Redstar Brand

GEORGE DREEMER

Research Process

The iterative process taking us from raw data to insights.

1
Our Dataset

2
Time Series Plots

3
Step 1:
Granger causalities

4
Step 2:
Stationary or Evolving?
& 4-Scenarios

5
Step 3:
VARx Model

6
Step 4:
Generating IRFs 1/2

7
Step 4:
Generating IRFs 2/2

8
Step 5:
Generating FEVD

Our Dataset

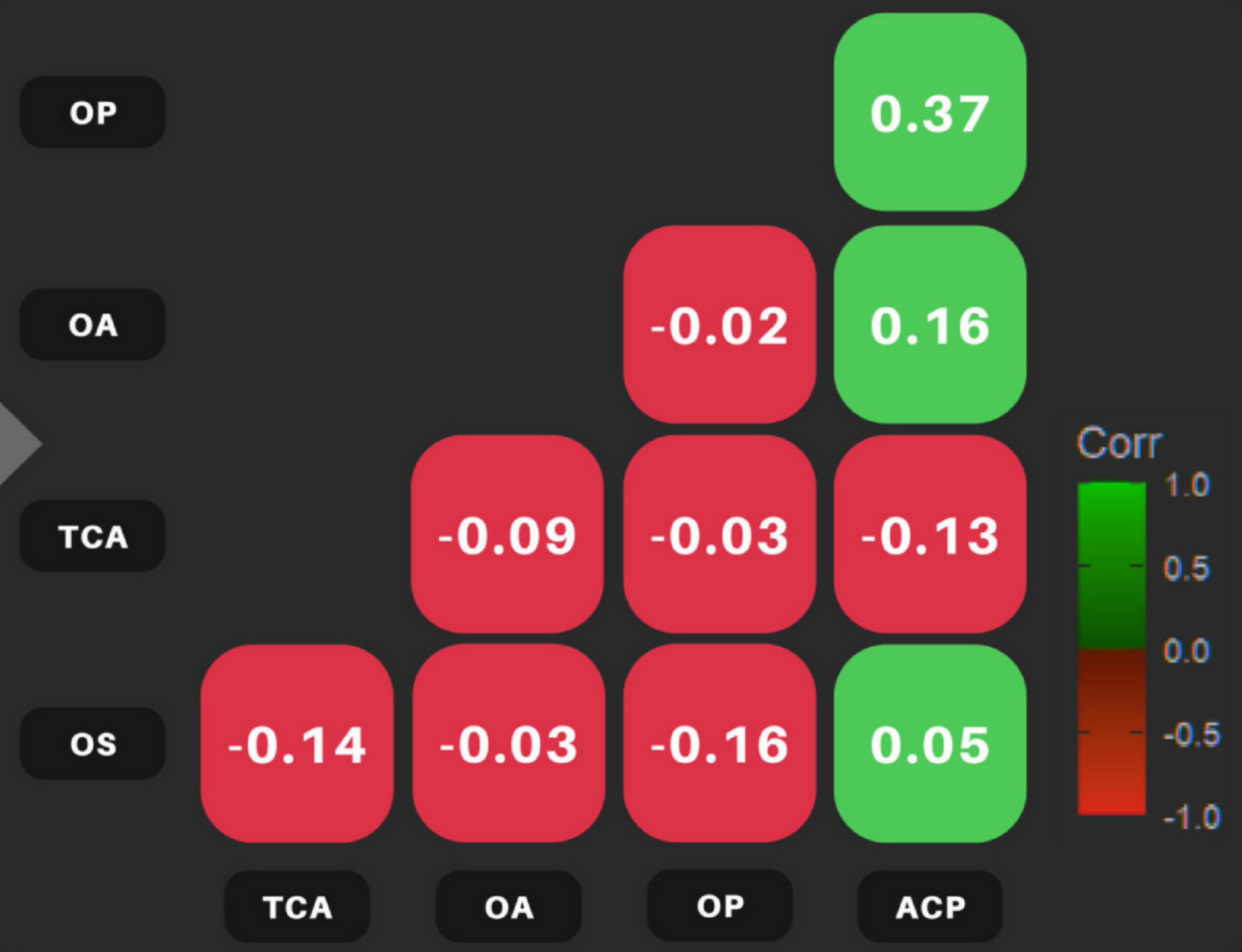
/our variables

Our dataset consists of data on our *own Sales (OS), Advertising (OA) & Price (OP)*, as well as, the *Total Competitor Advertising (TCA)* and *Average Competitor Price (ACP)* over a 207 week time period.

Correlations Findings

To get a rough idea of how the variables interact we created a correlation matrix and its corresponding plot.

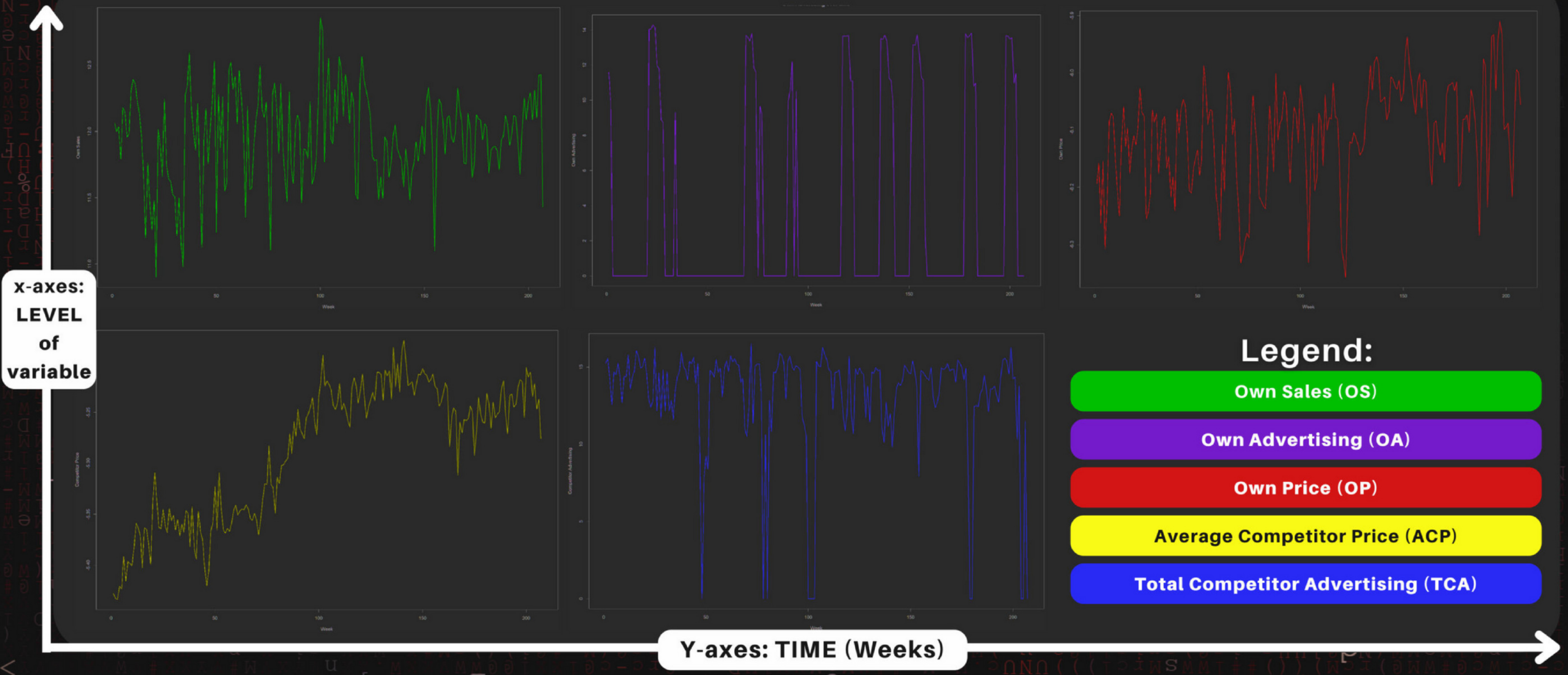
- **Own Price** has a positive correlation with **ACP**. *(Industry Price Setting?)*
- **Own Advertising** has a weak positive correlation with **ACP**. *(Seizing the opportunity?)*
- **Own Sales** has a weak negative correlation with **TCA**. *(More eyes on competitors less Sales for us?)*
- **Own Sales** has a weak negative correlation with **Own Price**. *(Price-sensitive customers?)*



See Appendix A-1.2

Time Series Plots

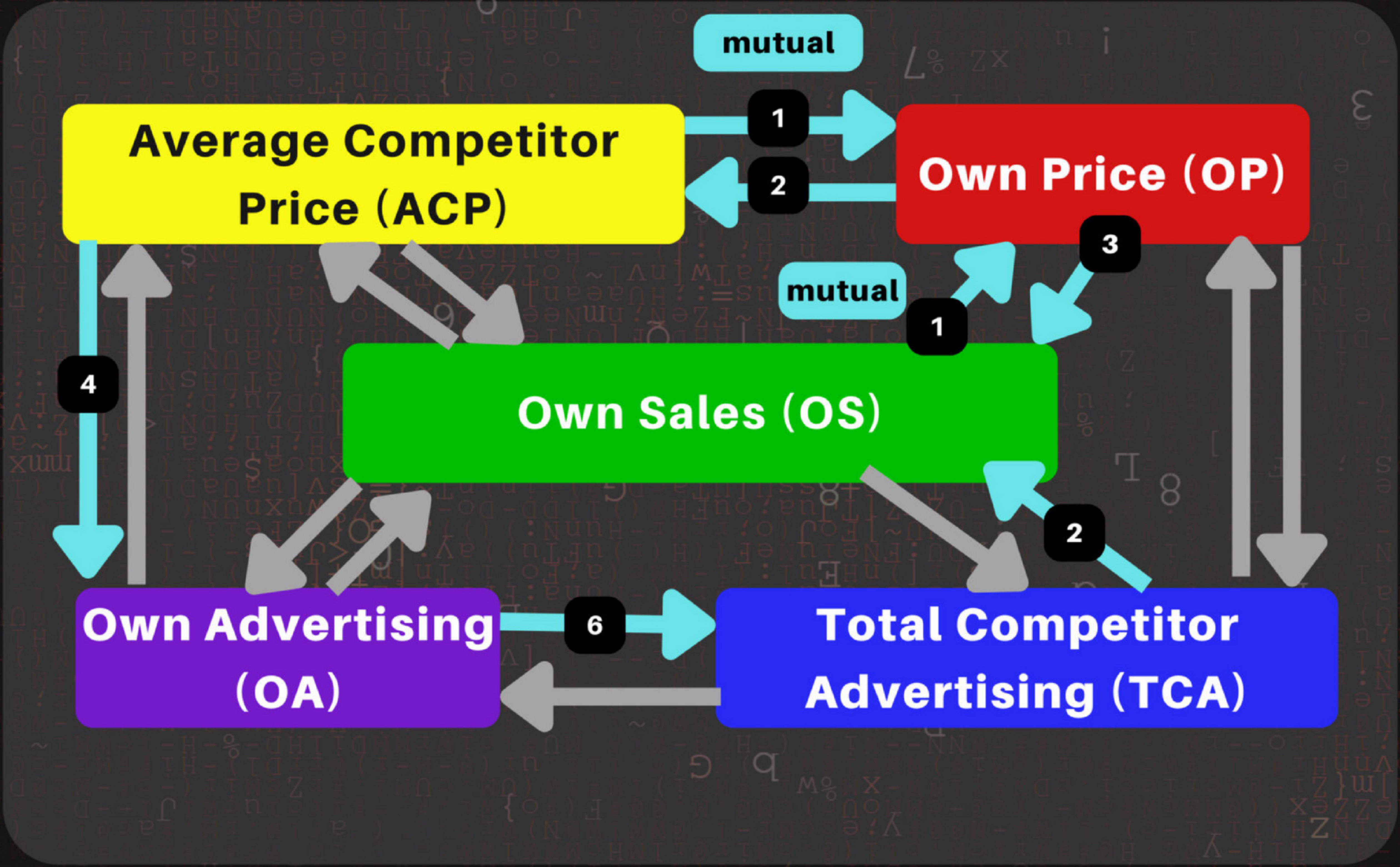
/Own Sales, Price & Advertising + TCA & ACP



See Appendix A-1.3

Granger Causalities

/which variable is temporally causing another? up to 13 lags



Legend & Definitions:

granger-causation

Variable x is granger-causing *variable y*, if knowing the past of *variable x* improves our forecast of *variable y* based on the past of *variable y*.

is granger-causing

is NOT granger causing

at how many lags? 2

See Appendix A-2

Stationary or Evolving?

/ADF-PP-KPSS results + 4-scenarios

Stationary Variables

	Own Advertising	Own Price	Own Sales	TCA
ADF*	1 2 3	1 2 3	1 2 3	1 2 3
PP	1 2 3	1 2 3	1 2 3	1 2 3
KPSS	-	-	-	-
Conclusion	0 mean-stationary	mean-stationary	mean-stationary	mean-stationary

Evolving Variable

	ACP
ADF*	1 2 3
PP	1 2 3
KPSS	1 2 3
Conclusion	evolving**

Legend: significant insignificant 1-3: Type 1-3 in the respective test

Own Advertising	< Stationary >	Own Sales	**Business as Usual:** effects of Advertising on Sales are temporary.
TCA	< Stationary >	Own Sales	**Business as Usual:** effects of TCA on Sales are temporary.
Own Price	< Stationary >	Own Sales	**Business as Usual:** effects of Price on Sales are temporary.
Evolving > ACP	Stationary >	Own Sales	**Escalation:** continued ACP changes have no permanent effect on Sales.

See Appendix A-3

* we ran ADF at 4-lags ** to be safe we named ACP as evolving & include it as differences, not levels

Were there immediate marketing effects or linear trend?

/VARx Model Specifications & Results

VARx Model Specifications

Lags:	Based on BIC criterion, we ran our model with **1 lag**
Endogenous variables:	Own **Sales**, **Advertising** & **Price** + **TCA** + **ACP**
Exogenous variables & Trend	**Quarters 2-4** (excl. Q1*) & Linear Trend

VARx Model Significant Results

- **Own Sales** had a **significant immediate marketing** effect during **Quarters 2, 3 & 4.**
- **Own Price** had a **significant linear trend.**
- **Own Advertising** had a **significant immediate marketing** effect during **Quarter 2.**
- **TCA** had a **significant immediate marketing** effect during **Quarters 2 & 4.**

- **ACP** had no significant immediate marketing effects, nor a linear trend.

See Appendix A-4

* by excluding quarter 1 we avoid the 'dummy variable trap'

Can we find significant dynamic impact of the different variables on each other?

/Generating IRFs*

With *stationary variables* when we generate the IRF, we are most interested in the cumulative effects, as immediate effects go back to the mean. This is the case with all of our stationary variables.

Own Sales

- + Sales has a <u>positive and significant</u> dynamic impact on **itself**
- + Advertising has a <u>positive and significant</u> dynamic impact on **Sales**
- Price has an insignificant on **Sales**
- − TCA has a <u>negative and significant</u> dynamic impact on **Sales**
- ACP has an insignificant dynamic impact on **Sales**

Own Price

- − Sales has a <u>negative and significant</u> dynamic impact on **Price**
- − Advertising has a <u>negative and significant</u> dynamic impact on **Price**
- + Price has a <u>positive and significant</u> dynamic impact on **itself**
- + TCA has a <u>positive and significant</u> dynamic impact on **Price**
- ACP has an insignificant dynamic impact on **Price**

*Please refer to the IRF graphs in Appendix A, we only include our interpretation here.

See Appendix A-5

(Continued)

/Generating IRFs*

Own Advertising

- **+** Sales has an insignificant dynamic impact on Advertising
- **+** Advertising has a positive and significant dynamic impact on itself
- **+** Price has a positive and significant on Advertising
- TCA has an insignificant dynamic impact on Advertising
- **+** ACP has a positive and significant dynamic impact on Advertising

TCA

- **−** Sales has an negative and significant dynamic impact on TCA
- Advertising has an insignificant dynamic impact on TCA
- Price has an insignificant dynamic impact on TCA
- **+** TCA has a positive and significant dynamic impact on itself
- ACP has an insignificant dynamic impact on TCA

ACP

- **+** Sales has a positive and significant dynamic impact on ACP
- Advertising has an insignificant dynamic impact on ACP
- **−** Price has a negative and significant on ACP
- TCA has an insignificant dynamic impact on ACP
- **+** ACP has a positive and significant on itself

*Please refer to the IRF graphs in Appendix A, we only include our interpretation here.

See Appendix A-5

What is the importance of each driver's past in explaining observed variance in Own Sales, Price and Advertising?

/Generating FEVD based on 13-week period

Observed variance in:

Own Sales is explained by (%), ranked by importance:

= 1. 94.4% by itself | 2. 3.2% by TCA | 3. 2% by OA | 0.3% by OP | 0.1% by ACP

Own Advertising is explained by (%), ranked by importance:

= 1. 96.2% by itself | 2. 2.2% by OP | 3. 0.6% by ACP | 0.5% by Sales | 0.5% by TCA

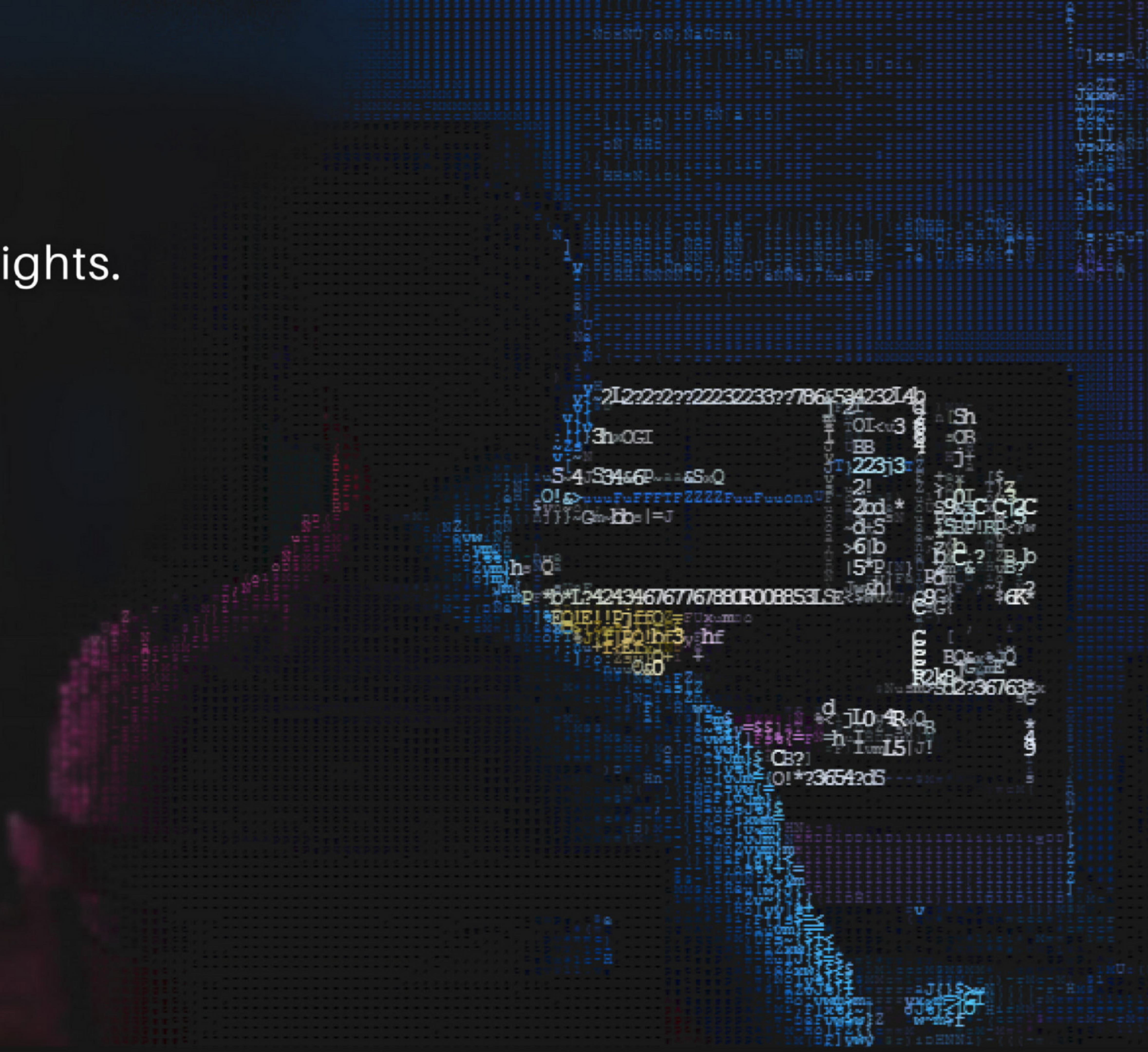
Own Price is explained by (%), ranked by importance:

= 1. 83.7% by itself | 2. 7.4% by Sales | 3. 5.2% by itself | 3.5% by TCA | 0.2% by ACP

See Appendix A-6

APPENDIX A

The raw data behind the insights.



APPENDIX A-1.1

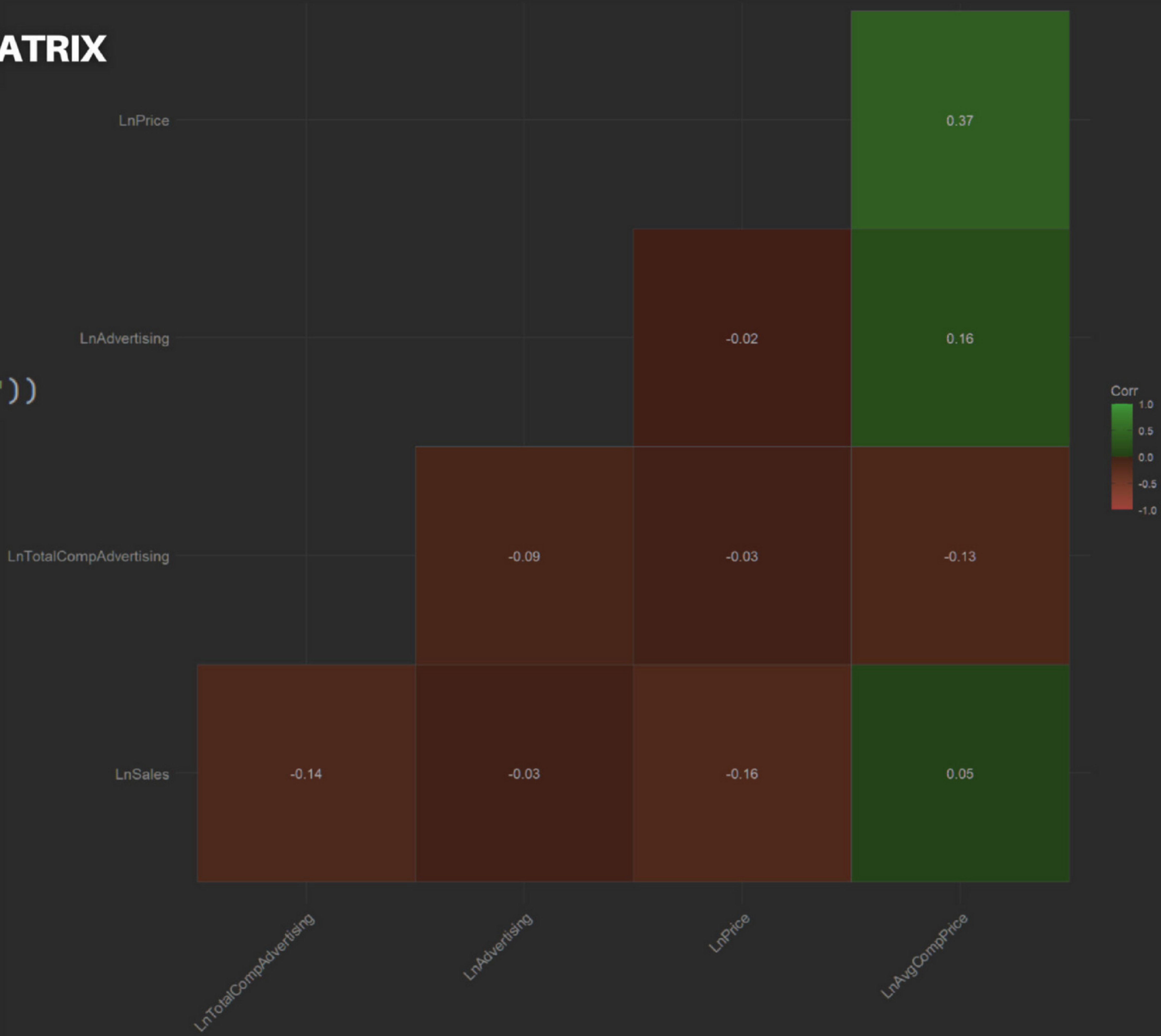
DESCRIPTIVE STATISTICS: SUMMARY OF VARIABLES

```
> summary(redstar.df)
    Student              Brand            Week            LnSales        LnAdvertising        LnPrice         LnTotalCompAdvertising
 Min.   :4513096   Min.   :105   Min.   :  1.0   Min.   :10.90   Min.   : 0.000   Min.   :-6.357   Min.   : 0.00
 1st Qu.:4513096   1st Qu.:105   1st Qu.: 52.5   1st Qu.:11.76   1st Qu.: 0.000   1st Qu.:-6.161   1st Qu.:12.45
 Median :4513096   Median :105   Median :104.0   Median :12.00   Median : 0.000   Median :-6.095   Median :14.46
 Mean   :4513096   Mean   :105   Mean   :104.0   Mean   :11.96   Mean   : 3.244   Mean   :-6.109   Mean   :13.08
 3rd Qu.:4513096   3rd Qu.:105   3rd Qu.:155.5   3rd Qu.:12.21   3rd Qu.: 9.072   3rd Qu.:-6.047   3rd Qu.:14.99
 Max.   :4513096   Max.   :105   Max.   :207.0   Max.   :12.86   Max.   :14.312   Max.   :-5.910   Max.   :16.53

 LnAvgCompPrice        Qrtr1             Qrtr2             Qrtr3             Qrtr4
 Min.   :-5.434   Min.   :0.0000   Min.   :0.0000   Min.   :0.0000   Min.   :0.0000
 1st Qu.:-5.350   1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.0000
 Median :-5.260   Median :0.0000   Median :0.0000   Median :0.0000   Median :0.0000
 Mean   :-5.287   Mean   :0.2464   Mean   :0.2512   Mean   :0.2512   Mean   :0.2512
 3rd Qu.:-5.231   3rd Qu.:0.0000   3rd Qu.:0.5000   3rd Qu.:0.5000   3rd Qu.:0.5000
 Max.   :-5.179   Max.   :1.0000   Max.   :1.0000   Max.   :1.0000   Max.   :1.0000
```

APPENDIX A-1.2

DESCRIPTIVE STATISTICS: CORRELATION MATRIX

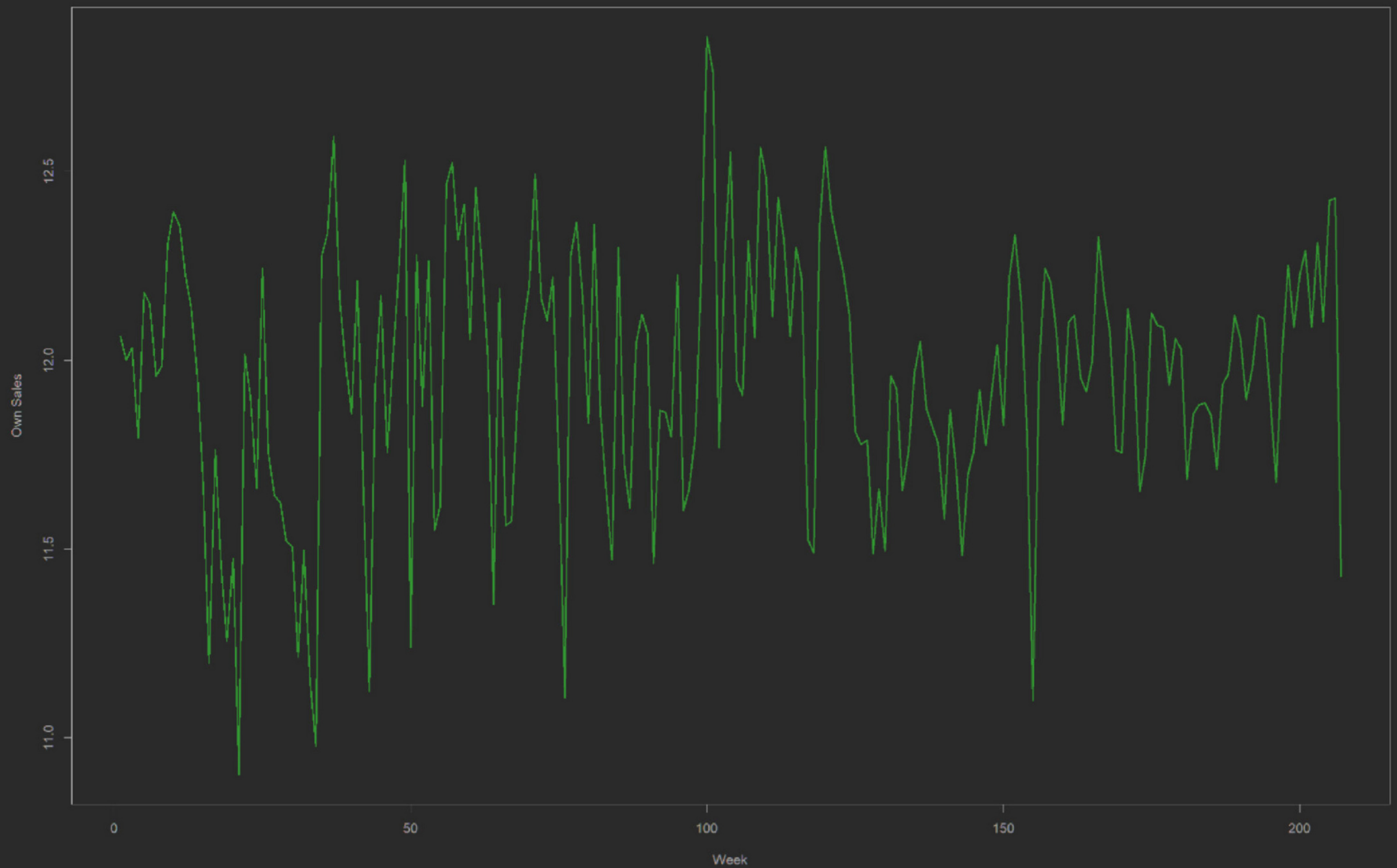
```
## Correlation Matrix ----
rs.cor <- cor(redstar.df[,c(4,5,6,7,8)])
ggcorrplot(rs.cor,
           hc.order = TRUE,
           type = "lower",
           lab = TRUE,
           colors = c("red", "white", "green"))
```



APPENDIX A-1.3.1

TIME SERIES PLOTS: OWN SALES OVER TIME

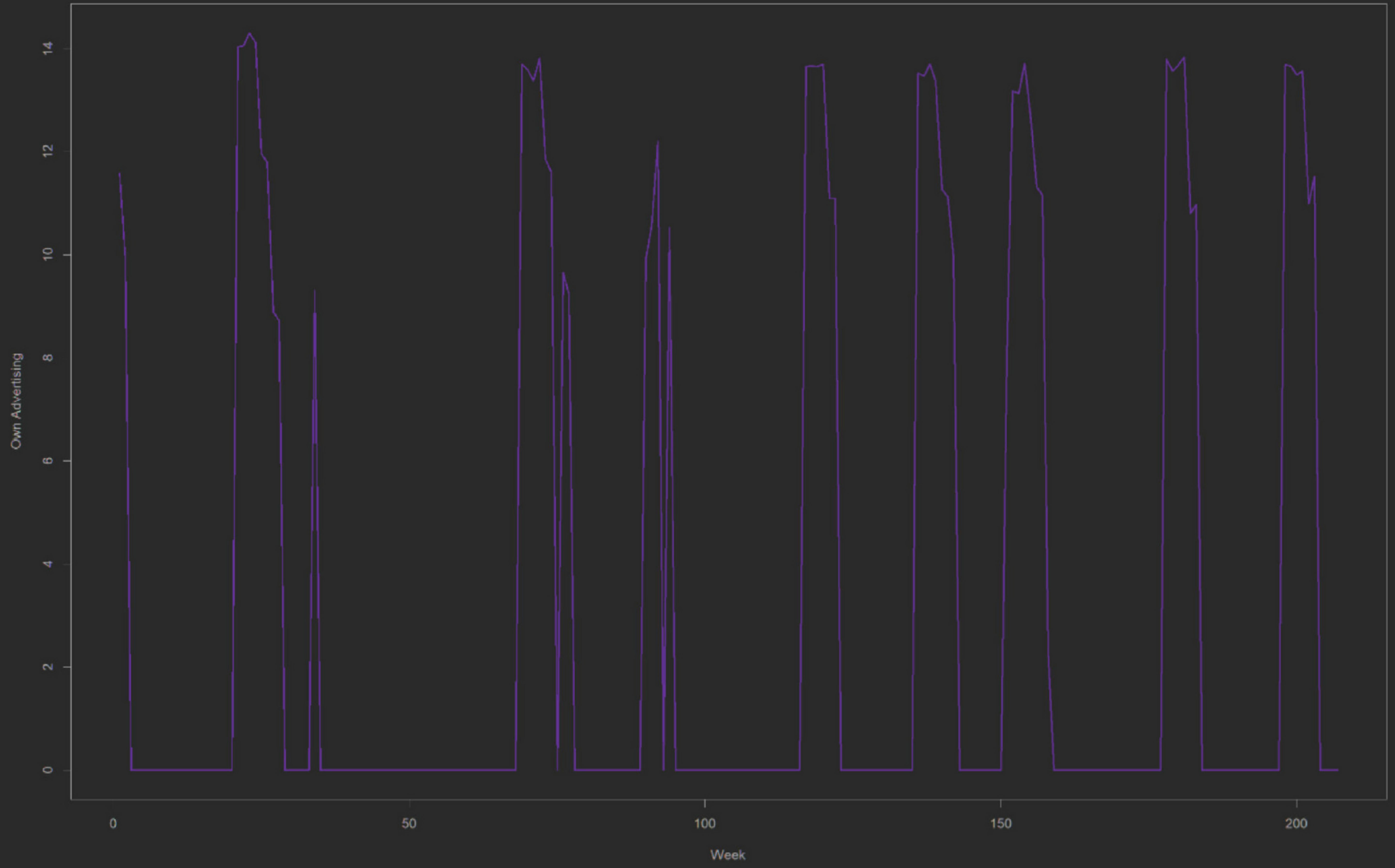
```
plot(redstar.df[,c(3)],redstar.df[,c(4)], type="l", col="green", lwd=2, xlab="Week", ylab="Own Sales", main="Own Sales over time")
```



APPENDIX A-1.3.2

TIME SERIES PLOTS: OWN ADVERTISING OVER TIME

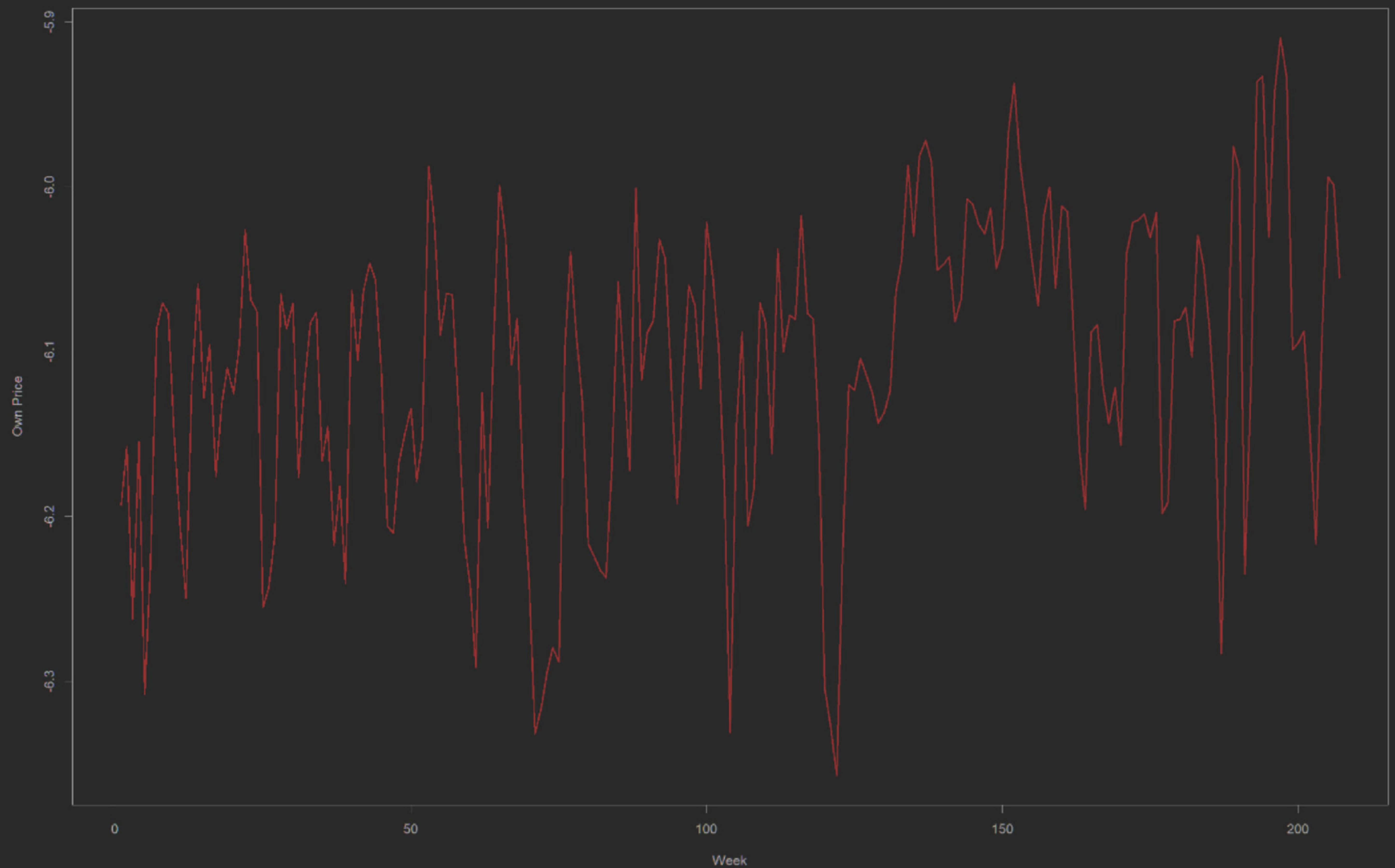
```
plot(redstar.df[,c(3)],redstar.df[,c(5)], type="l", col="purple", lwd=2, xlab="Week", ylab="Own Advertising", main="Own Advertising over time")
```



APPENDIX A-1.3.3

TIME SERIES PLOTS: OWN PRICE OVER TIME

```
plot(redstar.df[,c(3)],redstar.df[,c(6)], type="l", col="red", lwd=2, xlab="Week", ylab="Own Price", main="Own Price over time")
```



APPENDIX A-1.3.4

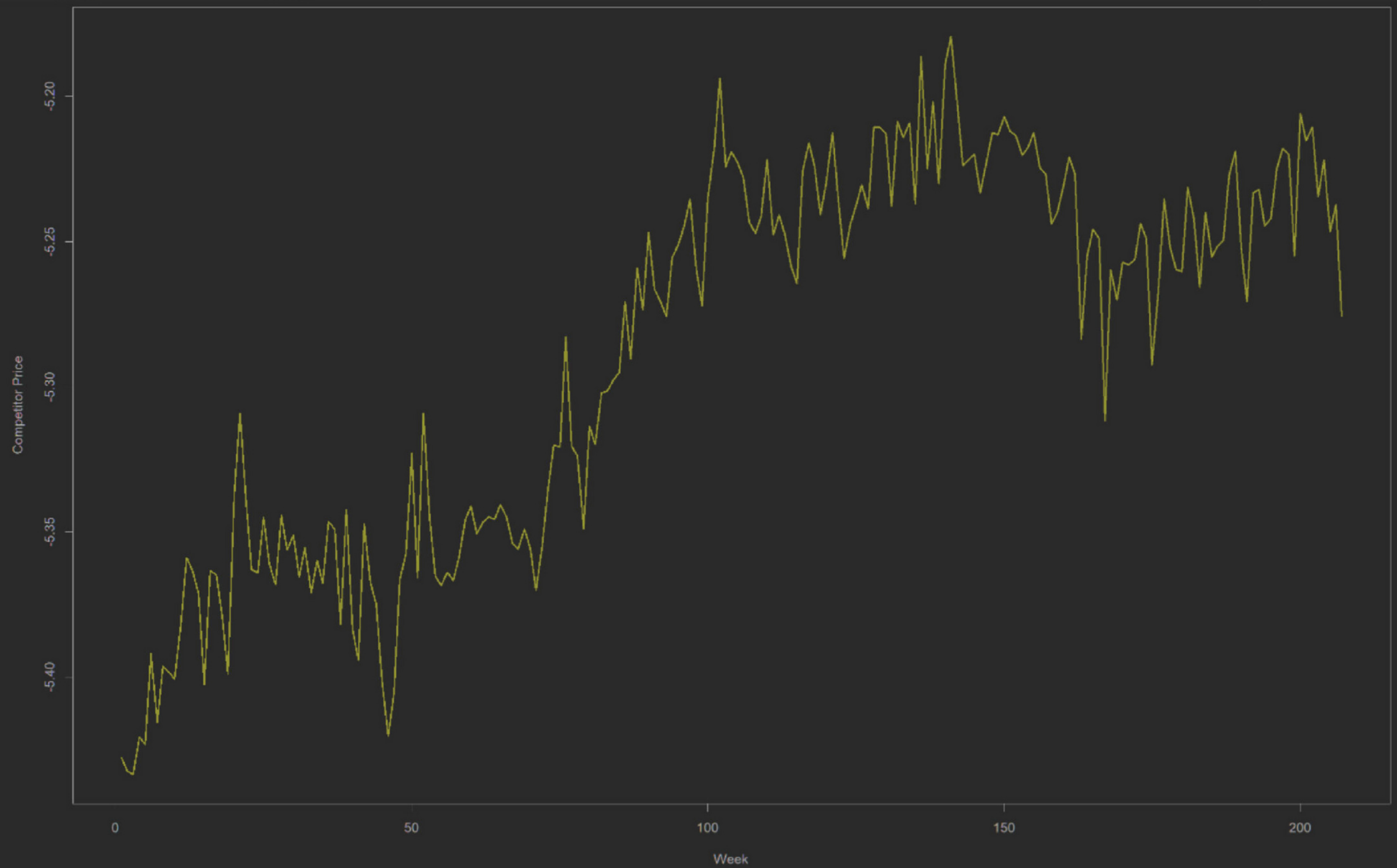
TIME SERIES PLOTS: TOTAL COMPETITOR ADVERTISING OVER TIME

```
plot(redstar.df[,c(3)],redstar.df[,c(7)], type="l", col="blue", lwd=2, xlab="Week", ylab="Competitor Advertising", main="Competitor Price over time")
```

APPENDIX A-1.3.5

TIME SERIES PLOTS: TOTAL COMPETITOR ADVERTISING OVER TIME

```
plot(redstar.df[,c(3)],redstar.df[,c(8)], type="l", col="orange", lwd=2, xlab="Week", ylab="Competitor Price", main="Competitor Price over time")
```



APPENDIX A-2.1

GRANGER CAUSALITIES: SELF-CODED GCT FUNCTION (line 66-93)

```
#Functionality: (1) It checks the relationship both ways
# (2) You can do as many lags as you wish
#Outputs: (1) neat df with all your pvalues with corresponding lags
# (2) An output in the console telling you at which lag the p-value is lowest
#How to use (example): gct_autolag(redstar.df$LnAdvertising,redstar.df$LnTotalCompAdvertising,13)
gct_autolag <- function(v1,v2,nlag){
  #outputdf.name takes the v1 v2 names, which include the input df name and $, removes that and leaves only the var names + at the end GCT.DF (note: paste function combines them and
  outputdf.name <- as.character(paste(str_extract(deparse(substitute(v1)), pattern: '\\b\\w+$'),str_extract(deparse(substitute(v2)), pattern: '\\b\\w+$'),'GCT.DF',sep = "_"))

  outputdf <- data.frame()
  for (lag in 1:nlag) { #loop runs granger in both directions up to and including specified nlag
    pval.v1v2 <- grangertest(v1~v2, order = lag)[2,4] #this saves ONLY the p-value result of GCT = [2,4]
    pval.v2v1 <- grangertest(v2~v1, order = lag)[2,4] #does the same for other direction
    outputdf[lag, 1] <- lag #assigns nth lag to a columns so we know which pval is for which lag
    outputdf[lag,2] <- pval.v1v2 #save the pvalues of first relationship into c2
    outputdf[lag,3] <- pval.v2v1 #save the pvalues of vice versa relationship into c3
  }
  colnames(outputdf) <- c("n.lag", "pval.v1v2", "pval.v2v1") #names the columns

  assign(outputdf.name,outputdf,envir = .GlobalEnv) #save the results in a df in the global environment incase we need the data

  #Also print the most meaningful part right away, so we don't have to lose our eyesight looking at p-values (+reduce human error)
  cat('Lowest p-values (per relationship) /w Lag:', '\n',
      str_extract(deparse(substitute(v1)), pattern: '\\b\\w+$'),'vs.',str_extract(deparse(substitute(v2)), pattern: '\\b\\w+$'),'- p-value:',min(outputdf$pval.v1v2), 'at Lag:', outputdf[out
      str_extract(deparse(substitute(v2)), pattern: '\\b\\w+$'),'vs.',str_extract(deparse(substitute(v1)), pattern: '\\b\\w+$'),'- p-value:',min(outputdf$pval.v2v1), 'at Lag:', outputdf[out
  ) ^gct_autolag
}
```

APPENDIX A-2.2

GRANGER CAUSALITIES: SIGNIFICANT GRANGER CAUSALITIES

```
> gct_autolag(redstar.df$LnTotalCompAdvertising,redstar.df$LnSales,13)
Lowest p-values (per relationship) /w Lag:
 LnTotalCompAdvertising vs. LnSales - p-value: 0.269818 at Lag: 1
 LnSales vs. LnTotalCompAdvertising - p-value: 0.01408014 at Lag: 2
```

```
> gct_autolag(redstar.df$LnAdvertising,redstar.df$LnTotalCompAdvertising,13)
Lowest p-values (per relationship) /w Lag:
 LnAdvertising vs. LnTotalCompAdvertising - p-value: 0.2196226 at Lag: 13
 LnTotalCompAdvertising vs. LnAdvertising - p-value: 0.004038217 at Lag: 6
```

```
> gct_autolag(redstar.df$LnAdvertising,redstar.df$LnAvgCompPrice,13)
Lowest p-values (per relationship) /w Lag:
 LnAdvertising vs. LnAvgCompPrice - p-value: 0.02128973 at Lag: 4
 LnAvgCompPrice vs. LnAdvertising - p-value: 0.1476304 at Lag: 2
```

```
> gct_autolag(redstar.df$LnPrice,redstar.df$LnAvgCompPrice,13)
Lowest p-values (per relationship) /w Lag:
 LnPrice vs. LnAvgCompPrice - p-value: 0.005180894 at Lag: 1
 LnAvgCompPrice vs. LnPrice - p-value: 0.03768227 at Lag: 2
```

```
> gct_autolag(redstar.df$LnPrice,redstar.df$LnSales,13)
Lowest p-values (per relationship) /w Lag:
 LnPrice vs. LnSales - p-value: 0.03731658 at Lag: 1
 LnSales vs. LnPrice - p-value: 0.0142497 at Lag: 3
```

APPENDIX A-2.2_alt

GRANGER CAUSALITIES: SIGNIFICANT GRANGER CAUSALITIES

```r
#### Total Comp. Advertising vs. Own Sales ----
gct_autolag(redstar.df$LnTotalCompAdvertising,redstar.df$LnSales, nlag: 13)
#Results: lowest pvalue for v1~v2 is 0.27 (at 1 lag), lowest pvalue for v2~v1 is 0.014 (at 2 lags); for up to 13 lags.
#Interpretation: Total Comp. Advertising is Granger-causing Own Sales.

#### Own Advertising vs Total Comp. Advertising ----
gct_autolag(redstar.df$LnAdvertising,redstar.df$LnTotalCompAdvertising, nlag: 13)
#Results: lowest pvalue for v1~v2 is 0.22 (at 13 lags), lowest pvalue for v2~v1 is 0.004 (at 6 lags); for up to 13 lags
#Interpretation: Own Advertising is Granger-causing Total Comp. Advertising.

#### Own Advertising vs Avg. Comp. Price ----
gct_autolag(redstar.df$LnAdvertising,redstar.df$LnAvgCompPrice, nlag: 13)
#Results: lowest pvalue for v1~v2 is 0.02 (at 4 lags), lowest pvalue for v2~v1 is 0.14 (at 2 lags); for up to 13 lags.
#Interpretation: Avg. Comp. Price is Granger-causing Own Advertising.

#### Own Price vs Avg. Comp. Price ----
gct_autolag(redstar.df$LnPrice,redstar.df$LnAvgCompPrice, nlag: 13)
#Results: lowest pvalue for v1~v2 is 0.005 (at 1 lag), lowest pvalue for v2~v1 is 0.038 (at 2 lags); for up to 13 lags.
#Interpretation: There is a significant mutual Granger causality, this is logical since
# usually firms in one industry determine their price externally.

#### Own Price vs Own Sales ----
gct_autolag(redstar.df$LnPrice,redstar.df$LnSales, nlag: 13)
#Results: lowest pvalue for v1~v2 is 0.037 (at 1 lag), lowest pvalue for v2~v1 is 0.014 (at 3 lags); for up to 13 lags.
#Interpretation: There is a significant mutual Granger causality, this is again logical because price-sales
# are in a similar relationship as supply-demand. Both influence each other.
```

APPENDIX A-2.3

GRANGER CAUSALITIES: INSIGNIFICANT GRANGER CAUSALITIES

```
> gct_autolag(redstar.df$LnAdvertising,redstar.df$LnSales,13)
Lowest p-values (per relationship) /w Lag:
 LnAdvertising vs. LnSales - p-value: 0.4324269 at Lag: 3
 LnSales vs. LnAdvertising - p-value: 0.1878825 at Lag: 2
```

```
> gct_autolag(redstar.df$LnPrice,redstar.df$LnTotalCompAdvertising,13)
Lowest p-values (per relationship) /w Lag:
 LnPrice vs. LnTotalCompAdvertising - p-value: 0.07323569 at Lag: 1
 LnTotalCompAdvertising vs. LnPrice - p-value: 0.4991668 at Lag: 3
```

```
> gct_autolag(redstar.df$LnAvgCompPrice,redstar.df$LnSales,13)
Lowest p-values (per relationship) /w Lag:
 LnAvgCompPrice vs. LnSales - p-value: 0.08689545 at Lag: 1
 LnSales vs. LnAvgCompPrice - p-value: 0.414173 at Lag: 1
```

APPENDIX A-2.3_alt

GRANGER CAUSALITIES: INSIGNIFICANT GRANGER CAUSALITIES

```
#### Own Advertising vs Own Sales ----
gct_autolag(redstar.df$LnAdvertising,redstar.df$LnSales, nlag: 13)
#Results: lowest pvalue for v1~v2 is 0.43 (at 3 lags), lowest pvalue for v2~v1 is 0.19 (at 2 lags); for up to 13 lags.
#Interpretation: There is no significant Granger causality present for up to 13 lags.

#### Own Price vs Total Comp. Advertising ----
gct_autolag(redstar.df$LnPrice,redstar.df$LnTotalCompAdvertising, nlag: 13)
#Results: lowest pvalue for v1~v2 is 0.07 (at 1 lag), lowest pvalue for v2~v1 is 0.499 (at 3 lags); for up to 13 lags.
#Interpretation: There is no significant Granger causality present for up to 13 lags.

#### Avg. Comp. Price vs. Own Sales ----
gct_autolag(redstar.df$LnAvgCompPrice,redstar.df$LnSales, nlag: 13)
#Results: lowest pvalue for v1~v2 is 0.087 (at 1 lag), lowest pvalue for v2~v1 is 0.414 (at 1 lag); for up to 13 lags.
#Interpretation: There is no significant Granger causality present for up to 13 lags.
```

APPENDIX A-3.1

STATIONARY OR EVOLVING - ADF, PP, KPSS: OWN SALES

```
adf.test(redstar.df$LnSales, nlag = 4, output = TRUE)
#Result: Type 1 (cannot reject), Type 2 and 3 (can reject!) = likely mean-stationary (4 lags)
pp.test(redstar.df$LnSales, output = TRUE)
#Result: Type 1 (cannot reject), Type 2 and 3 (can reject!) = probably mean-stationary (4 lags)
# CONCLUSION: ADF and PP sufficiently showcase we can reject unit root.  Own Sales is mean-stationary.
```

```
> adf.test(redstar.df$LnSales, nlag = 4, output = TRUE)
Augmented Dickey-Fuller Test
alternative: stationary

Type 1: no drift no trend
     lag    ADF p.value
[1,]   0 -0.341   0.546
[2,]   1 -0.254   0.571
[3,]   2 -0.182   0.591
[4,]   3 -0.099   0.615
Type 2: with drift no trend
     lag   ADF p.value
[1,]   0 -9.35    0.01
[2,]   1 -7.85    0.01
[3,]   2 -6.09    0.01
[4,]   3 -5.39    0.01
Type 3: with drift and trend
     lag   ADF p.value
[1,]   0 -9.40    0.01
[2,]   1 -7.91    0.01
[3,]   2 -6.16    0.01
[4,]   3 -5.45    0.01
```

```
> pp.test(redstar.df$LnSales, output = TRUE)
Phillips-Perron Unit Root Test
alternative: stationary

Type 1: no drift no trend
 lag   Z_rho p.value
   4 -0.0871   0.671
-----
 Type 2: with drift no trend
 lag Z_rho p.value
   4  -129    0.01
-----
 Type 3: with drift and trend
 lag Z_rho p.value
   4  -130    0.01
```

APPENDIX A-3.2

STATIONARY OR EVOLVING - ADF, PP, KPSS: OWN PRICE

```
adf.test(redstar.df$LnPrice, nlag = 4, output = TRUE)
#Result: Type 1 (cannot reject), Type 2 and 3 (can reject!) = likely mean-stationary (4 lags)
pp.test(redstar.df$LnPrice, output = TRUE)
#Result: Type 1 (cannot reject), Type 2 and 3 (can reject!) = probably mean-stationary (4 lags)
# CONCLUSION: ADF and PP sufficiently showcase we can reject unit root. Own Price is mean-stationary.
```

```
> adf.test(redstar.df$LnPrice, nlag = 4, output = TRUE)
Augmented Dickey-Fuller Test
alternative: stationary

Type 1: no drift no trend
     lag    ADF p.value
[1,]   0 -0.216   0.582
[2,]   1 -0.190   0.589
[3,]   2 -0.322   0.551
[4,]   3 -0.240   0.575
Type 2: with drift no trend
     lag   ADF p.value
[1,]   0 -6.65    0.01
[2,]   1 -6.76    0.01
[3,]   2 -5.88    0.01
[4,]   3 -5.07    0.01
Type 3: with drift and trend
     lag   ADF p.value
[1,]   0 -7.23    0.01
[2,]   1 -7.53    0.01
[3,]   2 -6.57    0.01
[4,]   3 -5.84    0.01
```

```
> pp.test(redstar.df$LnPrice, output = TRUE)
Phillips-Perron Unit Root Test
alternative: stationary

Type 1: no drift no trend
 lag    Z_rho p.value
   4 -0.0309   0.684
-----
 Type 2: with drift no trend
 lag Z_rho p.value
   4 -72.5    0.01
-----
 Type 3: with drift and trend
 lag Z_rho p.value
   4   -84    0.01
```

APPENDIX A-3.3

STATIONARY OR EVOLVING - ADF, PP, KPSS: OWN ADVERTISING

```
adf.test(redstar.df$LnAdvertising, nlag = 4, output = TRUE)
#Result: Type 1, Type 2 and 3 (can reject!) = zero-mean, stationarity (4 lags)
pp.test(redstar.df$LnAdvertising, output = TRUE)
#Result: Type 1, Type 2 and 3 (can reject!) = zero-mean, stationarity (4 lags)
# CONCLUSION: Seems there is overwhelming evidence already from ADF and did PP just in case.
# Own Advertising has a zero mean (no drift, no trend)
```

```
> adf.test(redstar.df$LnAdvertising, nlag = 4, output = TRUE)
Augmented Dickey-Fuller Test
alternative: stationary

Type 1: no drift no trend
     lag   ADF p.value
[1,]   0 -4.71    0.01
[2,]   1 -4.32    0.01
[3,]   2 -4.50    0.01
[4,]   3 -4.76    0.01
Type 2: with drift no trend
     lag   ADF p.value
[1,]   0 -5.43    0.01
[2,]   1 -5.04    0.01
[3,]   2 -5.53    0.01
[4,]   3 -6.07    0.01
Type 3: with drift and trend
     lag   ADF p.value
[1,]   0 -5.48    0.01
[2,]   1 -5.11    0.01
[3,]   2 -5.55    0.01
[4,]   3 -6.10    0.01
```

```
> pp.test(redstar.df$LnAdvertising, output = TRUE)
Phillips-Perron Unit Root Test
alternative: stationary

Type 1: no drift no trend
 lag Z_rho p.value
   4 -44.1    0.01
-----
 Type 2: with drift no trend
 lag Z_rho p.value
   4 -60.6    0.01
-----
 Type 3: with drift and trend
 lag Z_rho p.value
   4 -61.6    0.01
```

APPENDIX A-3.4

STATIONARY OR EVOLVING - ADF, PP, KPSS: TOTAL COMP. ADVERTISING (TCA)

```
adf.test(redstar.df$LnTotalCompAdvertising, nlag = 4, output = TRUE)
#Result: Type 1 (can reject at 1 lag, cannot 3-4), Type 2 and 3 (can reject!) = likely mean-stationarity (4 lags)
pp.test(redstar.df$LnTotalCompAdvertising, output = TRUE)
#Result: Type 1 (cannot reject), Type 2 and 3 (can reject!) = very likely mean-stationarity (4 lags)
# CONCLUSION: ADF and PP sufficiently showcase we can reject unit root. Total Comp. Advertising is mean-stationary
```

```
> adf.test(redstar.df$LnTotalCompAdvertising, nlag = 4, output = TRUE)
Augmented Dickey-Fuller Test
alternative: stationary

Type 1: no drift no trend
     lag   ADF p.value
[1,]   0 -1.99  0.0467
[2,]   1 -1.53  0.1305
[3,]   2 -1.45  0.1613
[4,]   3 -1.37  0.1902
Type 2: with drift no trend
     lag   ADF p.value
[1,]   0 -7.15    0.01
[2,]   1 -5.38    0.01
[3,]   2 -5.51    0.01
[4,]   3 -5.97    0.01
Type 3: with drift and trend
     lag   ADF p.value
[1,]   0 -7.26    0.01
[2,]   1 -5.48    0.01
[3,]   2 -5.62    0.01
[4,]   3 -6.08    0.01
```

```
> pp.test(redstar.df$LnTotalCompAdvertising, output = TRUE)
Phillips-Perron Unit Root Test
alternative: stationary

Type 1: no drift no trend
 lag Z_rho p.value
   4  -4.3   0.214
-----
 Type 2: with drift no trend
 lag Z_rho p.value
   4 -94.2    0.01
-----
 Type 3: with drift and trend
 lag Z_rho p.value
   4 -96.2    0.01
```

APPENDIX A-3.5

STATIONARY OR EVOLVING - ADF, PP, KPSS: AVG. COMP. PRICE (ACP) + CREATING FIRST DIFF. ACP

```r
adf.test(redstar.df$LnAvgCompPrice, nlag = 4, output = TRUE)
#Result: Type 1 and Type 2 (cannot reject) and 3 (can reject only at lag 1, 3-4 cannot reject) = unclear
pp.test(redstar.df$LnAvgCompPrice, output = TRUE)
#Result: Type 1 and Type 2 (cannot reject) and 3 (can reject!) = it seems adding the trend made it significant, unclear
kpss.test(redstar.df$LnAvgCompPrice, output = TRUE)
#Result: Type 1 (cannot reject!), Type 2 & 3 (can reject) => unclear
# CONCLUSION: It's not very clear if it's stationary, therefore it is perhaps safer to choose a unit root.

# We will use the first difference for Avg.Comp.Price in the VAR model, others as levels
redstar.df$LnAvgCompPrice.diff <- c(0, diff(redstar.df$LnAvgCompPrice, lag = 1, differences=1))
#Note: We do not test for cointegration, as we have one evolving variable.
```

```
> adf.test(redstar.df$LnAvgCompPrice, nlag = 4, output = TRUE
Augmented Dickey-Fuller Test
alternative: stationary

Type 1: no drift no trend
     lag    ADF p.value
[1,]   0 -0.532   0.489
[2,]   1 -0.800   0.392
[3,]   2 -0.967   0.333
[4,]   3 -1.376   0.186
Type 2: with drift no trend
     lag   ADF p.value
[1,]   0 -2.83  0.0589
[2,]   1 -2.39  0.1756
[3,]   2 -2.31  0.2082
[4,]   3 -2.16  0.2663
Type 3: with drift and trend
     lag   ADF p.value
[1,]   0 -4.16   0.010
[2,]   1 -2.93   0.186
[3,]   2 -2.49   0.370
[4,]   3 -1.68   0.709
```

```
> pp.test(redstar.df$LnAvgCompPrice, output = TRUE
Phillips-Perron Unit Root Test
alternative: stationary

Type 1: no drift no trend
 lag    Z_rho p.value
   4 -0.0297   0.684
-----
 Type 2: with drift no trend
 lag Z_rho p.value
   4 -7.37    0.32
-----
 Type 3: with drift and trend
 lag Z_rho p.value
   4 -23.7  0.0293
```

```
> kpss.test(redstar.df$LnAvgCompPrice, ou
KPSS Unit Root Test
alternative: nonstationary

Type 1: no drift no trend
 lag  stat p.value
   3 0.147     0.1
-----
 Type 2: with drift no trend
 lag  stat p.value
   3 0.819    0.01
-----
 Type 1: with drift and trend
 lag  stat p.value
   3 0.475    0.01
```

APPENDIX A-4.1

VARX MODEL - SPECIFICATIONS: VARIABLES, LAG AND MODEL

```r
#Endogenous: LnSales, LnAdvertising, LnPrice, LnTotalCompAdvertising, LnAvgCompPrice.diff
rs.endogenous <- redstar.df[,c(4,5,6,7,13)]
#Exogenous: Qrtr2, Qrtr3, Qrtr4 (excl. Qrtr1 to avoid dummy-trap) - we interpret relative to the first quarter.
rs.exogenous <- redstar.df[,c(10,11,12)]

#Lag-length: 4 (based on business knowledge that after the end of 4 weeks effects are not there)
VARselect(rs.endogenous,lag.max = 4, type = "both", exogen = rs.exogenous)
#AIC and FPE say 3; HQ and BIC say 1
# CONCLUSION: BIC is more reliable, also previous MR shows 1 lag as optimal, so we go for 1 lag (p=1 in our VAR() model).

#VARx Model (we use type "both" as it includes both the intercept and linear trend):
rs.varx <- VAR(rs.endogenous, p=1, type = "both", exogen = rs.exogenous)
```

```
> VARselect(rs.endogenous,lag.max = 4, type = "both", exogen = rs.exogenous)
$selection
AIC(n)  HQ(n)  SC(n) FPE(n)
     3      1      1      3

$criteria
                   1             2             3             4
AIC(n) -1.077910e+01 -1.084407e+01 -1.086715e+01 -1.082213e+01
HQ(n)  -1.044896e+01 -1.034885e+01 -1.020686e+01 -9.996764e+00
SC(n)  -9.963042e+00 -9.619981e+00 -9.235032e+00 -8.781976e+00
FPE(n)  2.083860e-05  1.954633e-05  1.913586e-05  2.007883e-05
```

APPENDIX A-4.2.1

VARX MODEL - RESULTS: SUMMARY OUTPUTS - OWN SALES

```
summary(rs.varx, "LnSales")
#Sales had a significant immediate marketing effect during Quarter 2 and Quarter 3 (p<0.001) & Quarter 4 (p~0.008)
```

```
Estimation results for equation LnSales:
========================================
LnSales = LnSales.l1 + LnAdvertising.l1 + LnPrice.l1 + LnTotalCompAdvertising.l1 + LnAvgCompPrice.diff.l1 + const + trend + Qrtr2 +
  Qrtr3 + Qrtr4

                            Estimate Std. Error t value Pr(>|t|)
LnSales.l1                 0.2603406  0.0705963   3.688 0.000293 ***
LnAdvertising.l1           0.0059500  0.0040949   1.453 0.147810
LnPrice.l1                -0.2721145  0.2750184  -0.989 0.323667
LnTotalCompAdvertising.l1 -0.0136551  0.0063414  -2.153 0.032515 *
LnAvgCompPrice.diff.l1     0.4924151  1.0230604   0.481 0.630829
const                      7.4390616  1.7265931   4.309  2.6e-05 ***
trend                      0.0006653  0.0004047   1.644 0.101820
Qrtr2                     -0.2337615  0.0661878  -3.532 0.000515 ***
Qrtr3                     -0.2470847  0.0661530  -3.735 0.000246 ***
Qrtr4                     -0.1775513  0.0658641  -2.696 0.007634 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1


Residual standard error: 0.3065 on 196 degrees of freedom
Multiple R-Squared: 0.2386, Adjusted R-squared: 0.2036
F-statistic: 6.825 on 9 and 196 DF,  p-value: 1.614e-08
```

APPENDIX A-4.2.2

VARX MODEL - RESULTS: SUMMARY OUTPUTS - OWN PRICE

```
summary(rs.varx,"LnPrice")
#Price had no significant immediate marketing effect these 4 quarters and a significant linear trend.
Estimation results for equation LnPrice:
========================================
LnPrice = LnSales.l1 + LnAdvertising.l1 + LnPrice.l1 + LnTotalCompAdvertising.l1 + LnAvgCompPrice.diff.l1 + const + trend + Qrtr2 +
  Qrtr3 + Qrtr4

                           Estimate Std. Error t value Pr(>|t|)
LnSales.l1               -3.806e-02  1.497e-02  -2.542 0.011794 *
LnAdvertising.l1         -1.761e-03  8.684e-04  -2.028 0.043948 *
LnPrice.l1                5.215e-01  5.833e-02   8.941 2.88e-16 ***
LnTotalCompAdvertising.l1  2.418e-03  1.345e-03   1.798 0.073732 .
LnAvgCompPrice.diff.l1    1.679e-01  2.170e-01   0.774 0.439920
const                    -2.526e+00  3.662e-01  -6.898 7.09e-11 ***
trend                     2.971e-04  8.583e-05   3.462 0.000658 ***
Qrtr2                    -1.110e-02  1.404e-02  -0.791 0.430178
Qrtr3                     1.994e-03  1.403e-02   0.142 0.887146
Qrtr4                     1.504e-02  1.397e-02   1.077 0.282880
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1


Residual standard error: 0.065 on 196 degrees of freedom
Multiple R-Squared: 0.4895, Adjusted R-squared: 0.4661
F-statistic: 20.88 on 9 and 196 DF,  p-value: < 2.2e-16
```

APPENDIX A-4.2.3

VARX MODEL - RESULTS: SUMMARY OUTPUTS - OWN ADVERTISING

```
summary(rs.varx, "LnAdvertising")
#Advertising had a significant immediate marketing effect during Quarter 2 (p~0.003)
```

```
Estimation results for equation LnAdvertising:
==============================================
LnAdvertising = LnSales.l1 + LnAdvertising.l1 + LnPrice.l1 + LnTotalCompAdvertising.l1 + LnAvgCompPrice.diff.l1 + const + trend +
 Qrtr2 + Qrtr3 + Qrtr4

                           Estimate Std. Error t value Pr(>|t|)
LnSales.l1                 0.316647   0.805900   0.393  0.69481
LnAdvertising.l1           0.721429   0.046746  15.433  < 2e-16 ***
LnPrice.l1                 5.304798   3.139502   1.690  0.09268 .
LnTotalCompAdvertising.l1  0.030773   0.072391   0.425  0.67124
LnAvgCompPrice.diff.l1    17.076411  11.678856   1.462  0.14530
const                     28.025568  19.710110   1.422  0.15665
trend                      0.001064   0.004620   0.230  0.81813
Qrtr2                      2.262248   0.755574   2.994  0.00311 **
Qrtr3                      0.701434   0.755177   0.929  0.35412
Qrtr4                      0.657244   0.751879   0.874  0.38311
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1


Residual standard error: 3.499 on 196 degrees of freedom
Multiple R-Squared: 0.6038, Adjusted R-squared: 0.5856
F-statistic: 33.19 on 9 and 196 DF,  p-value: < 2.2e-16
```

APPENDIX A-4.2.4

VARX MODEL - RESULTS: SUMMARY OUTPUTS - TOTAL COMP. ADVERTISING (TCA)

```
summary(rs.varx, "LnTotalCompAdvertising")
#Total Competitor Advertising had a significant immediate marketing effect during Quarter 3 (p~0.046) and Quarter 4 (p<0.001)
```

```
Estimation results for equation LnTotalCompAdvertising:
=======================================================
LnTotalCompAdvertising = LnSales.l1 + LnAdvertising.l1 + LnPrice.l1 + LnTotalCompAdvertising.l1 + LnAvgCompPrice.diff.l1 + const +
 trend + Qrtr2 + Qrtr3 + Qrtr4

                          Estimate Std. Error t value Pr(>|t|)
LnSales.l1               -1.051118   0.693317  -1.516 0.131112
LnAdvertising.l1         -0.031795   0.040215  -0.791 0.430114
LnPrice.l1                2.004732   2.700917   0.742 0.458830
LnTotalCompAdvertising.l1  0.499912   0.062278   8.027 9.01e-14 ***
LnAvgCompPrice.diff.l1    0.191975  10.047333   0.019 0.984775
const                    32.768585  16.956629   1.932 0.054740 .
trend                    -0.002137   0.003975  -0.538 0.591437
Qrtr2                    -0.967056   0.650021  -1.488 0.138429
Qrtr3                    -1.302508   0.649679  -2.005 0.046355 *
Qrtr4                    -2.182809   0.646842  -3.375 0.000891 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1


Residual standard error: 3.01 on 196 degrees of freedom
Multiple R-Squared: 0.3557, Adjusted R-squared: 0.3262
F-statistic: 12.03 on 9 and 196 DF,  p-value: 4.82e-15
```

APPENDIX A-4.2.5

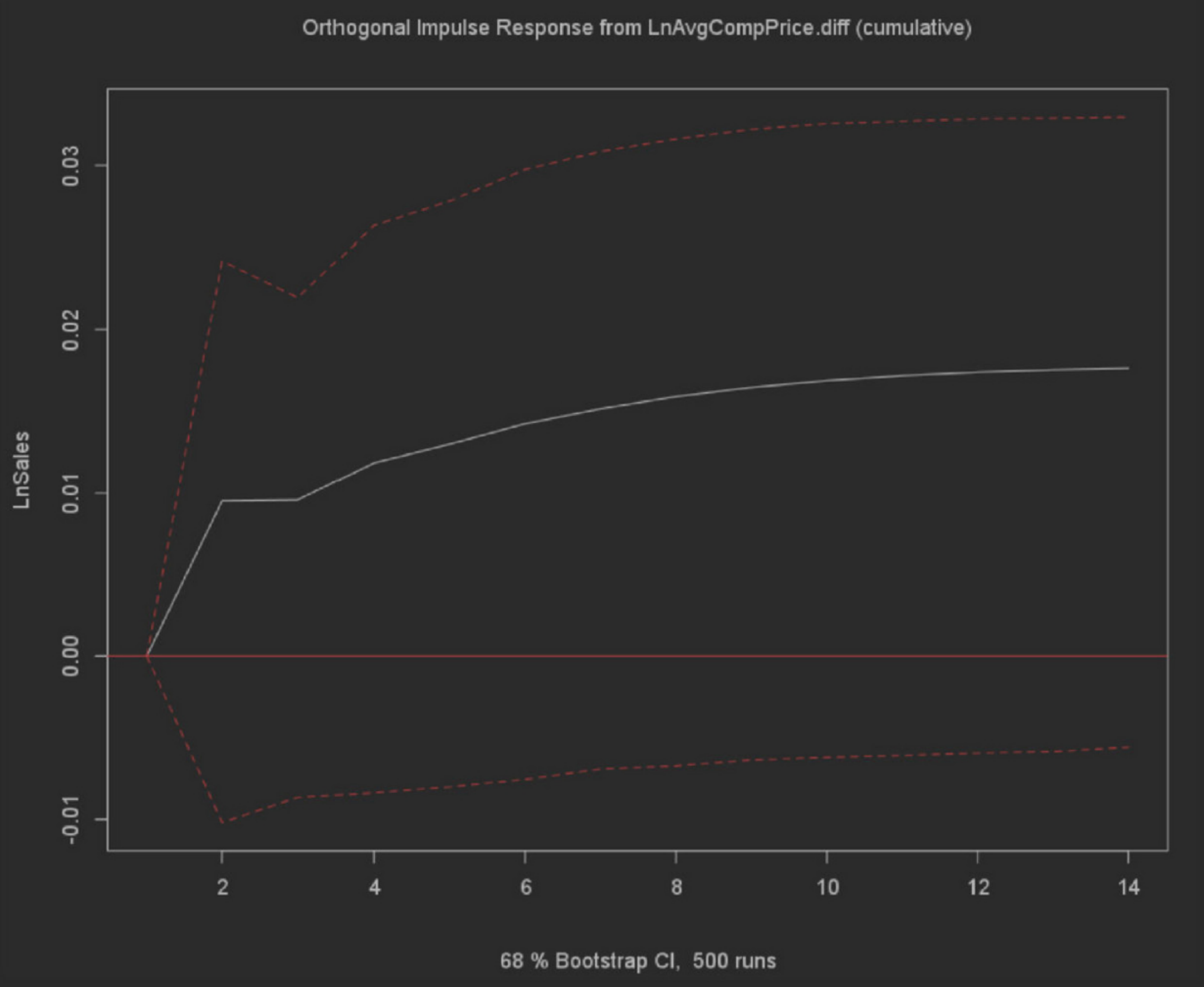
VARX MODEL - RESULTS: SUMMARY OUTPUTS - AVG. COMP. PRICE (ACP)

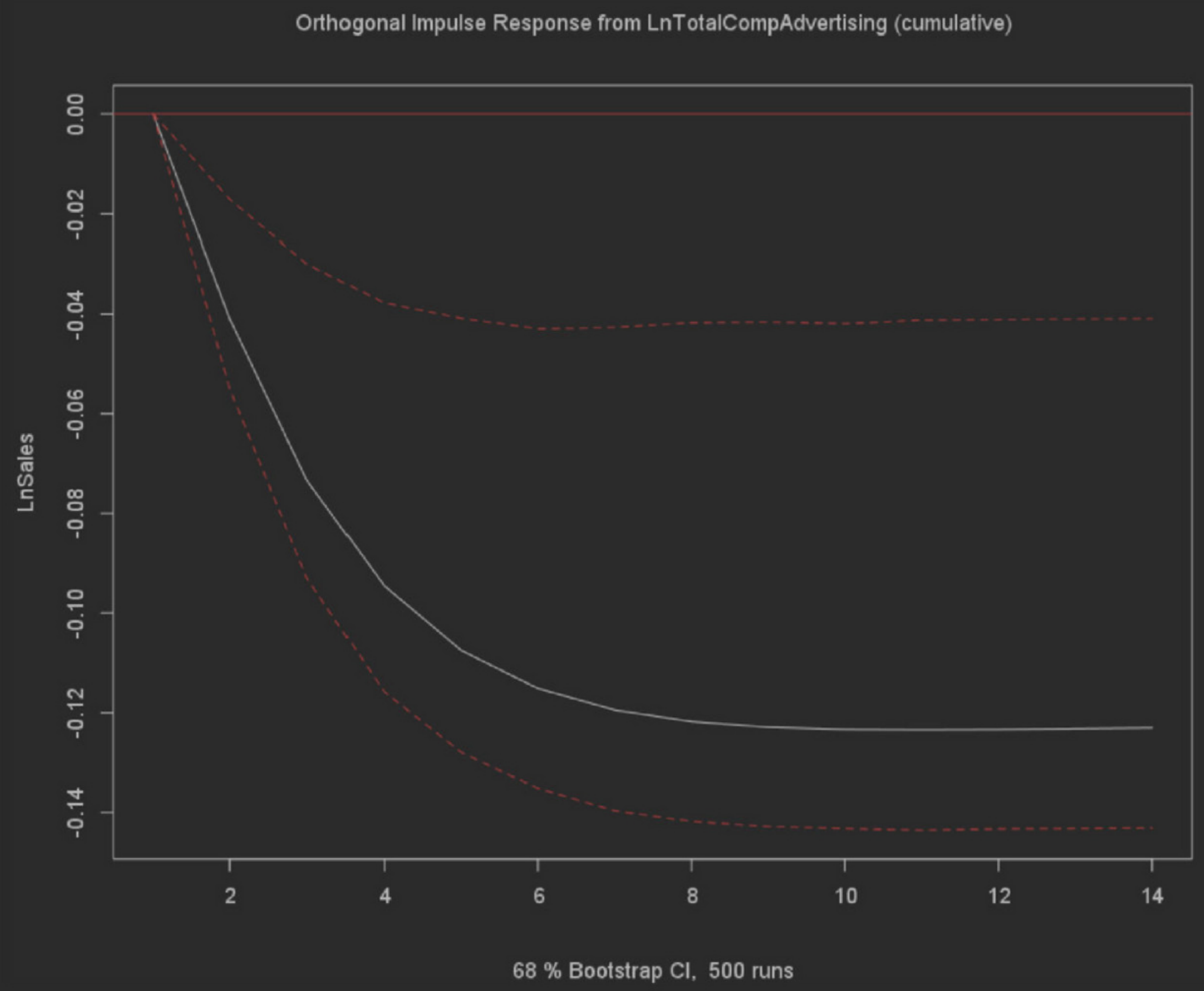
```
summary(rs.varx, "LnAvgCompPrice.diff")

#Average Competitor Price had no significant marketing effect these 4 quarters.
```

```
Estimation results for equation LnAvgCompPrice.diff:
====================================================
LnAvgCompPrice.diff = LnSales.l1 + LnAdvertising.l1 + LnPrice.l1 + LnTotalCompAdvertising.l1 + LnAvgCompPrice.diff.l1 + const +
 trend + Qrtr2 + Qrtr3 + Qrtr4

                            Estimate Std. Error t value Pr(>|t|)
LnSales.l1                 5.179e-03  4.554e-03   1.137  0.25683
LnAdvertising.l1          -5.647e-04  2.642e-04  -2.138  0.03378 *
LnPrice.l1                -4.362e-02  1.774e-02  -2.458  0.01482 *
LnTotalCompAdvertising.l1  3.781e-05  4.091e-04   0.092  0.92646
LnAvgCompPrice.diff.l1    -3.618e-01  6.600e-02  -5.483 1.28e-07 ***
const                     -3.300e-01  1.114e-01  -2.963  0.00343 **
trend                     -1.789e-06  2.611e-05  -0.069  0.94545
Qrtr2                      6.249e-03  4.270e-03   1.464  0.14493
Qrtr3                      5.133e-03  4.268e-03   1.203  0.23050
Qrtr4                      5.295e-03  4.249e-03   1.246  0.21417
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1


Residual standard error: 0.01977 on 196 degrees of freedom
Multiple R-Squared: 0.1794, Adjusted R-squared: 0.1417
F-statistic: 4.761 on 9 and 196 DF,  p-value: 9.627e-06
```

APPENDIX A-5.1.1

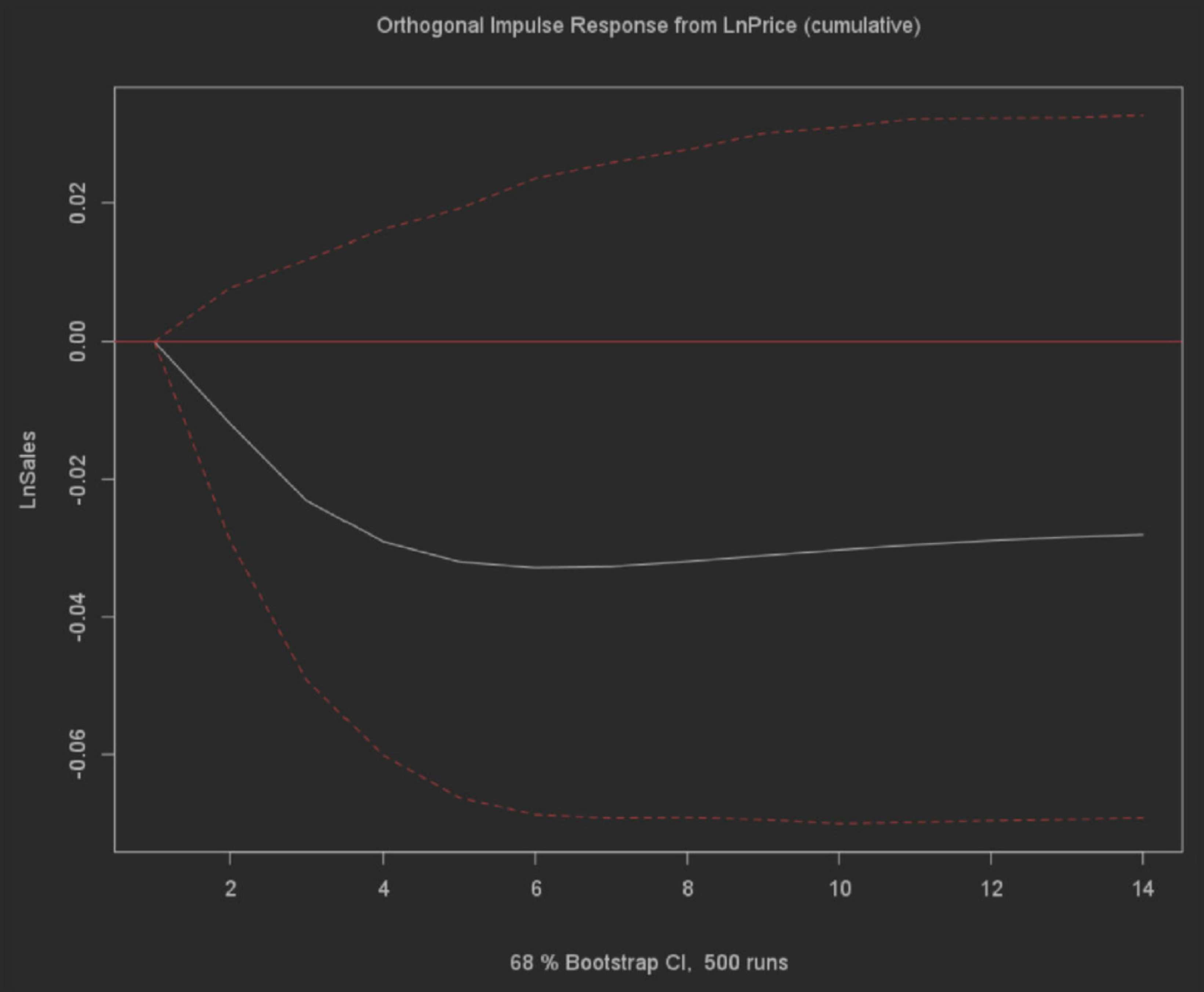
IRF PLOTS: OWN SALES CUMULATIVE

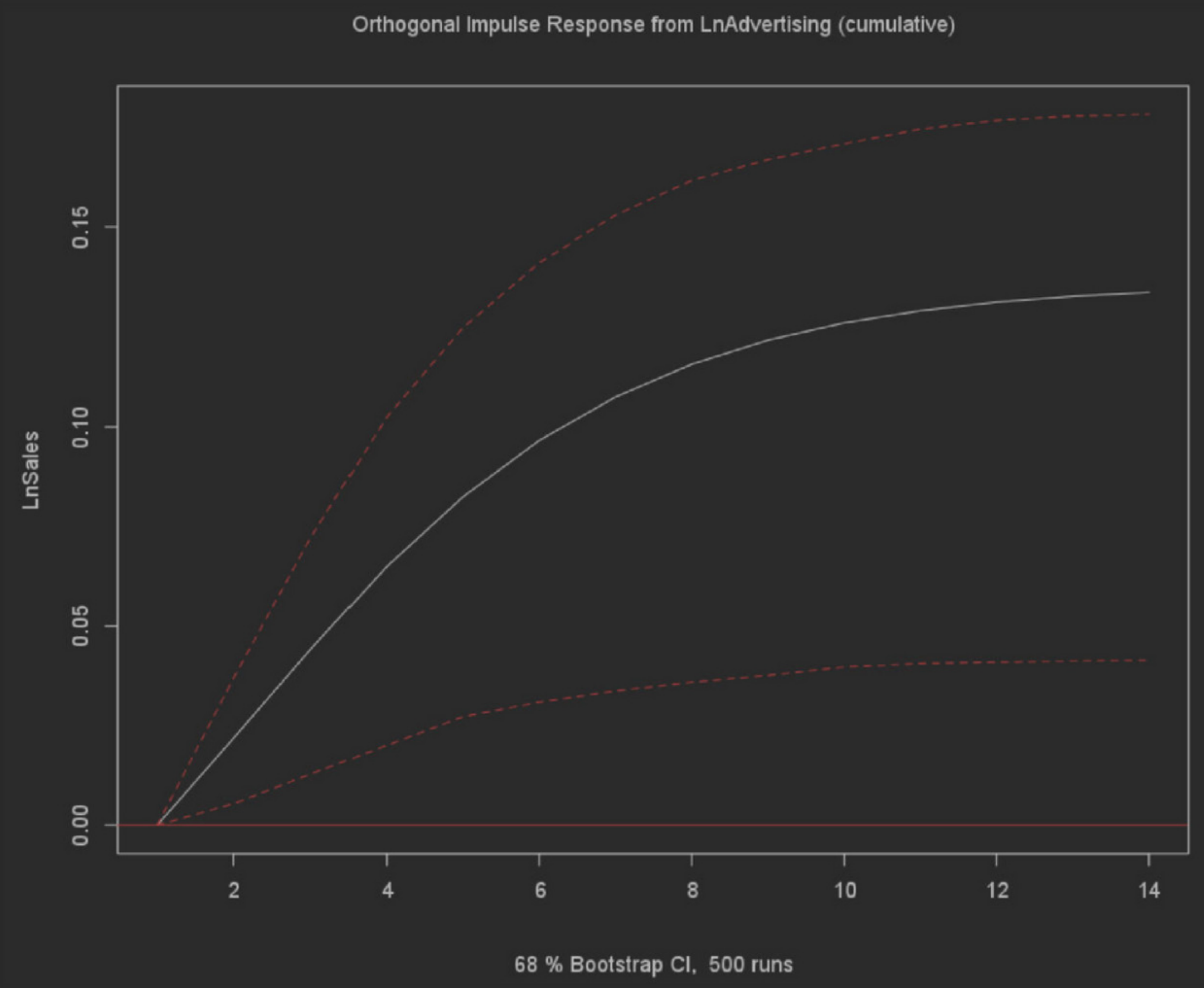




APPENDIX A-5.1.2

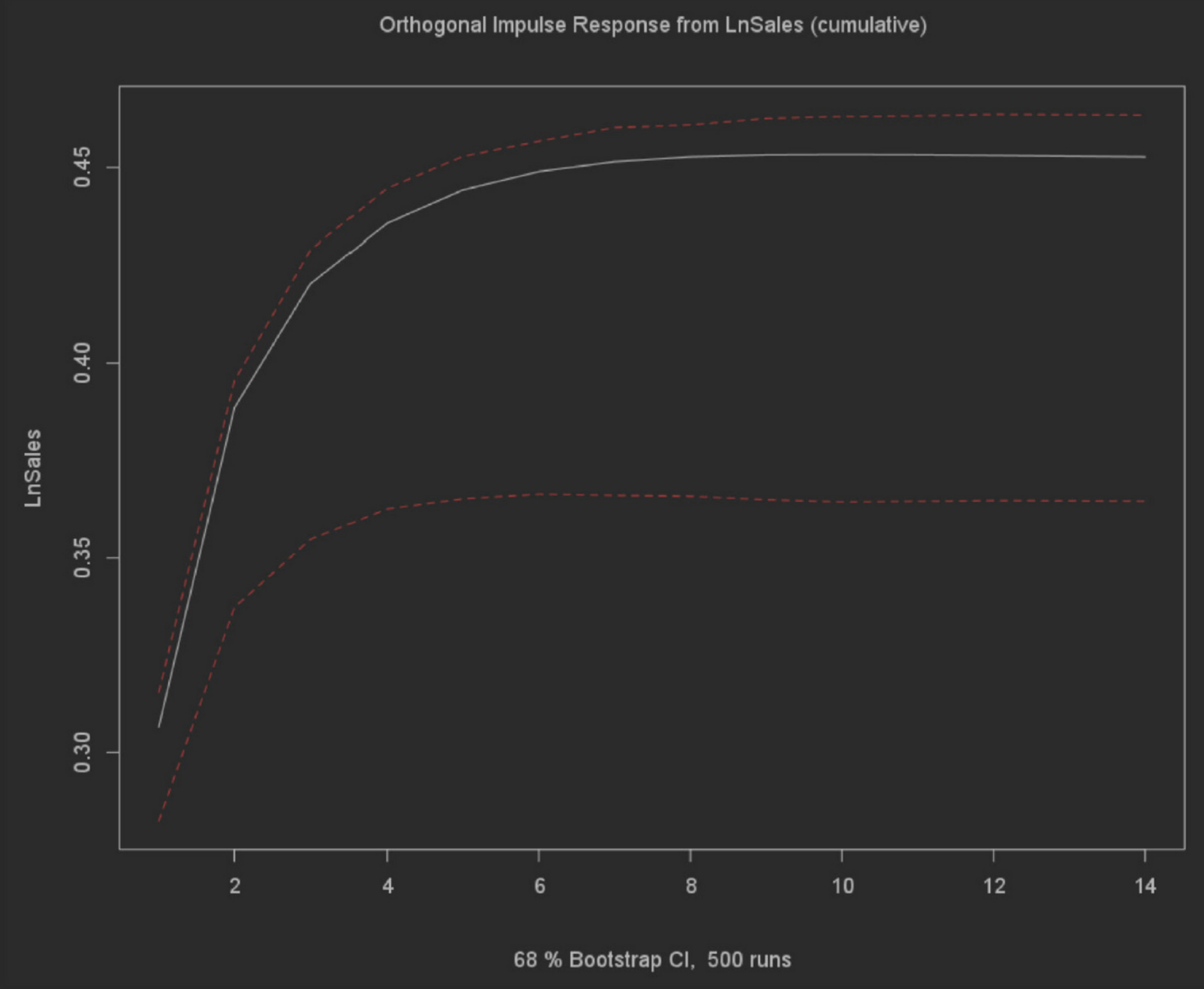
IRF PLOTS: OWN SALES CUMULATIVE





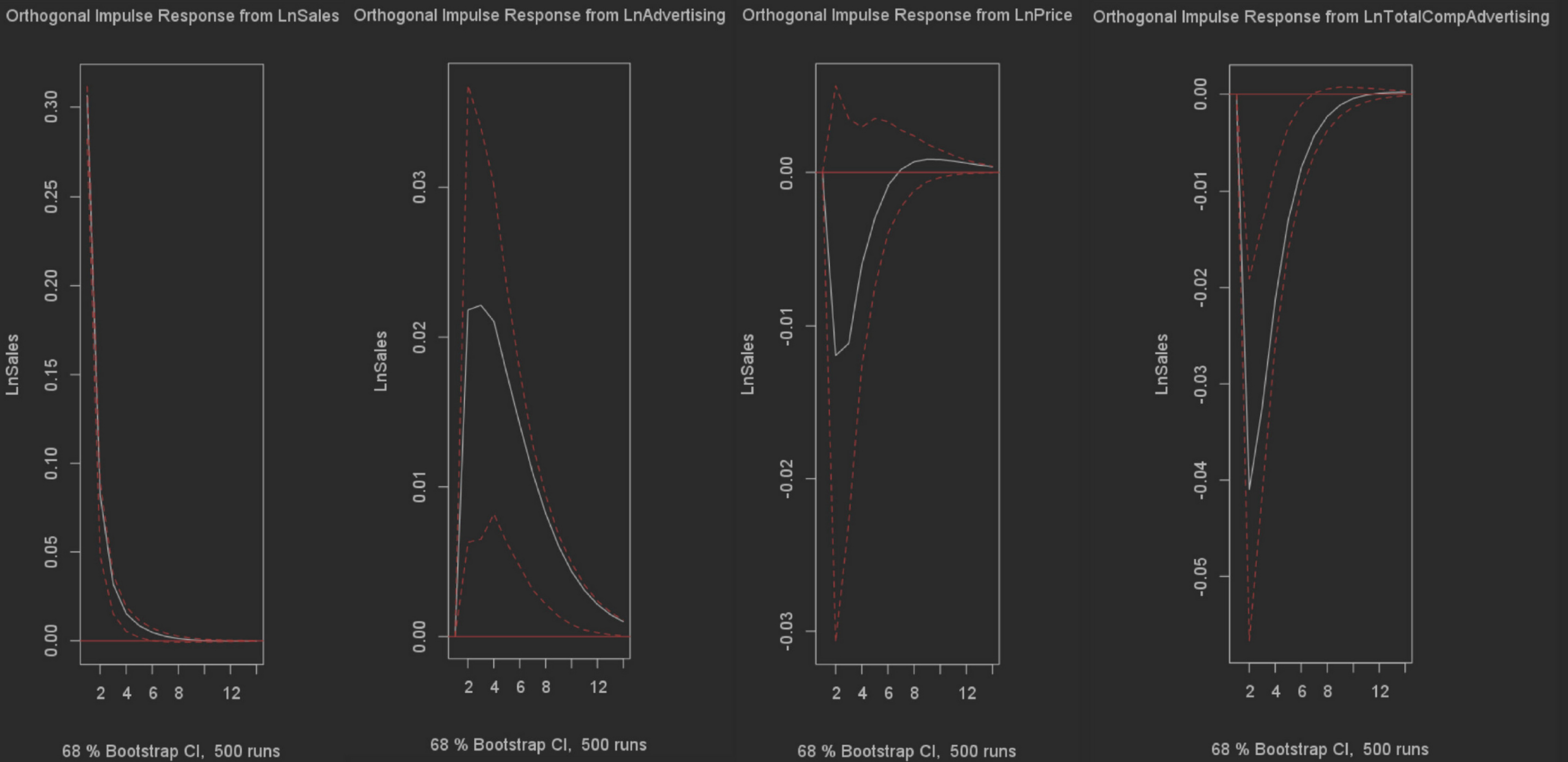
APPENDIX A-5.1.3

IRF PLOTS: OWN SALES CUMULATIVE



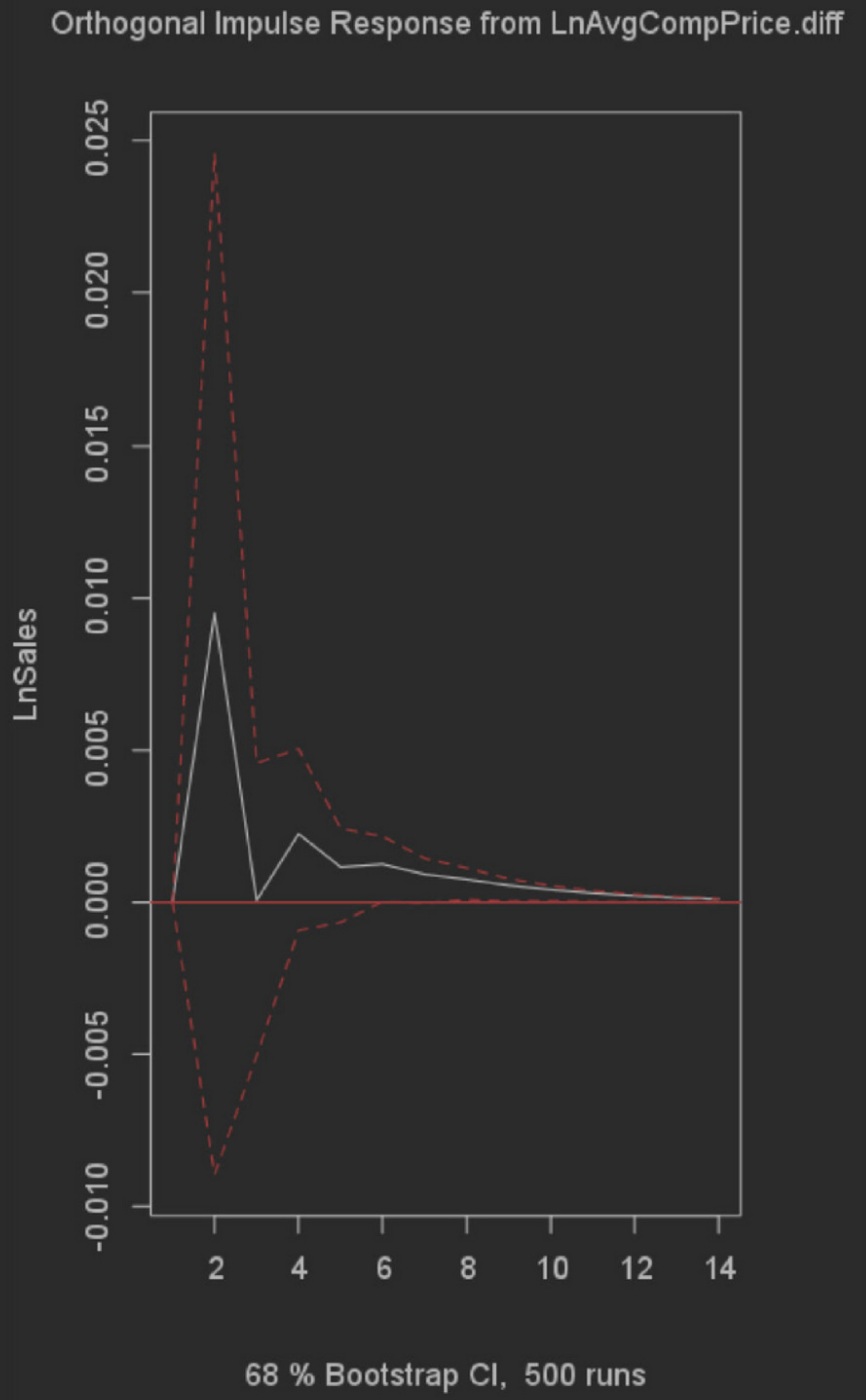
APPENDIX A-5.1.4

IRF PLOTS: OWN SALES IMMEDIATE



APPENDIX A-5.1.5

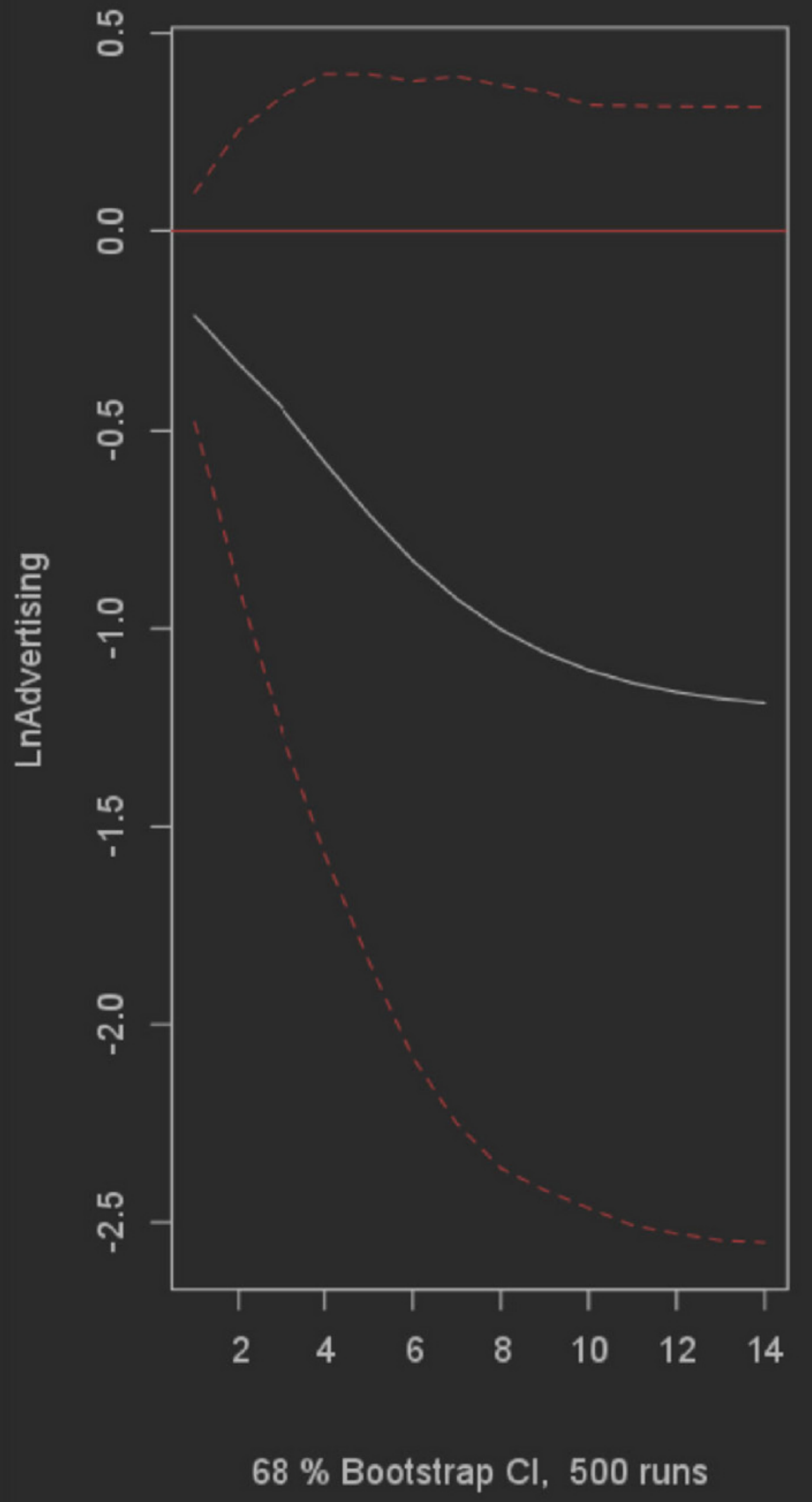
IRF PLOTS: OWN SALES IMMEDIATE



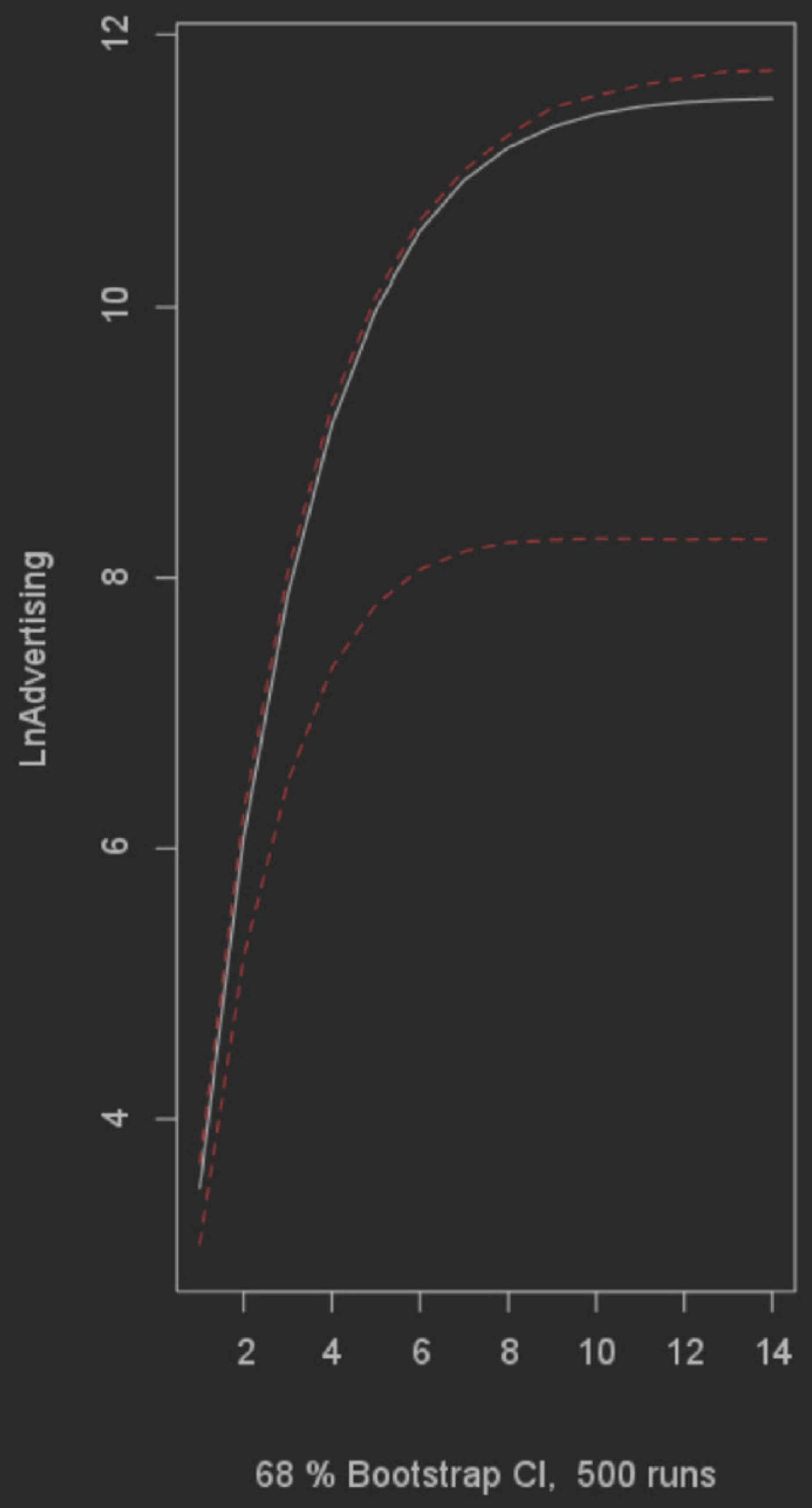
APPENDIX A-5.2.1

IRF PLOTS: OWN ADVERTISING CUMULATIVE

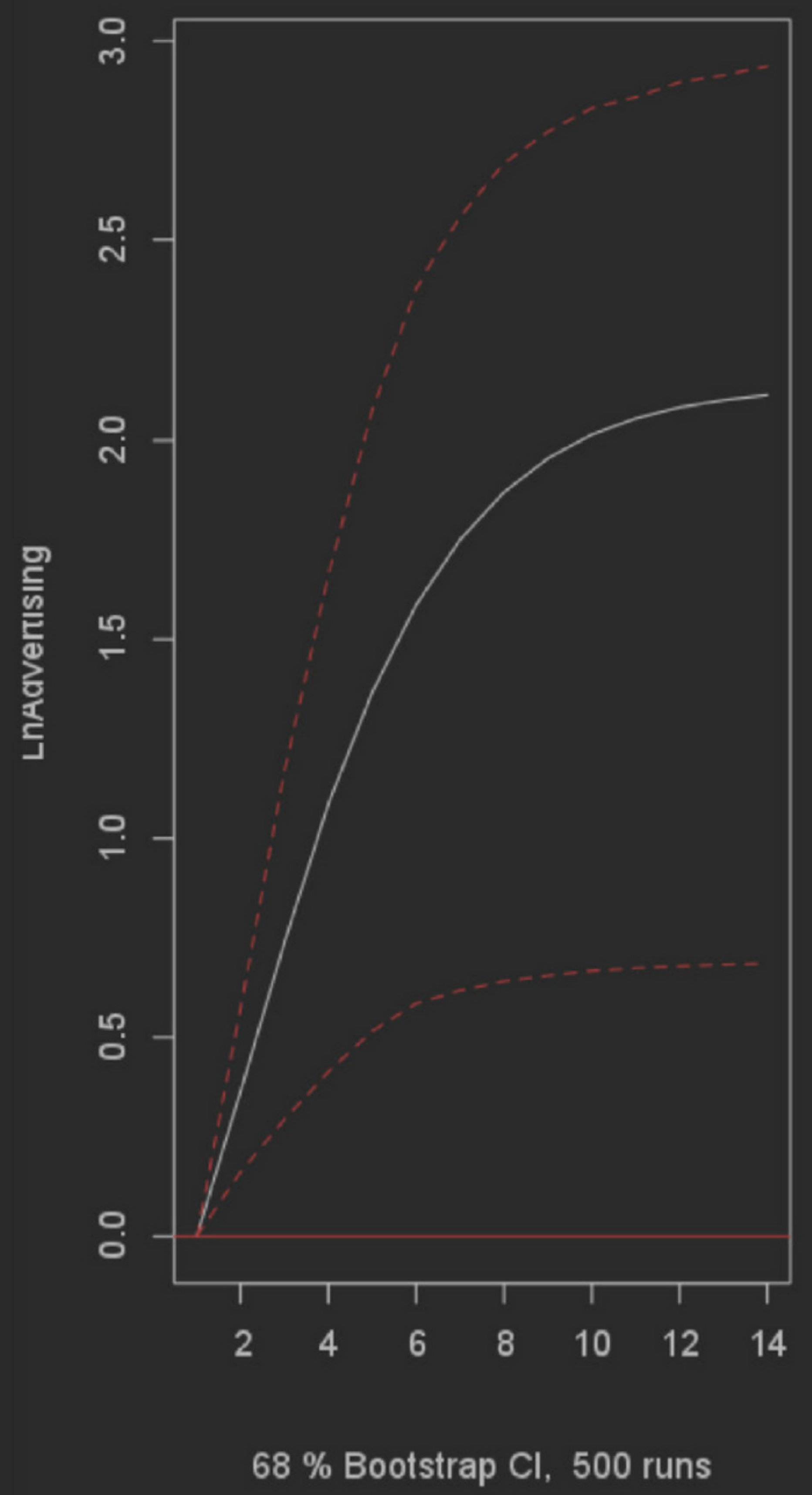
Orthogonal Impulse Response from LnSales (cumulative)



Orthogonal Impulse Response from LnAdvertising (cumulative)

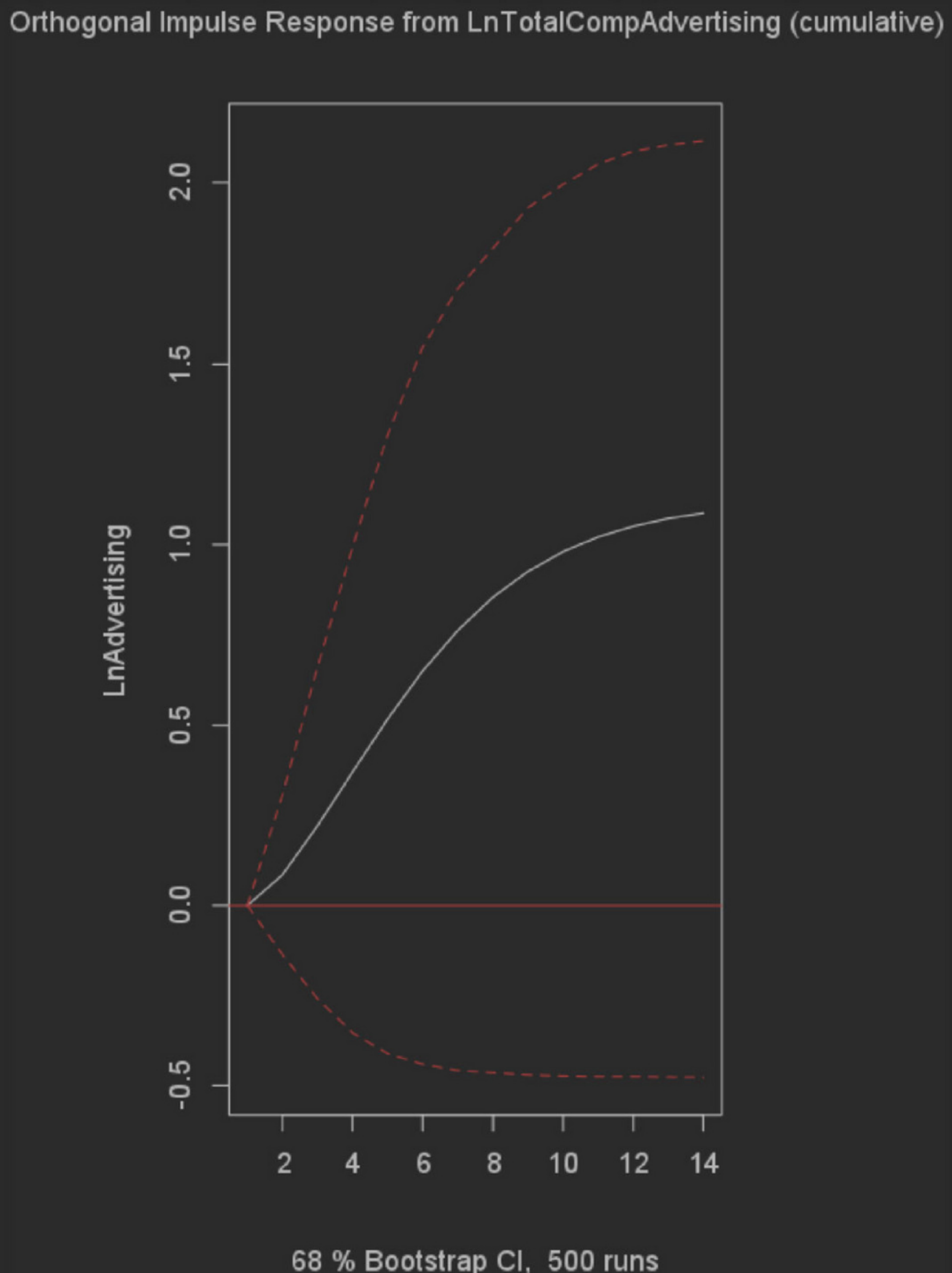


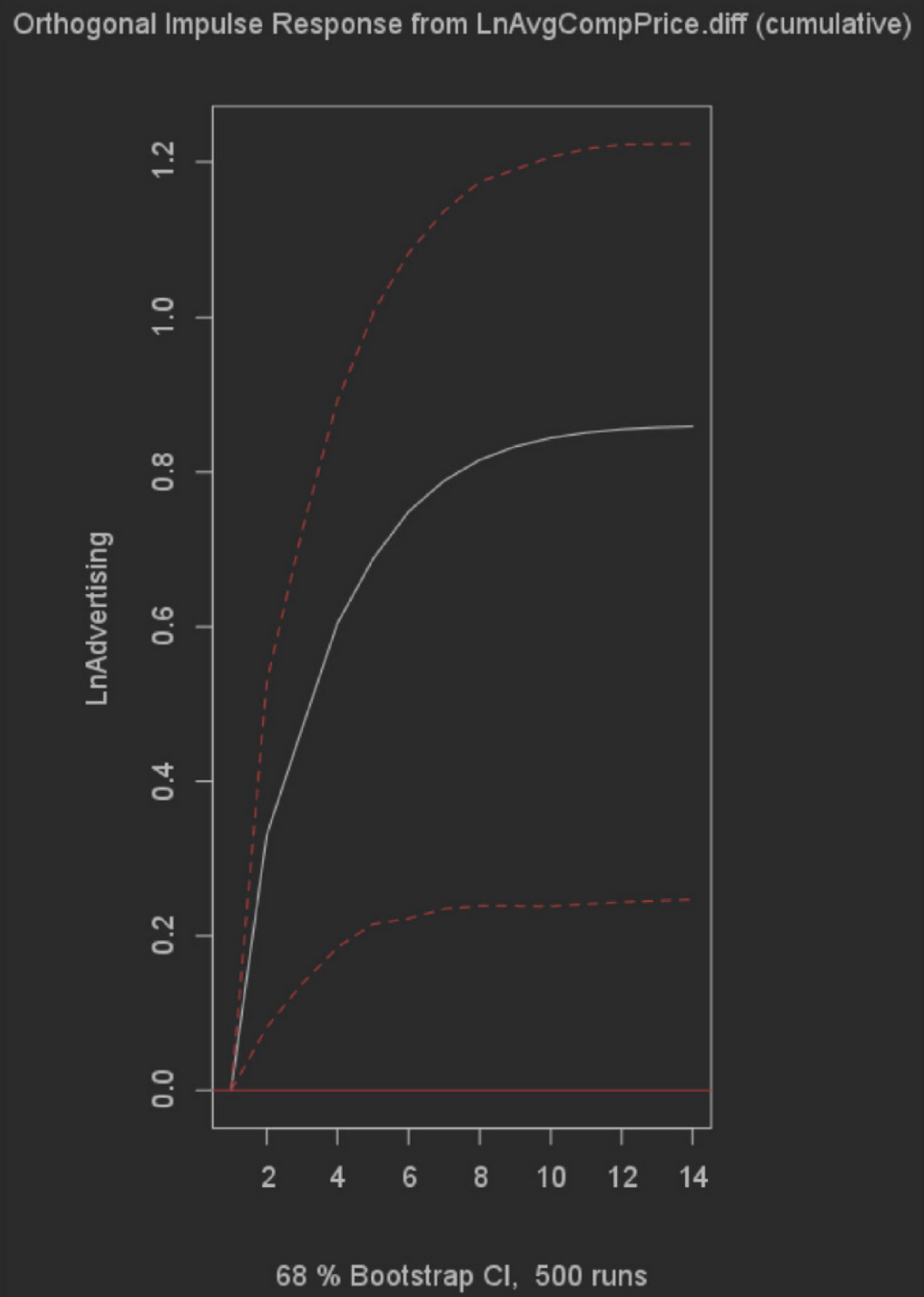
Orthogonal Impulse Response from LnPrice (cumulative)



APPENDIX A-5.2.2

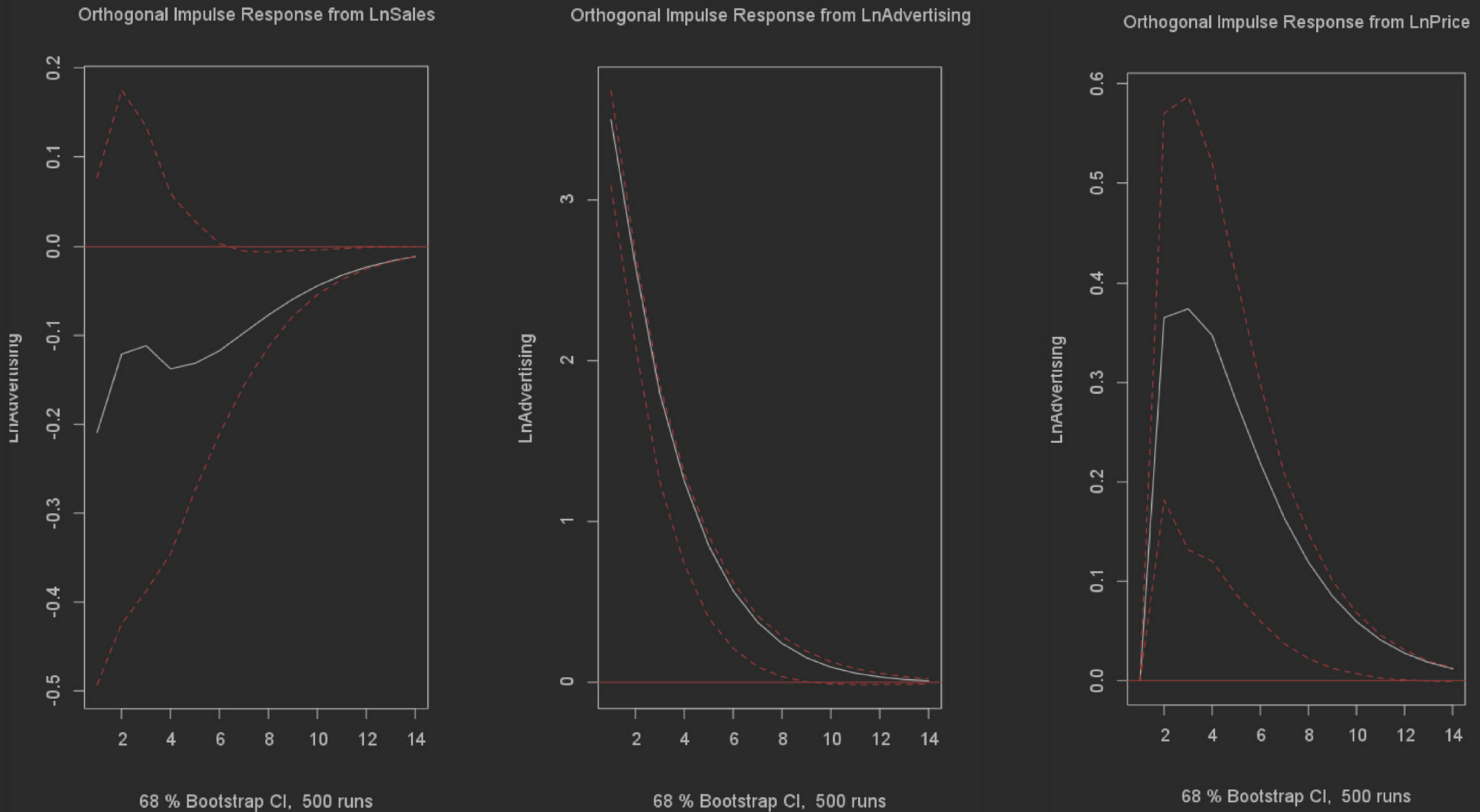
IRF PLOTS: OWN ADVERTISING CUMULATIVE





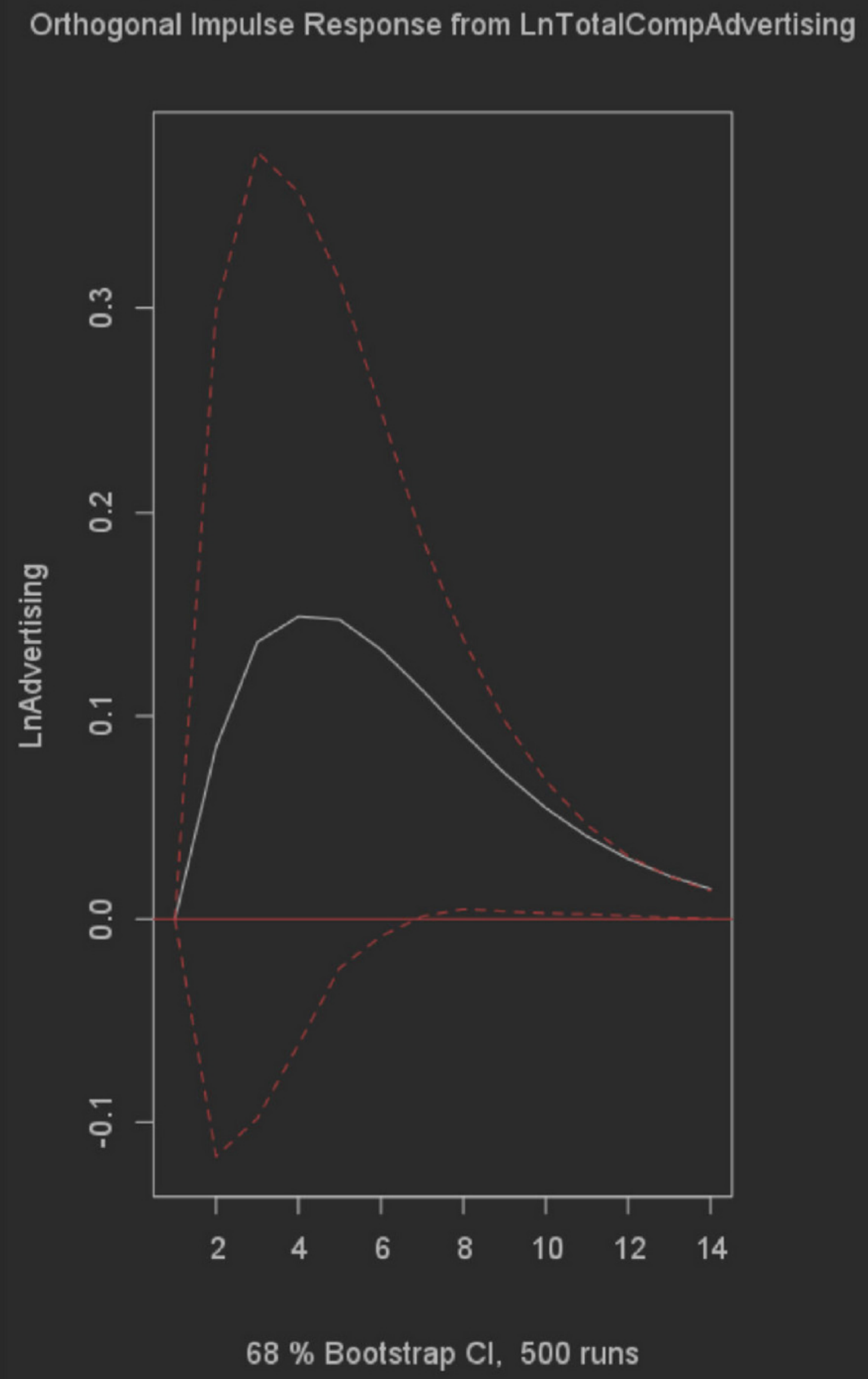
APPENDIX A-5.2.3

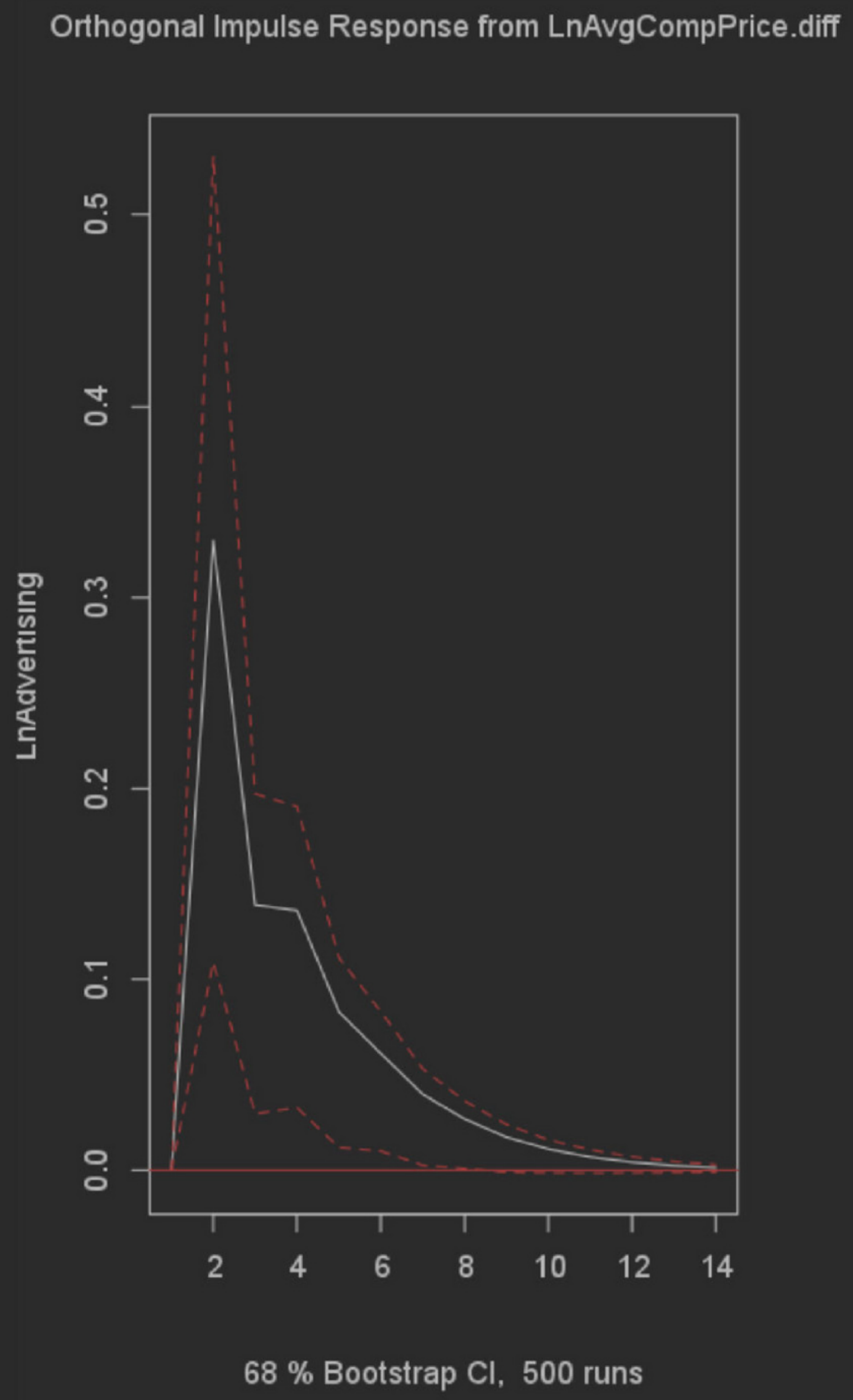
IRF PLOTS: OWN ADVERTISING IMMEDIATE



APPENDIX A-5.2.4

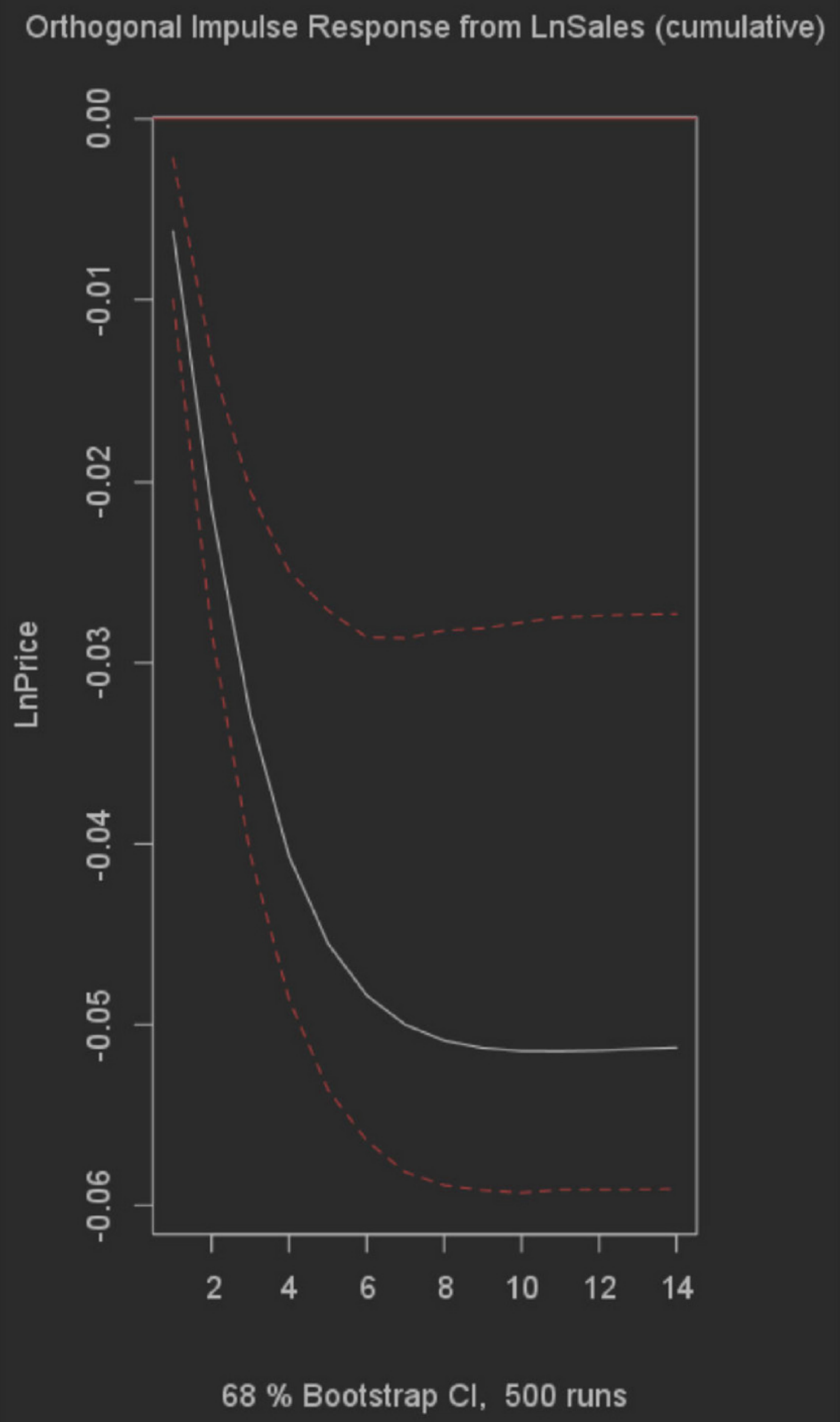
IRF PLOTS: OWN ADVERTISING IMMEDIATE

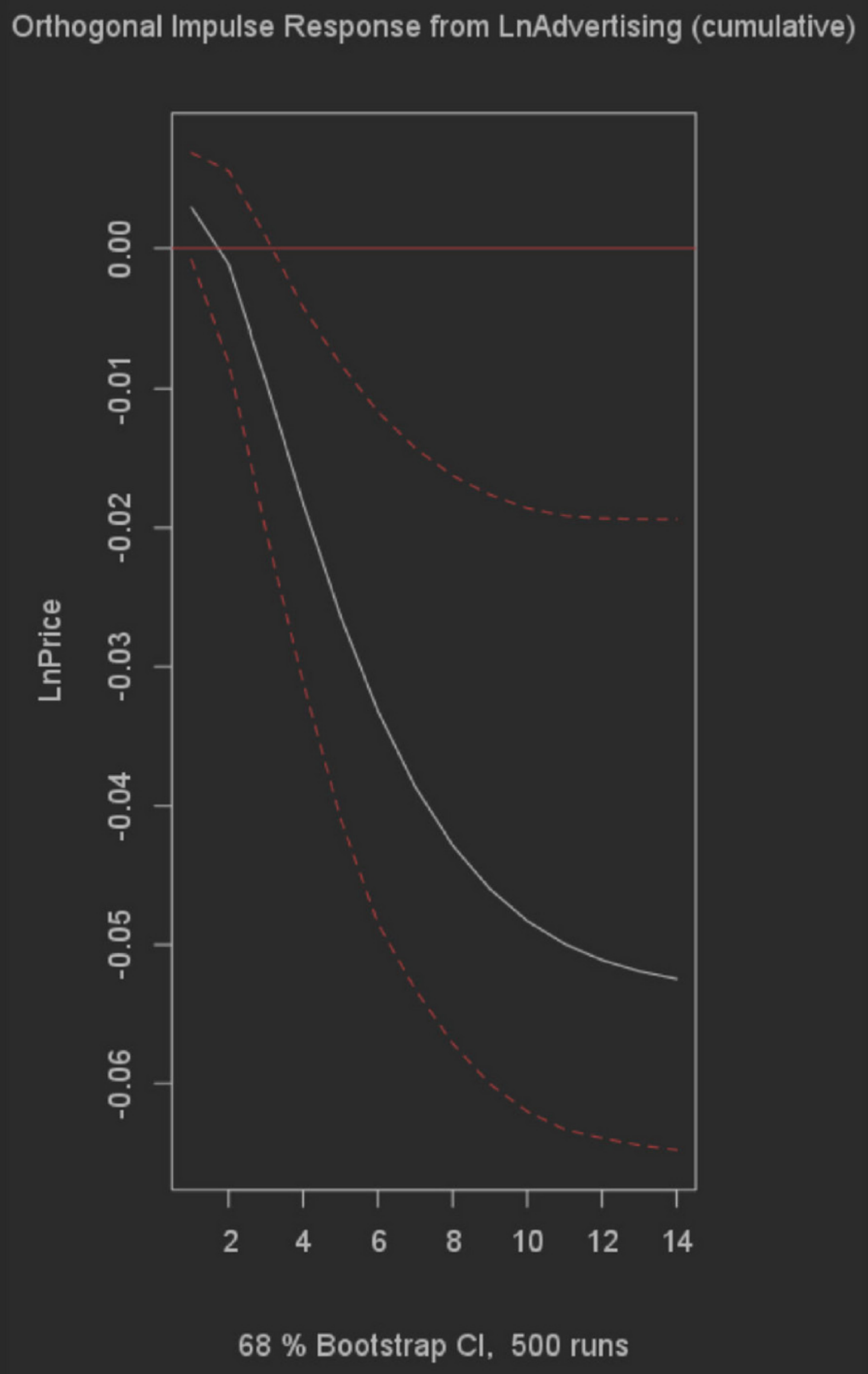


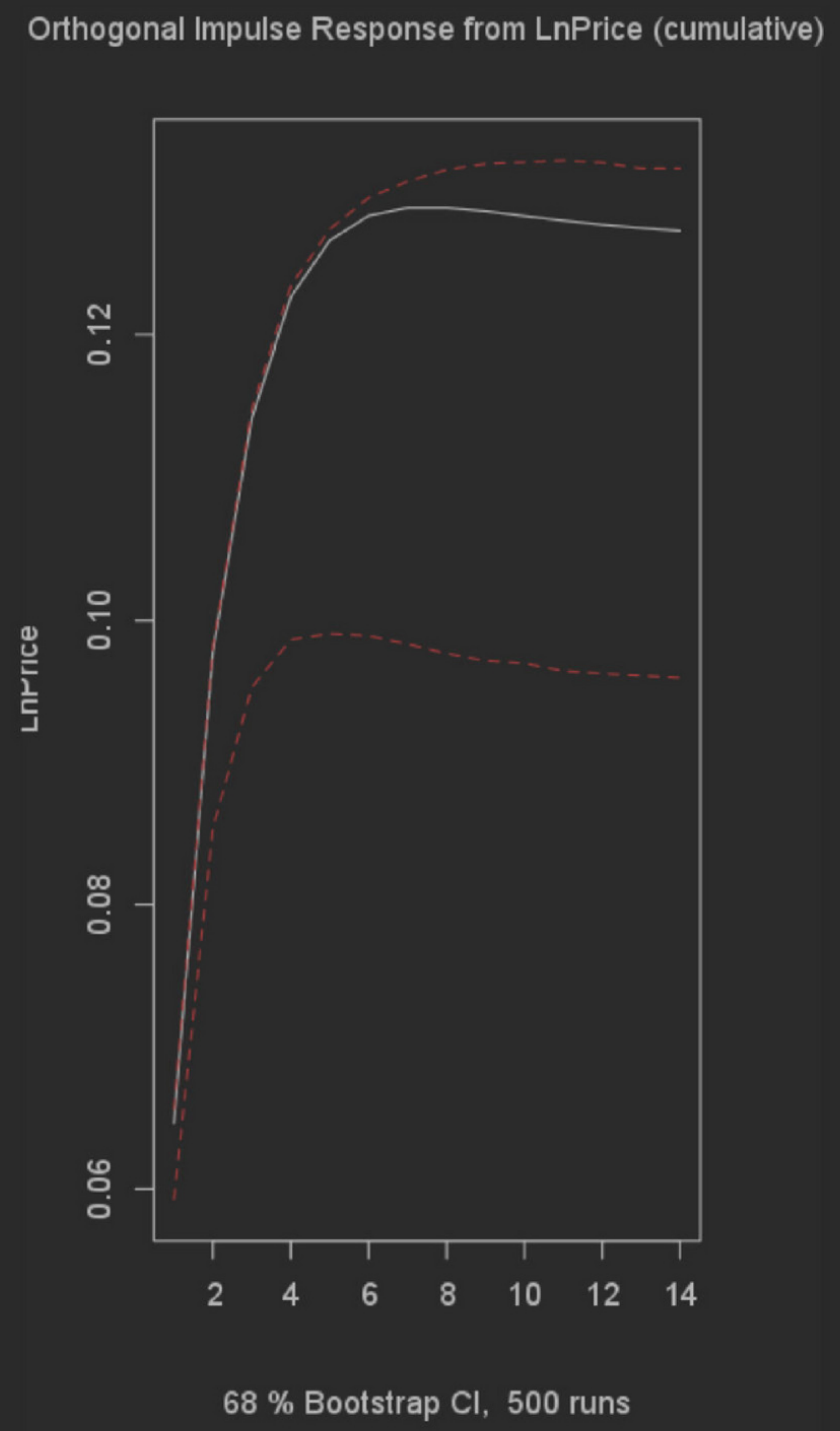


APPENDIX A-5.3.1

IRF PLOTS: OWN PRICE CUMULATIVE

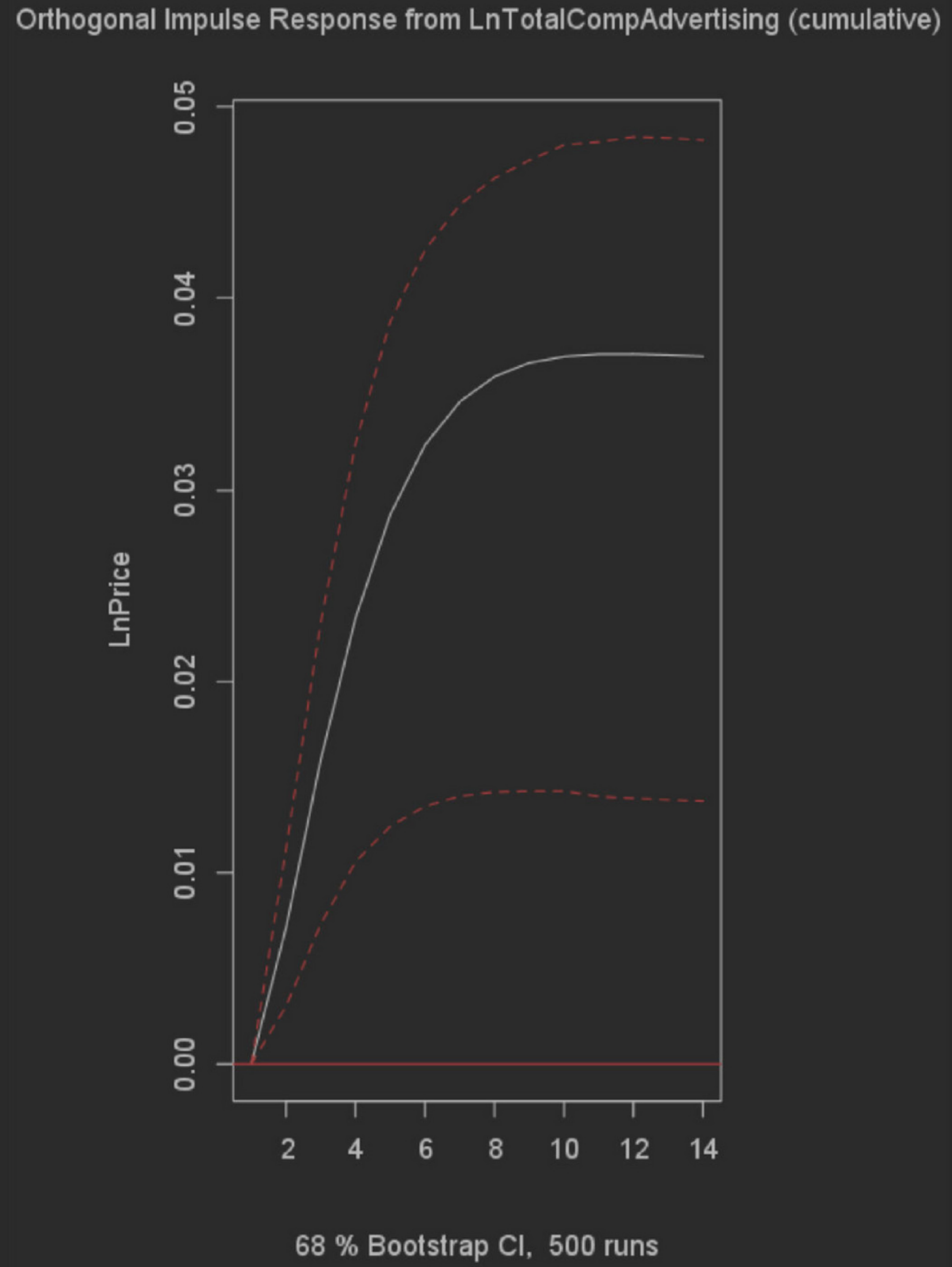


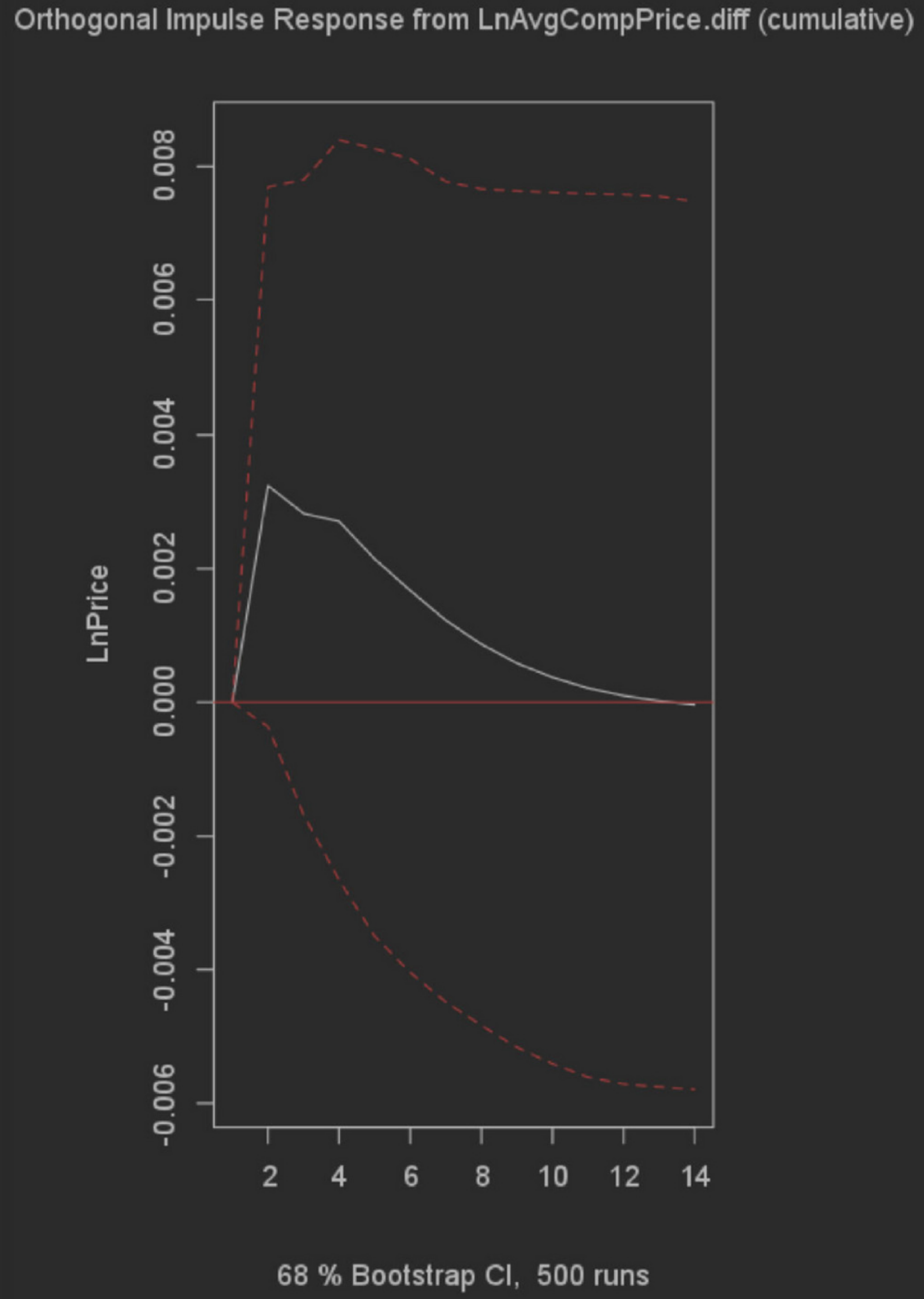




APPENDIX A-5.3.2

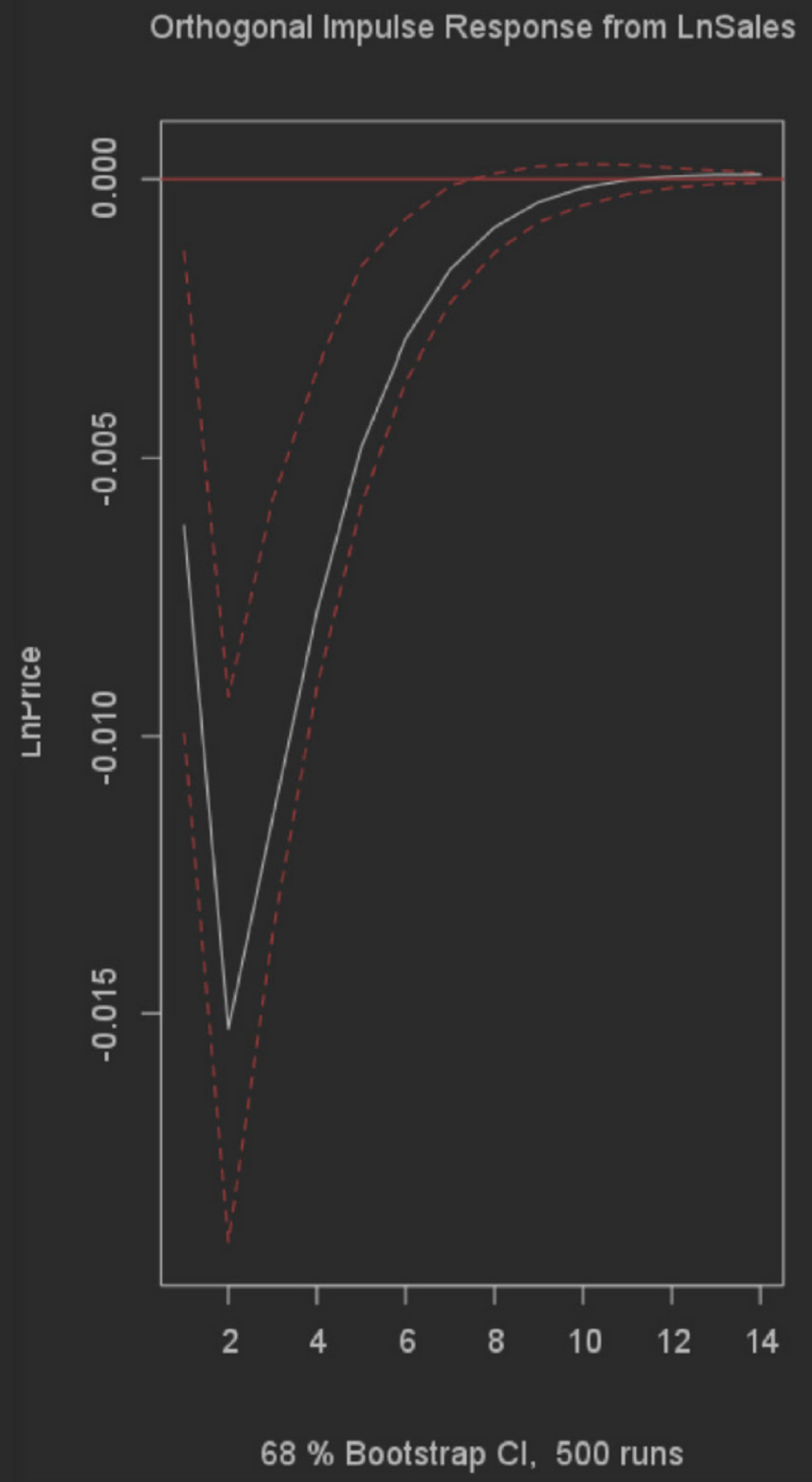
IRF PLOTS: OWN PRICE CUMULATIVE

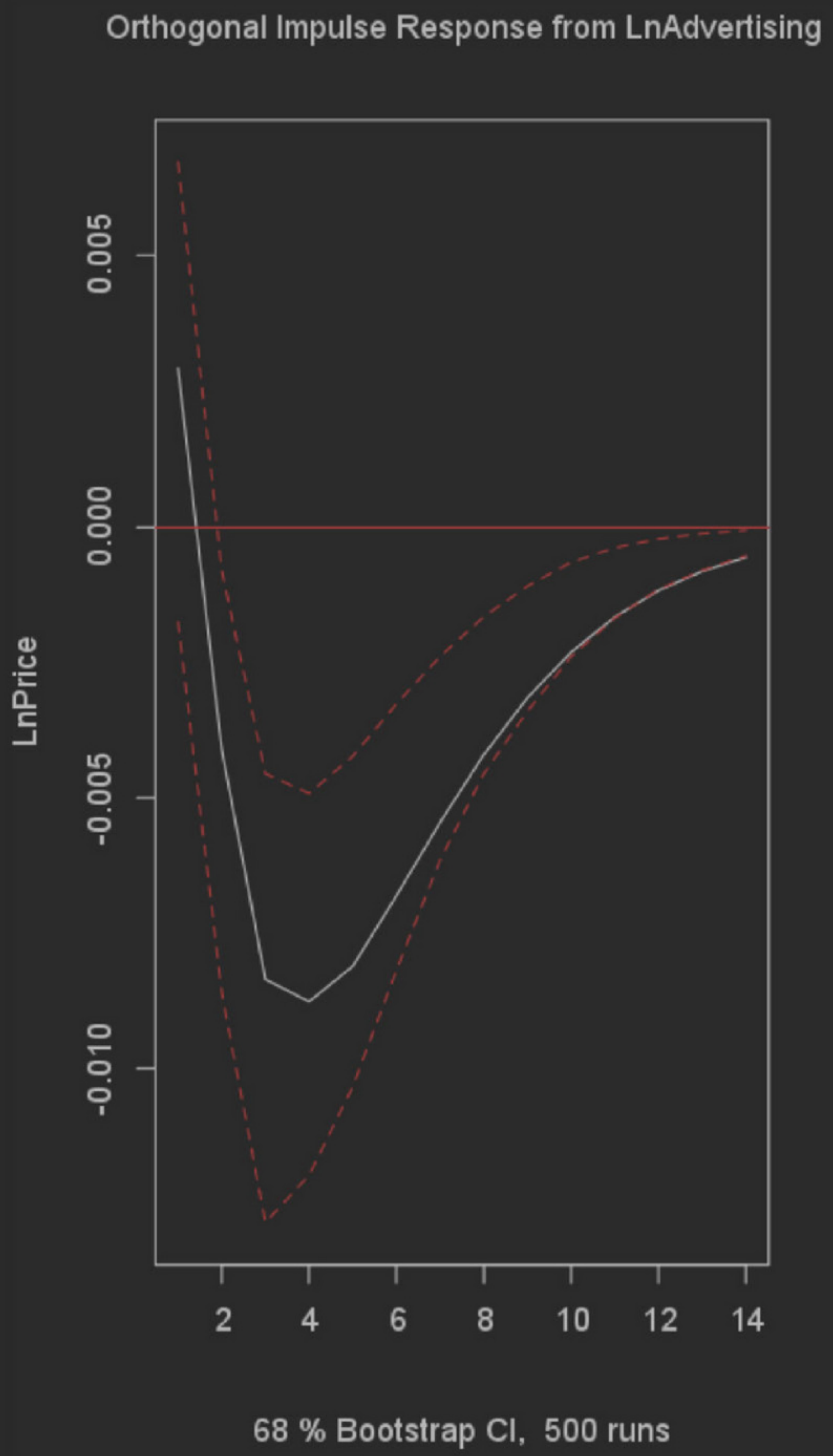


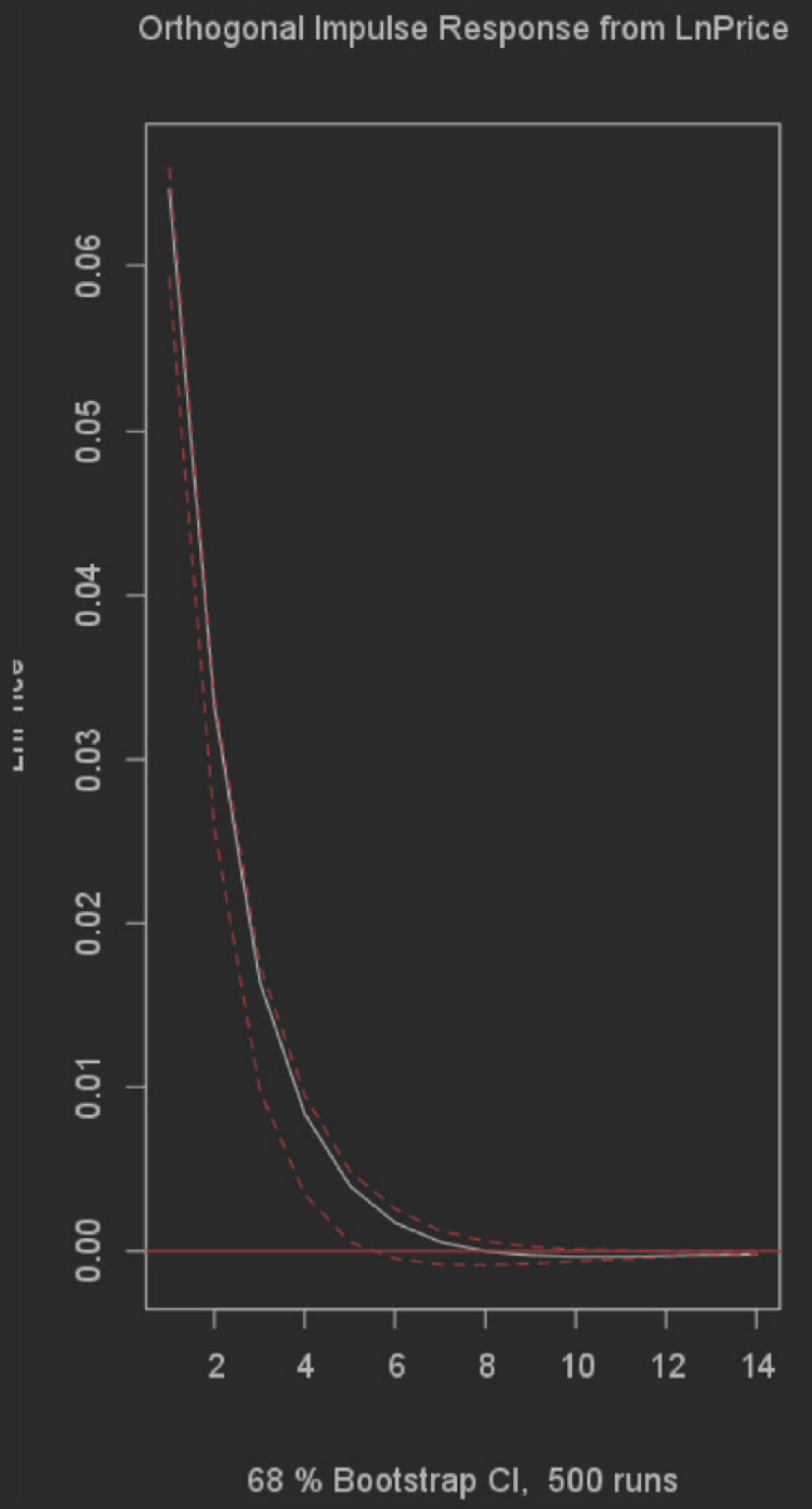


APPENDIX A-5.3.3

IRF PLOTS: OWN PRICE IMMEDIATE

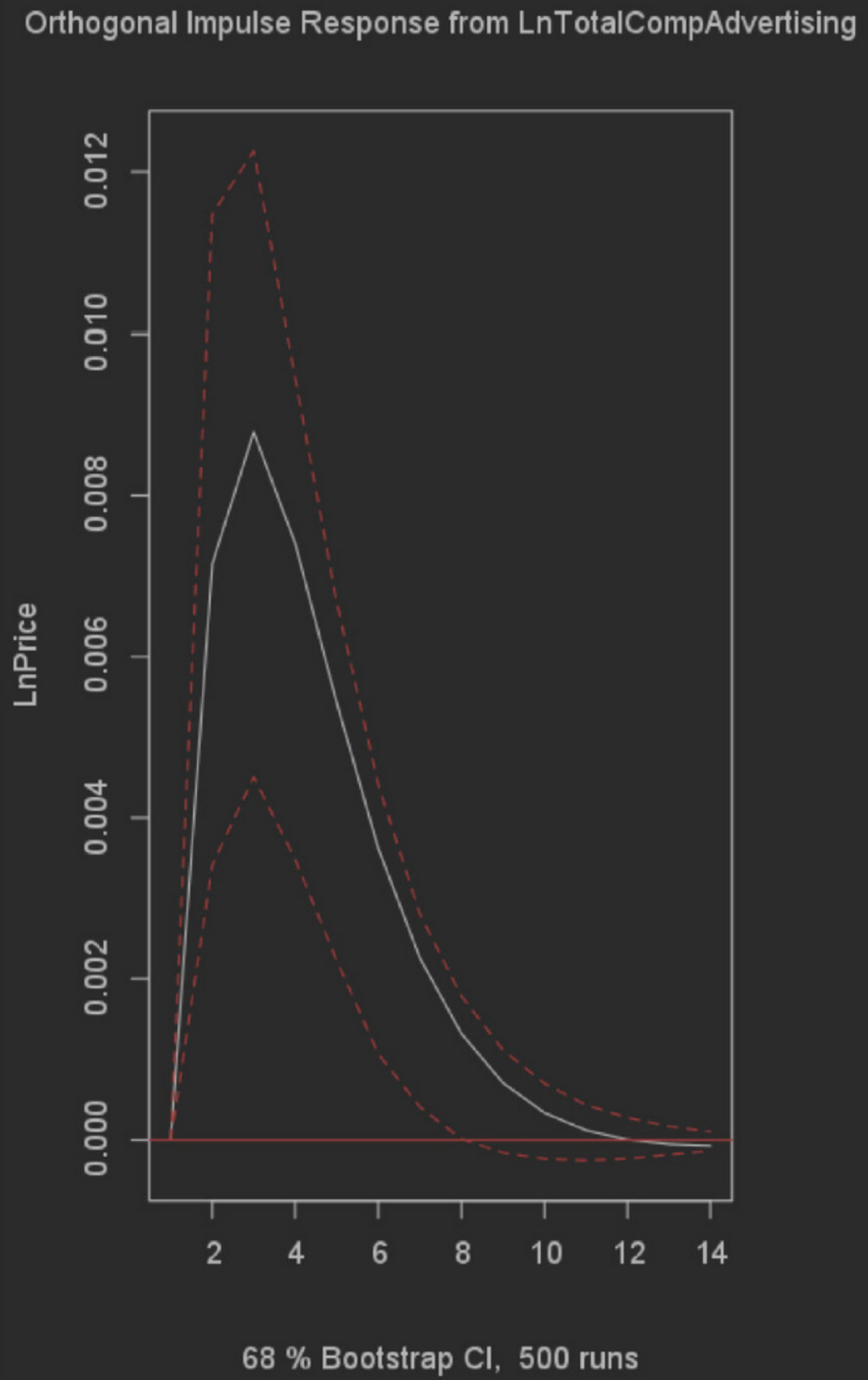


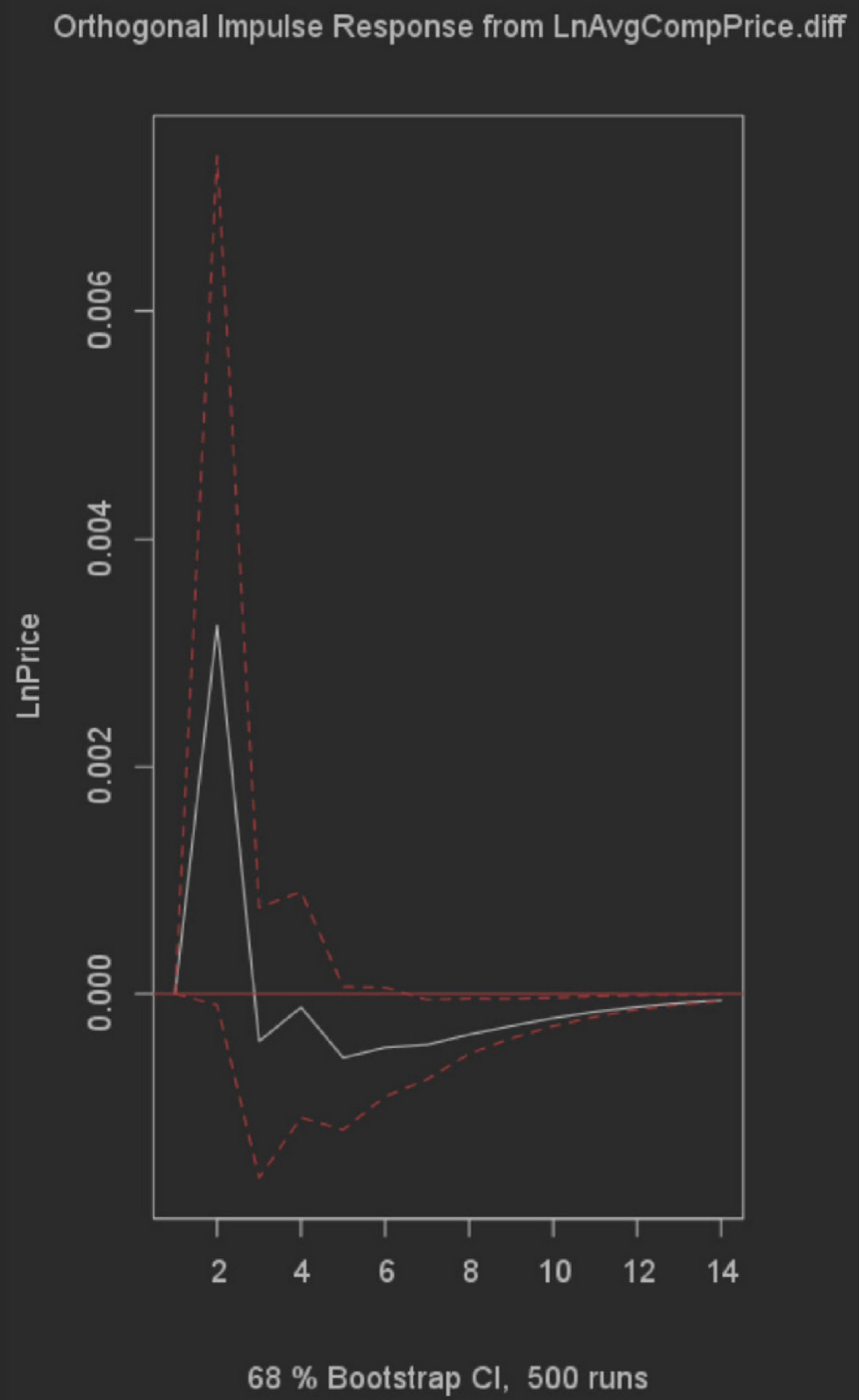




APPENDIX A-5.3.4

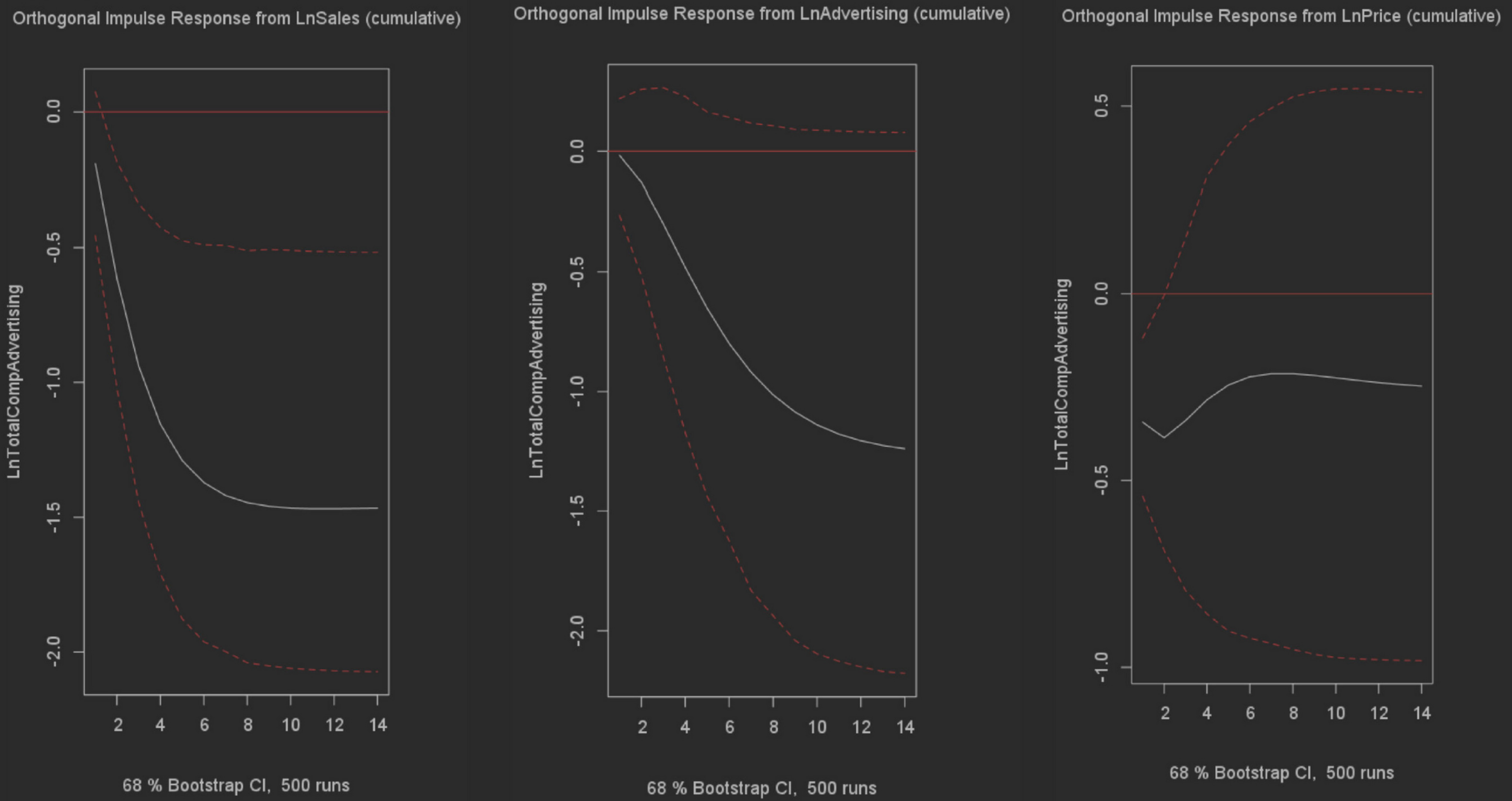
IRF PLOTS: OWN PRICE IMMEDIATE





APPENDIX A-5.4.1

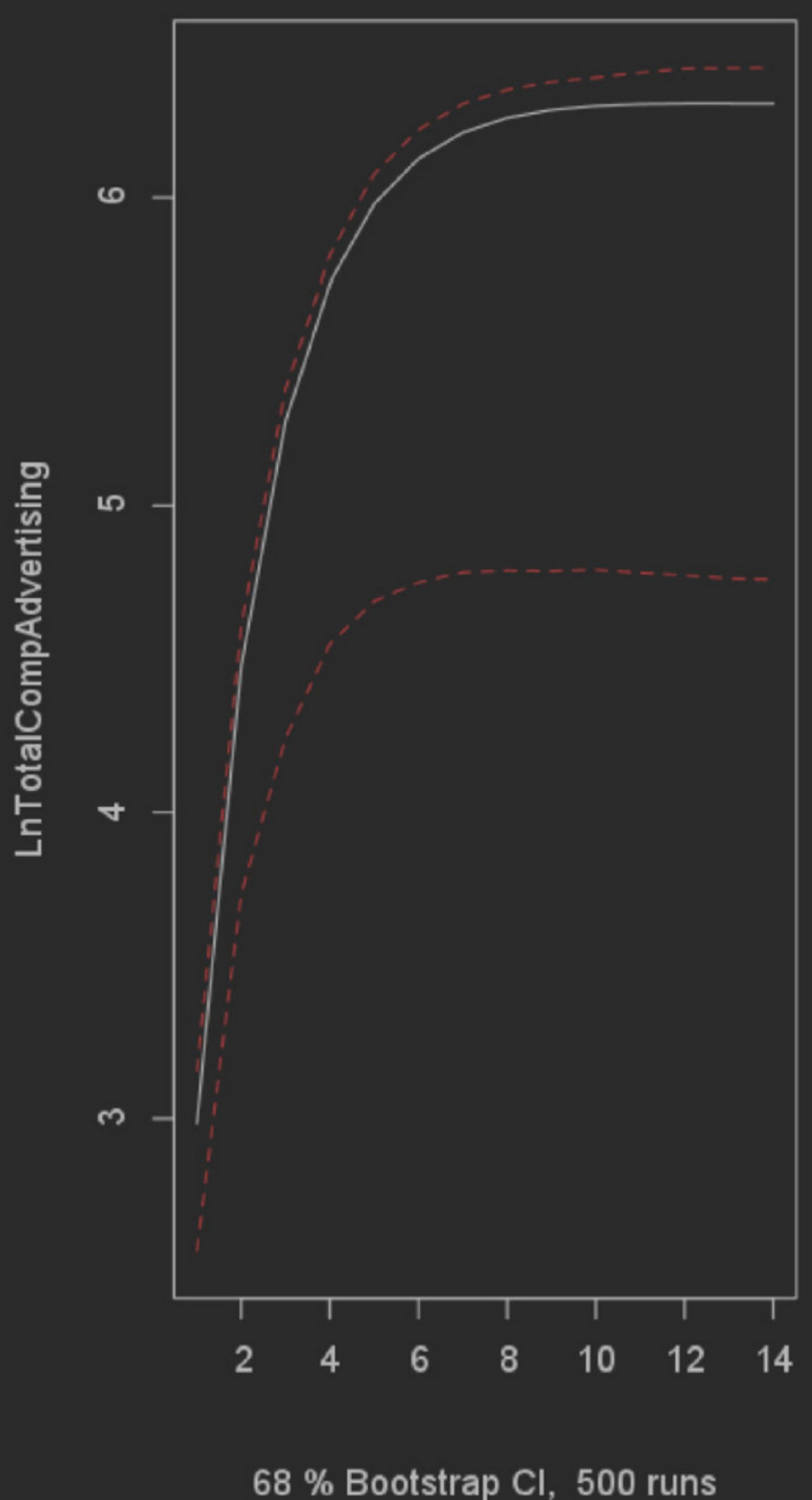
IRF PLOTS: TCA CUMULATIVE



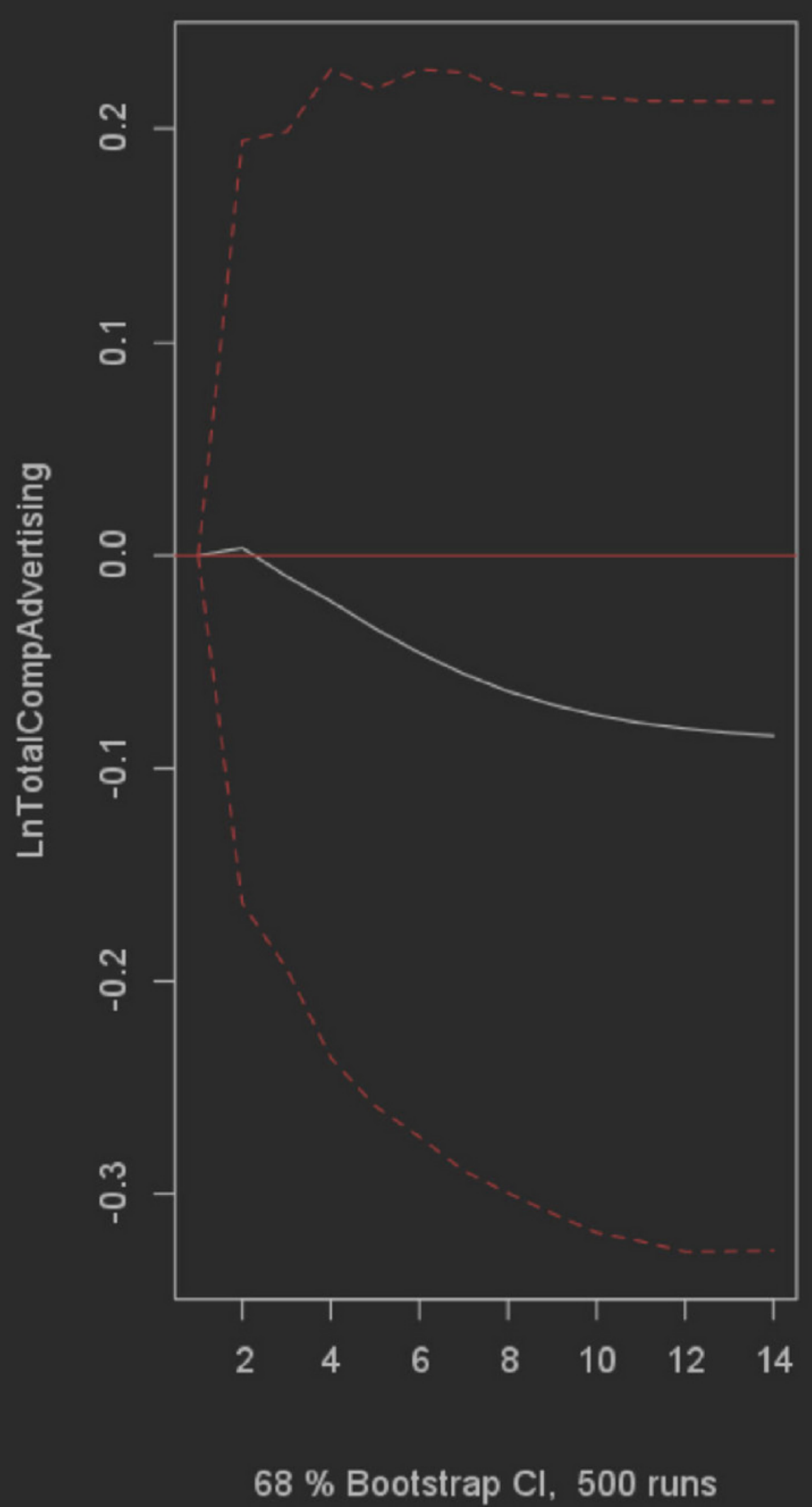
APPENDIX A-5.4.2

IRF PLOTS: TCA CUMULATIVE

Orthogonal Impulse Response from LnTotalCompAdvertising (cumulative)

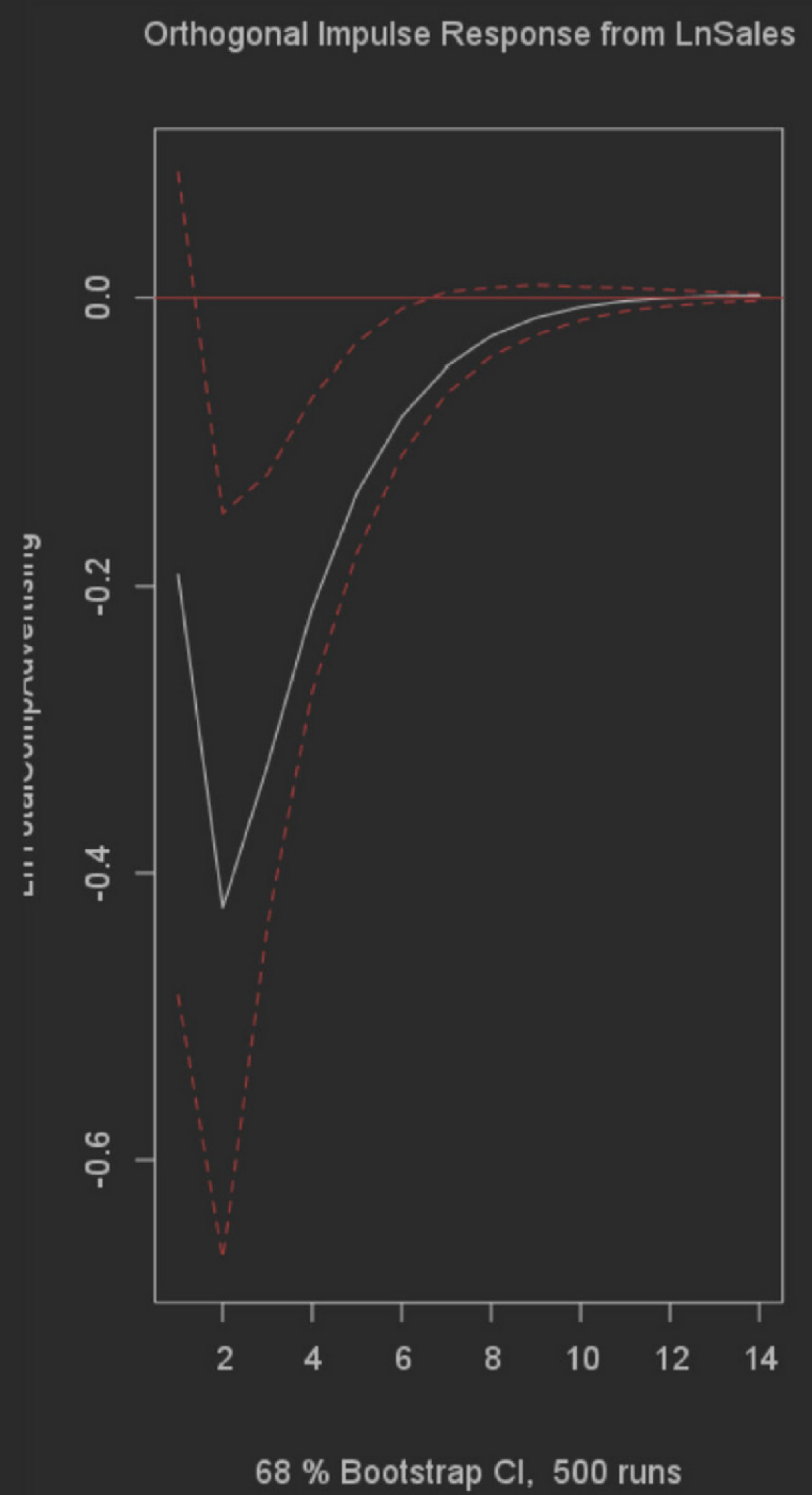


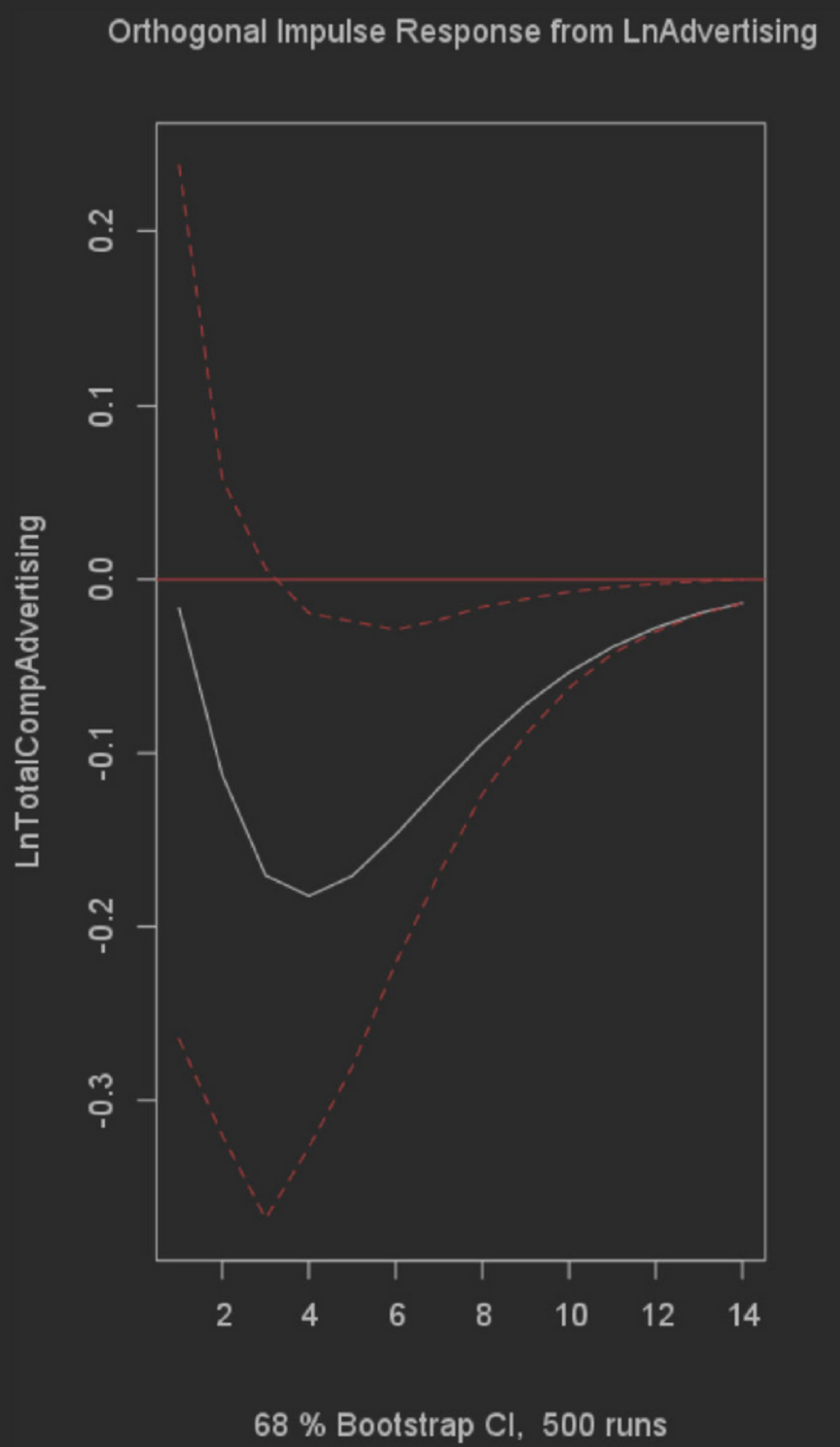
Orthogonal Impulse Response from LnAvgCompPrice.diff (cumulative)

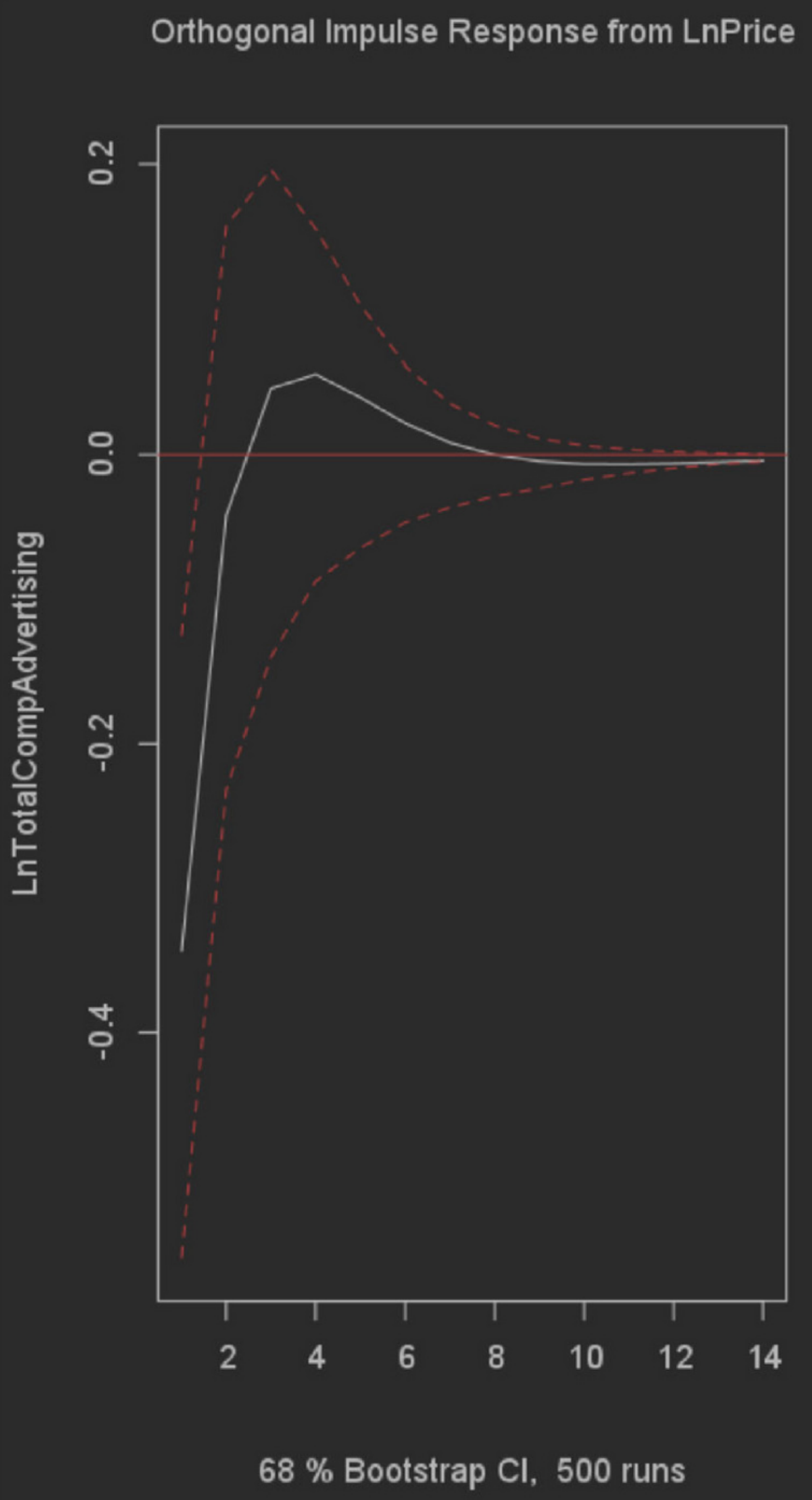


APPENDIX A-5.4.3

IRF PLOTS: TCA IMMEDIATE

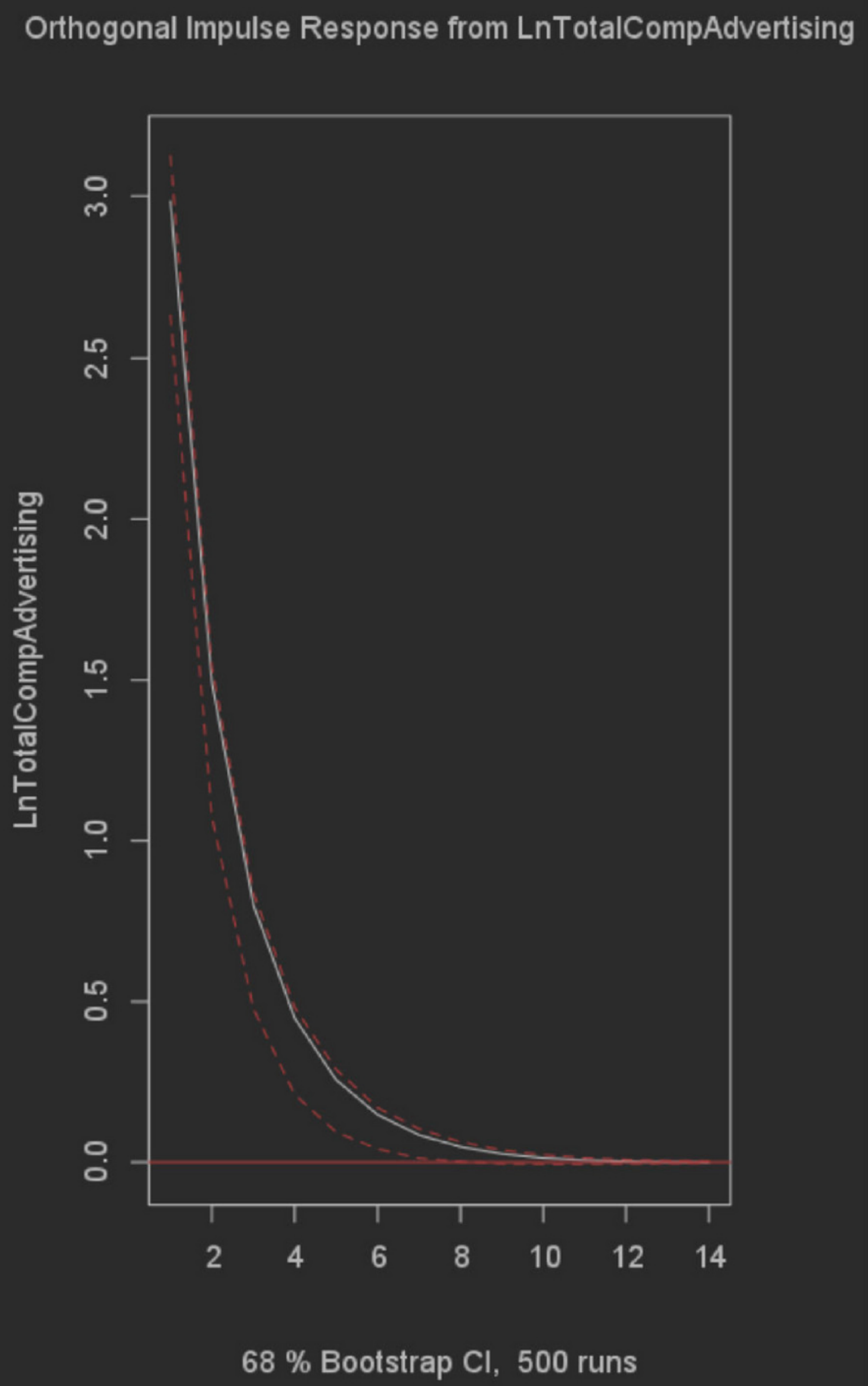


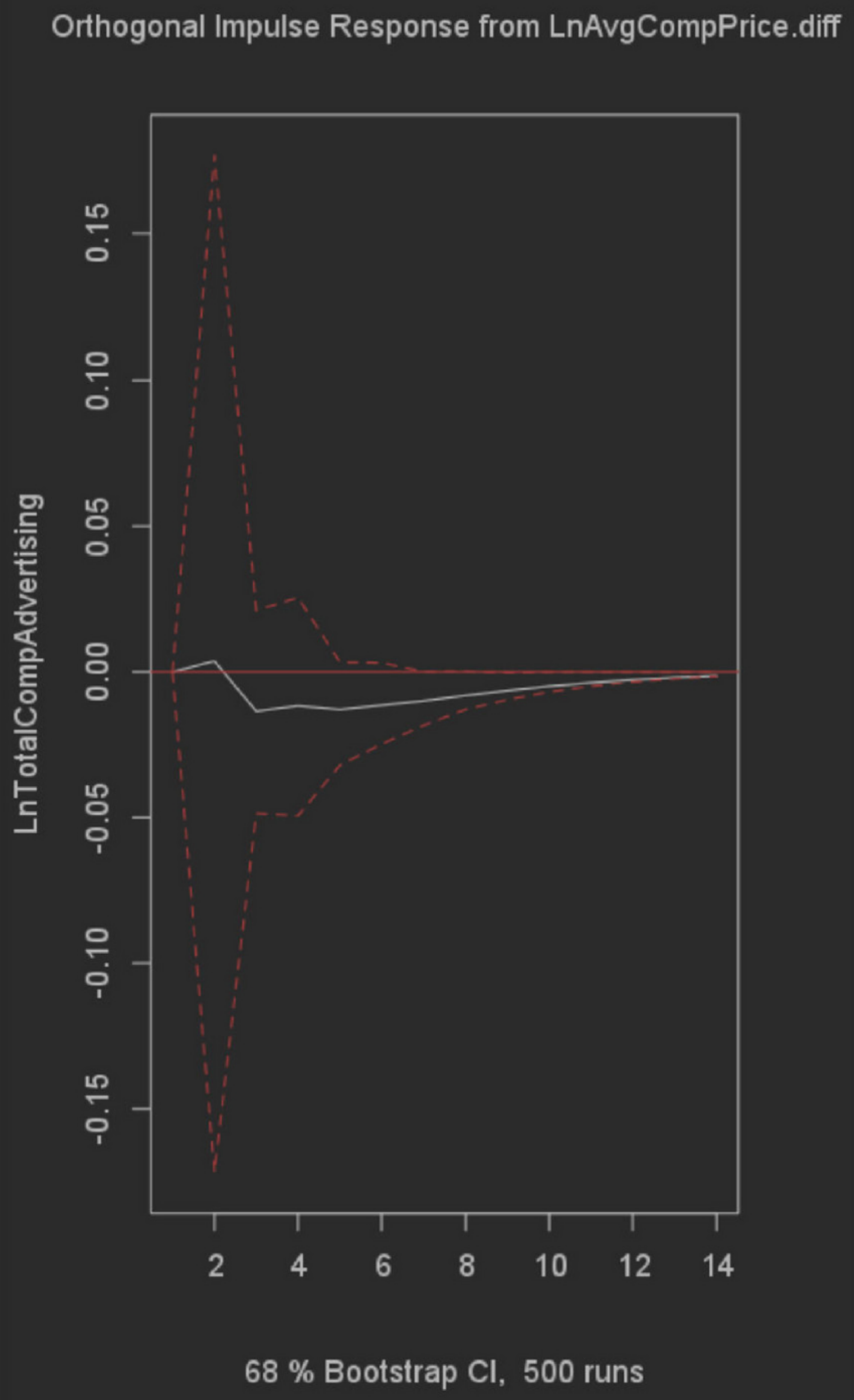




APPENDIX A-5.4.4

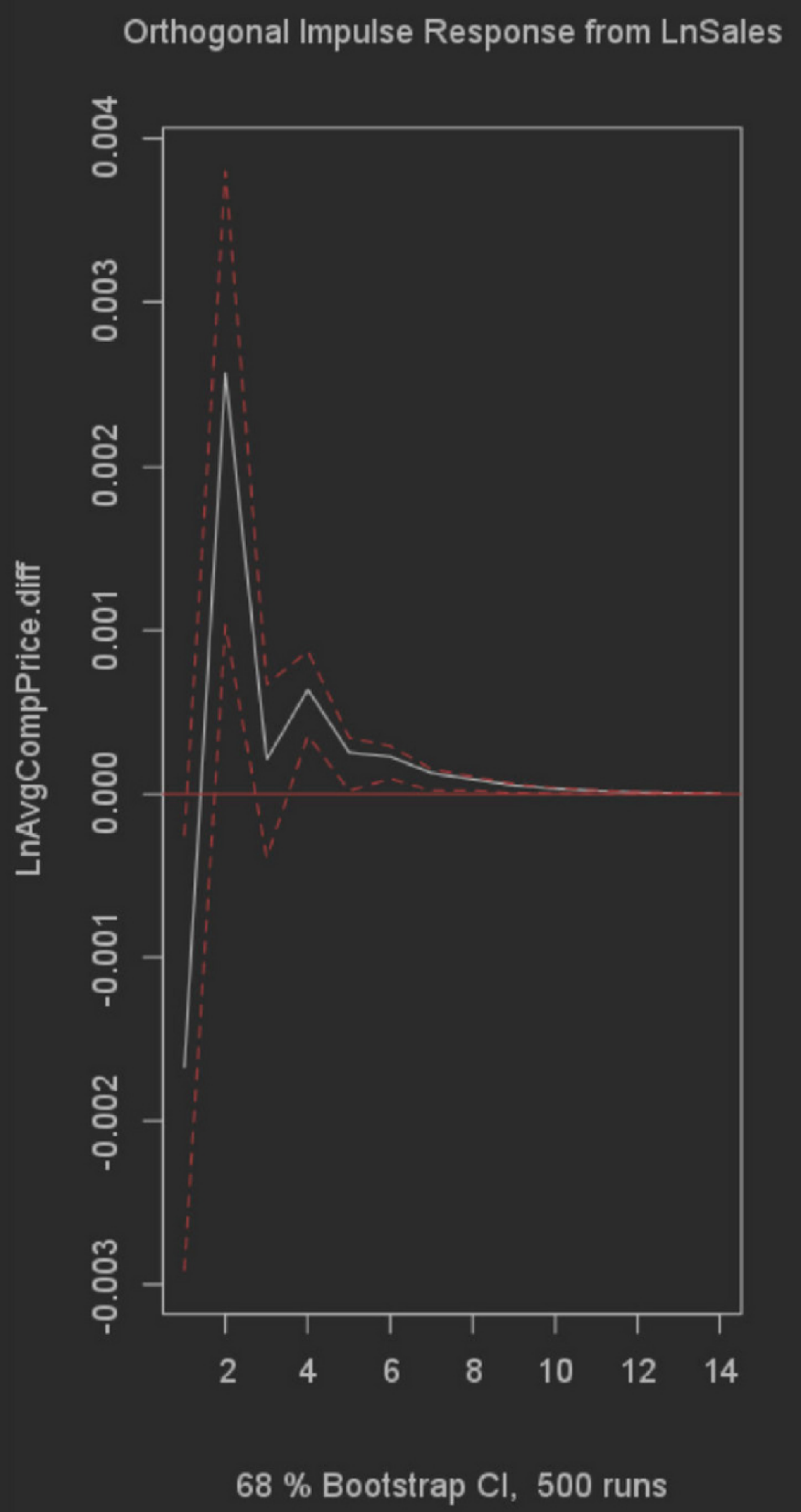
IRF PLOTS: TCA IMMEDIATE

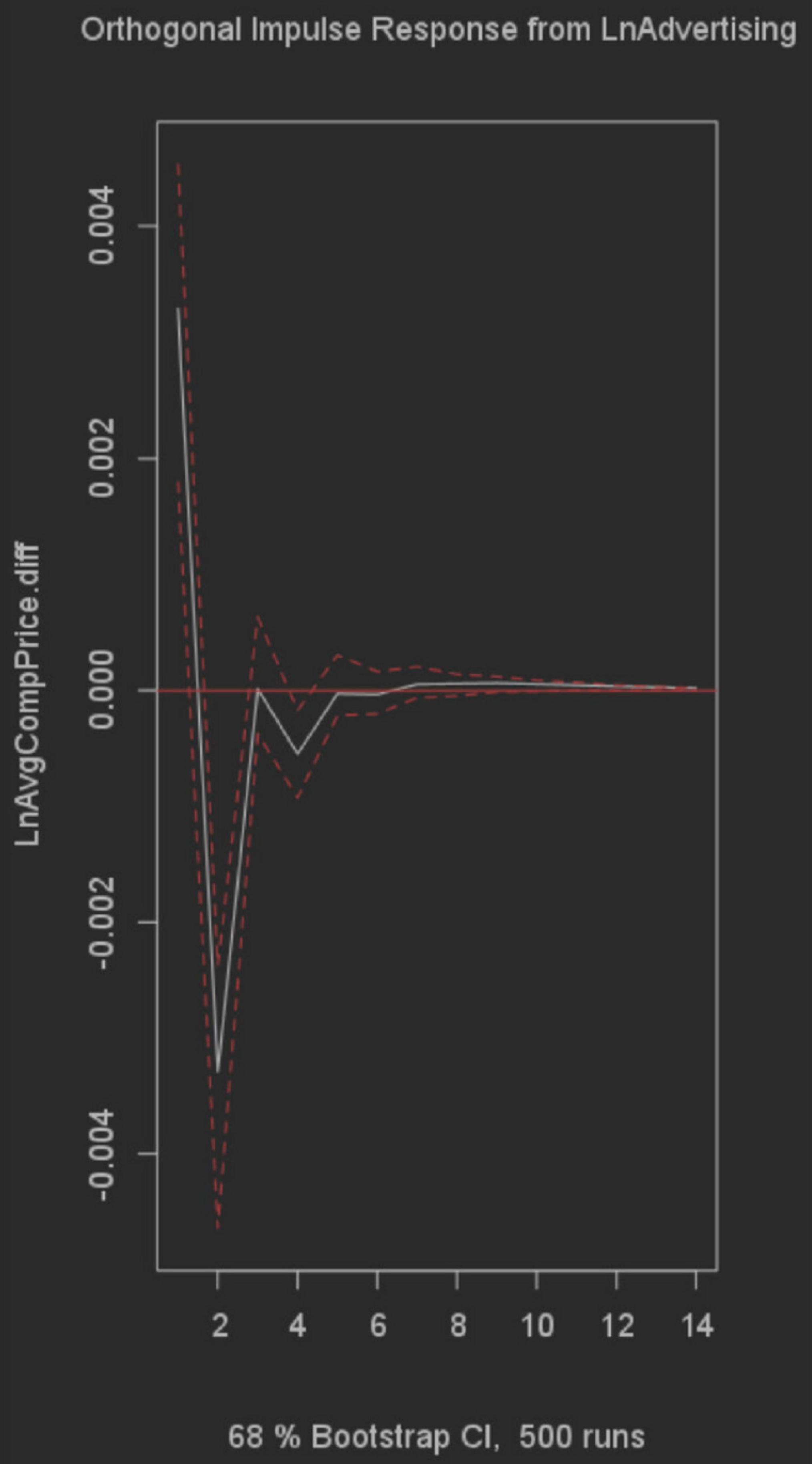


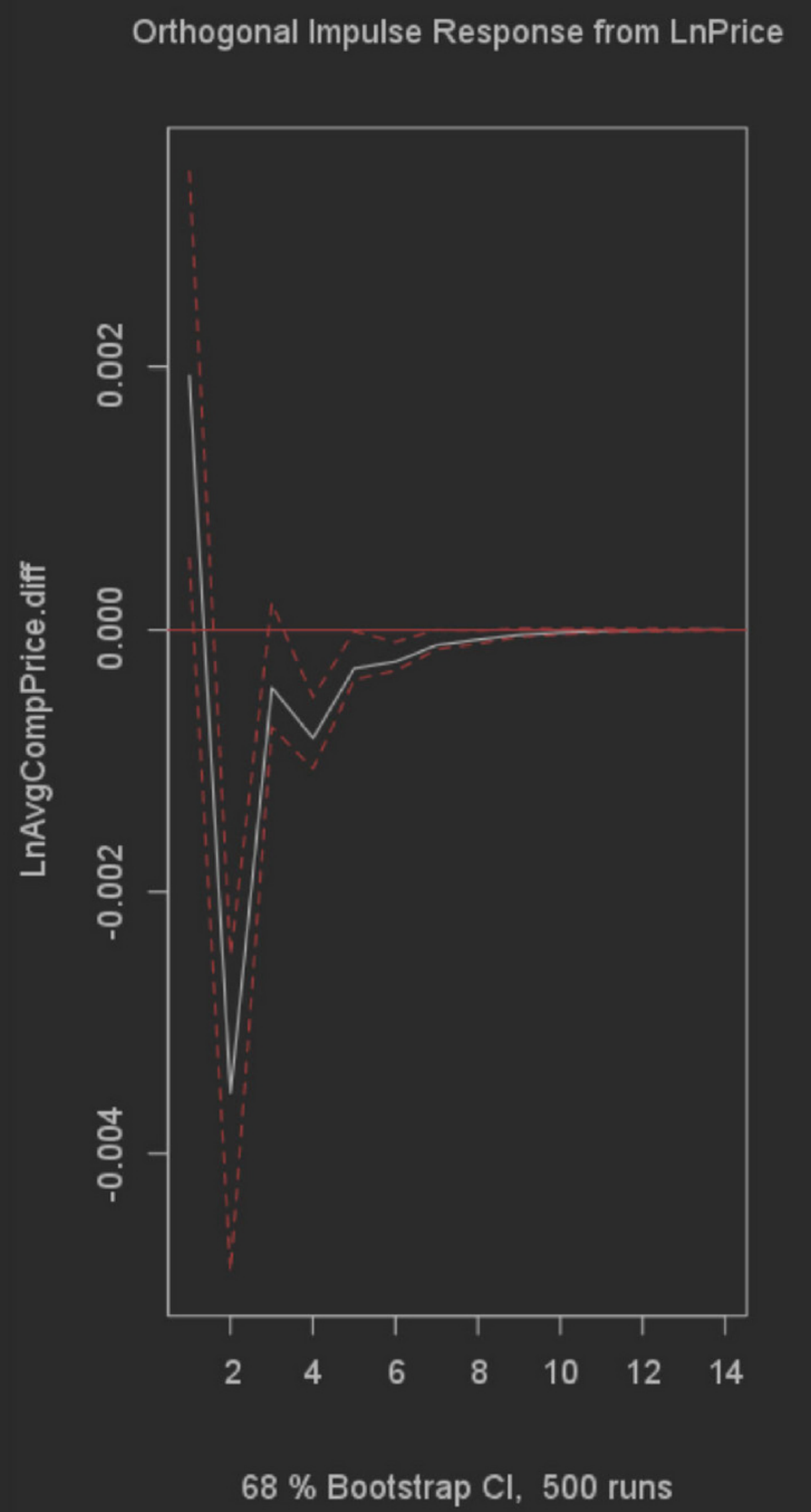


APPENDIX A-5.5.1

IRF PLOTS: ACP IMMEDIATE

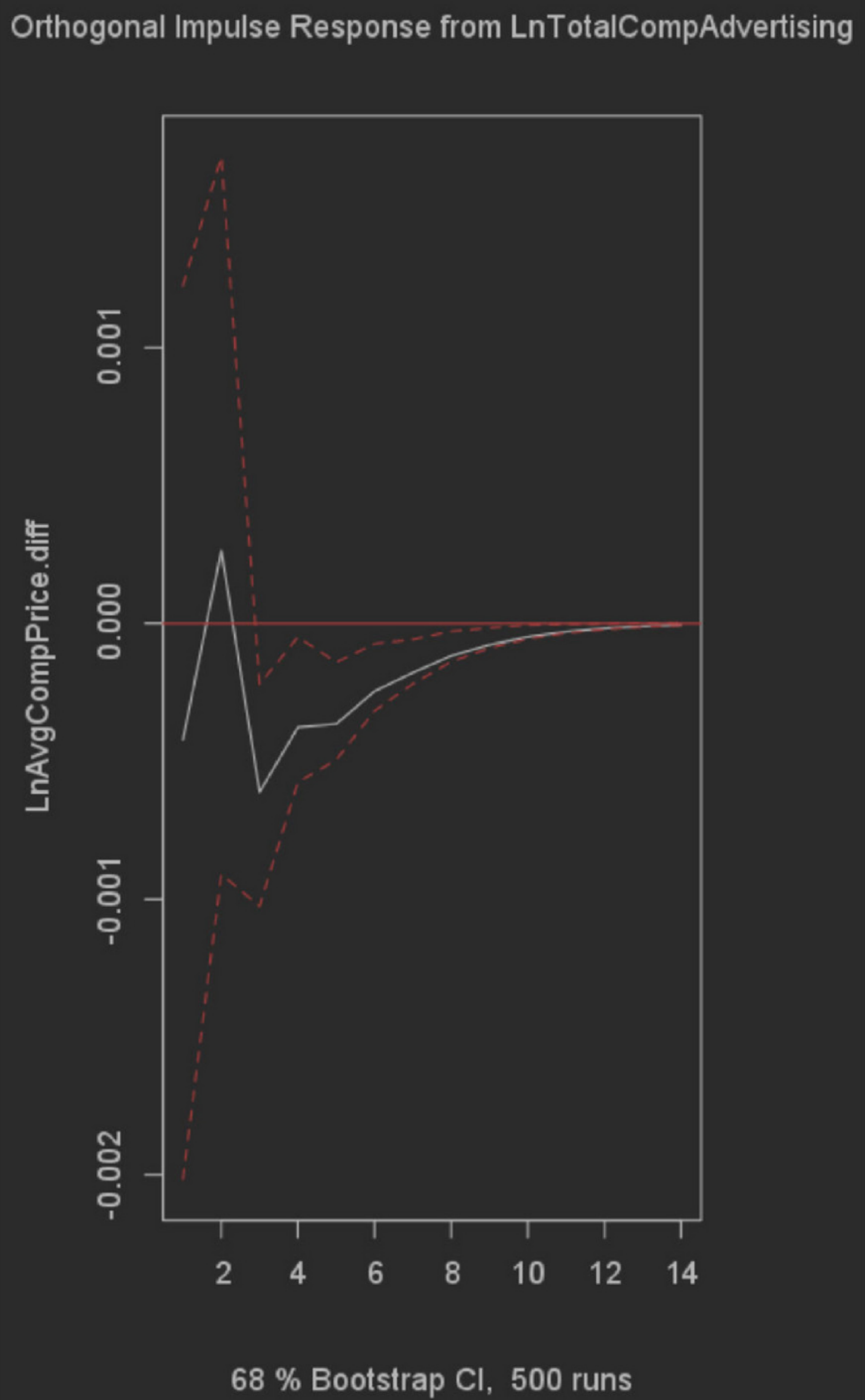






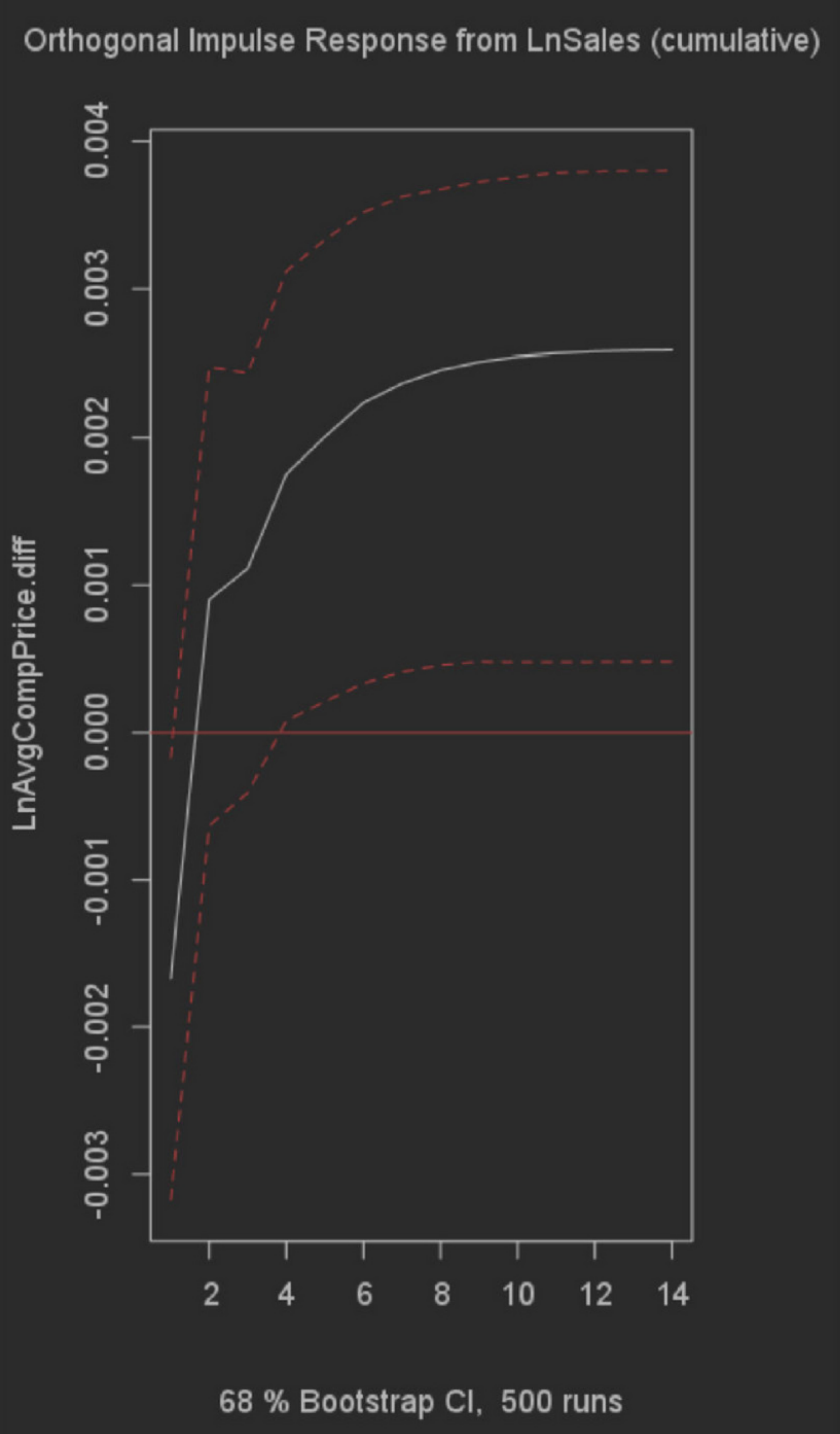
APPENDIX A-5.5.2

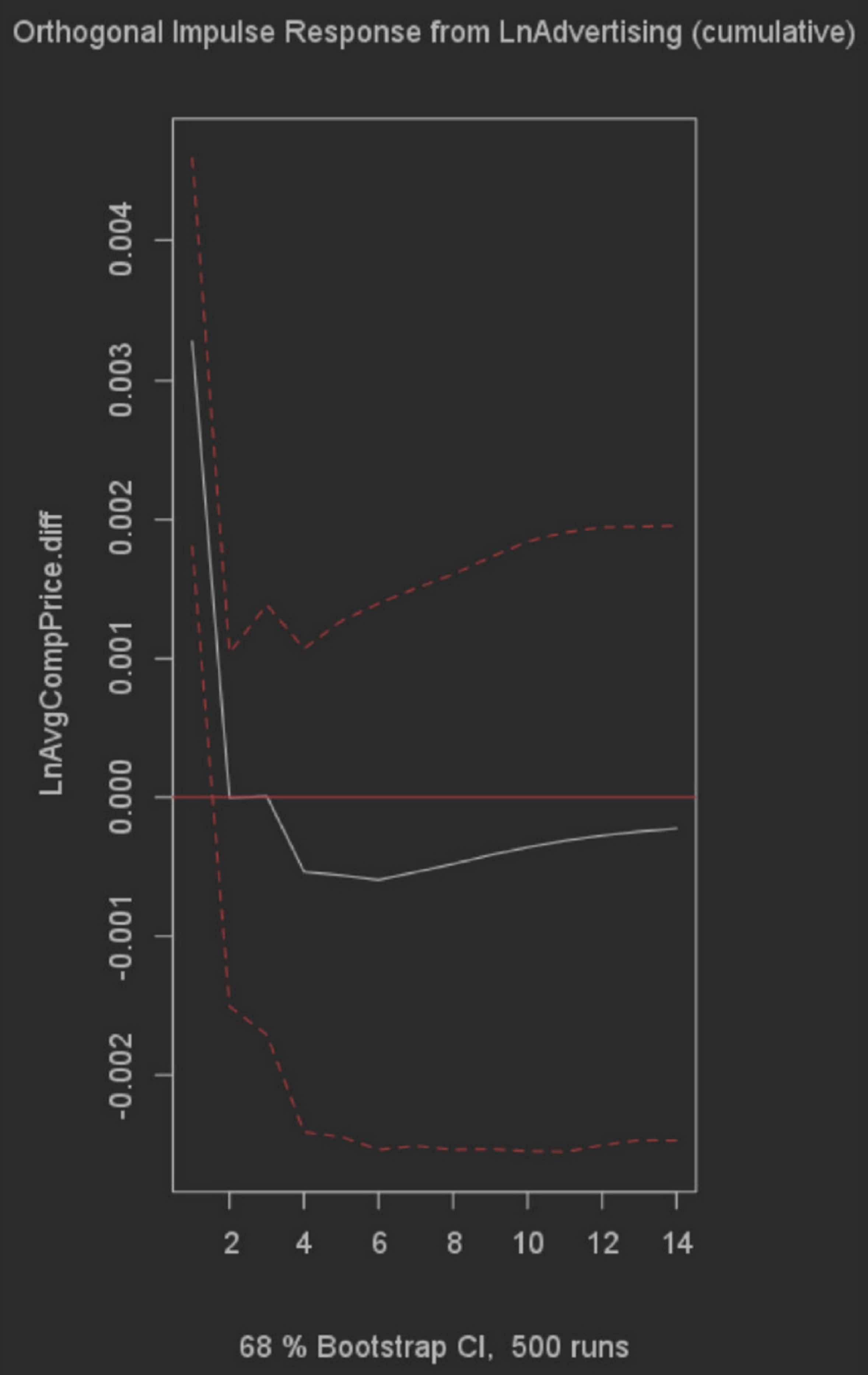
IRF PLOTS: ACP IMMEDIATE

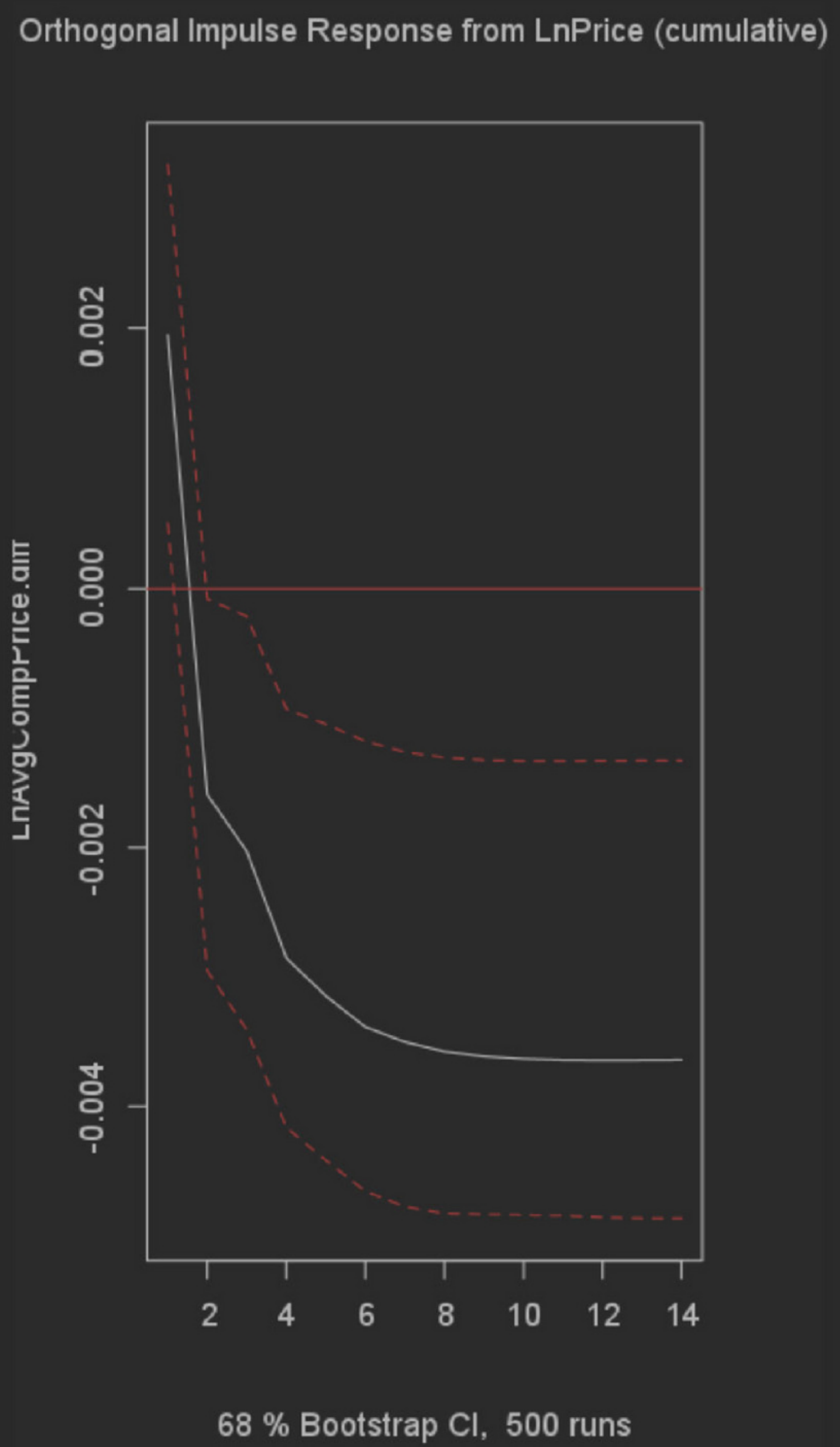


APPENDIX A-5.5.3

IRF PLOTS: ACP CUMULATIVE

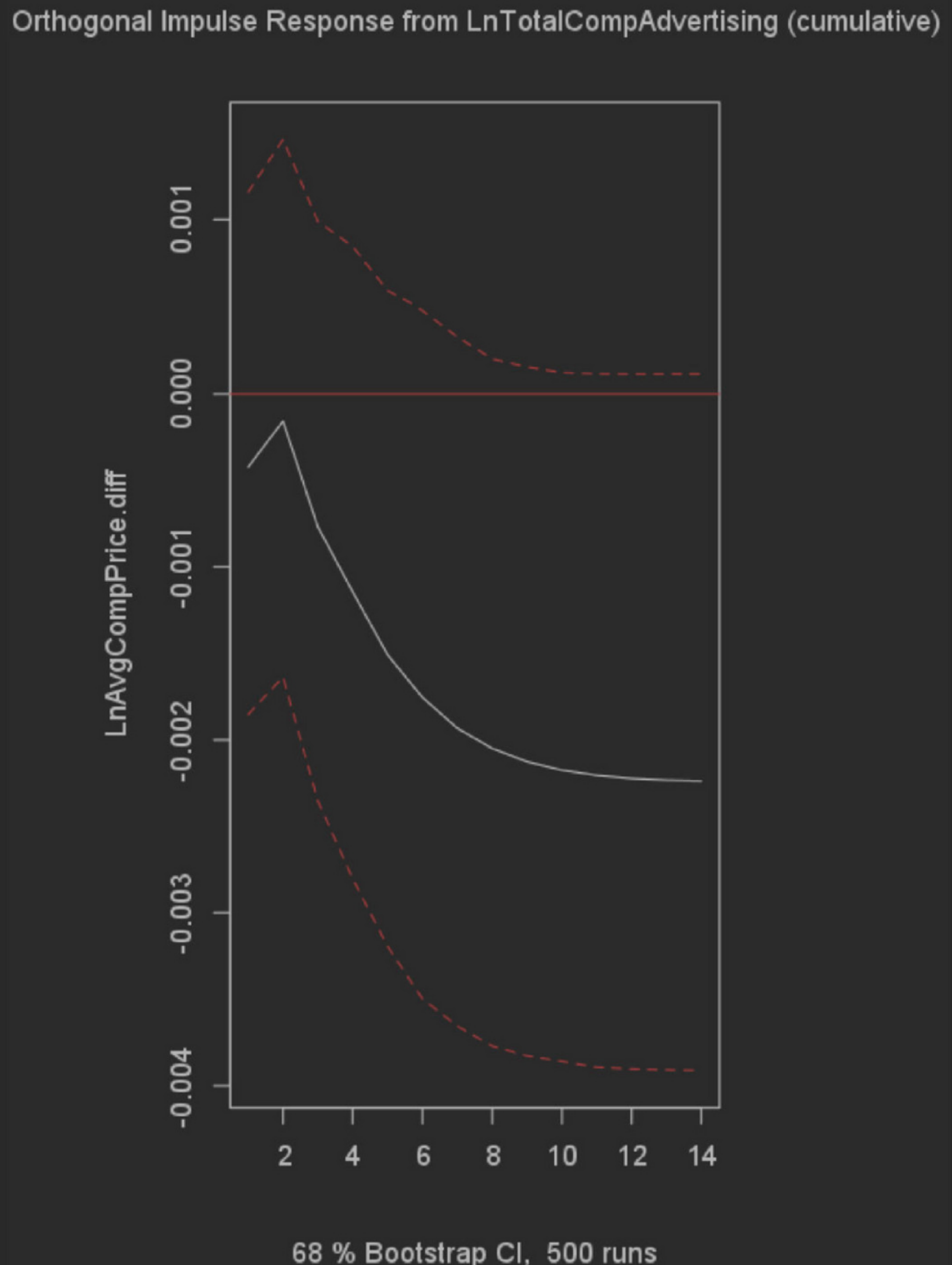


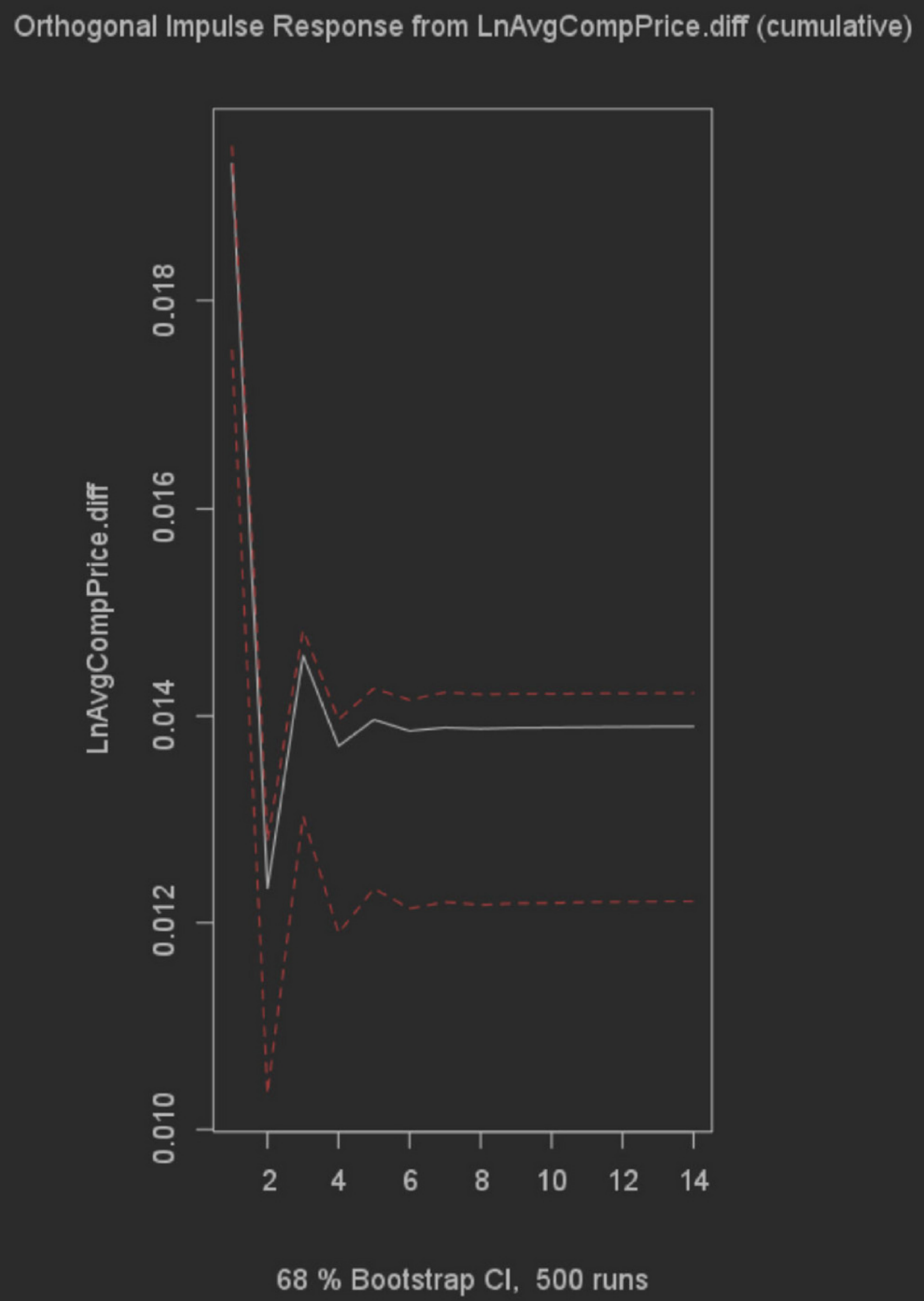




APPENDIX A-5.5.4

IRF PLOTS: ACP CUMULATIVE





APPENDIX A-6.1

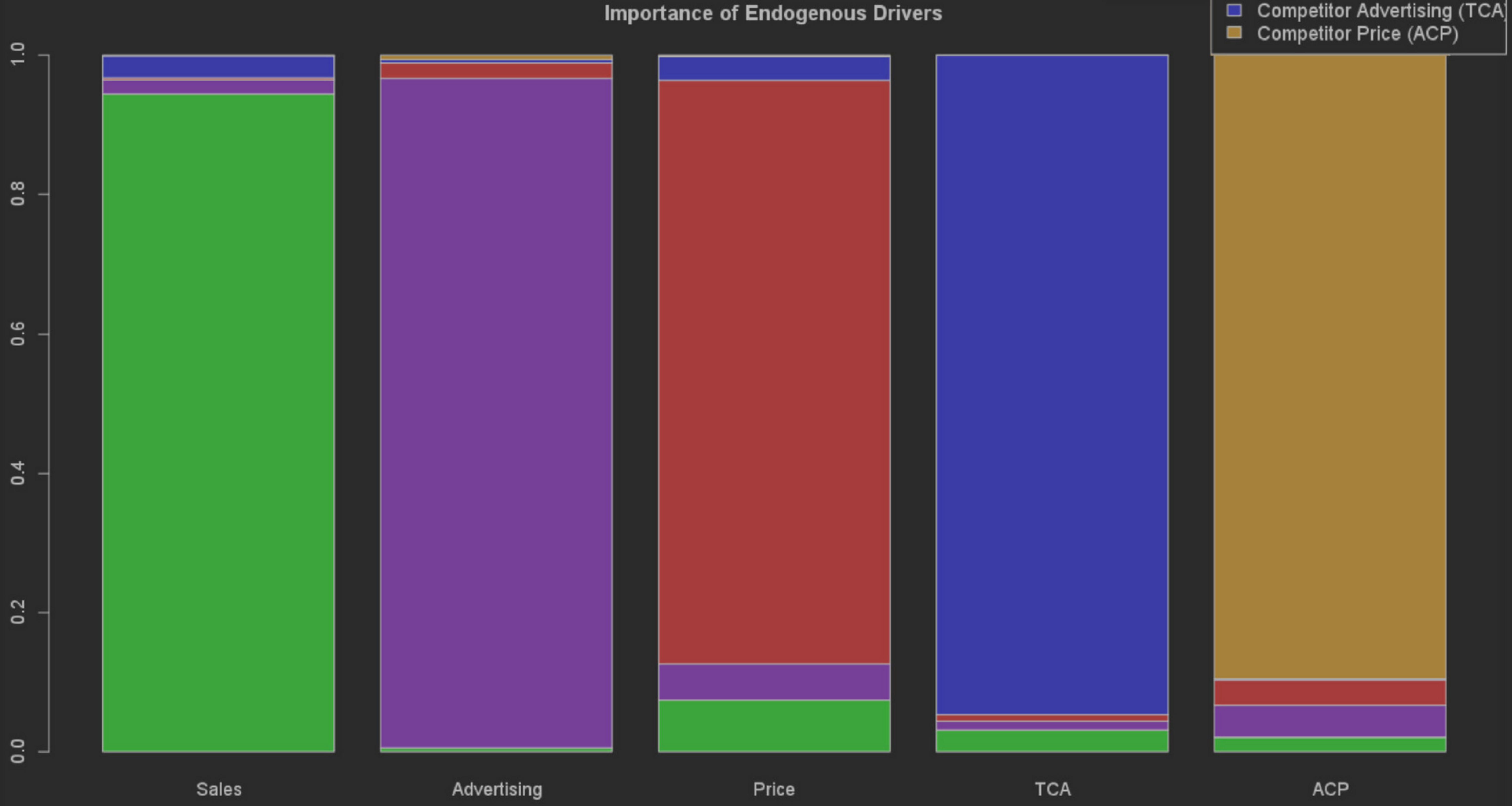
FEVD MODEL: TABLE

```r
bt2 <- bt2[,c(13,26,39,52,65)]
colnames(bt2) <- c("Sales", "Advertising", "Price", "Competitor Advertising (TCA)", "Competitor Price (ACP)")

#Look at % figures
bt2percent <- bt2 * 100
bt2percent
#                          Sales       Advertising    Price       Competitor Advertising (TCA)    Competitor Price (ACP)
#LnSales                   94.43140074   0.5513635    7.4029357              3.106185898                2.0938992
#LnAdvertising             2.01284178   96.1540349    5.2218000              1.254972571                4.5821679
#LnPrice                   0.29149036    2.2177624   83.7388025              0.998105220                3.6209289
#LnTotalCompAdvertising    3.17119314    0.4602686    3.4619734             94.633836358                0.2114916
#LnAvgCompPrice.diff       0.09307399    0.6165705    0.1744884              0.006899953               89.4915124
```

APPENDIX A-6.1

FEVD MODEL: PLOT

```r
barplot(bt2,  col = c("Green", "Purple", "Red", "Blue", "Orange"),
        main="Importance of Endogenous Drivers",
        names.arg = c("Sales","Advertising","Price","TCA","ACP"))
legend( x: "topright",
       legend = c("Sales", "Advertising", "Price", "Competitor Advertising (TCA)", "Competitor Price (ACP)"),
```



APPENDIX B

The R script behind the raw data.

R scripts available upon formal request.