ARIMA model for forecasting median home value (per sqft) in San Francisco city

Jigyasa Bhagat

December 12, 2015

Data and Plot We have data used is from April 1996 to October 2015

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(tidyr)
zhvi=read.csv(file="~/Desktop/Reserach Study/City_MedianValuePerSqft_AllHomes.cs
v") %>%
    filter(RegionName=="San Francisco") %>%
    gather(Years,median.ppsf,7:ncol(.))
zhvi$Years=gsub("^X","",zhvi$Years)
tail(zhvi)
```

```
##     RegionID    RegionName State         Metro    CountyName SizeRank
## 230    20330 San Francisco    CA San Francisco San Francisco       12
## 231    20330 San Francisco    CA San Francisco San Francisco       12
## 232    20330 San Francisco    CA San Francisco San Francisco       12
## 233    20330 San Francisco    CA San Francisco San Francisco       12
## 234    20330 San Francisco    CA San Francisco San Francisco       12
## 235    20330 San Francisco    CA San Francisco San Francisco       12
##       Years median.ppsf
## 230 2015.05         851
## 231 2015.06         866
## 232 2015.07         878
## 233 2015.08         889
## 234 2015.09         894
## 235 2015.10         895
```

```
dim(zhvi)
```

```
## [1] 235   8
```

```
ts_pricepersqft=ts(zhvi$median.ppsf,frequency=12,start = c(1996,4))
ts_pricepersqft
```

```
##      Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
## 1996                 198 199 201 202 203 203 203 204 206
## 1997 208 210 212 214 216 218 220 223 225 228 231 233
## 1998 237 240 243 245 247 249 252 255 257 260 264 267
## 1999 270 273 276 278 282 287 292 297 304 311 319 328
## 2000 337 347 355 363 370 377 383 388 393 396 399 401
## 2001 401 401 404 405 404 403 403 404 405 405 403 403
## 2002 405 407 407 409 415 424 431 438 442 444 445 447
## 2003 446 446 445 446 447 449 452 455 459 464 471 480
## 2004 489 498 508 519 527 531 534 538 543 550 559 567
## 2005 576 586 596 603 607 610 614 618 622 625 627 627
## 2006 625 623 621 620 621 621 620 619 619 617 615 613
## 2007 615 618 623 629 634 639 643 647 649 649 646 641
## 2008 638 633 627 621 615 608 601 594 587 583 584 582
## 2009 575 568 561 554 548 544 542 541 541 541 542 545
## 2010 548 551 552 549 547 545 542 540 539 537 535 534
## 2011 532 528 523 520 518 518 517 517 518 519 519 520
## 2012 523 526 529 533 537 543 550 559 569 580 589 597
## 2013 606 615 625 634 645 658 670 681 690 699 704 710
## 2014 717 725 736 748 758 765 770 774 777 780 785 790
## 2015 795 803 816 834 851 866 878 889 894 895
```

```
start(ts_pricepersqft)
```

```
## [1] 1996    4
```

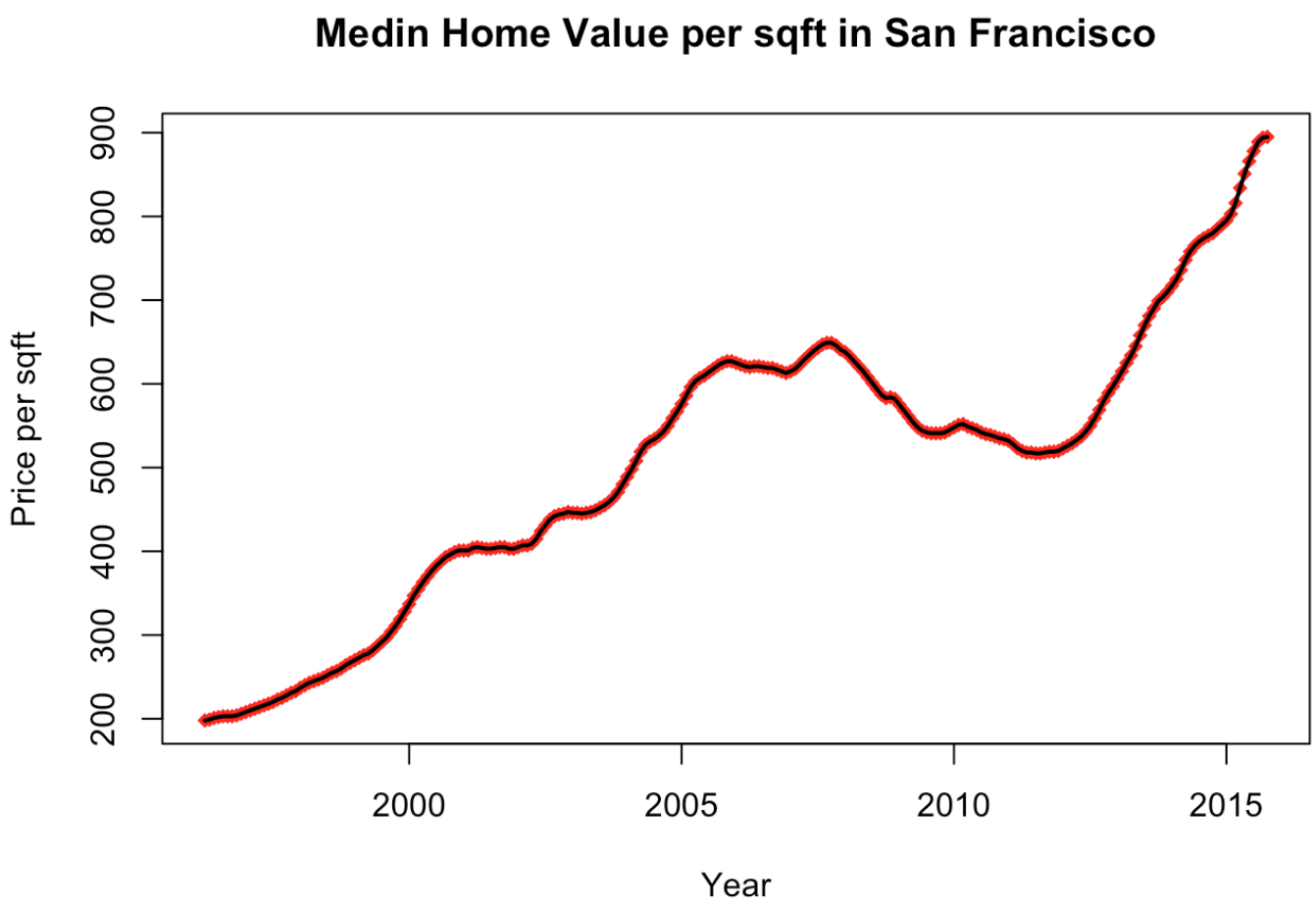
```
end(ts_pricepersqft)
```

```
## [1] 2015   10
```

```
frequency(ts_pricepersqft)
```

```
## [1] 12
```

```
par(mfrow=c(1,1))
plot(ts_pricepersqft,type="b",main="Medin Home Value per sqft in San Francisco",xl
ab="Year",ylab="Price per sqft",pch=18,col="red") # Non stationary series
lines(ts_pricepersqft,lwd=2)
```



First order differences

```
#install.packages("tseries")
#install.packages("astsa")
#install.packages("forecast")
library(tseries)
library(astsa)
library(forecast)
```

```
## Loading required package: zoo
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
## Loading required package: timeDate
```

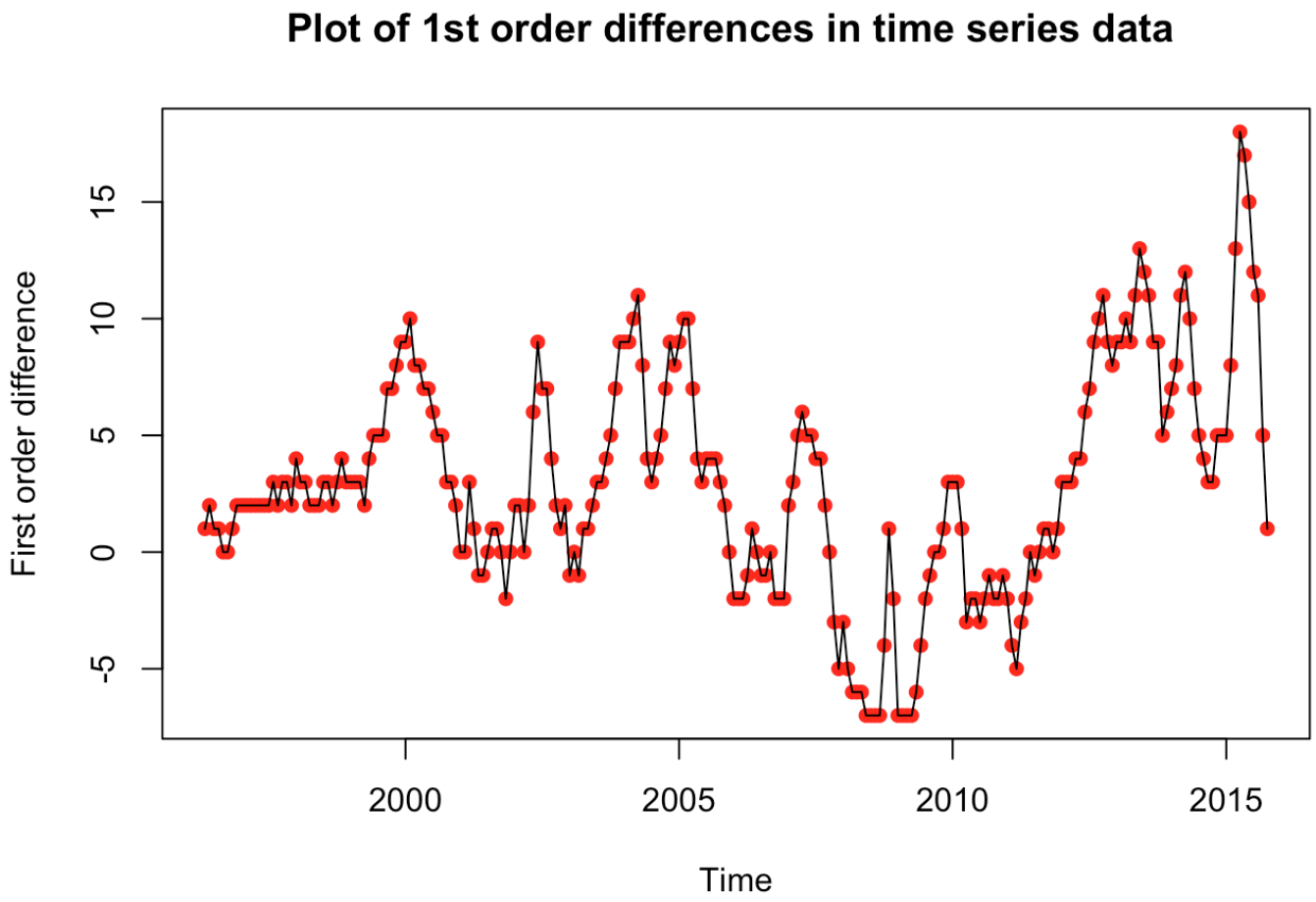
```
## This is forecast 6.2
```

```
##
## Attaching package: 'forecast'
```

```
## The following object is masked from 'package:astsa':
##
##     gas
```

```
diff1_mppsq=diff(ts_pricepersqft,1)

par(mfrow=c(1,1))
plot.ts(diff1_mppsq,type="p",col="red",pch=16,ylab="First order difference",main
="Plot of 1st order differences in time series data ") # still non stationary
lines(diff1_mppsq)
```



ADF Test

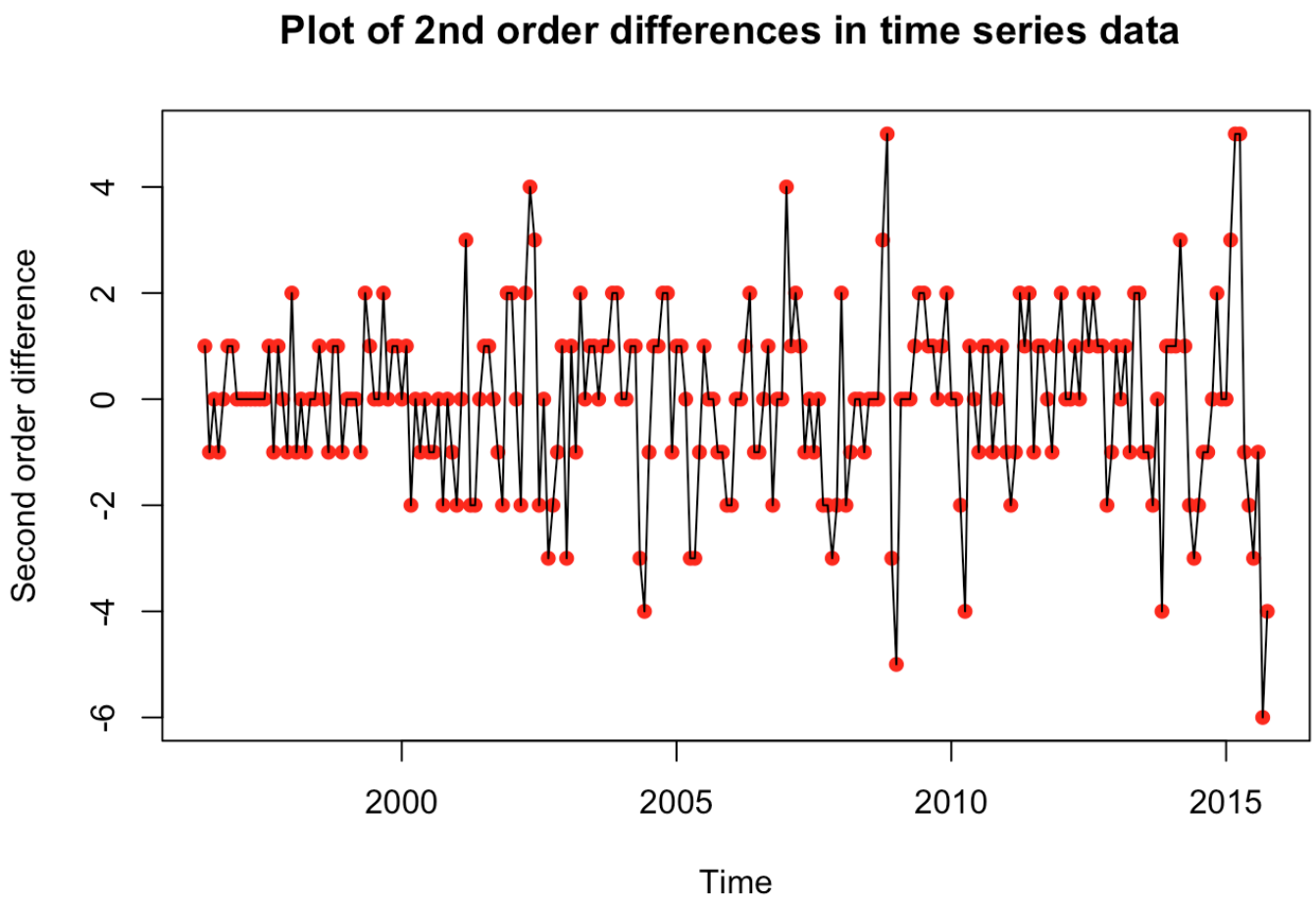
```
## Warning in kpss.test(diff1_mppsq, null = "Trend"): p-value smaller than
## printed p-value
```

```
##
##  KPSS Test for Trend Stationarity
##
## data:  diff1_mppsq
## KPSS Trend = 0.50195, Truncation lag parameter = 3, p-value = 0.01
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  diff1_mppsq
## Dickey-Fuller = -2.757, Lag order = 6, p-value = 0.2577
## alternative hypothesis: stationary
```

Still non stationary, 2nd order diff

```
diff1_mppsq=diff(ts_pricepersqft,differences = 2)
plot.ts(diff(ts_pricepersqft,differences = 2),type="p",col="red",pch=16,ylab="Seco
nd order difference",main="Plot of 2nd order differences in time series data ") #
still non stationary
lines(diff(ts_pricepersqft,differences = 2))
```




```
kpss.test(diff(ts_pricepersqft,differences = 2),null = "Trend") # for Trend
```

```
## Warning in kpss.test(diff(ts_pricepersqft, differences = 2), null =
## "Trend"): p-value greater than printed p-value
```

```
##
##  KPSS Test for Trend Stationarity
##
## data:  diff(ts_pricepersqft, differences = 2)
## KPSS Trend = 0.028802, Truncation lag parameter = 3, p-value = 0.1
```

```
adf.test(diff(ts_pricepersqft,differences=2),alternative = "stationary") # for sta
tionarity
```

```
## Warning in adf.test(diff(ts_pricepersqft, differences = 2), alternative =
## "stationary"): p-value smaller than printed p-value
```

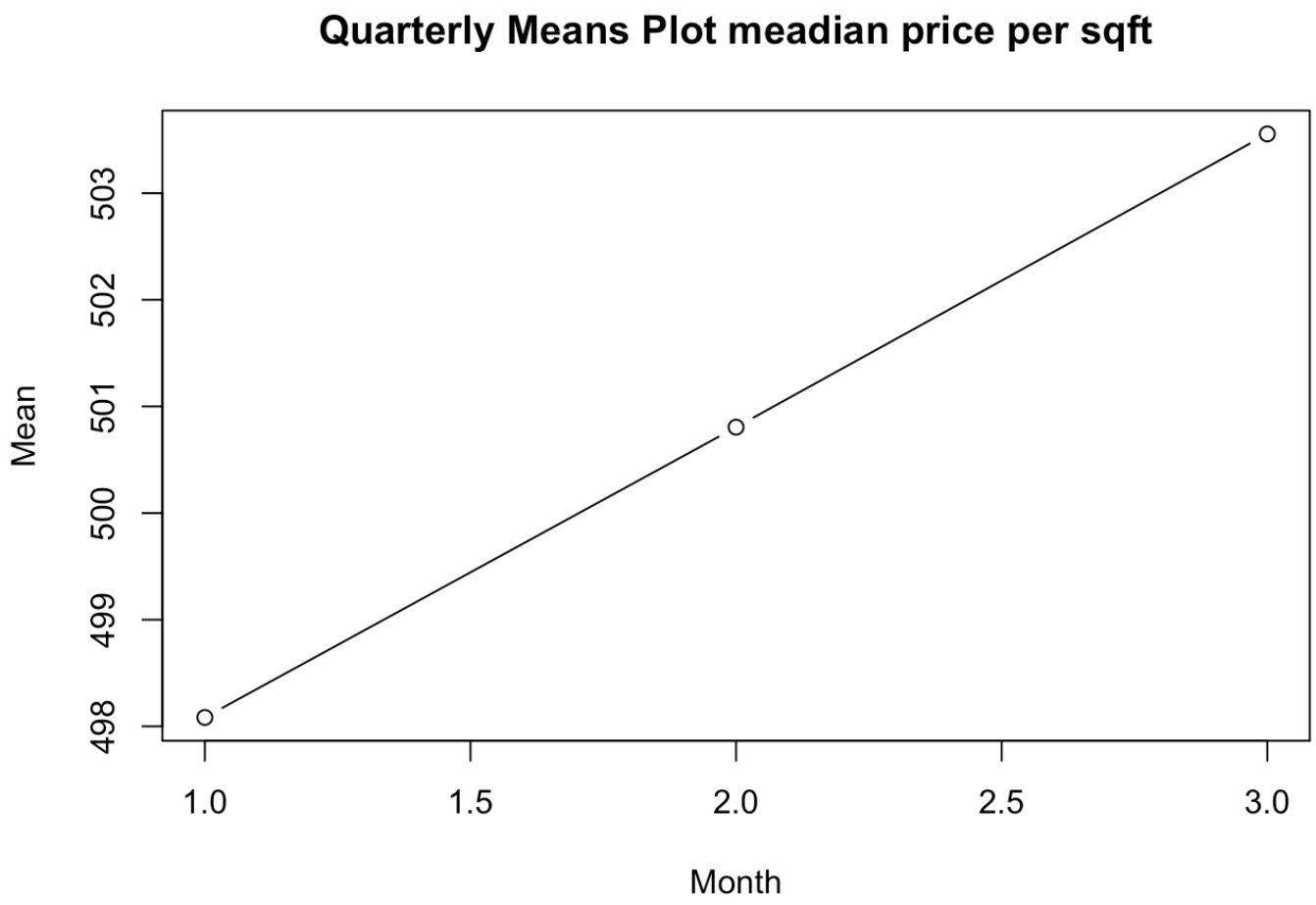
```
##
##  Augmented Dickey-Fuller Test
##
## data:  diff(ts_pricepersqft, differences = 2)
## Dickey-Fuller = -6.7551, Lag order = 6, p-value = 0.01
## alternative hypothesis: stationary
```

Check Monthly Seasonality

```
ppsm=matrix(ts_pricepersqft[10:225],ncol=12,byrow=T)
col.means=apply(ppsm,2,mean)
plot(col.means,type="b", main="Monthly Means Plot meadian price per sqft", xlab="M
onth", ylab="Mean") # This shows there is just trend  and nomonthly seasonality
```

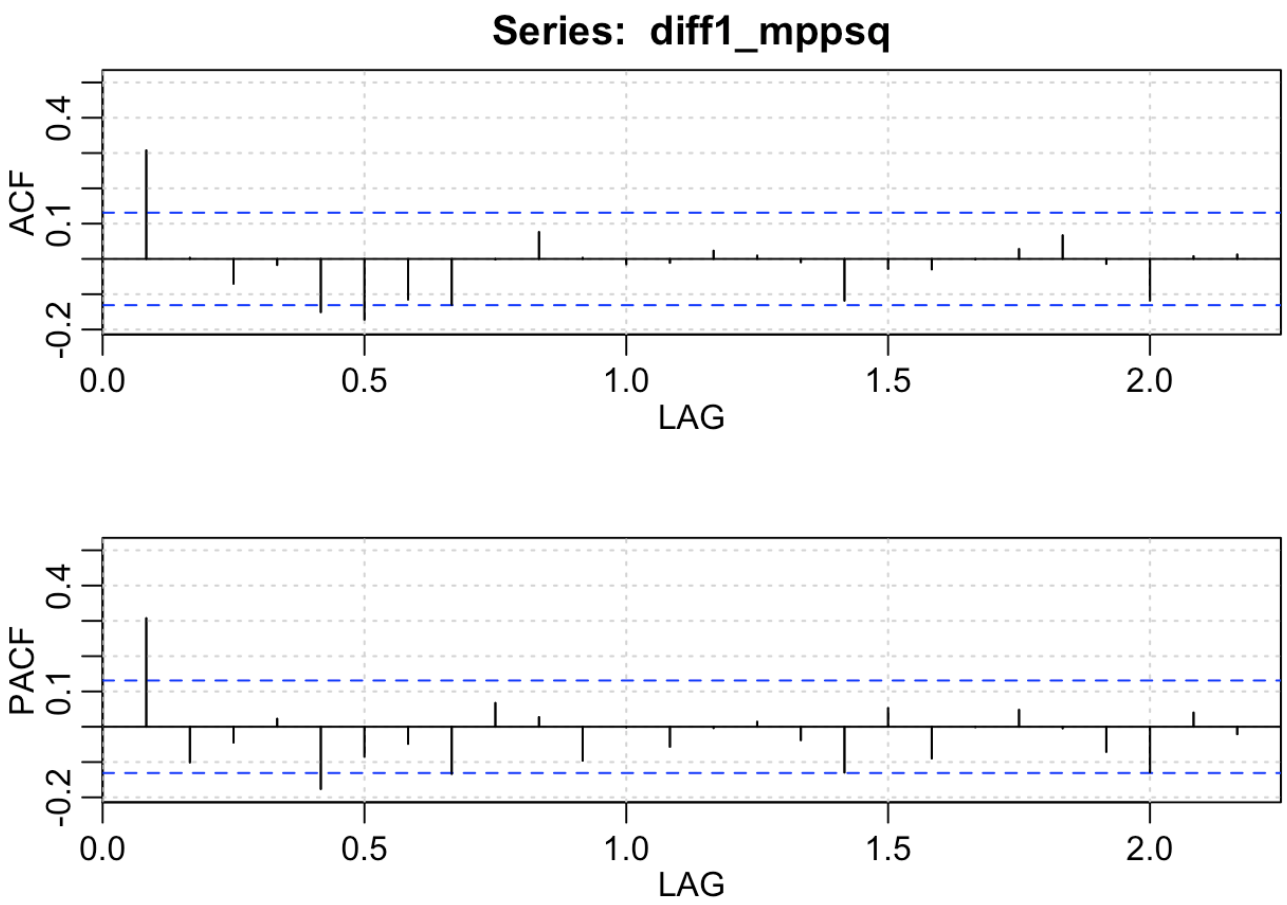
Check quarterly seasonality

```
ppsq=matrix(ts_pricepersqft[10:225],ncol=3,byrow=T)
col.meansq=apply(ppsq,2,mean)
plot(col.meansq,type="b", main="Quarterly Means Plot meadian price per sqft", xlab
="Month", ylab="Mean")
```



ACF and PACF

```
acf2(diff1_mppsq)
```


Series: diff1_mppsq
ACF
0.4
0.1
-0.2
0.0
0.5
1.0
1.5
2.0
LAG
PACF
0.4
0.1
-0.2
0.0
0.5
1.0
1.5
2.0
LAG

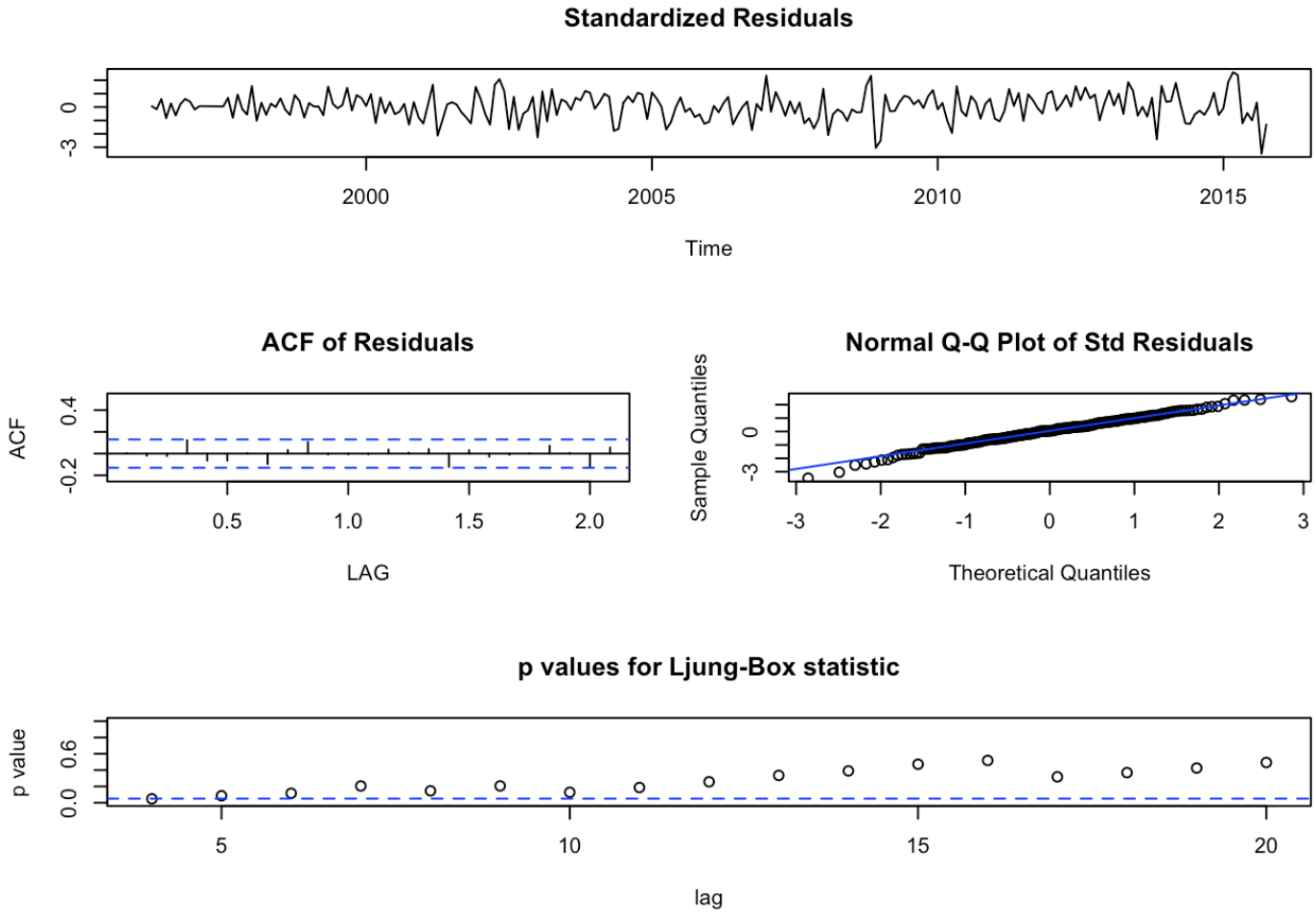
```
##          ACF  PACF
##  [1,]  0.31  0.31
##  [2,]  0.00 -0.10
##  [3,] -0.07 -0.04
##  [4,] -0.02  0.02
##  [5,] -0.15 -0.18
##  [6,] -0.17 -0.09
##  [7,] -0.11 -0.05
##  [8,] -0.13 -0.13
##  [9,]  0.00  0.07
## [10,]  0.08  0.03
## [11,]  0.00 -0.10
## [12,] -0.02  0.00
## [13,] -0.01 -0.06
## [14,]  0.02  0.00
## [15,]  0.01  0.01
## [16,] -0.01 -0.04
## [17,] -0.12 -0.13
## [18,] -0.03  0.05
## [19,] -0.03 -0.09
## [20,]  0.00  0.00
## [21,]  0.03  0.05
## [22,]  0.07  0.00
## [23,] -0.01 -0.07
## [24,] -0.12 -0.13
## [25,]  0.01  0.04
## [26,]  0.01 -0.02
```

Final ARIMA model after multipe trials

```
sarima(ts_pricepersqft,2,2,1)
```

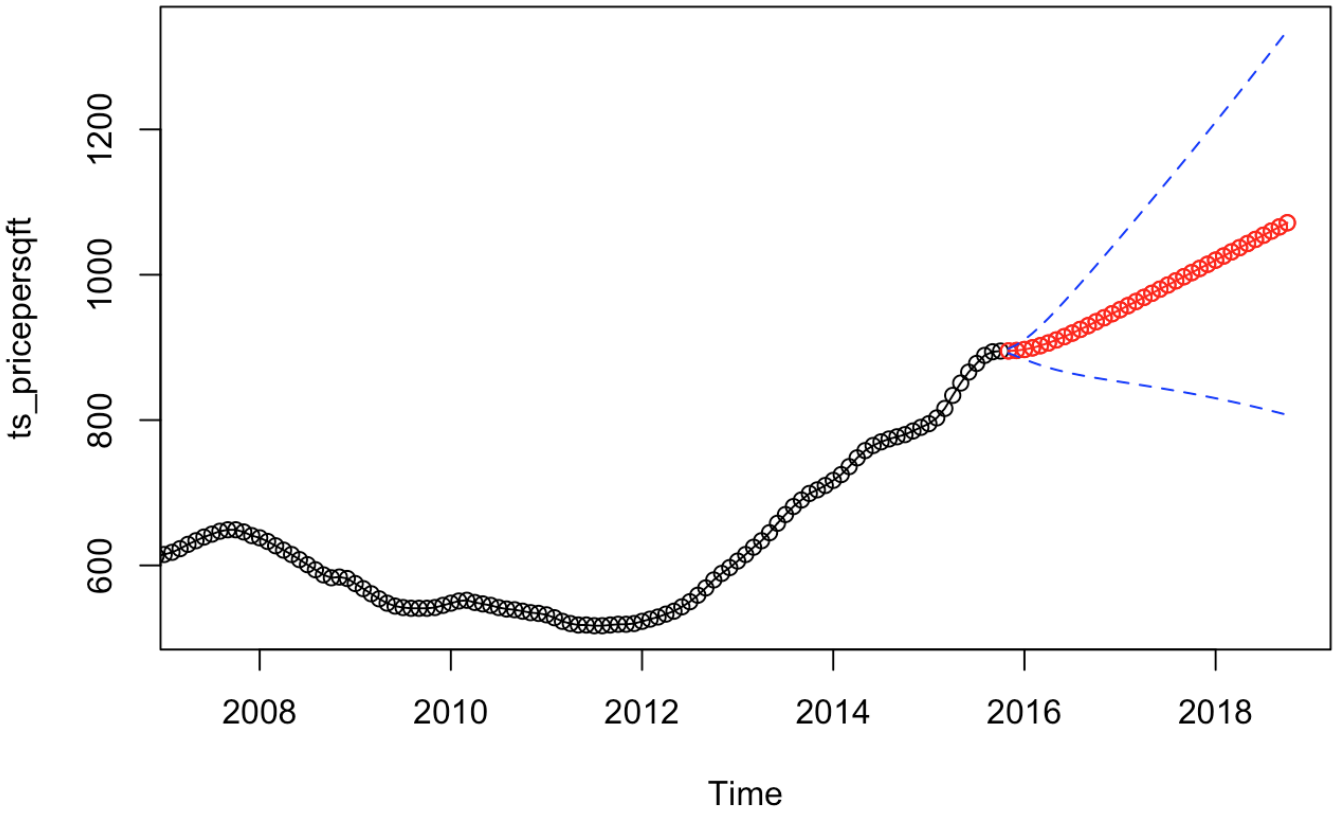
```
## initial  value 0.511085
## iter   2 value 0.494723
## iter   3 value 0.456838
## iter   4 value 0.455611
## iter   5 value 0.455567
## iter   6 value 0.455546
## iter   7 value 0.455198
## iter   8 value 0.454896
## iter   9 value 0.454811
## iter  10 value 0.454660
## iter  11 value 0.454652
## iter  12 value 0.454637
## iter  13 value 0.454575
## iter  14 value 0.454459
## iter  15 value 0.454035
## iter  16 value 0.453896
## iter  17 value 0.453718
## iter  18 value 0.453685
## iter  19 value 0.453666
## iter  20 value 0.453244
## iter  21 value 0.452698
## iter  22 value 0.452381
## iter  23 value 0.450493
## iter  24 value 0.450026
## iter  25 value 0.449660
## iter  26 value 0.448271
## iter  27 value 0.447552
## iter  28 value 0.447012
## iter  29 value 0.446778
## iter  30 value 0.446251
## iter  31 value 0.445225
## iter  32 value 0.444958
## iter  33 value 0.444788
## iter  34 value 0.444702
## iter  35 value 0.444658
## iter  36 value 0.444654
## iter  37 value 0.444654
## iter  38 value 0.444654
## iter  39 value 0.444654
## iter  39 value 0.444654
## iter  39 value 0.444654
## final  value 0.444654
## converged
## initial  value 0.437437
## iter   2 value 0.437149
## iter   3 value 0.436562
## iter   4 value 0.435987
## iter   5 value 0.435275
## iter   6 value 0.434863
## iter   7 value 0.434795
```

```
## iter   8 value 0.434784
## iter   9 value 0.434783
## iter  10 value 0.434783
## iter  11 value 0.434783
## iter  12 value 0.434783
## iter  13 value 0.434783
## iter  14 value 0.434783
## iter  15 value 0.434783
## iter  15 value 0.434783
## final  value 0.434783
## converged
```



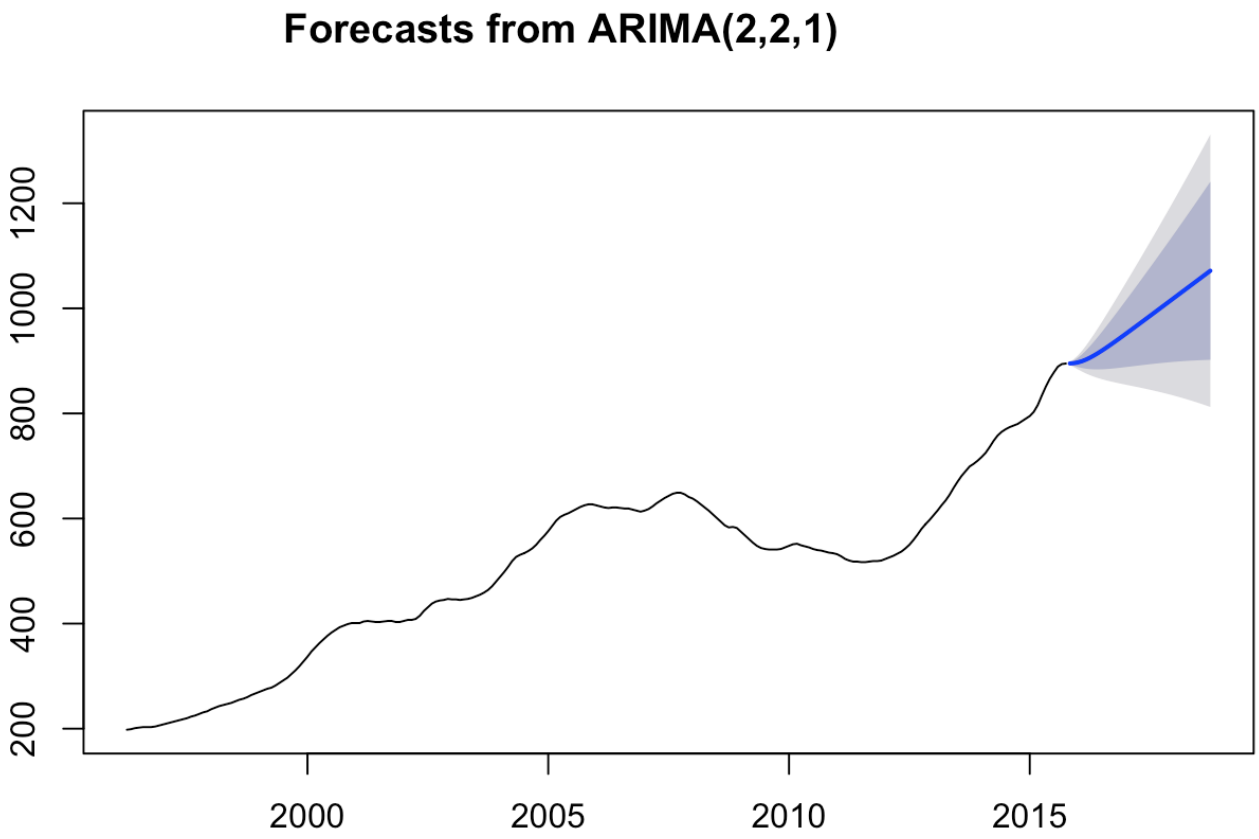
```
## $fit
##
## Call:
## stats::arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D,
##     Q), period = S), include.mean = !no.constant, optim.control = list(trace =
trc,
##     REPORT = 1, reltol = tol))
##
## Coefficients:
##          ar1      ar2      ma1
##       1.2357  -0.3668  -0.9411
## s.e.  0.0843   0.0636   0.0715
##
## sigma^2 estimated as 2.381:  log likelihood = -431.92,  aic = 871.83
##
## $AIC
## [1] 1.892877
##
## $AICc
## [1] 1.902128
##
## $BIC
## [1] 0.9370419
```

```
sarima.for(ts_pricepersqft,36,2,2,1)
```


ts_pricepersqft
600
800
1000
1200
2008
2010
2012
2014
2016
2018
Time

```
## $pred
##            Jan       Feb       Mar       Apr       May       Jun       Jul
## 2015                                                                       
## 2016  897.1198  899.3885  902.4286  906.1027  910.2771  914.8374  919.6908
## 2017  951.9155  957.5254  963.1630  968.8211  974.4943  980.1788  985.8716
## 2018 1020.1140 1025.8276 1031.5419 1037.2568 1042.9720 1048.6876 1054.4034
##            Aug       Sep       Oct       Nov       Dec
## 2015                                 895.1514  895.7213
## 2016  924.7649  930.0042  935.3667  940.8208  946.3429
## 2017  991.5706  997.2740 1002.9809 1008.6903 1014.4015
## 2018 1060.1194 1065.8355 1071.5517                     
##
## $se
##             Jan        Feb        Mar        Apr        May        Jun
## 2015                                                                  
## 2016   6.747676   9.980676  13.415091  16.959502  20.559093  24.182627
## 2017  49.608973  53.264603  56.936010  60.626322  64.338343  68.074544
## 2018  95.053035  99.039332 103.060989 107.118459 111.212104 115.342212
##             Jul        Aug        Sep        Oct        Nov        Dec
## 2015                                                 1.542914   3.861890
## 2016  27.813631  31.444663  35.073647  38.701585  42.331125  45.965693
## 2017  71.837084  75.627830  79.448383  83.300110  87.184171  91.101540
## 2018 119.508999 123.712626 127.953205 132.230803
```

```
plot(forecast(Arima(ts_pricepersqft,order = c(2,2,1)), h=36))
```




```
fit6=Arima(ts_pricepersqft,order = c(2,2,1))
```

Diagnostics

```
library(FitAR)
```

```
## Loading required package: lattice
```

```
## Loading required package: leaps
```

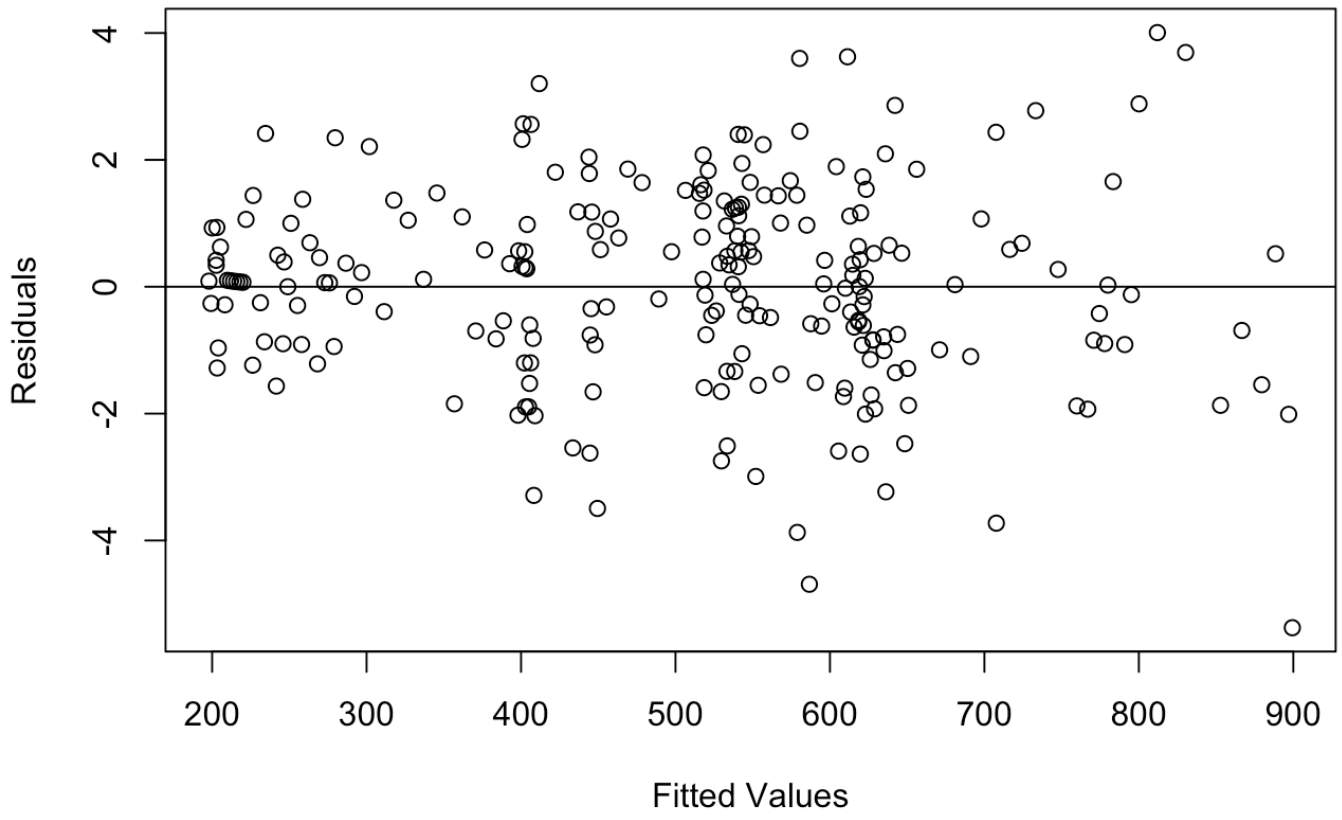
```
## Loading required package: ltsa
```

```
## Loading required package: bestglm
```

```
##
## Attaching package: 'FitAR'
```

```
## The following object is masked from 'package:forecast':
##
##     BoxCox
```

```
fit6=Arima(ts_pricepersqft,order = c(2,2,1))
plot(fitted(fit6), fit6$residuals,xlab="Fitted Values",ylab="Residuals") # Constan
t variance
abline(h=0)
```




```
cbind(fitted(fit6),fit6$x,residuals(fit6))
```

```
##          fitted(fit6) fit6$x residuals(fit6)
## Apr 1996     197.9115    198    0.0885482816
## May 1996     199.2634    199   -0.2634078002
## Jun 1996     200.0730    201    0.9270234923
## Jul 1996     203.2808    202   -1.2807829481
## Aug 1996     202.5824    203    0.4175950525
## Sep 1996     203.9646    203   -0.9646377582
## Oct 1996     202.6614    203    0.3385963804
## Nov 1996     203.0653    204    0.9347325156
## Dec 1996     205.3745    206    0.6254814026
## Jan 1997     208.2840    208   -0.2840224299
## Feb 1997     209.8987    210    0.1012510328
## Mar 1997     211.9062    212    0.0937789268
## Apr 1997     213.9130    214    0.0870352304
## May 1997     215.9191    216    0.0809181884
## Jun 1997     217.9247    218    0.0753451084
## Jul 1997     219.9298    220    0.0702480391
## Aug 1997     221.9383    223    1.0617405454
## Sep 1997     226.2361    225   -1.2360893584
## Oct 1997     226.5612    228    1.4388038617
## Nov 1997     231.2517    231   -0.2517442873
## Dec 1997     233.8675    233   -0.8674925284
## Jan 1998     234.5838    237    2.4162114052
## Feb 1998     241.5660    240   -1.5660015366
## Mar 1998     242.5027    243    0.4973395421
## Apr 1998     245.8982    245   -0.8982393261
## May 1998     246.6090    247    0.3910100242
## Jun 1998     248.9993    249    0.0007222281
## Jul 1998     251.0003    252    0.9997143139
## Aug 1998     255.2955    255   -0.2954684655
## Sep 1998     257.9103    257   -0.9103388296
## Oct 1998     258.6213    260    1.3786514169
## Nov 1998     263.3063    264    0.6937275178
## Dec 1998     268.2157    267   -1.2157252551
## Jan 1999     269.5413    270    0.4587140950
## Feb 1999     272.9353    273    0.0646681426
## Mar 1999     275.9392    276    0.0608127117
## Apr 1999     278.9425    278   -0.9424878114
## May 1999     279.6517    282    2.3482882485
## Jun 1999     286.6288    287    0.3711630154
## Jul 1999     292.1528    292   -0.1527911796
## Aug 1999     296.7770    297    0.2230000785
## Sep 1999     301.7906    304    2.2094409608
## Oct 1999     311.3923    311   -0.3922845973
## Nov 1999     317.6357    319    1.3642987619
## Dec 1999     326.9520    328    1.0479972905
## Jan 2000     336.8827    337    0.1172988421
## Feb 2000     345.5230    347    1.4770435951
## Mar 2000     356.8456    355   -1.8455730351
## Apr 2000     361.8987    363    1.1012900237
```

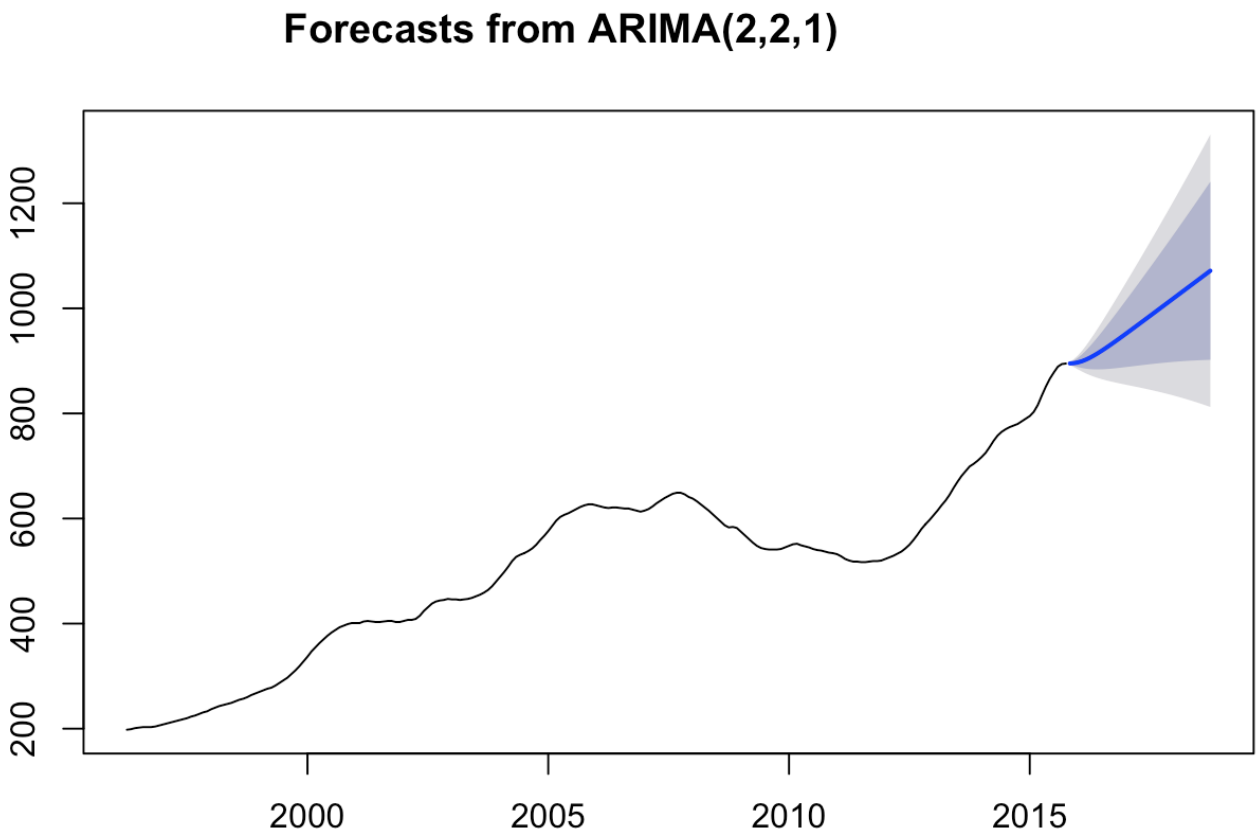
```
## May 2000     370.6972    370   -0.6972099698
## Jun 2000     376.4205    377    0.5795115529
## Jul 2000     383.8214    383   -0.8214123541
## Aug 2000     388.5373    388   -0.5373210236
## Sep 2000     392.6368    393    0.3631783171
## Oct 2000     398.0250    396   -2.0249567133
## Nov 2000     398.4343    399    0.5656527911
## Dec 2000     402.2013    401   -1.2012533193
## Jan 2001     402.8948    401   -1.8947770713
## Feb 2001     400.6787    401    0.3213479182
## Mar 2001     401.4312    404    2.5687562745
## Apr 2001     408.2895    405   -3.2894703774
## May 2001     405.5239    404   -1.5239478878
## Jun 2001     402.6965    403    0.3035133157
## Jul 2001     402.4480    403    0.5520183254
## Aug 2001     403.7162    404    0.2838478847
## Sep 2001     405.6017    405   -0.6017121355
## Oct 2001     406.1995    405   -1.1994547995
## Nov 2001     404.8931    403   -1.8931399819
## Dec 2001     400.6772    403    2.3228492667
## Jan 2002     404.0189    405    0.9811166330
## Feb 2002     407.8144    407   -0.8143583080
## Mar 2002     409.0328    407   -2.0327748555
## Apr 2002     406.4418    409    2.5582431681
## May 2002     411.7974    415    3.2026460837
## Jun 2002     422.1950    424    1.8050196937
## Jul 2002     433.5410    431   -2.5410166153
## Aug 2002     436.8196    438    1.1803532388
## Sep 2002     444.6228    442   -2.6227690009
## Oct 2002     444.7613    444   -0.7613447404
## Nov 2002     445.3456    445   -0.3456169279
## Dec 2002     445.8232    447    1.1767778581
## Jan 2003     449.4950    446   -3.4949796480
## Feb 2003     444.2154    446    1.7846128032
## Mar 2003     446.6566    445   -1.6565552439
## Apr 2003     443.9565    446    2.0434611151
## May 2003     447.9150    447   -0.9149935770
## Jun 2003     448.1275    449    0.8725036037
## Jul 2003     451.4145    452    0.5854667219
## Aug 2003     455.3179    455   -0.3178598898
## Sep 2003     457.9323    459    1.0676644924
## Oct 2003     463.2309    464    0.7691338971
## Nov 2003     469.1450    471    1.8549894647
## Dec 2003     478.3588    480    1.6412437394
## Jan 2004     489.1931    489   -0.1931081522
## Feb 2004     497.4481    498    0.5518740201
## Mar 2004     506.4806    508    1.5193713416
## Apr 2004     517.8058    519    1.1942380972
## May 2004     529.7449    527   -2.7449411343
## Jun 2004     533.5095    531   -2.5095229507
## Jul 2004     533.5195    534    0.4804617520
```

```
## Aug 2004     536.7794    538    1.2205970006
## Sep 2004     542.4537    543    0.5462555526
## Oct 2004     548.3548    550    1.6452381907
## Nov 2004     557.5562    559    1.4438465761
## Dec 2004     568.3789    567   -1.3788792996
## Jan 2005     574.3284    576    1.6715884802
## Feb 2005     585.0293    586    0.9706863074
## Mar 2005     595.9553    596    0.0446726241
## Apr 2005     605.5912    603   -2.5911535404
## May 2005     608.7316    607   -1.7315979394
## Jun 2005     610.0231    610   -0.0230792631
## Jul 2005     612.8865    614    1.1135164696
## Aug 2005     618.5545    618   -0.5545183602
## Sep 2005     622.1551    622   -0.1550557816
## Oct 2005     626.1459    625   -1.1459238688
## Nov 2005     627.8428    627   -0.8427850700
## Dec 2005     628.9243    627   -1.9243040306
## Jan 2006     626.7065    625   -1.7064773008
## Feb 2006     622.8683    623    0.1317160044
## Mar 2006     621.6097    621   -0.6096506190
## Apr 2006     619.5737    620    0.4262542471
## May 2006     619.8345    621    1.1654998167
## Jun 2006     623.0076    621   -2.0076377022
## Jul 2006     620.9201    620   -0.9201396817
## Aug 2006     618.9971    619    0.0028970456
## Sep 2006     618.3641    619    0.6359217595
## Oct 2006     619.6372    617   -2.6371808354
## Nov 2006     614.6438    615    0.3562394517
## Dec 2006     613.3984    613   -0.3983502985
## Jan 2007     611.3749    615    3.6251101965
## Feb 2007     618.5310    618   -0.5309897526
## Mar 2007     621.2681    623    1.7318505668
## Apr 2007     628.4746    629    0.5253582803
## May 2007     635.0076    634   -1.0076234554
## Jun 2007     638.3458    639    0.6541748751
## Jul 2007     643.7512    643   -0.7511568699
## Aug 2007     646.4713    647    0.5287324122
## Sep 2007     650.8692    649   -1.8692115020
## Oct 2007     650.2878    649   -1.2878247974
## Nov 2007     648.4743    646   -2.4742877022
## Dec 2007     642.3552    641   -1.3552243408
## Jan 2008     635.9045    638    2.0954775262
## Feb 2008     636.2328    633   -3.2328442472
## Mar 2008     627.8375    627   -0.8375382079
## Apr 2008     621.2862    621   -0.2861708946
## May 2008     615.6361    615   -0.6361217614
## Jun 2008     609.5987    608   -1.5986579010
## Jul 2008     601.2689    601   -0.2688557539
## Aug 2008     594.6198    594   -0.6198263816
## Sep 2008     587.5833    587   -0.5833222247
## Oct 2008     580.5490    583    2.4510320494
```

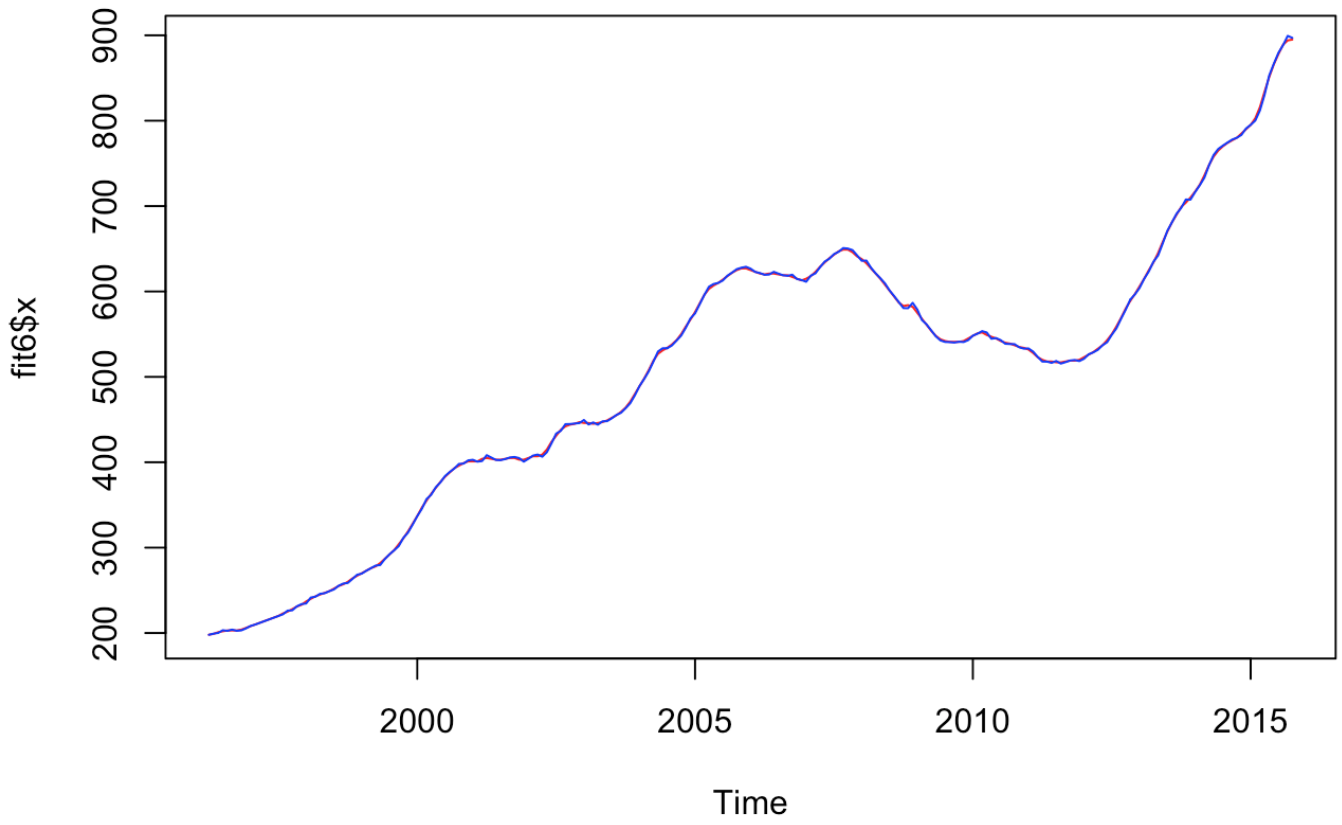
```
## Nov 2008     580.4003    584    3.5997289869
## Dec 2008     586.6901    582   -4.6901126824
## Jan 2009     578.8729    575   -3.8729175239
## Feb 2009     566.5670    568    1.4330133566
## Mar 2009     561.4854    561   -0.4854060812
## Apr 2009     554.4568    554   -0.4568184958
## May 2009     547.4299    548    0.5700854475
## Jun 2009     542.6991    544    1.3008602156
## Jul 2009     540.8802    542    1.1197507518
## Aug 2009     540.6839    541    0.3161122414
## Sep 2009     540.2045    541    0.7954538750
## Oct 2009     541.1202    541   -0.1202395790
## Nov 2009     540.7464    542    1.2536464933
## Dec 2009     543.0558    545    1.9441635053
## Jan 2010     548.2748    548   -0.2748327677
## Feb 2010     550.5250    551    0.4749626040
## Mar 2010     553.5530    552   -1.5530099213
## Apr 2010     551.9902    549   -2.9902456487
## May 2010     544.6051    547    2.3948551463
## Jun 2010     545.4491    545   -0.4490570224
## Jul 2010     543.0558    542   -1.0558055216
## Aug 2010     538.7580    540    1.2420257876
## Sep 2010     538.4336    539    0.5664225286
## Oct 2010     538.3358    537   -1.3357823156
## Nov 2010     534.6547    535    0.3453426557
## Dec 2010     533.0418    534    0.9581993259
## Jan 2011     533.3339    532   -1.3338835481
## Feb 2011     529.6529    528   -1.6528704031
## Mar 2011     523.4510    523   -0.4510295939
## Apr 2011     517.9224    520    2.0775746171
## May 2011     517.8829    518    0.1171118826
## Jun 2011     516.3918    518    1.6081734760
## Jul 2011     518.5910    517   -1.5910349451
## Aug 2011     515.5281    517    1.4719275792
## Sep 2011     517.2172    518    0.7827844465
## Oct 2011     519.1322    519   -0.1321628520
## Nov 2011     519.7576    519   -0.7575745685
## Dec 2011     518.4773    520    1.5226926782
## Jan 2012     521.1694    523    1.8305597774
## Feb 2012     526.3817    526   -0.3817458990
## Mar 2012     528.6257    529    0.3743460323
## Apr 2012     531.6477    533    1.3522992357
## May 2012     536.9630    537    0.0370061771
## Jun 2012     540.5984    543    2.4016313921
## Jul 2012     549.2111    550    0.7888882677
## Aug 2012     556.7596    559    2.2403861526
## Sep 2012     567.9961    569    1.0039440946
## Oct 2012     578.5572    580    1.4427764465
## Nov 2012     590.5110    589   -1.5110405289
## Dec 2012     596.5839    597    0.4160566542
## Jan 2013     604.1064    606    1.8935945456
```

```
## Feb 2013     614.8204    615    0.1796176895
## Mar 2013     623.4642    625    1.5358439187
## Apr 2013     634.7903    634   -0.7902588505
## May 2013     642.1413    645    2.8587379724
## Jun 2013     656.1477    658    1.8522692753
## Jul 2013     670.9945    670   -0.9945103009
## Aug 2013     680.9667    681    0.0333204033
## Sep 2013     691.0998    690   -1.0997961109
## Oct 2013     697.9305    699    1.0694718503
## Nov 2013     707.7271    704   -3.7271231257
## Dec 2013     707.5650    710    2.4349847778
## Jan 2014     716.4113    717    0.5887092081
## Feb 2014     724.3148    725    0.6851918085
## Mar 2014     733.2240    736    2.7759921469
## Apr 2014     747.7276    748    0.2723554958
## May 2014     759.8789    758   -1.8789211907
## Jun 2014     766.9302    765   -1.9301579837
## Jul 2014     770.8431    770   -0.8431406912
## Aug 2014     774.4226    774   -0.4225975620
## Sep 2014     777.8957    777   -0.8956678318
## Oct 2014     779.9741    780    0.0259276393
## Nov 2014     783.3424    785    1.6575959898
## Dec 2014     790.9113    790   -0.9113277918
## Jan 2015     795.1240    795   -0.1240465778
## Feb 2015     800.1167    803    2.8832590417
## Mar 2015     811.9935    816    4.0065003337
## Apr 2015     830.3073    834    3.6927022058
## May 2015     852.8690    851   -1.8690057181
## Jun 2015     866.6893    866   -0.6892583489
## Jul 2015     879.5442    878   -1.5441686659
## Aug 2015     888.4799    889    0.5201161085
## Sep 2015     899.3753    894   -5.3752791575
## Oct 2015     897.0116    895   -2.0116080937
```

```
plot(forecast(fit6, h=36))
```




```
plot(fit6$x, col='red')
lines(fitted(fit6), col='blue')
```

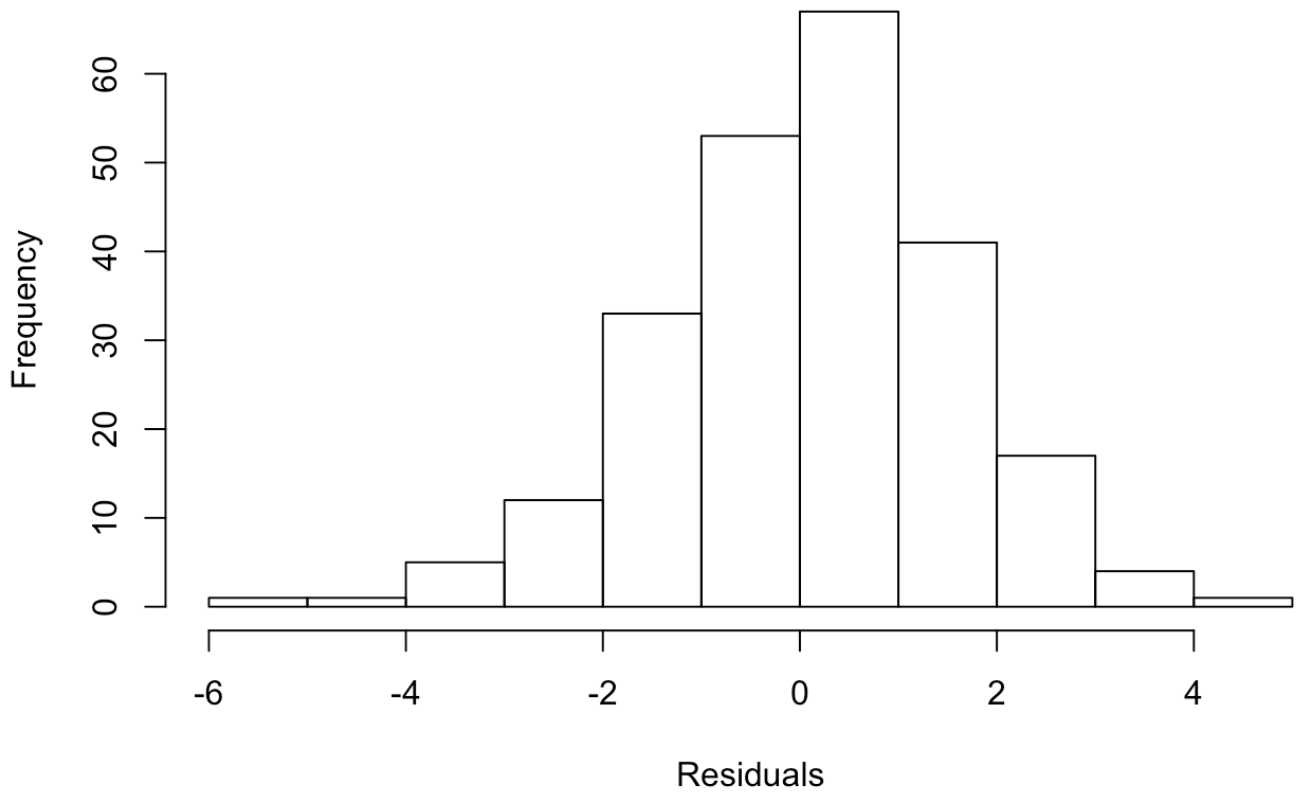



```
library(car)
```

```
##
## Attaching package: 'car'
```

```
## The following object is masked from 'package:FitAR':
##
##     Boot
```

```
hist(residuals(fit6),xlab="Residuals",main="Histogram of forecast residuals of ARI
MA(2,2,1)" )
```



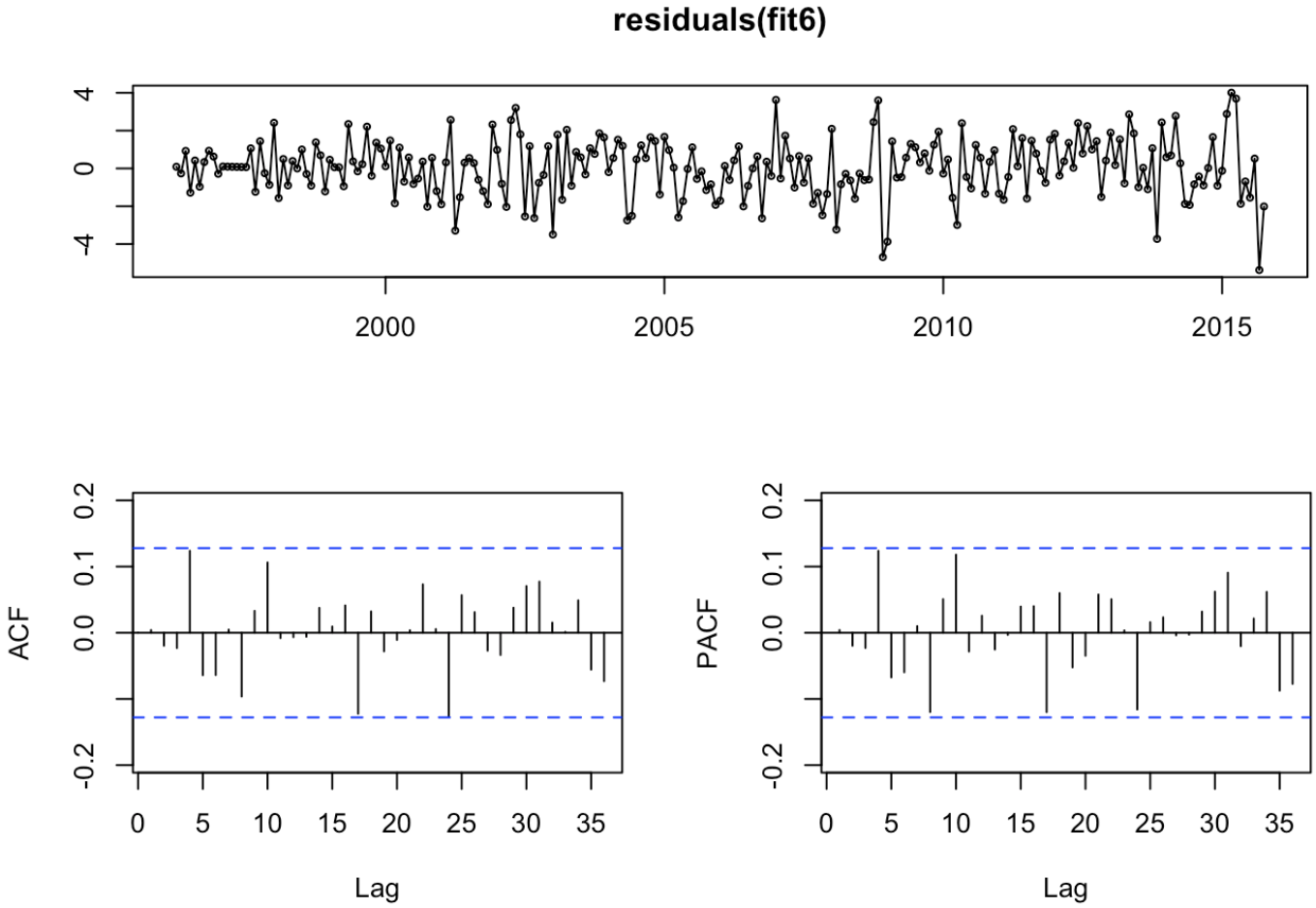

```
shapiro.test(residuals(fit6))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(fit6)
## W = 0.99269, p-value = 0.2989
```

```
dwt(as.vector(residuals(fit6)), alt="two.sided") # DUrbing watson test
```

```
## [1] 1.982399
```

```
d = sum((fit6$residuals - lag(fit6$residuals))^2, na.rm = TRUE) /
       sum(fit6$residuals^2, na.rm = TRUE)
tsdisplay(residuals(fit6))
```



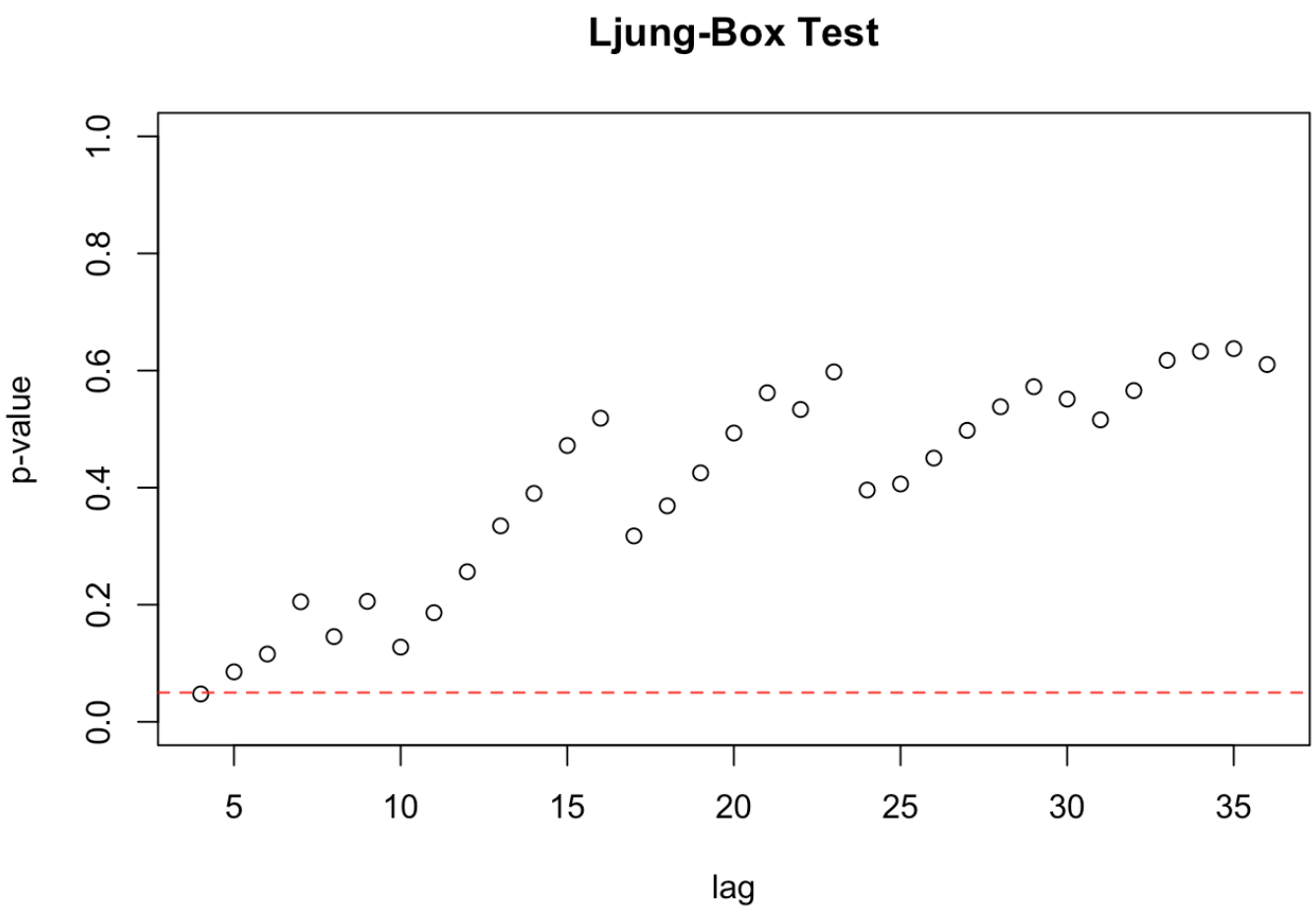

```
Box.test(residuals(Arima(ts_pricepersqft,order = c(2,2,1))),lag = 36,fitdf =3,type
= "Ljung-Box")
```

```
##
##  Box-Ljung test
##
## data:  residuals(Arima(ts_pricepersqft, order = c(2, 2, 1)))
## X-squared = 30.139, df = 33, p-value = 0.6103
```

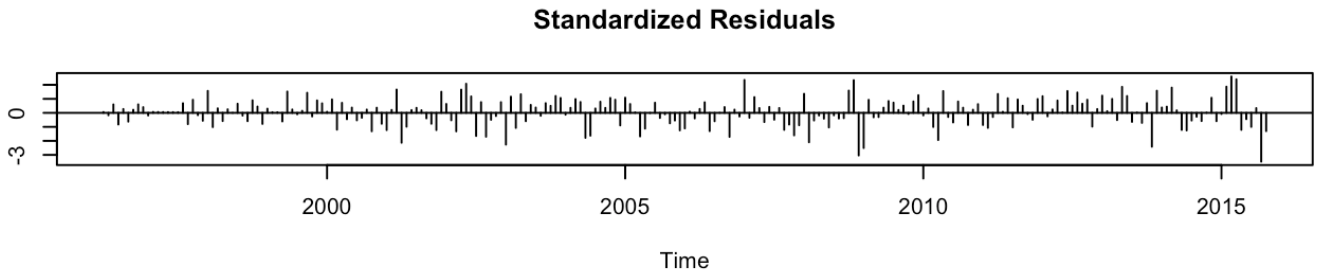
```
Box.test(residuals(Arima(ts_pricepersqft,order = c(2,2,1))),lag = 36,fitdf =3,type
= "Box-Pierce")
```

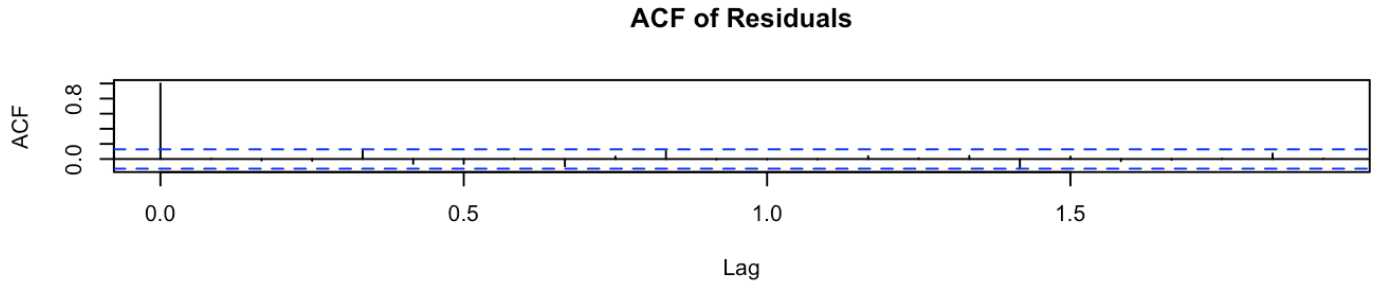
```
##
##  Box-Pierce test
##
## data:  residuals(Arima(ts_pricepersqft, order = c(2, 2, 1)))
## X-squared = 27.575, df = 33, p-value = 0.734
```

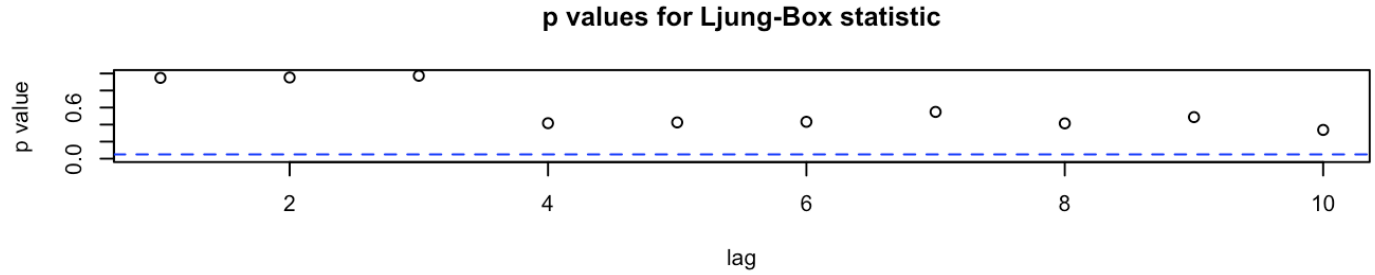
```
LBQPlot(residuals(Arima(ts_pricepersqft,order = c(2,2,1))), lag.max = 36,k = 3)
```



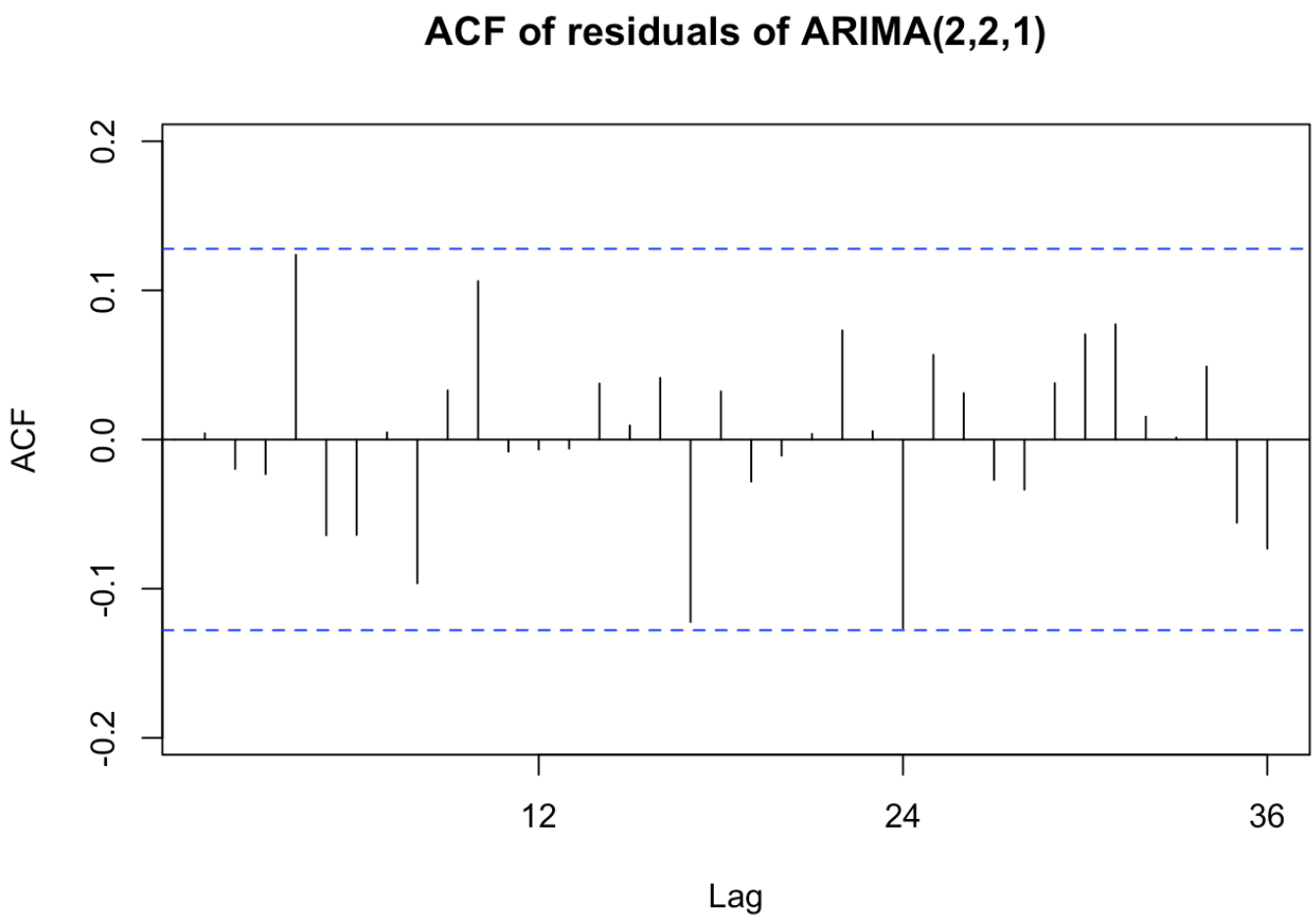

```
tsdiag(Arima(ts_pricepersqft,order = c(2,2,1)),which=1)
```



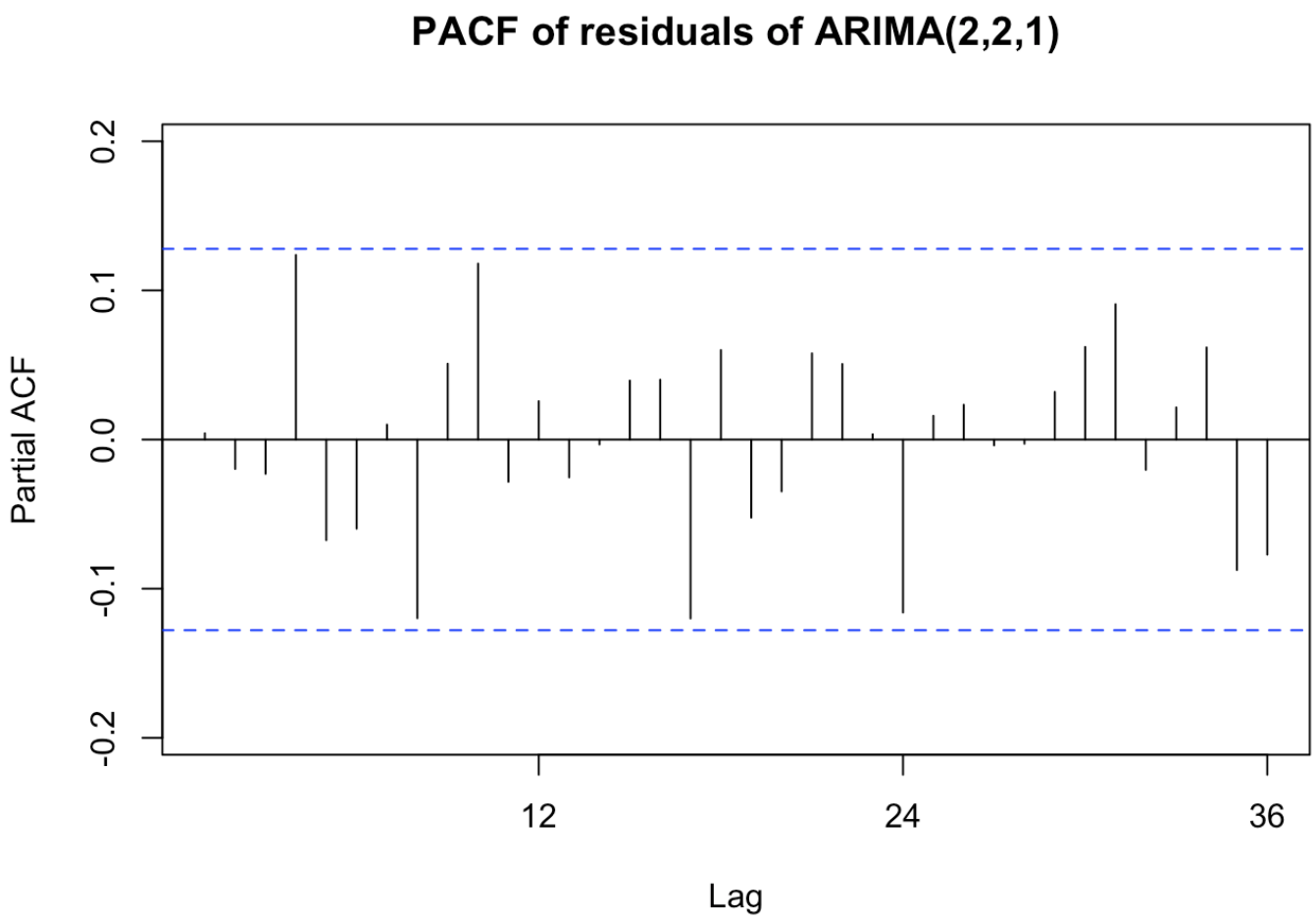




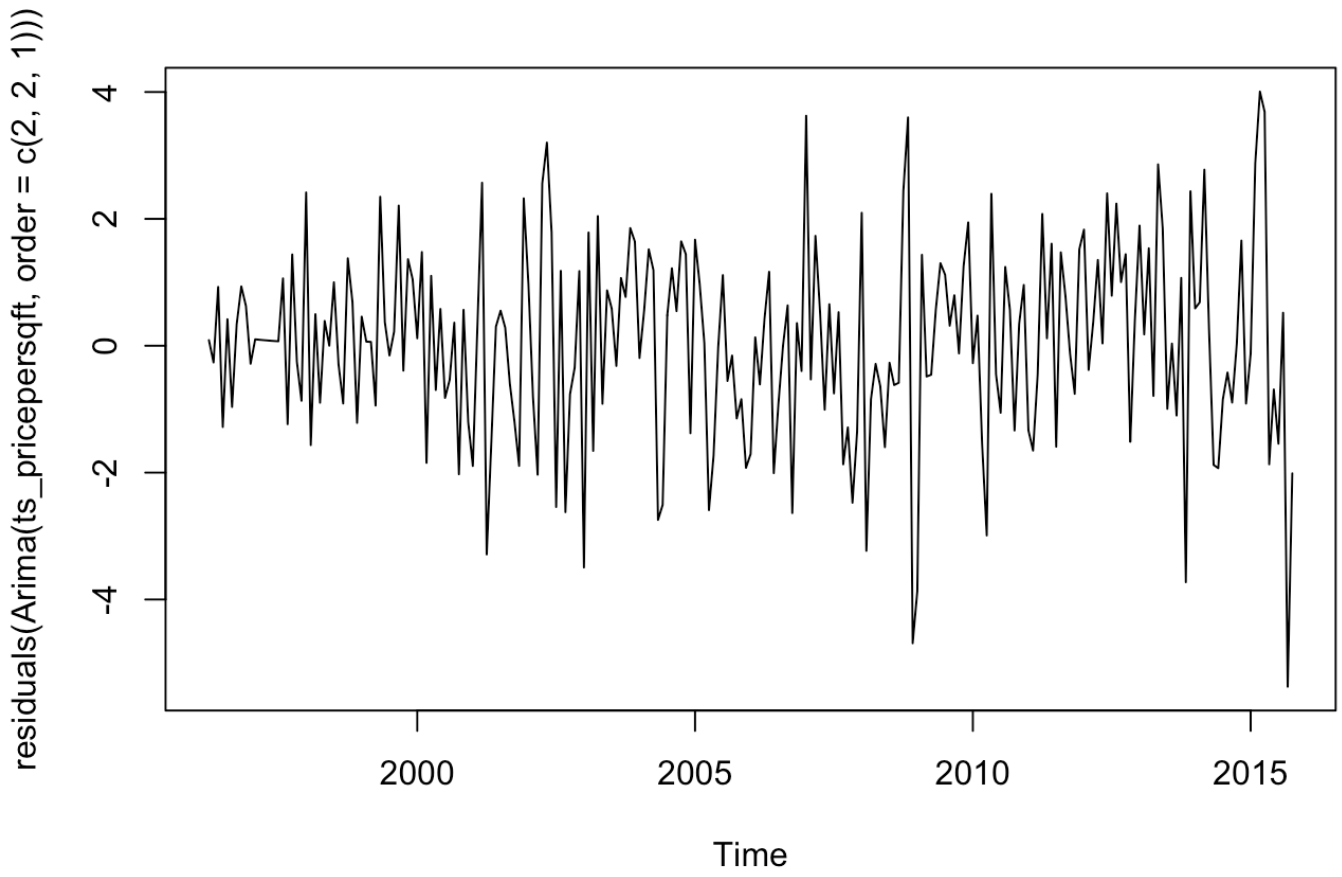

```
Acf(residuals(Arima(ts_pricepersqft,order = c(2,2,1))),lag.max = 36,main="ACF of r
esiduals of ARIMA(2,2,1)")
```




```
Pacf(residuals(Arima(ts_pricepersqft,order = c(2,2,1))),lag.max = 36,main="PACF of
residuals of ARIMA(2,2,1)")
```



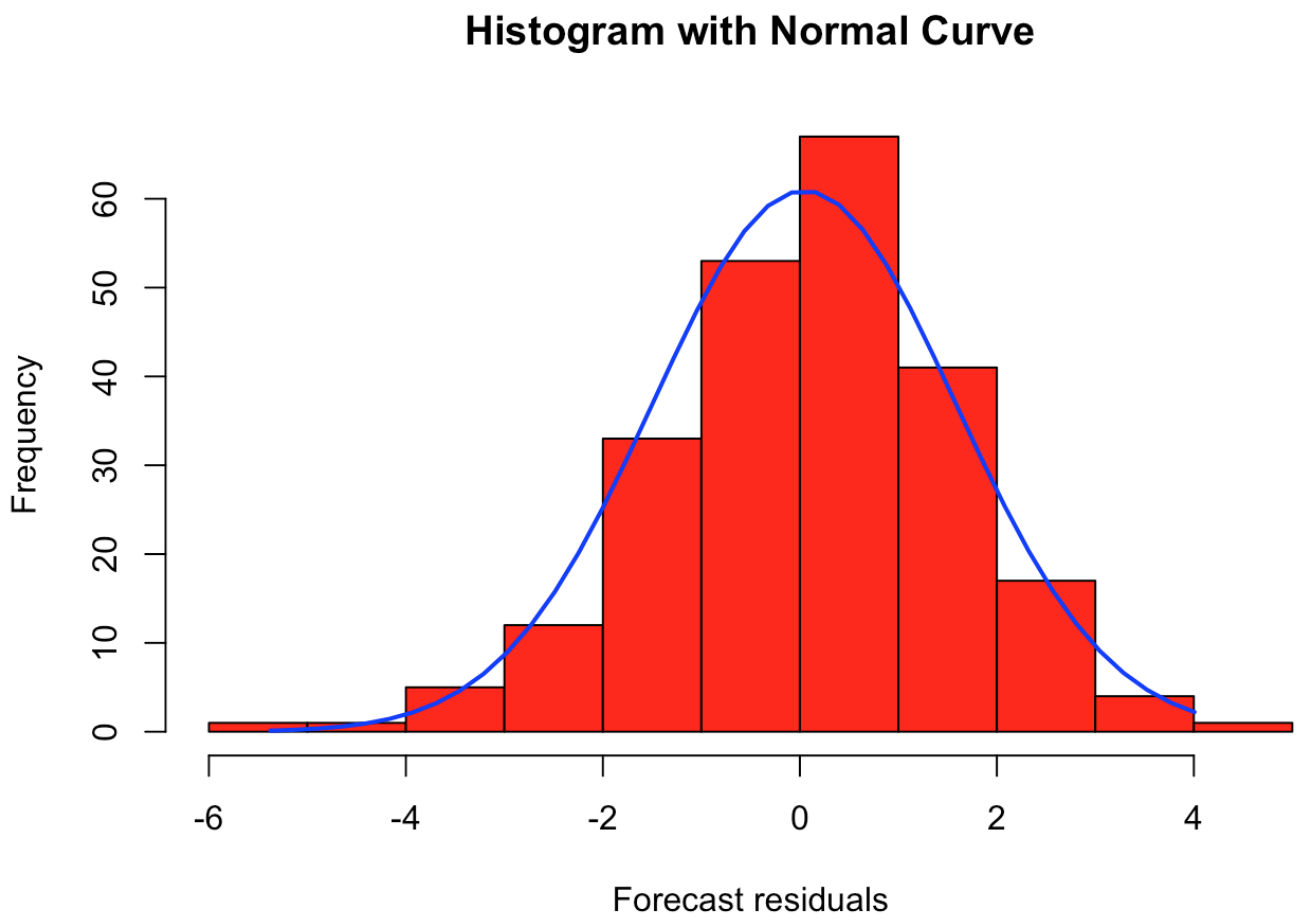

```
plot(residuals(Arima(ts_pricepersqft,order = c(2,2,1))),type="l")
```




```
2*(1-pnorm(abs(fit6$coef)/sqrt(diag(fit6$var.coef))))# check if the coeffocoents o
f AR and MA are significant
```

```
##          ar1          ar2          ma1
## 0.000000e+00 8.255667e-09 0.000000e+00
```

```
x <- residuals(fit6)
h<-hist(x, breaks=10, col="red", xlab="Forecast residuals",
      main="Histogram with Normal Curve")
xfit<-seq(min(x),max(x),length=40)
yfit<-dnorm(xfit,mean=mean(x),sd=sd(x))
yfit <- yfit*diff(h$mids[1:2])*length(x)
lines(xfit, yfit, col="blue", lwd=2)
```


Histogram with Normal Curve
Frequency
0
10
20
30
40
50
60
-6
-4
-2
0
2
4
Forecast residuals