan 2014 actual 26281759 NA NA NA NA  
## 2 Feb 2014 actual 25886334 NA NA NA NA  
## 3 Mar 2014 actual 37873874 NA NA NA NA  
## 4 Apr 2014 actual 50445595 NA NA NA NA  
## 5 May 2014 actual 58355947 NA NA NA NA  
## 6 Jun 2014 actual 49492064 NA NA NA NA  
## 7 Jul 2014 actual 49130087 NA NA NA NA  
## 8 Aug 2014 actual 29280277 NA NA NA NA  
## 9 Sep 2014 actual 32785593 NA NA NA NA  
## 10 Oct 2014 actual 36523133 NA NA NA NA  
## # ... with 98 more rows



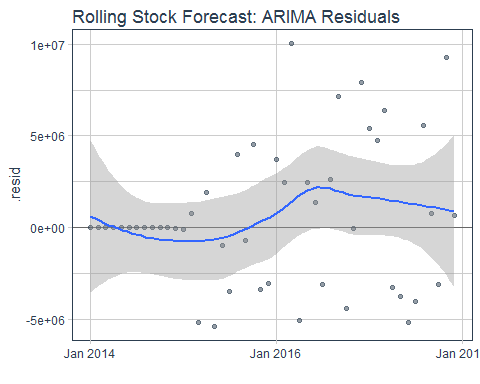
## # A tibble: 3 x 2  
## term estimate  
## <chr> <dbl>  
## 1 ar1 -0.352  
## 2 ma1 -0.660  
## 3 xreg 0.361

## sigma logLik AIC BIC  
## 1 4750782 -586.7387 1181.477 1187.699

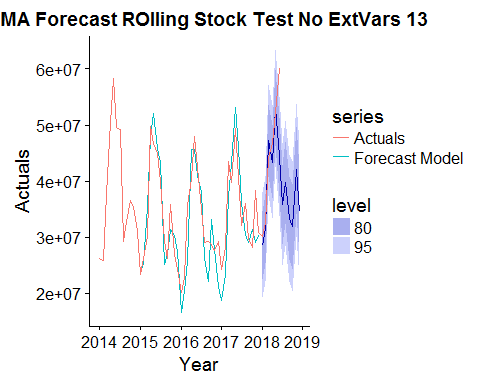
## # A tibble: 48 x 4  
## index .actual .fitted .resid  
## <S3: yearmon> <dbl> <dbl> <dbl>  
## 1 Jan 2014 26281759 26270709 11050  
## 2 Feb 2014 25886334 25879705 6629  
## 3 Mar 2014 37873874 37862579 11295  
## 4 Apr 2014 50445595 50427244 18351  
## 5 May 2014 58355947 58335534 20413  
## 6 Jun 2014 49492064 49482329 9735  
## 7 Jul 2014 49130087 49121276 8811  
## 8 Aug 2014 29280277 29290110 - 9833  
## 9 Sep 2014 32785593 32790727 - 5134  
## 10 Oct 2014 36523133 36525591 - 2458  
## # ... with 38 more rows



##   
## Ljung-Box test  
##   
## data: Residuals from Regression with ARIMA(1,1,1)(0,1,0)[12] errors  
## Q\* = 16.165, df = 21, p-value = 0.7603  
##   
## Model df: 3. Total lags used: 24



## # A tibble: 108 x 7  
## index key value lo.80 lo.95 hi.80 hi.95  
## <S3: yearmon> <chr> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 Jan 2014 actual 26281759 NA NA NA NA  
## 2 Feb 2014 actual 25886334 NA NA NA NA  
## 3 Mar 2014 actual 37873874 NA NA NA NA  
## 4 Apr 2014 actual 50445595 NA NA NA NA  
## 5 May 2014 actual 58355947 NA NA NA NA  
## 6 Jun 2014 actual 49492064 NA NA NA NA  
## 7 Jul 2014 actual 49130087 NA NA NA NA  
## 8 Aug 2014 actual 29280277 NA NA NA NA  
## 9 Sep 2014 actual 32785593 NA NA NA NA  
## 10 Oct 2014 actual 36523133 NA NA NA NA  
## # ... with 98 more rows



print(Accuracy.df)

## Model AIC BIC ME RMSE MAE MPE MAPE  
## 1 No Xreg 1180.449 1185.115 524417.7 3932851 2870723 1.986190 8.691725  
## 2 Net Sales 1181.337 1187.559 488056.9 3872354 2830420 1.791008 8.561064  
## 3 CHEM 1182.884 1187.550 630037.4 4076102 2941025 2.158776 8.853765  
## 4 FERT 1182.884 1187.550 556174.4 3941413 2869156 1.804395 8.621303  
## 5 SEED 1181.789 1186.455 566671.9 4021337 2994217 1.879556 9.101208  
## 6 APPL 1181.477 1187.699 575386.5 3878997 2833662 2.100791 8.628610  
## ACF1 Theil's U  
## 1 -0.02451171 0.5740400  
## 2 -0.02668544 0.5712914  
## 3 -0.25571309 0.5823443  
## 4 -0.25225285 0.5800947  
## 5 -0.23524011 0.5791552  
## 6 -0.04560999 0.5761874

CHEM FERT TEST