SALES FORECASTING

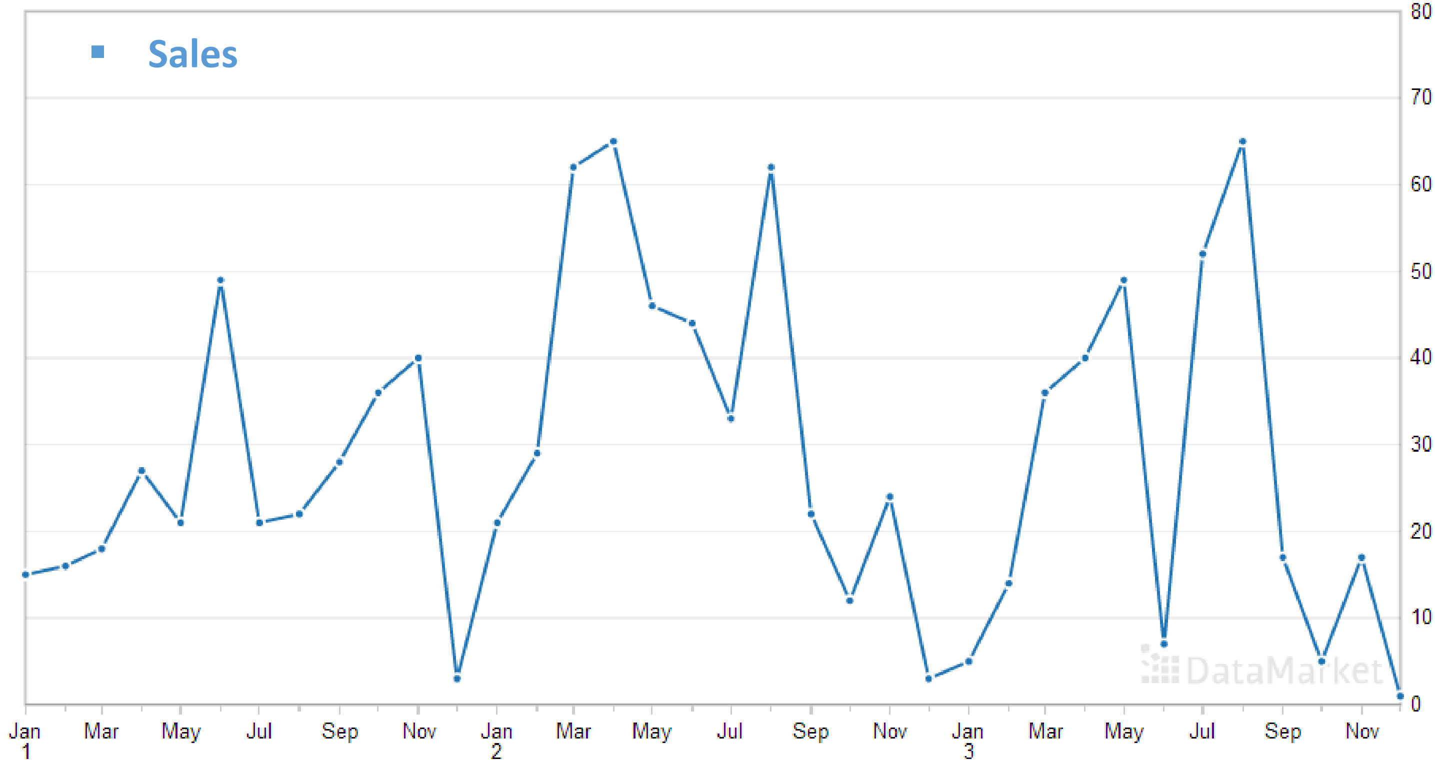
Yen-lin Lin

Mar/19/2015

Business Understanding:
Forecast monthly sales of a dietary weight control product

Data Description: 36 consecutive **monthly sales** of a dietary weight control product

Goal: Apply different methods to forecast monthly sales of a dietary weight control product



https://datamarket.com/data/set/22kw/advertising-and-sales-data-36-consecutive-monthly-sales-and-advertising-expenditures-of-a-dietary-weight-control-product#!ds=22kw!2ekl&display=line

Business Understanding:

Sales Forecasting. Why is it necessary?

To increase the profitability.

To increase the revenue.

To retain more and more customer.

To establish capacity and output levels

To raise the necessary cash for operations

To acquire and stock the right amount of supplies

To hire the required number of people

Business Understanding

Data Understanding

Modeling

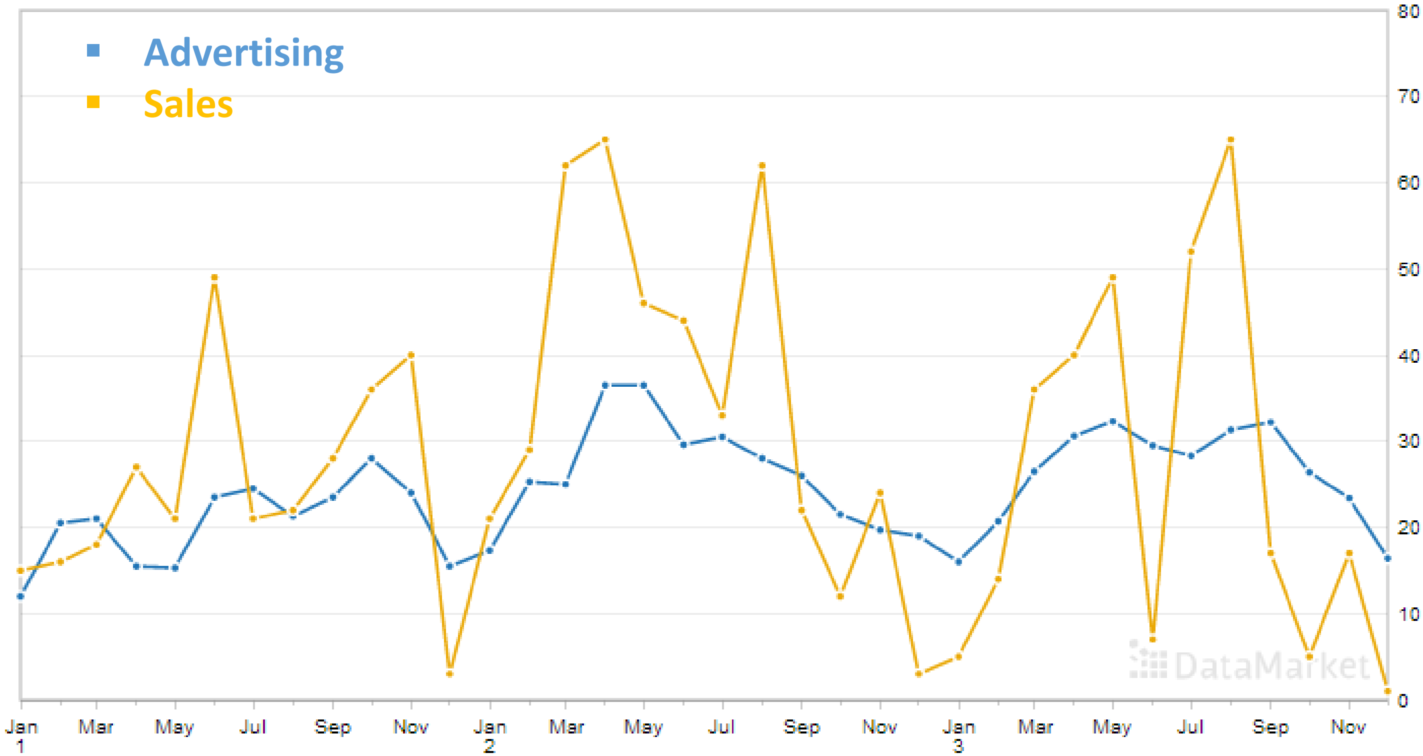
Model Evaluation

Forecast

Data Understanding:
Description of Data

Advertising and Sales data:

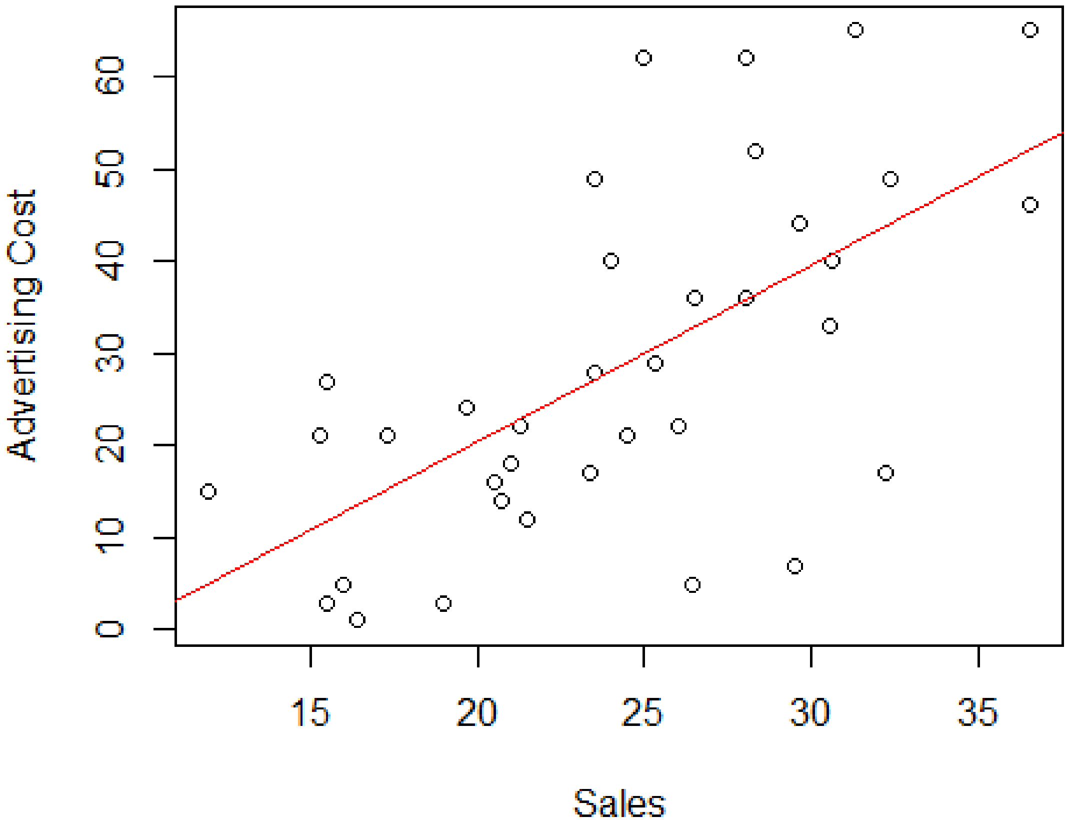
monthly sales and advertising expenditures of a dietary weight control product



https://datamarket.com/data/set/22kw/advertising-and-sales-data-36-consecutive-monthly-sales-and-advertising-expenditures-of-a-dietary-weight-control-product#!ds=22kw!2ekl&display=line

Exploratory Data Analysis:

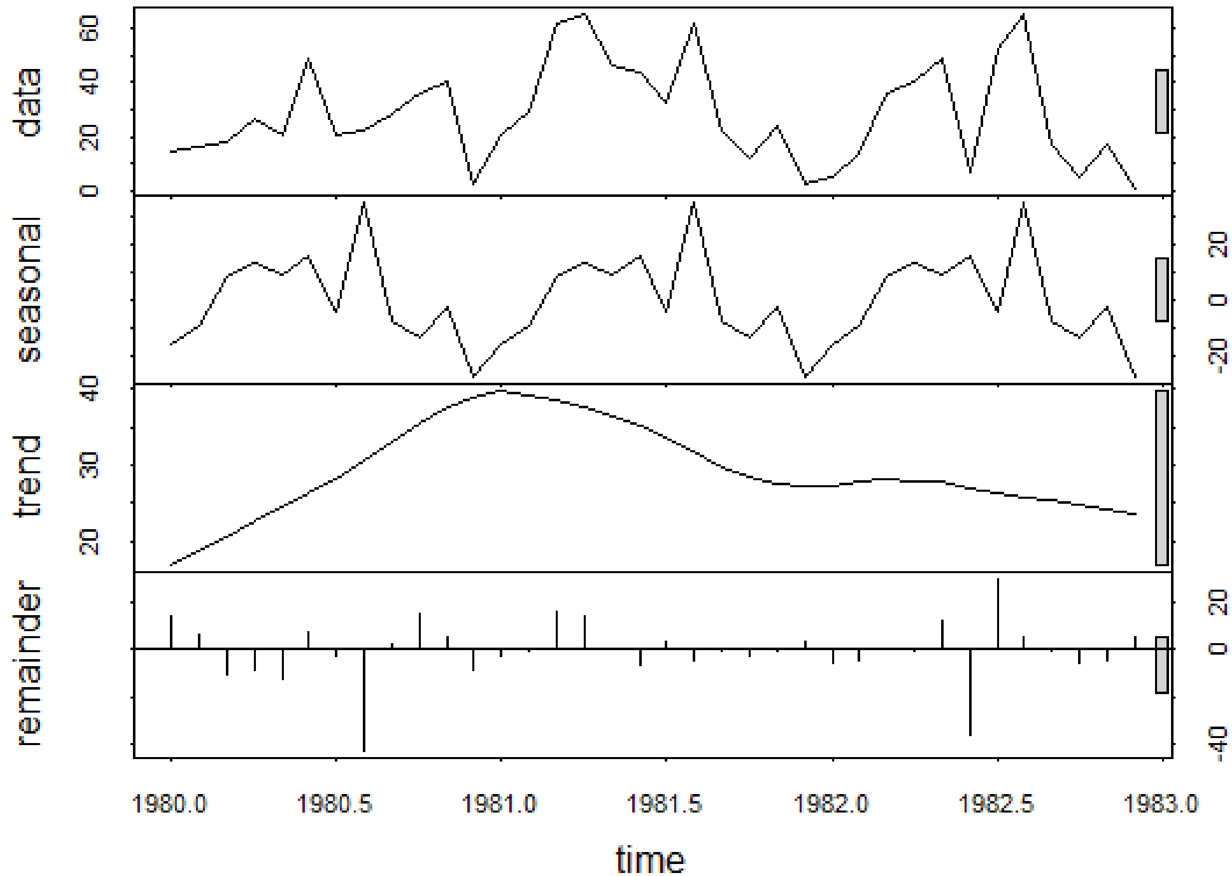
Sales and Advertising are positively correlated (*r* = 0.63)

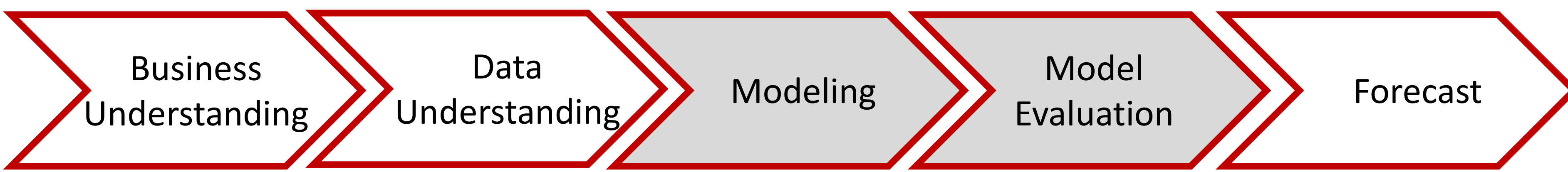


***Corr.*=0.63**

Exploratory Data Analysis:
Sales shows seasonality and trend

- Seasonal Decomposition of Sales:




Business Understanding
Data Understanding
Modeling
Model Evaluation
Forecast

Modeling:

Steps for generating models for forecasting Sales

- Select Modeling Techniques
- Build Model Parameter Settings
- Model Selection Criteria
- Model Evaluation

Modeling:

Steps for generating models for forecasting Sales

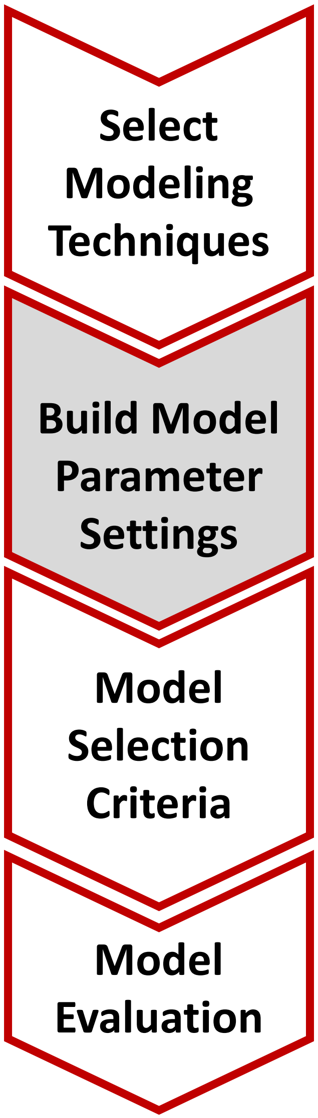
- Select Modeling Techniques
- Build Model Parameter Settings
- Model Selection Criteria
- Model Evaluation

- **Modeling Assumption**
  - Linear relationship between Sales and Advertising
  - Seasonal pattern in Sales

- **Modeling Technique**
  - Linear regression (Sales ~ Advertising) with random forest method
  - Linear regression (Sales ~ Advertising + Month) with random forest approach
  - Regression with ARIMA errors
  - Seasonal ARIMA
  - SVM (machine learning)
  - Ensemble models

Modeling:

Steps for generating models for forecasting Sales



- **Modeling Assumption**
  - Linear relationship between Sales and Advertising
  - Seasonal pattern in Sales

- **Modeling Technique**
  - Linear regression (Sales ~ Advertising) with random forest method
  - Linear regression (Sales ~ Advertising + Month) with random forest approach
  - Regression with ARIMA errors
  - Seasonal ARIMA
  - SVM (machine learning)
  - Ensemble models

Modeling:

Steps for generating models for forecasting Sales

Select Modeling Techniques

Build Model Parameter Settings

Model Selection Criteria

Model Evaluation

- **For sARIMA models or regression w/ ARIMA error:**

  Step a: run auto.arima() in R

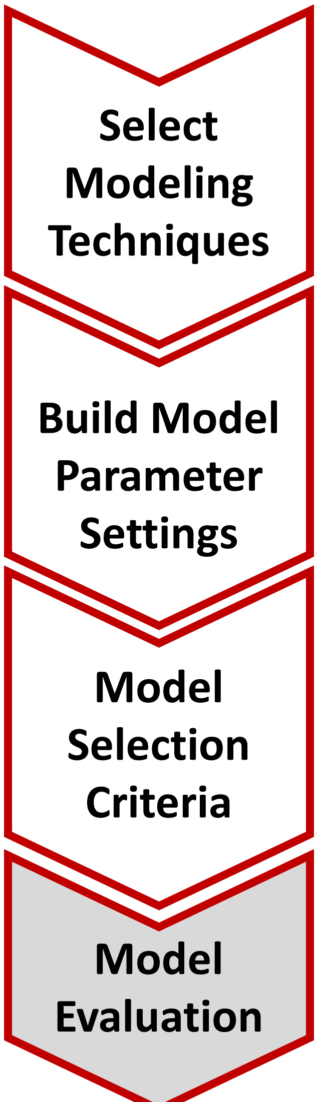
  Step b: sequentially change the p and q parameters of the auto.arima result

  Step c: compute and compare AICc and BIC values of models as well as the RMSE of the fitted values

  Step d: select the model with the lowest AICc, BIC, and RMSE values

Modeling:

Steps for generating models for forecasting Sales



- **Residual Diagnosis**
  ACF and PACF
  Ljung-Box test
  Durbin-Watson test

- **Time series cross-validation (TS-CV)**
  Forecast evaluation with a rolling origin:

  Step a: define and select training window of flexible length (start from size=18 and then add 1 datapoint afterward)

  Step b: build model using the data in the training window

  Step c: forecast the next 12 monthly data

  Step d: compare 1-step, 2-step, …, 12-step forecasts using RMSE

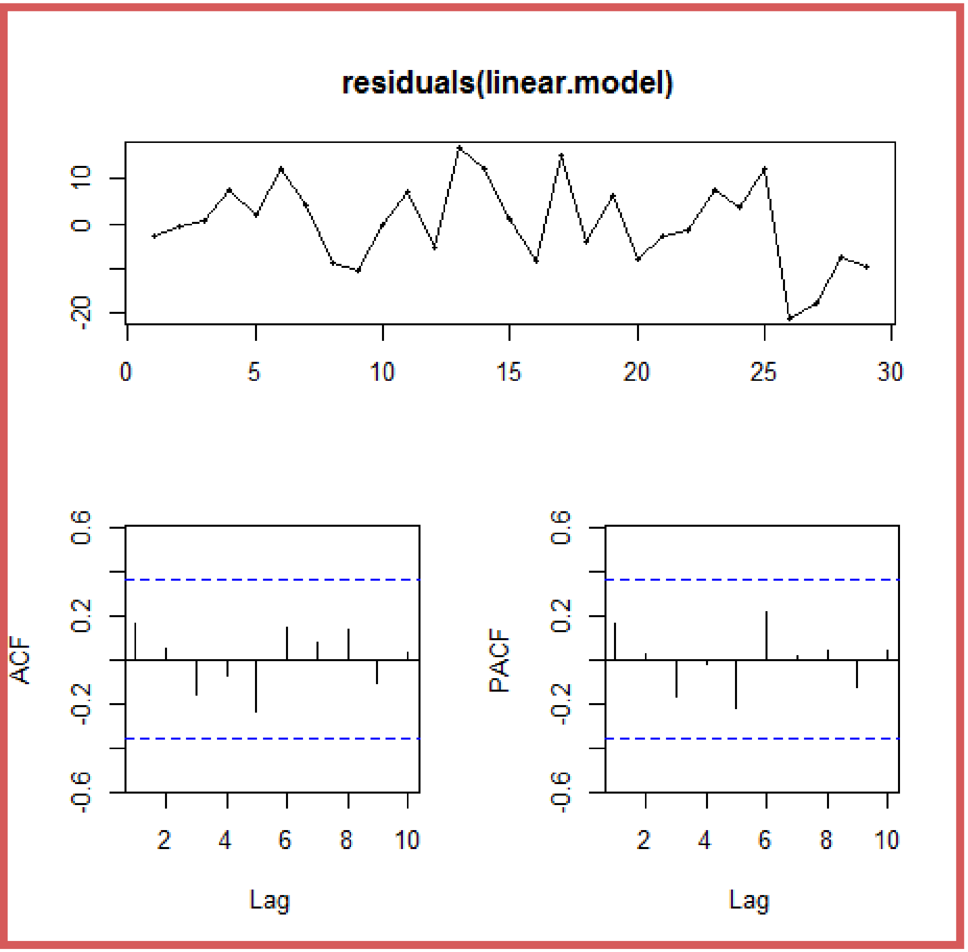
Model 1: Linear Regression (Sales ~ Advertising)

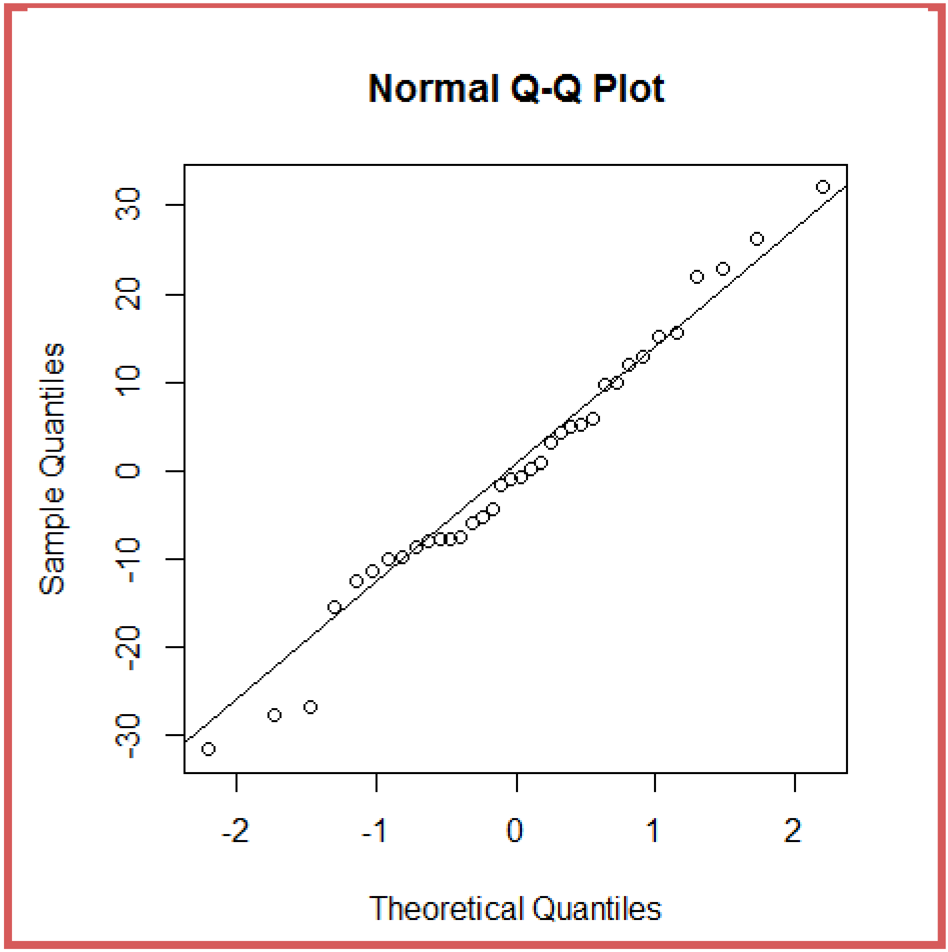
R Output of linear regression model

```
##
## Call:
## lm(formula = Sales ~ Advertising, data = data)
##
## Residuals:
##     Min       1Q   Median       3Q      Max
## -31.593  -8.186  -0.783   9.727  32.039
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -17.992     10.102  -1.781   0.0839 .
## Advertising    1.918      0.404   4.748 3.64e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1
##
## Residual standard error: 14.77 on 34 degrees of freedom
## Multiple R-squared:  0.3987, Adjusted R-squared:  0.381
## F-statistic: 22.54 on 1 and 34 DF,  p-value: 3.635e-05
```

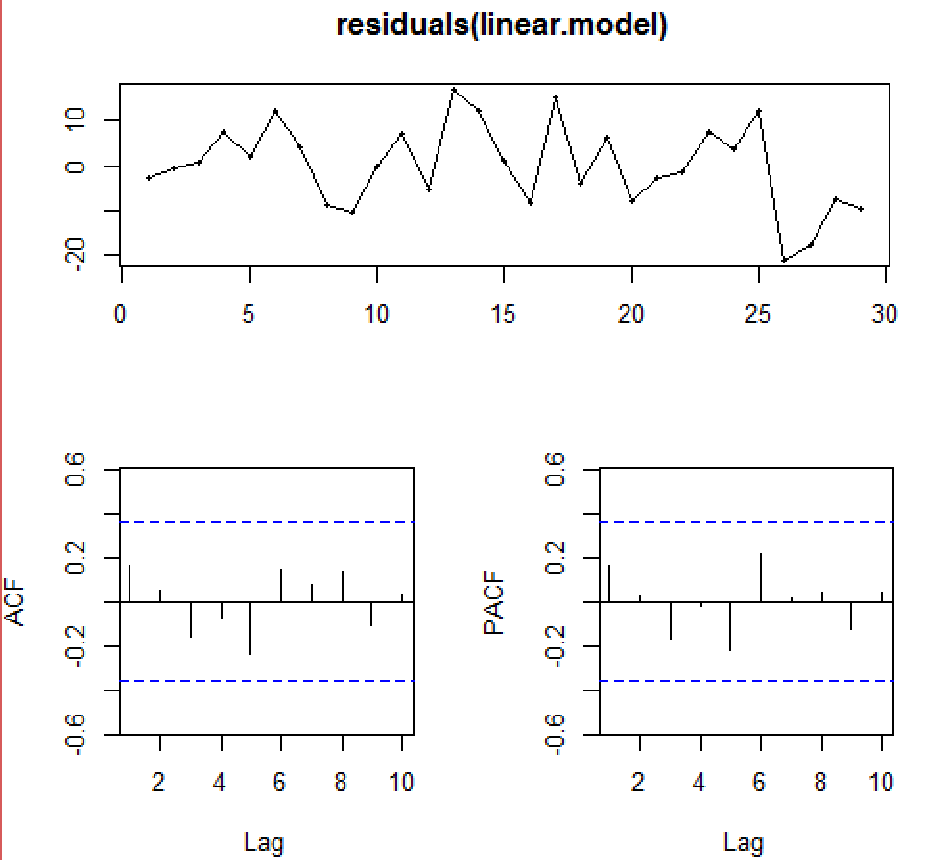
Model 1: Linear Regression (Sales ~ Advertising)

Residual diagnosis of the linear regression model





Model 1: Linear Regression (Sales ~ Advertising)
Residual diagnosis of the linear regression model



- **Ljung-Box test:**

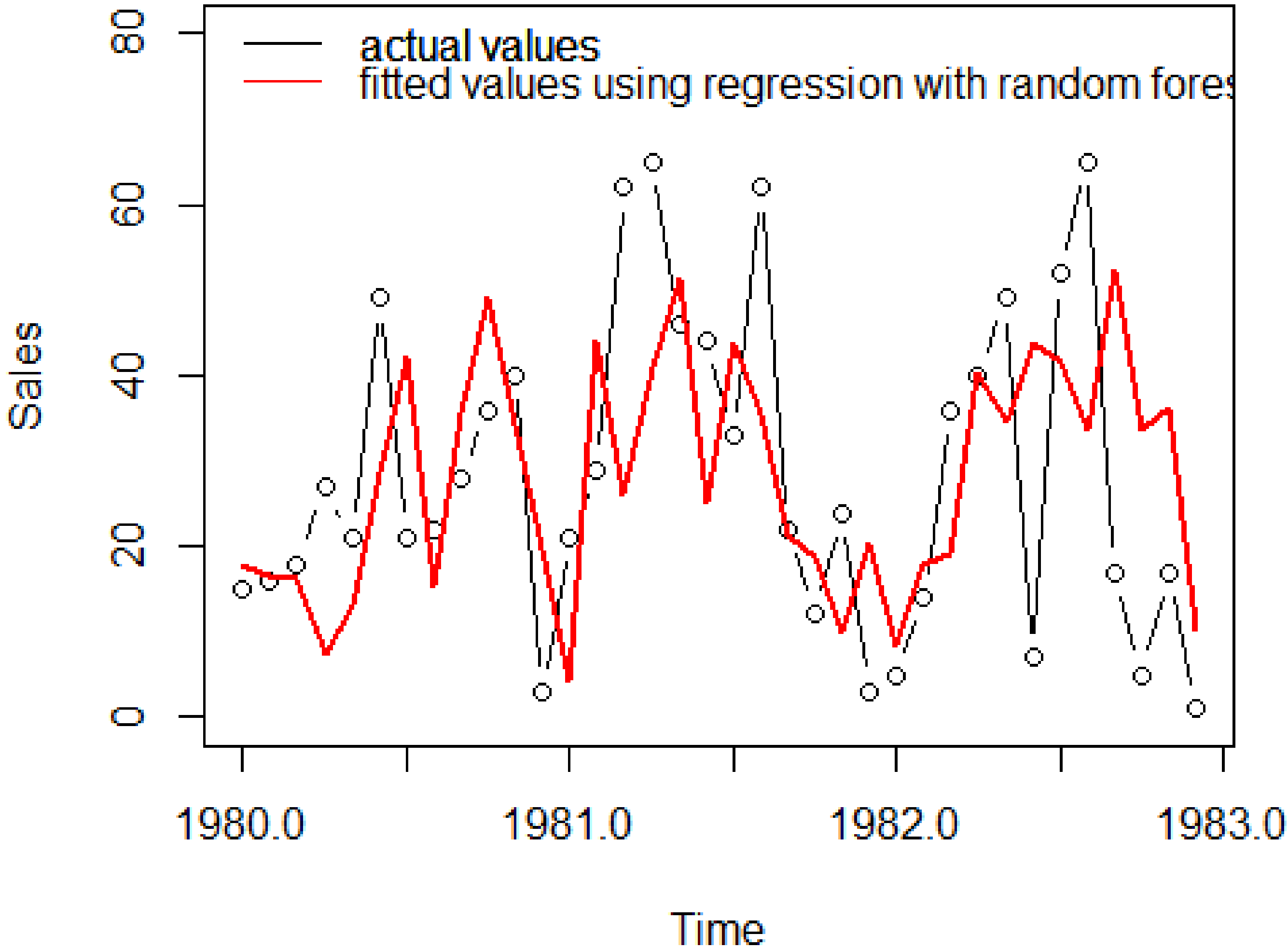
```
>Box.test(residuals(linear.model),fitdf=2,lag=20,type="Ljung") Box-Ljung test data: residuals(linear.model) X-squared = 18.557, df = 18, p-value = 0.4196
```

- **Durbin-Watson test**:

```
> durbinWatsonTest(linear.model, alternative=c("two.sided")) lag Autocorrelation D-W Statistic p-value 1 0.009233116 1.947187 0.77 Alternative hypothesis: rho != 0
```

Model 1: Linear Regression (Sales ~ Advertising) with random forest

Performance of model 1



Model 1: Linear Regression (Sales ~ Advertising) with random forest

RMSE for Forecast of Sales

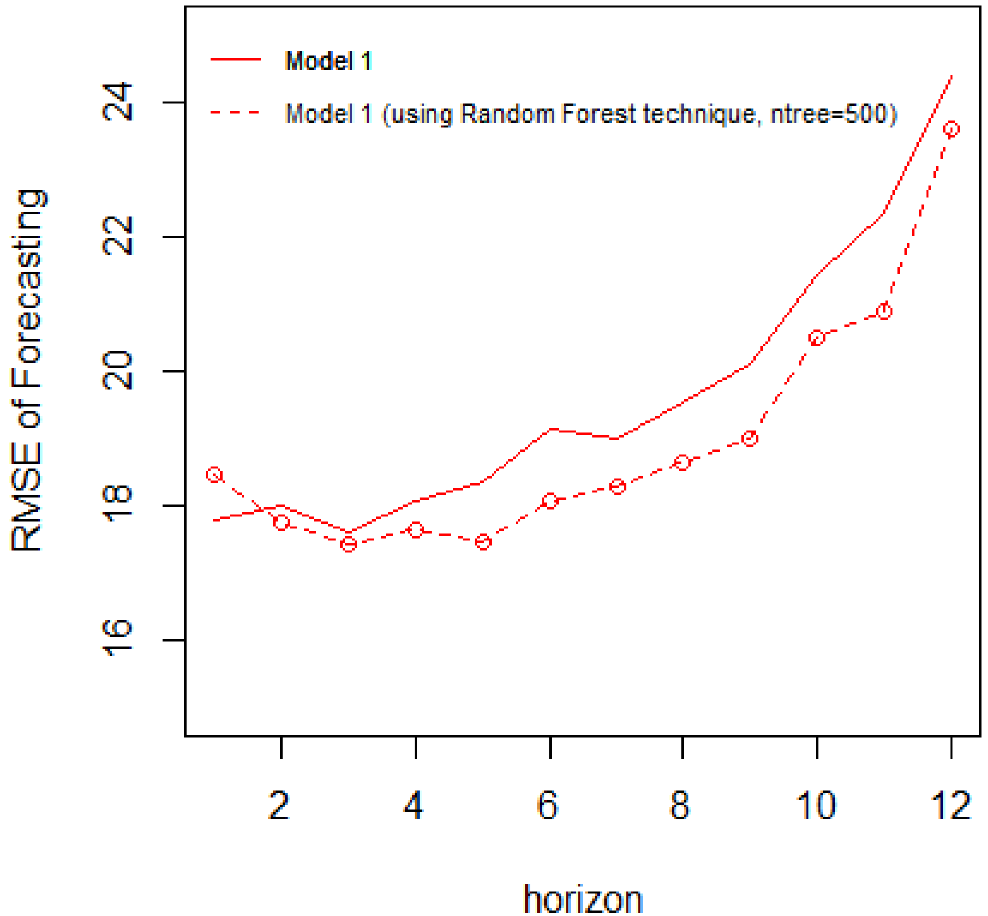
Regression Model 1 (with/without random forest method)

↓

Time Series Cross Validation **(TS-CV)**

↓

Calculate **RMSE** for the prediction of Sales



Model 2: Linear Regression (Sales ~ Advertising + Month)

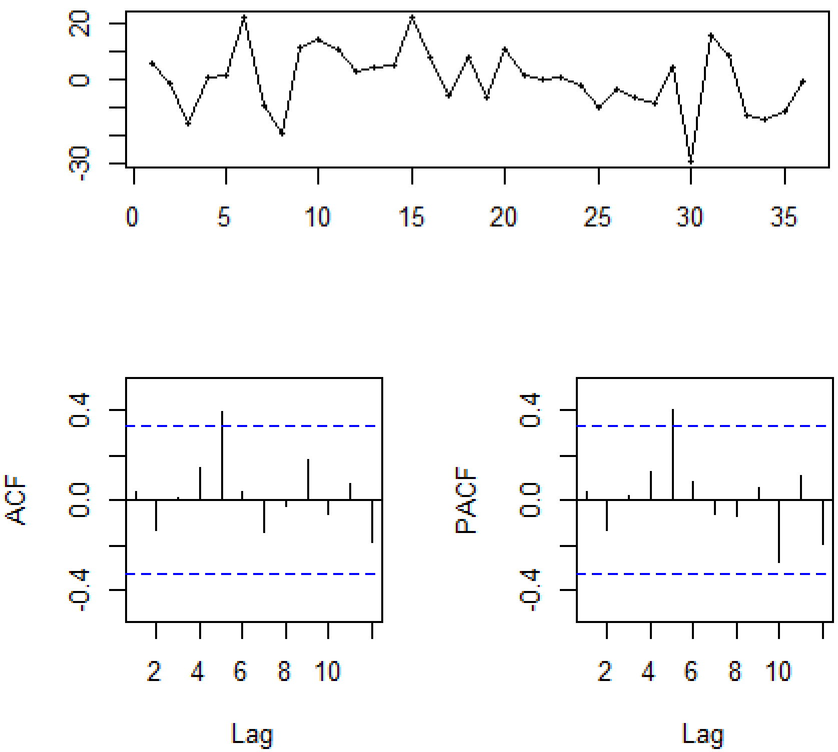
R Output of Model 2

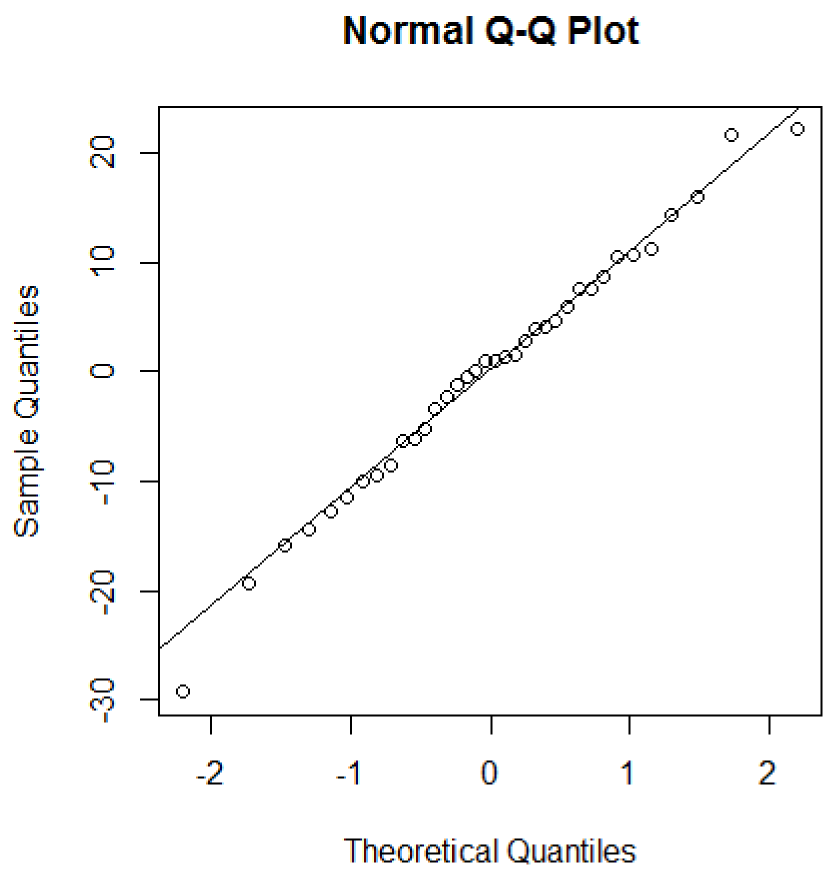
```
##
## Call:
## glm(formula = Sales ~ Advertising + factor(Month), data = data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -29.279   -6.969    1.008    7.571   22.085
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)      -8.9493    11.2429  -0.796  0.43417
## Advertising       1.4977     0.5219   2.870  0.00866 **
## factor(Month)2   -4.5841    11.9243  -0.384  0.70419
## factor(Month)3   11.4204    12.2874   0.929  0.36231
## factor(Month)4   11.7114    13.0651   0.896  0.37933
## factor(Month)5    5.6292    13.1966   0.427  0.67367
## factor(Month)6    1.0447    13.0651   0.080  0.93696
## factor(Month)7    2.6952    13.1260   0.205  0.83912
## factor(Month)8   18.3765    12.8958   1.425  0.16759
## factor(Month)9   -9.5060    12.9880  -0.732  0.47162
## factor(Month)10 -11.2770    12.5271  -0.900  0.37734
## factor(Month)11   2.4497    11.9570   0.205  0.83947
## factor(Month)12 -14.1291    11.3814  -1.241  0.22696
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 192.8808)
##
##     Null deviance: 12341.0  on 35  degrees of freedom
## Residual deviance:  4436.3  on 23  degrees of freedom
## AIC: 303.47
##
## Number of Fisher Scoring iterations: 2
```

Model 2: Linear Regression (Sales ~ Advertising + Month)

Residual diagnosis of Model 2

Model 2: (Sales ~ Advertising + **Month**)

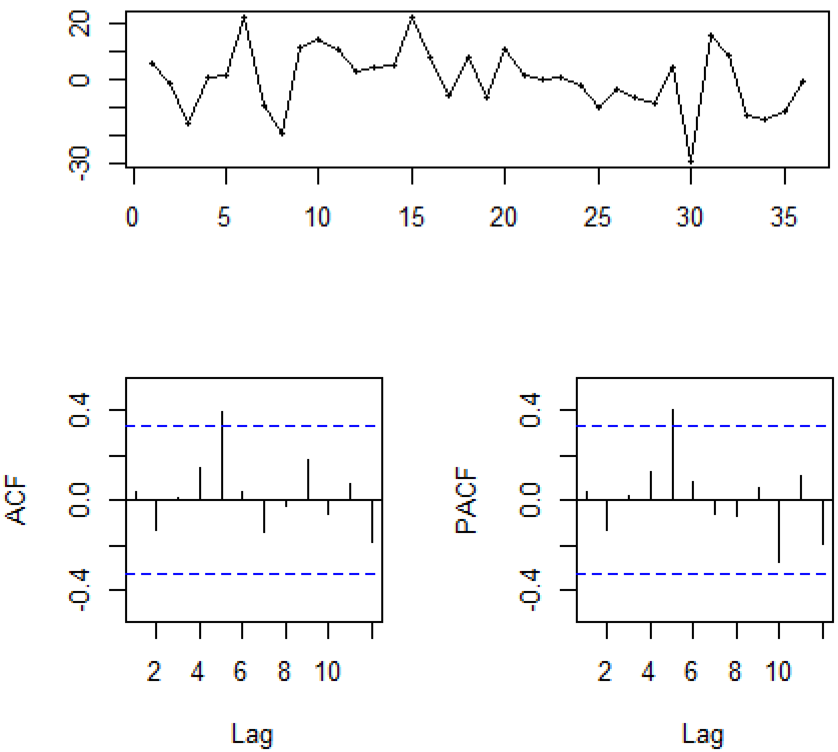




Model 2: Linear Regression (Sales ~ Advertising + Month)

Residual diagnosis of Model 2

Model 2: (Sales ~ Advertising + **Month**)



- **Ljung-Box test:**

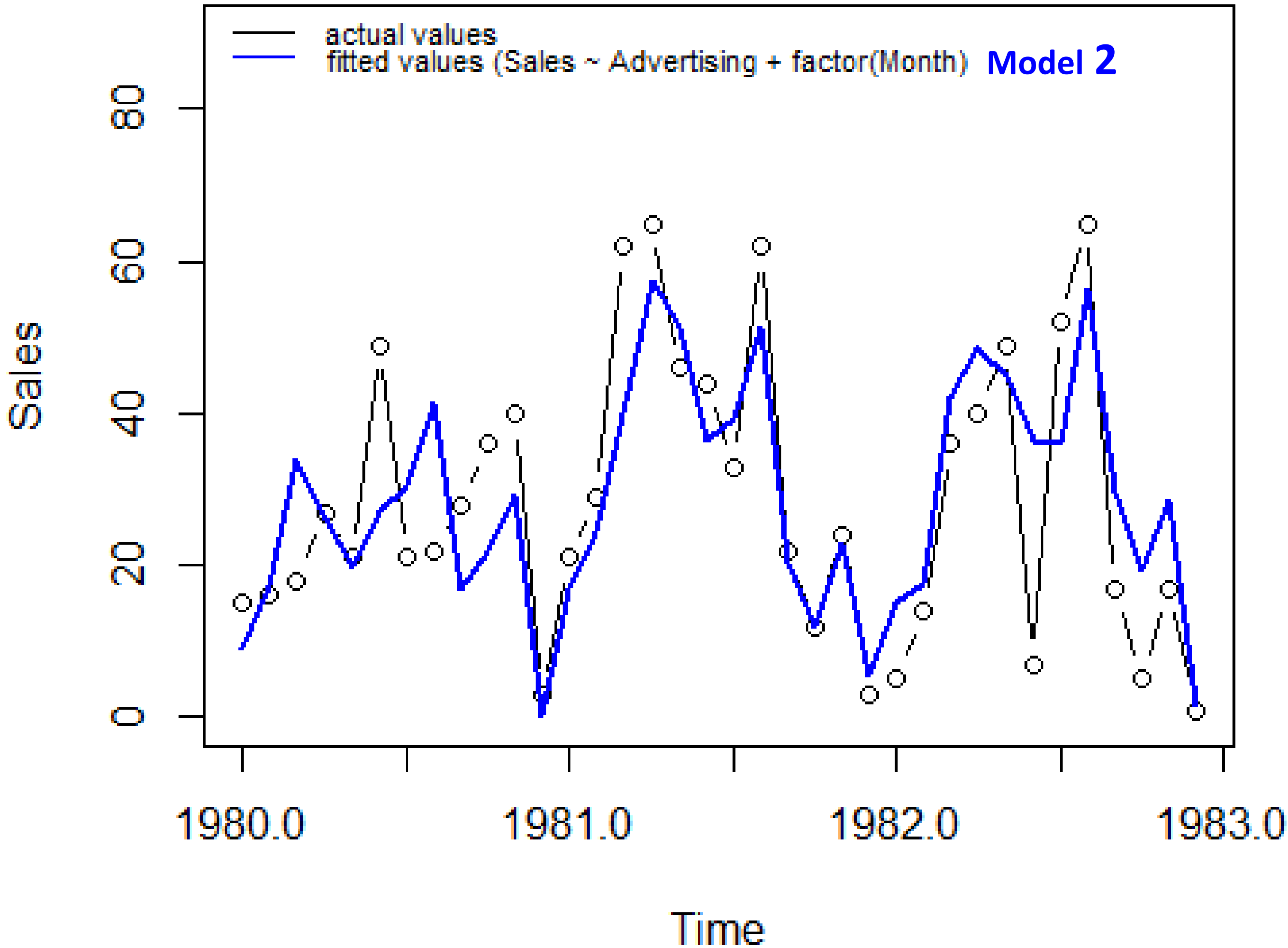
```
>Box.test(residuals(linear.model.glm),fitdf=
13,lag=20,type="Ljung") Box-Ljung test data:
residuals(linear.model.glm) X-squared =
21.6473, df = 7, p-value = 0.002921
```

- **Durbin-Watson test**:

```
> durbinWatsonTest(linear.model.glm,
alternative=c("two.sided")) lag
Autocorrelation D-W Statistic p-value 1
0.0325731 1.92675 0.832 Alternative
hypothesis: rho != 0
```

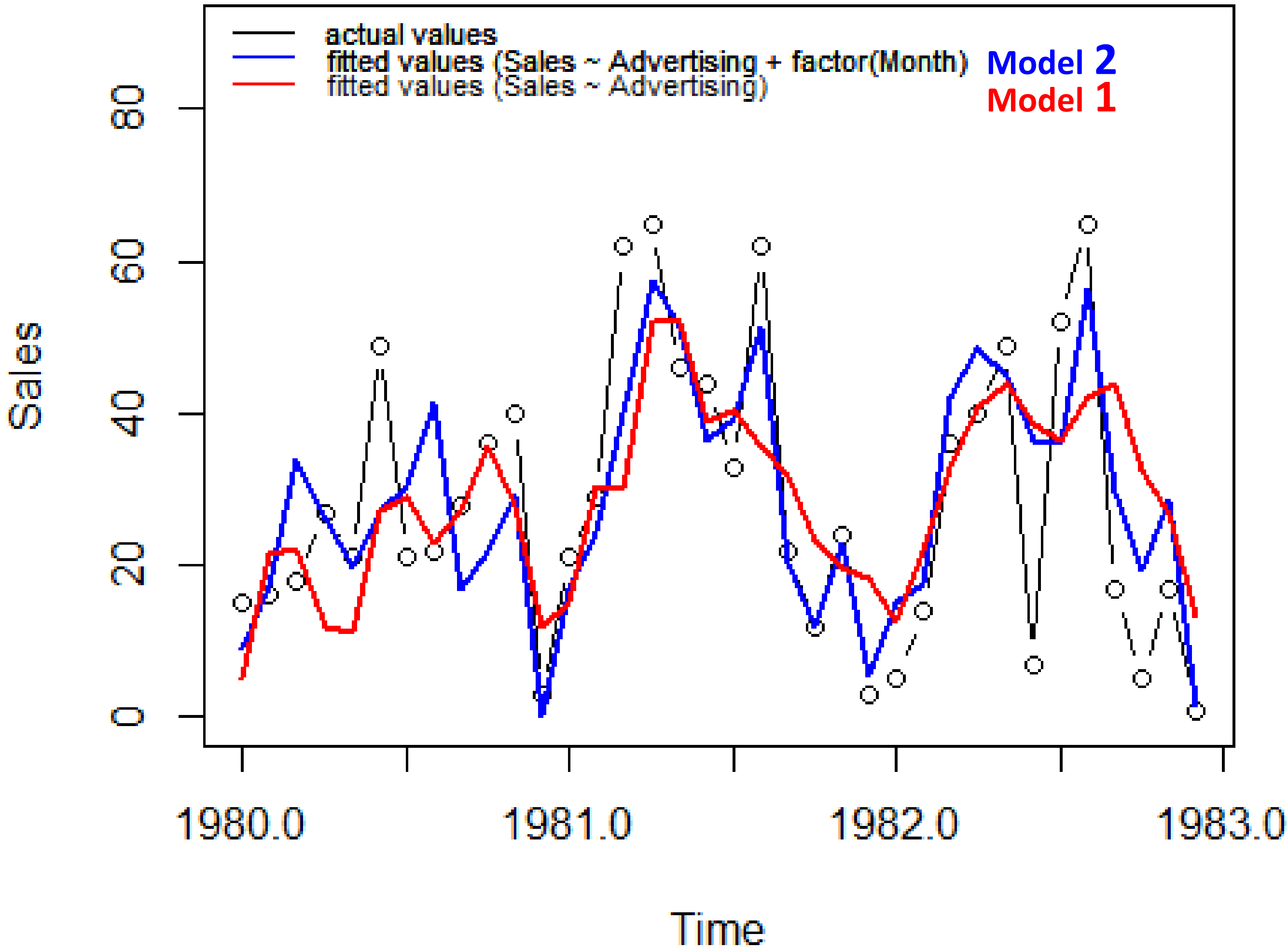
Model 2: Linear Regression (Sales ~ Advertising + Month) with random forest

Performance of Model 2



Model 2: Linear Regression (Sales ~ Advertising + Month) with random forest

Performance of Model 2



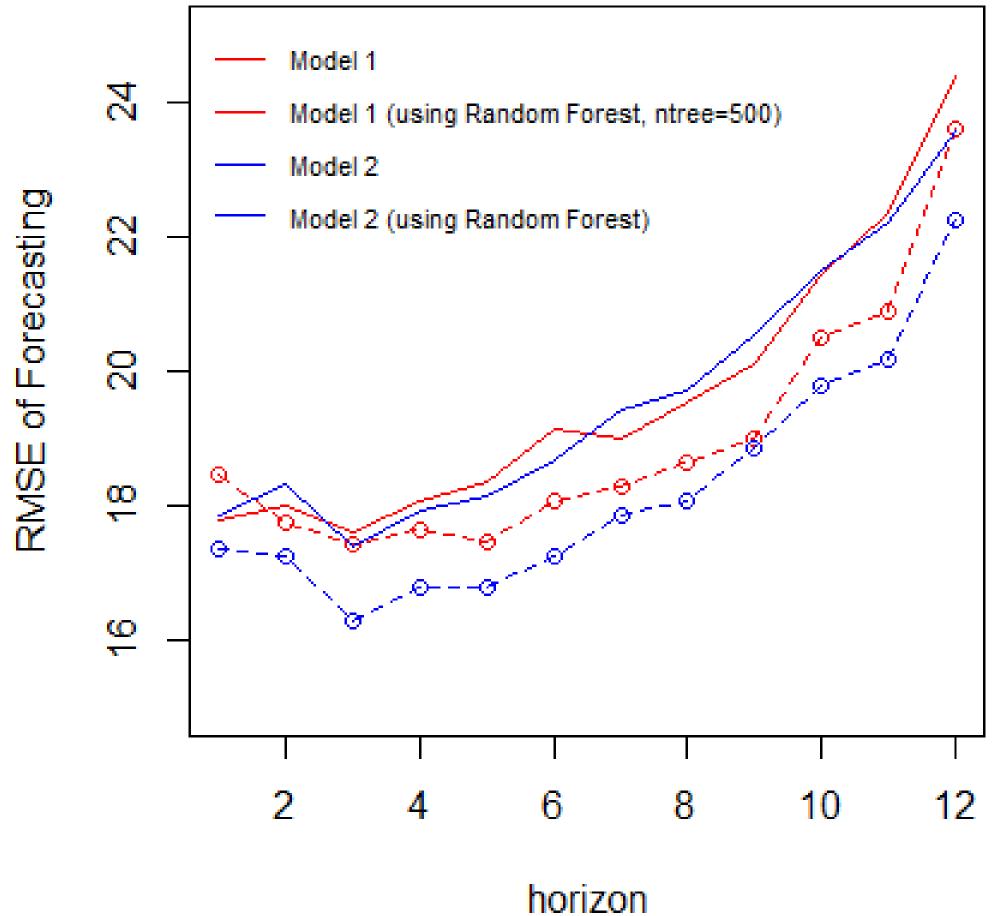
Model 2: Linear Regression (Sales ~ Advertising + Month)

RMSE for Forecast of Sales

Regression Model 2 with/without random forest

Time Series Cross Validation (TS-CV)

Calculate **RMSE** for the prediction of Sales



Model 3: Model 1 w/ ARIMA errors
Model Selection

Regression Model 1

↓

Fit residuals of Model 1 using auto.arima()

↓

Sequentially change the p and q parameters from the previous step

↓

Evaluate models based in AICc, BIC, and RMSE of fitted values

↓

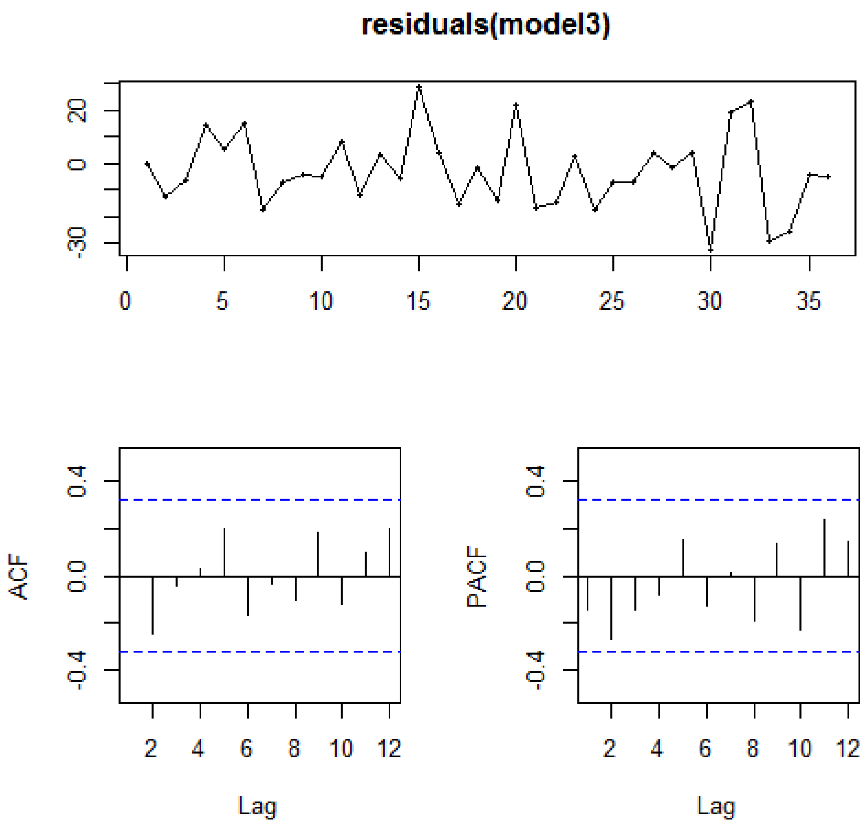
Check the performance of the selected model

Model 1: (Sales ~ Advertising)

Model 1 w/ ARIMA error	**AICc**	**BIC**	**RMSE**
ARIMA(0,1,1)	287.134	**291.111**	**252.542**
ARIMA(1,1,1)	289.388	294.389	265.444
ARIMA(1,1,2)	290.434	296.282	296.024
ARIMA(0,1,2)	288.908	293.908	298.393
ARIMA(2,1,1)	289.949	295.798	308.353
ARIMA(1,1,0)	300.650	304.626	357.606
ARIMA(1,1,3)	290.186	296.687	362.963
ARIMA(0,1,3)	**286.846**	292.694	378.282
ARIMA(2,1,2)	289.057	295.558	381.550
ARIMA(2,1,3)	289.909	296.845	402.262
ARIMA(2,1,0)	296.478	301.479	419.117
ARIMA(0,1,0)	305.150	307.942	490.398

Model 3: Model 1 w/ ARIMA(0,1,1) errors

Residual diagnosis of Model 3



- **Ljung-Box test:**

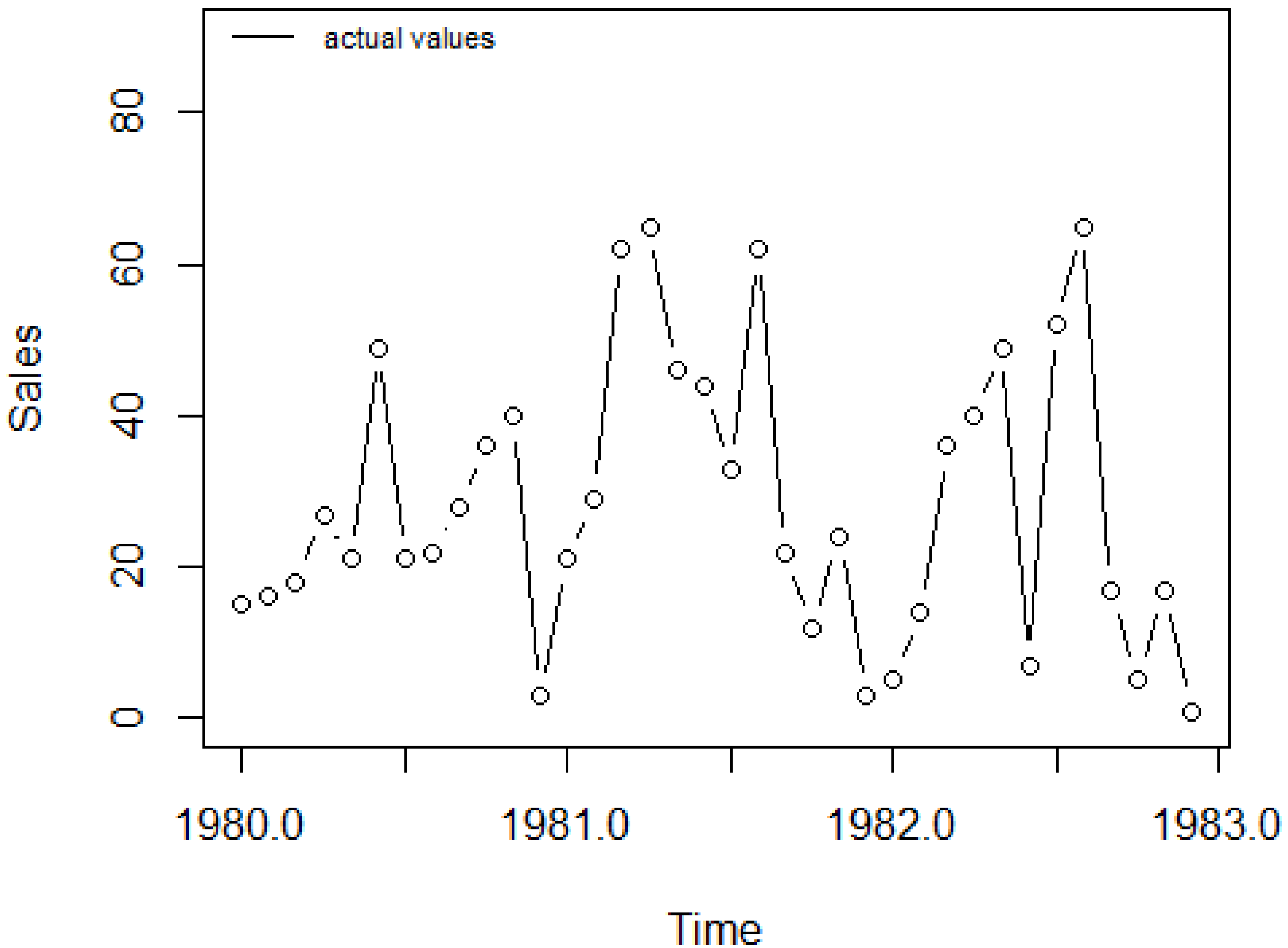
```
>
Box.test(residuals(model3),fitdf=4,lag=20,type="Ljung") Box-Ljung test data:
residuals(model3) X-squared = 21.8558, df =
16, p-value = 0.1479
```

- **Durbin-Watson test**:

```
>durbinWatsonTest(as.vector(model3$residuals
), alternative=c("two.sided"))
[1] 2.187782
```

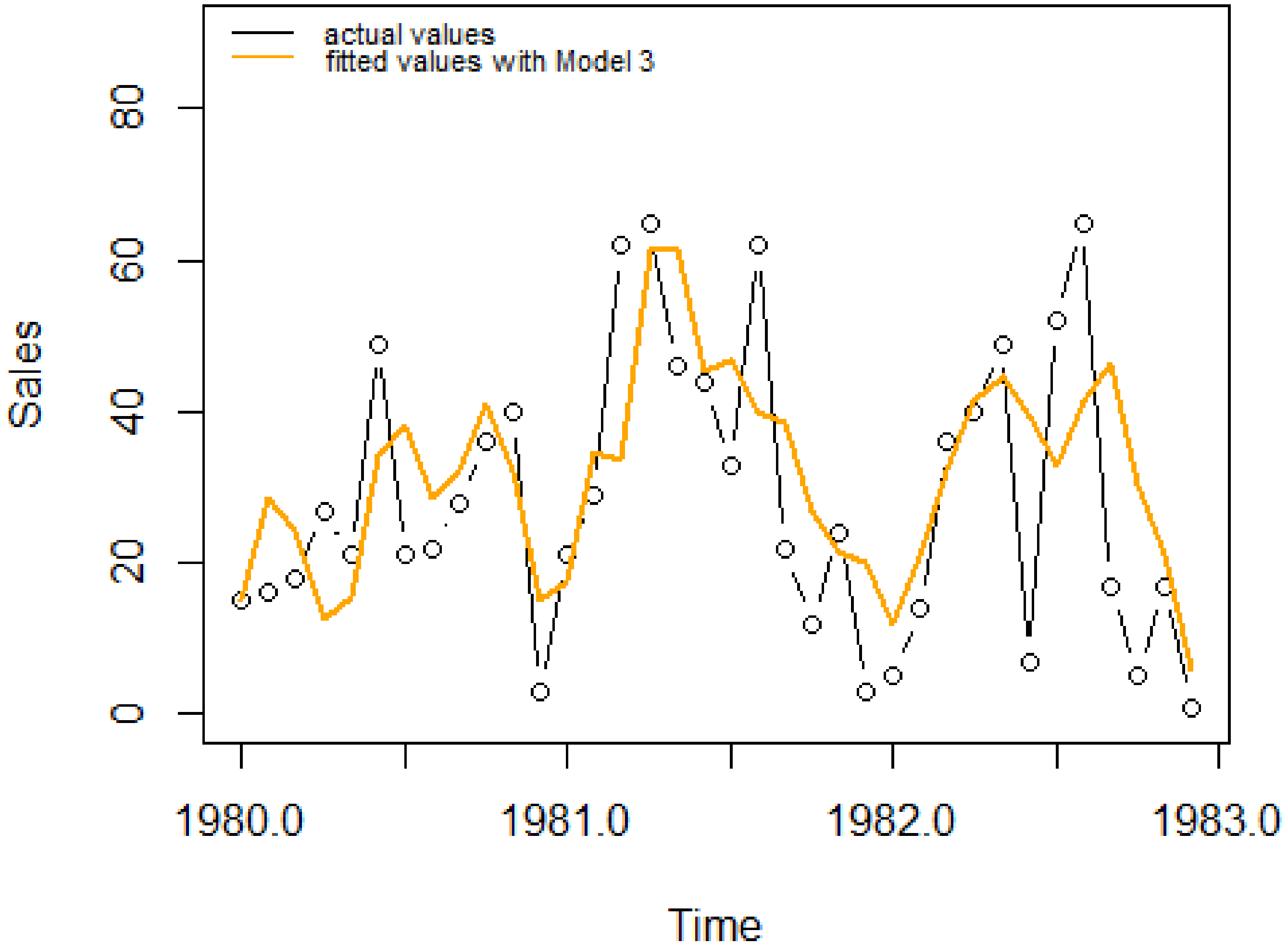
Model 3: Model 1 w/ ARIMA(0,1,1) errors

Performance of Model 3



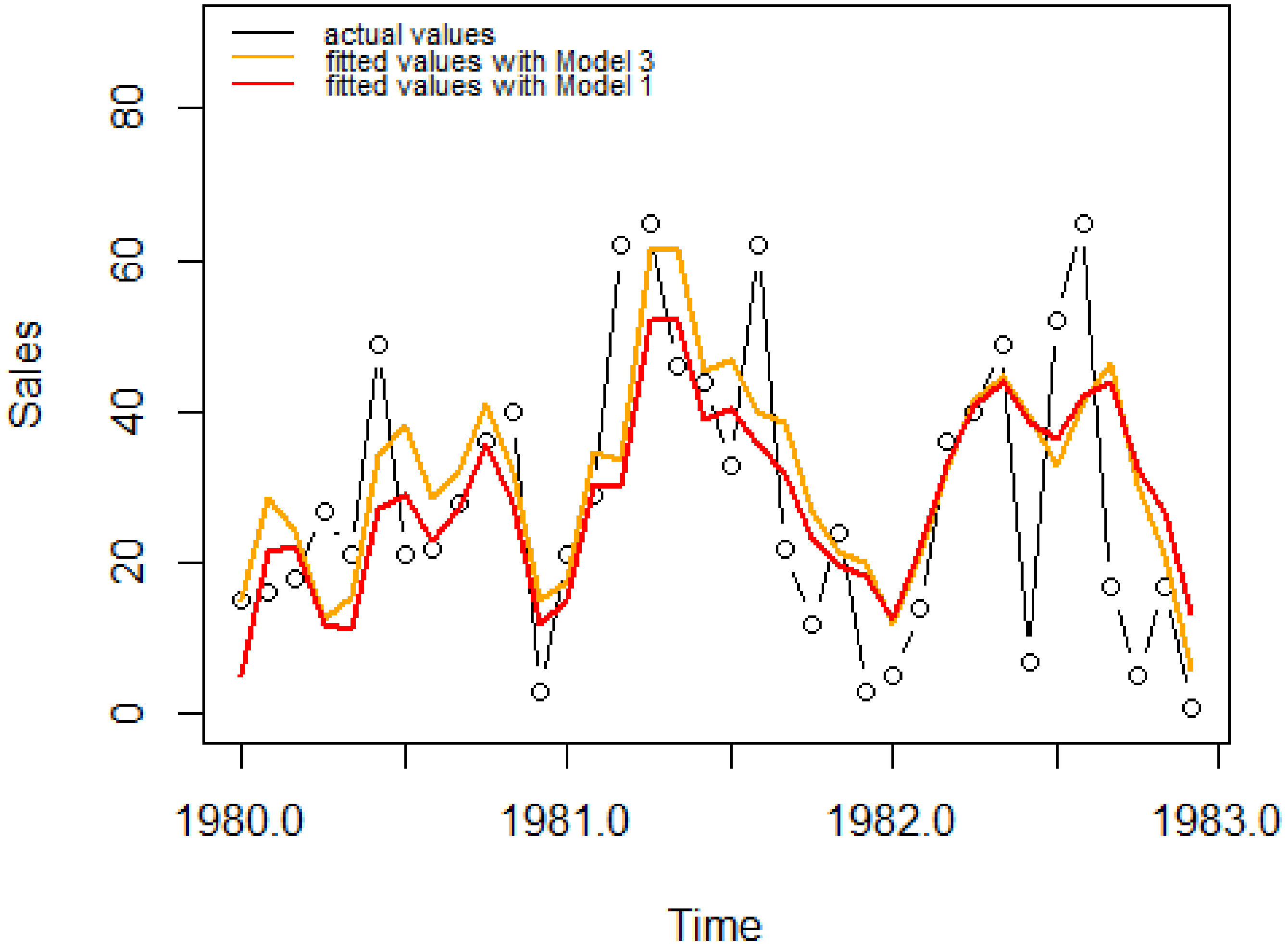
Model 3: Model 1 w/ ARIMA(0,1,1) errors

Performance of Model 3



Model 3: Model 1 w/ ARIMA(0,1,1) errors

Performance of Model 3



Model 3: Linear Regression w/ ARIMA(0,1,1) errors
RMSE for Forecast of Sales

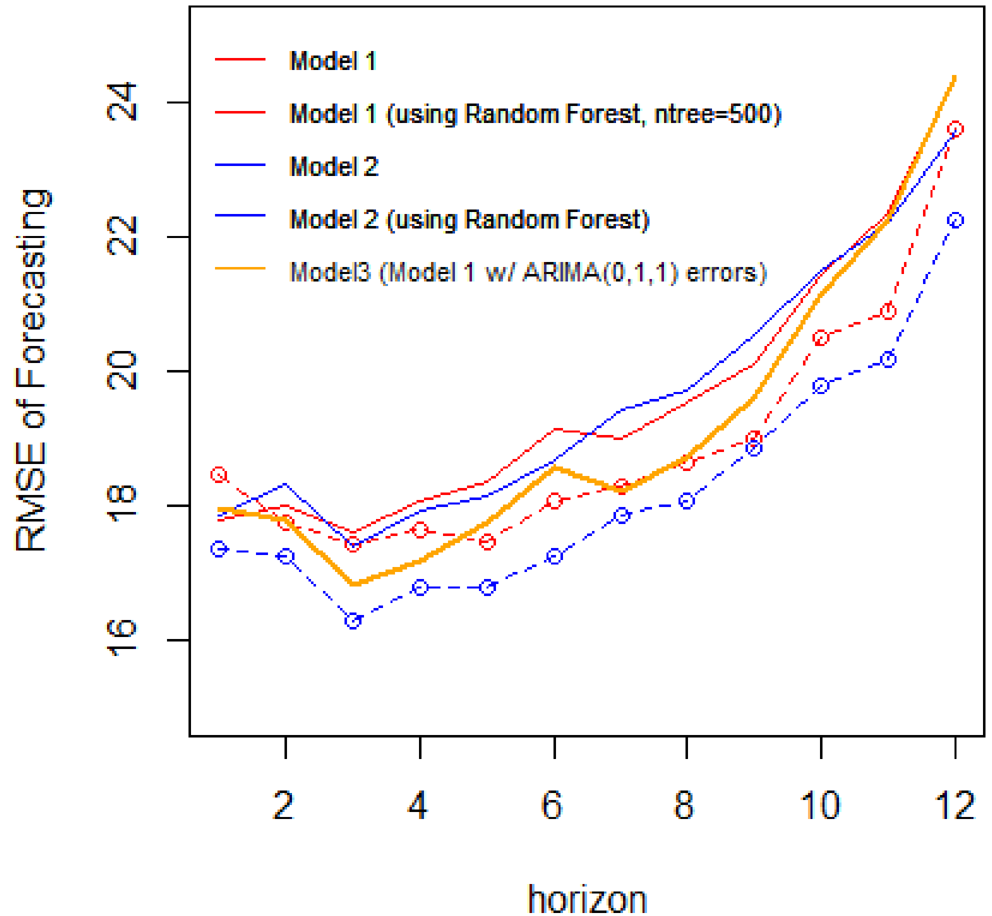
Regression Model 1 w/ ARIMA(0,1,1) errors

↓

Time series Cross Validation (TS-CV)

↓

Calculate **RMSE** for the prediction of Sales



Model 4: Model 2 w/ ARIMA errors

Model Selection

Regression Model 2

↓

Fit residuals of Model 2 using auto.arima()

↓

Sequentially change the p and q parameters from the previous step

↓

Evaluate models based in AICc, BIC, and RMSE of fitted values

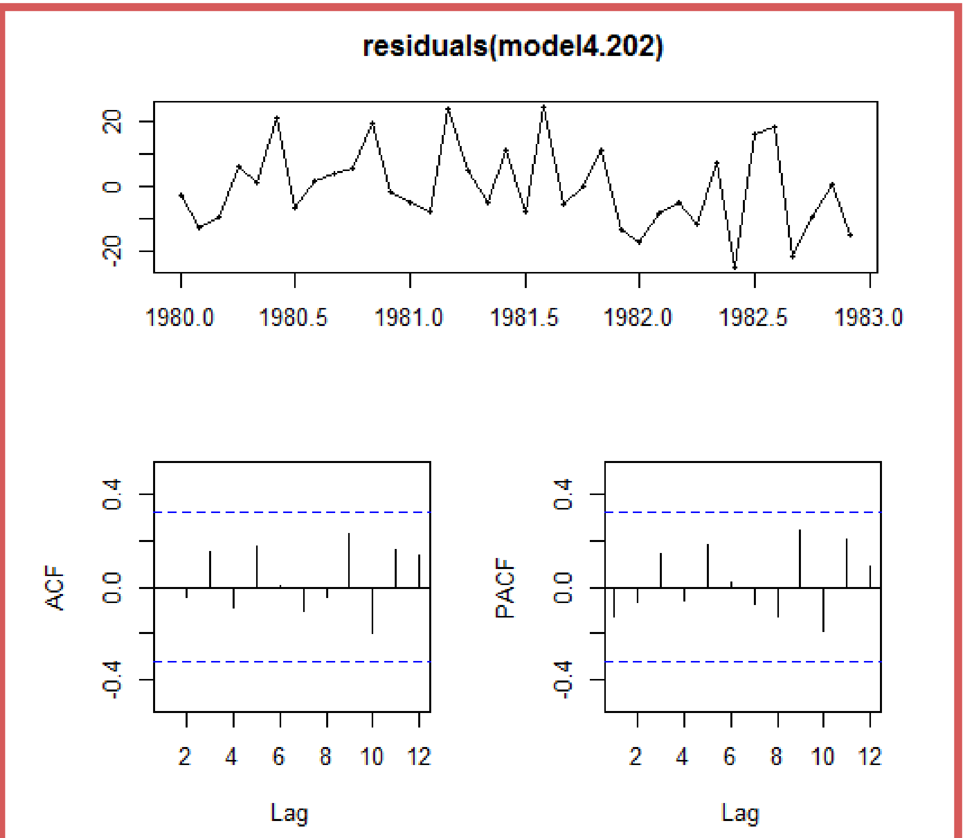
↓

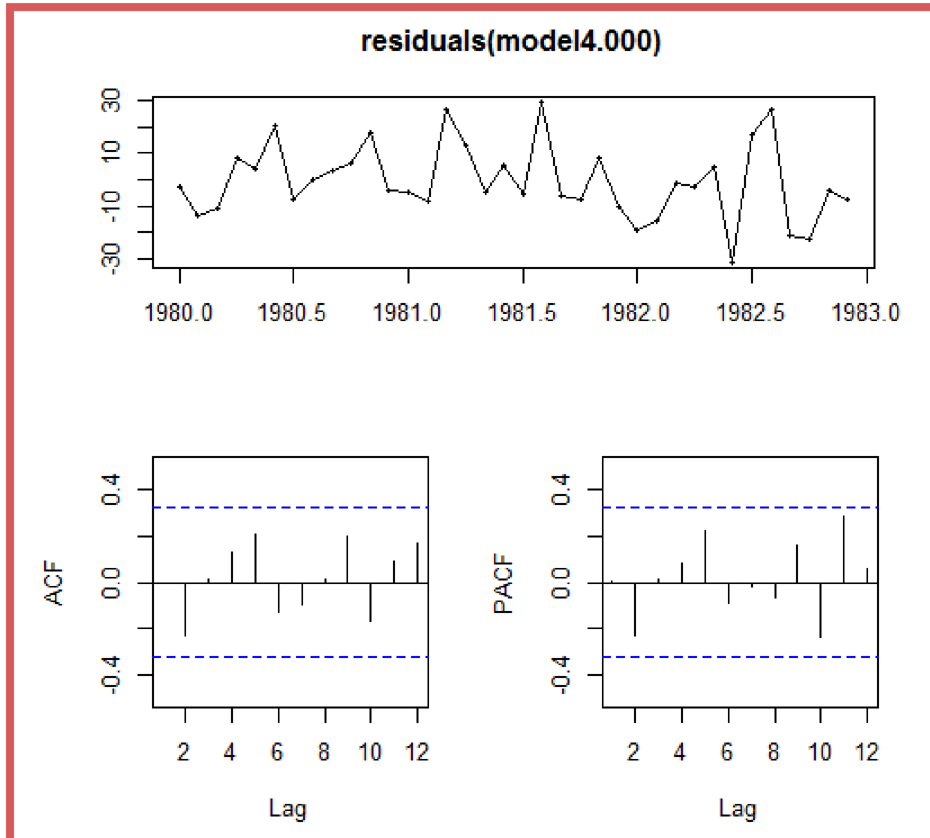
Check the performance of the selected model

Model 2: (Sales ~ Advertising + Month)

Model 2 w/ ARIMA error	**AICc**	**BIC**	**RMSE**
ARIMA(2,0,2)	305.8	312.9	**174.8**
ARIMA(0,0,1)	301.6	306.7	196.9
ARIMA(1,0,0)	301.6	306.7	196.9
ARIMA(0,0,0)	**299.1**	**303.1**	196.9
ARIMA(1,0,1)	301.1	307.1	197.1
ARIMA(2,0,1)	305.2	311.8	198.8
ARIMA(2,0,0)	302.3	308.3	198.8
ARIMA(1,0,2)	303.1	309.8	198.9
ARIMA(0,0,2)	302.6	308.5	199.4

Model 4: Model 2 w/ ARIMA(2,0,2) or ARIMA(0,0,0) errors
Residual diagnosis of Model 4

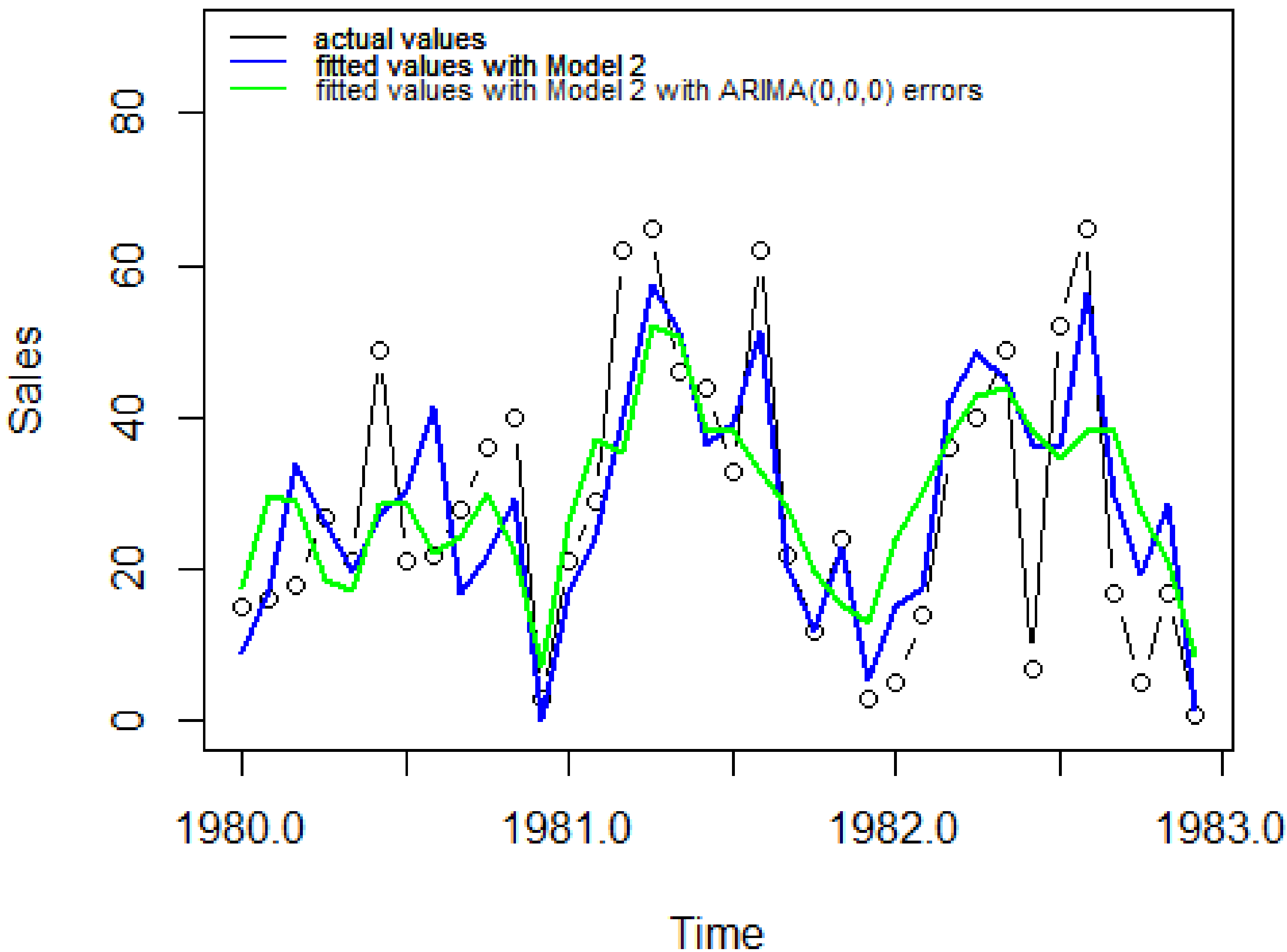




	p-value for BL test	DW	AICc	BIC
Model4.202	0.0406	2.21	305.8	312.9
Model4.000	**0.1817**	**1.98**	**299.1**	**303.1**

Model 4: Model 2 w/ ARIMA(0,0,0) errors

Performance of Model 4



Model 4: Model 2 w/ ARIMA(0,0,0) errors
RMSE for Forecast of Sales

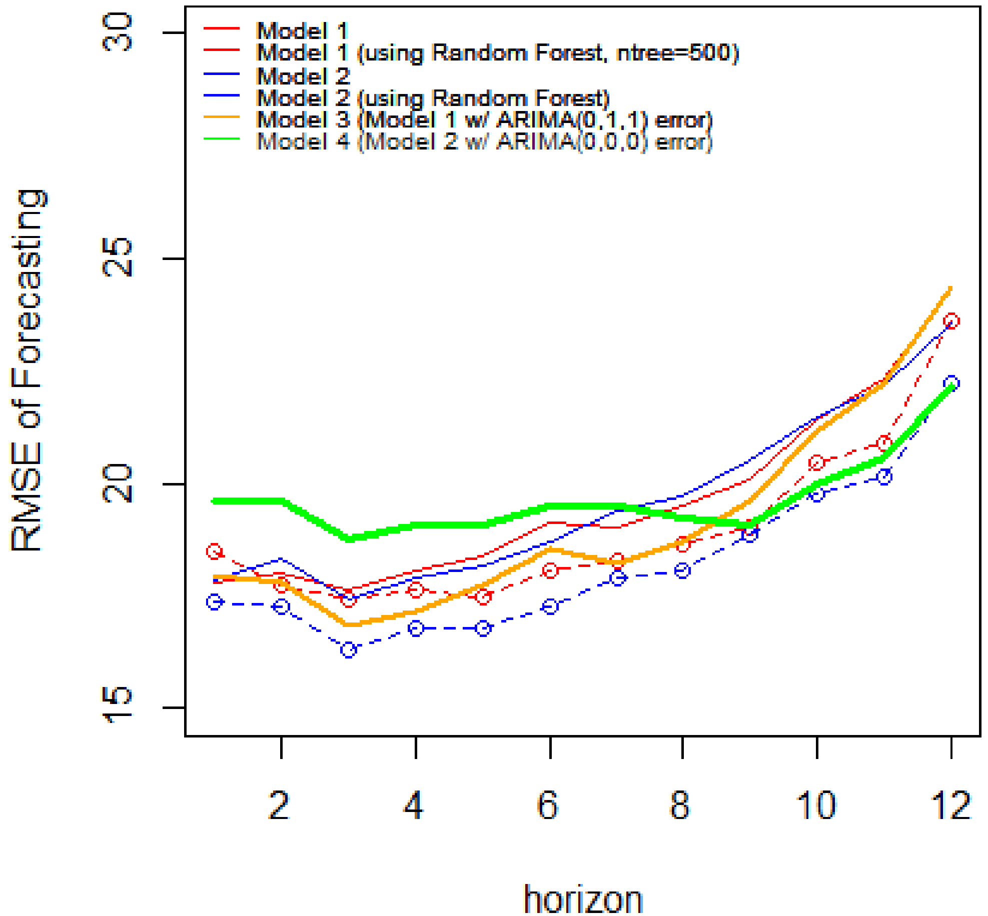
Regression Model 2 w/ ARIMA(0,0,0) errors

↓

Time series
Cross Validation
(TS-CV)

↓

Calculate **RMSE** for the prediction of Sales



Model 5: Seasonal ARIMA (sARIMA)

Model Selection

Use auto.arima() to fit data

↓

Sequentially change the p and q parameters from the previous step

↓

Evaluate models based in AICc, BIC, and RMSE of fitted values

↓

Check the performance of the selected model

Model 5: Seasonal ARIMA (sARIMA)

Model Selection

sARIMA model	AICc	BIC	Forecast RMSE
(3,0,1)[4]	217.599	207.250	405.8
(3,0,0)[4]	213.993	206.153	405.8
(1,0,0)[4]	208.173	204.269	407.6
(2,0,1)[4]	214.204	206.364	408.3
(1,0,1)[4]	211.046	205.335	408.3
(2,0,0)[4]	211.028	205.317	408.8
(0,0,2)[4]	210.338	204.627	409.0
(1,0,2)[4]	213.412	205.572	413.1
(0,0,1)[4]	**208.880**	**204.975**	**420.1**
(3,0,2)[4]	218.961	205.651	432.3
(2,0,2)[4]	214.992	204.642	445.4
(0,0,0)[4]	208.892	206.517	451.3

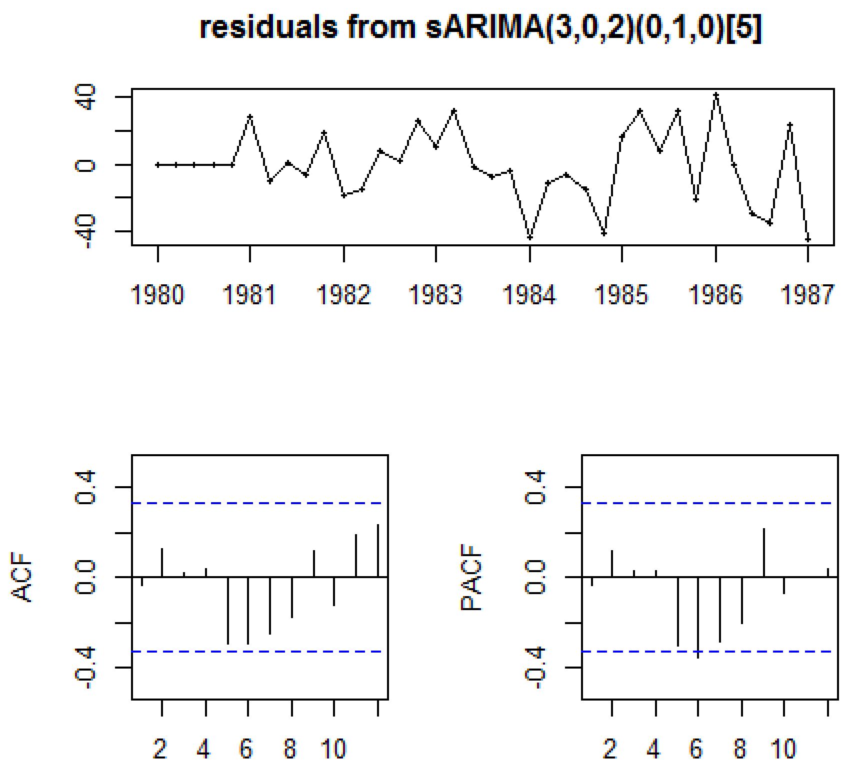
sARIMA model	AICc	BIC	Forecast RMSE
(3,0,2)(0,1,0)[5]	**176.428**	**164.019**	**883.6**
(0,0,1)(0,1,0)[5]	169.241	166.689	951.0
(3,0,1)(0,1,0)[5]	173.940	164.818	960.8
(3,0,0)(0,1,0)[5]	171.960	165.497	975.3
(0,0,2)(0,1,0)[5]	170.820	166.515	988.7
(2,0,2)(0,1,0)[5]	172.234	163.112	1037.6
(2,0,0)(0,1,0)[5]	169.743	165.439	1067.5
(2,0,1)(0,1,0)[5]	171.636	165.174	1075.4
(1,0,1)(0,1,0)[5]	169.774	165.470	1091.1
(1,0,0)(0,1,0)[5]	166.972	164.419	1124.6
(1,0,2)(0,1,0)[5]	172.401	165.939	1126.8
(0,0,0)(0,1,0)[5]	174.881	173.745	1292.8

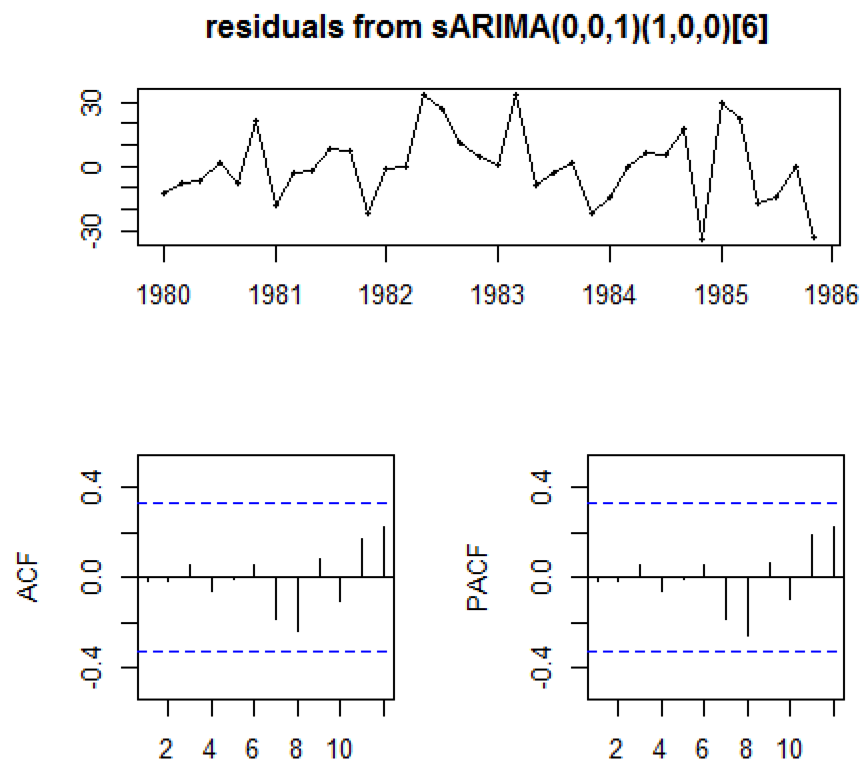
sARIMA model	AICc	BIC	Forecast RMSE
(0,0,1)(1,0,0)[6]	**206.377**	**200.666**	**279.3**
(1,0,0)(1,0,0)[6]	207.825	202.114	341.4
(0,0,0)(1,0,0)[6]	209.461	205.557	418.0
(2,0,0)(1,0,0)[6]	209.712	201.872	279.6
(0,0,2)(1,0,0)[6]	210.211	202.371	306.0
(1,0,1)(1,0,0)[6]	210.325	202.485	301.6
(3,0,0)(1,0,0)[6]	213.318	202.968	281.2
(2,0,1)(1,0,0)[6]	213.319	202.970	280.2
(1,0,2)(1,0,0)[6]	213.815	203.465	306.8
(2,0,2)(1,0,0)[6]	214.118	200.808	424.0
(3,0,1)(1,0,0)[6]	217.339	204.029	284.5
(3,0,2)(1,0,0)[6]	218.157	201.346	384.4

	(3,0,0)[4]	(3,0,2)(0,1,0)[5]	(0,0,1)(1,0,0)[6]
p-value for BJ test	0.3358	0.02853	**0.5239**
DW	1.916259	**1.947457**	1.906277

Seasonal ARIMA Models:

Residual Diagonsis



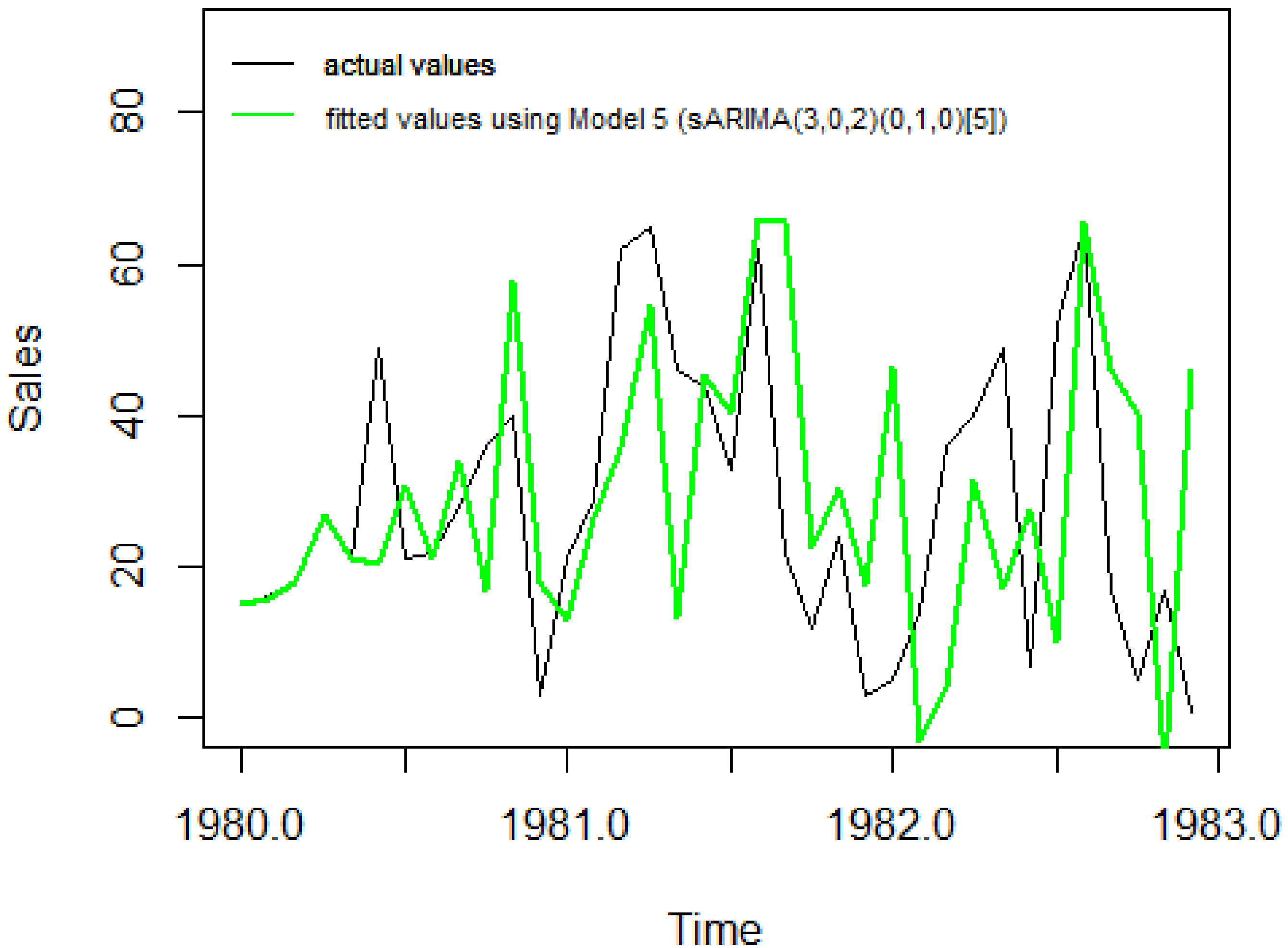



```
➢ Box.test(residuals(model5.5), fitdf=6, lag=20,
  type="Ljung")

Box-Ljung test data: residuals(model5.5) X-
squared = 25.666, df = 14, p-value = 0.02853
```

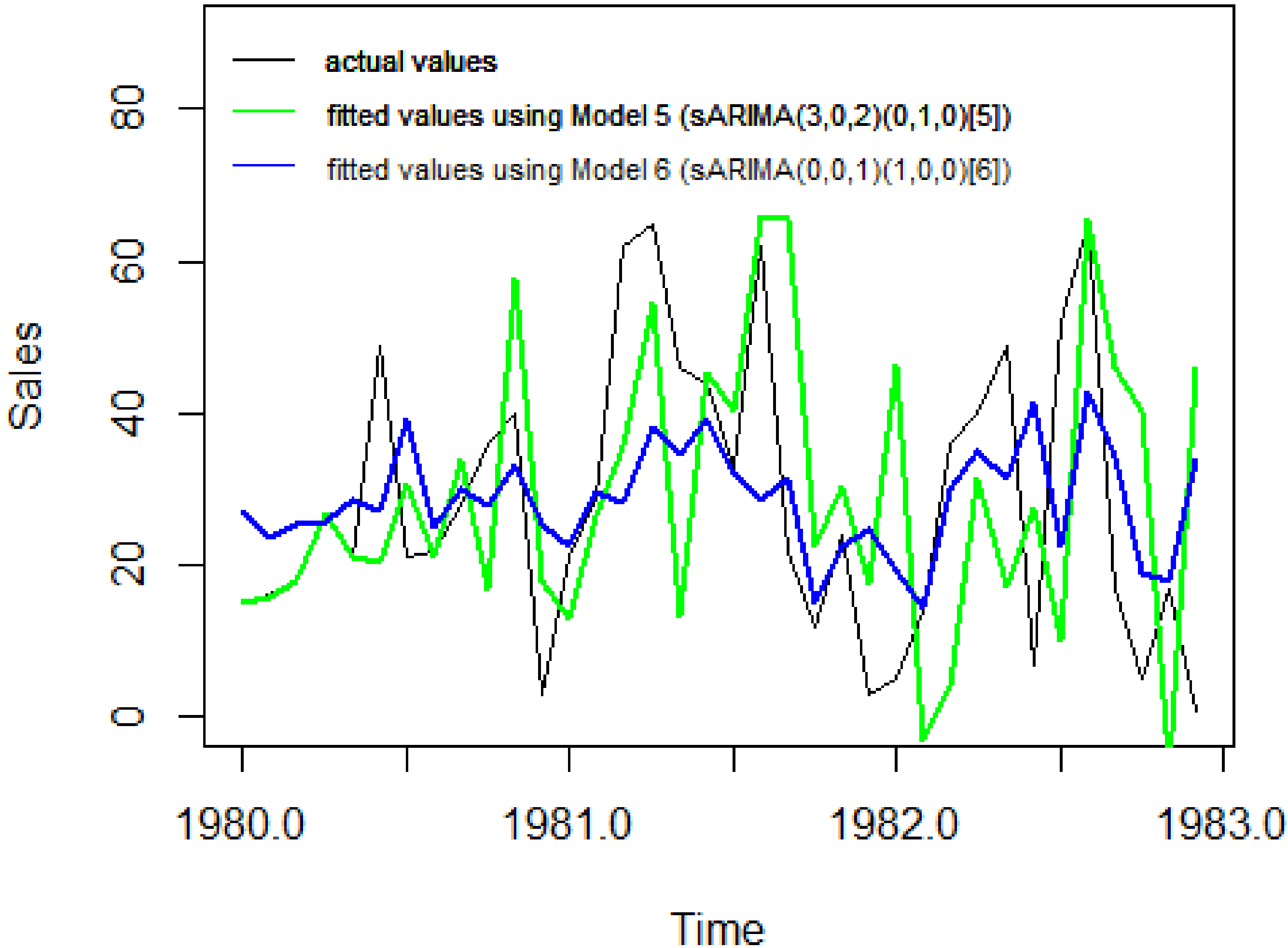
```
➢ Box.test(residuals(model5.6), fitdf=2, lag=20,
  type="Ljung")

Box-Ljung test data: residuals(model5.6) X-squared
= 16.9885, df = 18, p-value = 0.5239
```

Models 5: sARIMA (3,0,2)(0,1,0)[5]

Performance of Model 5

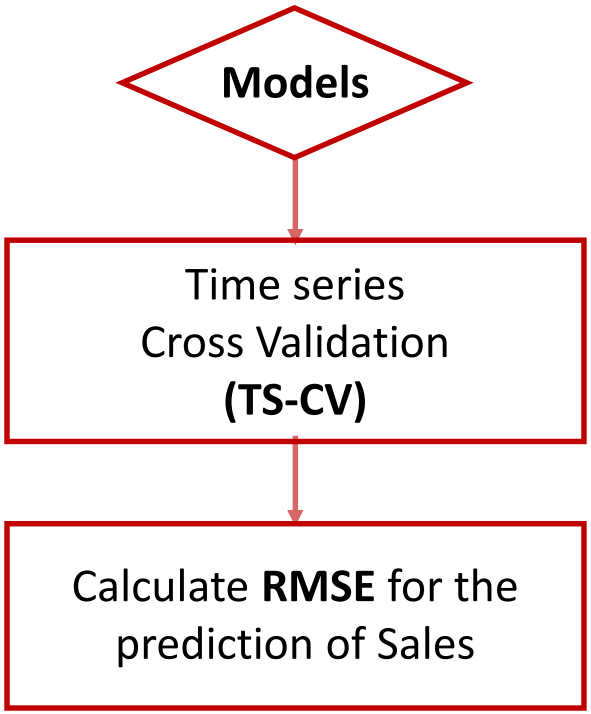


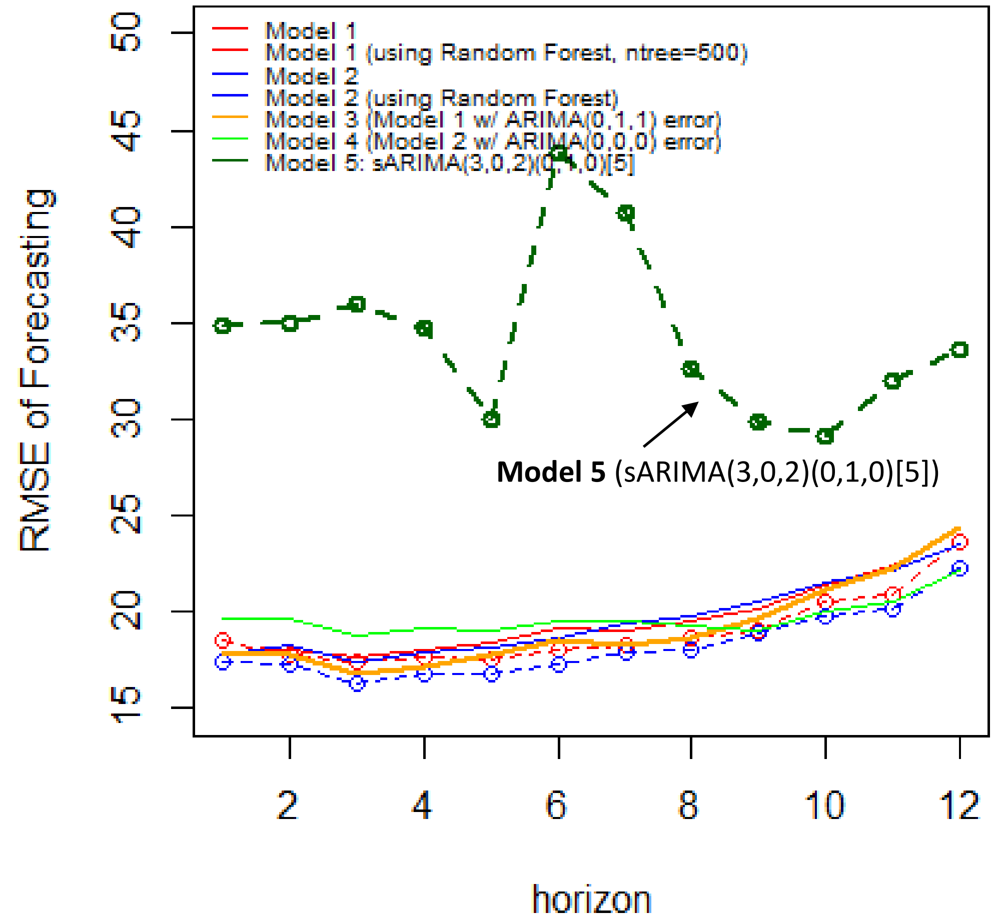
Models 6: sARIMA (0,0,1)(1,0,0)[6]
Performance of Model 6



Models 5 & 6: Seasonal ARIMA (sARIMA)

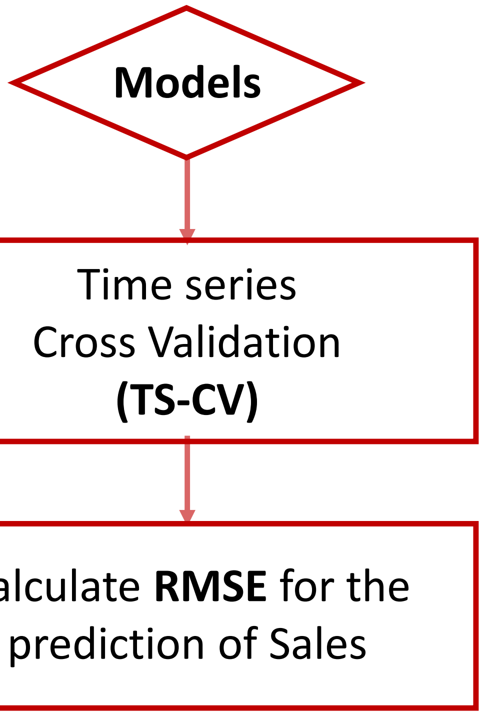
RMSE for Forecast of Sales

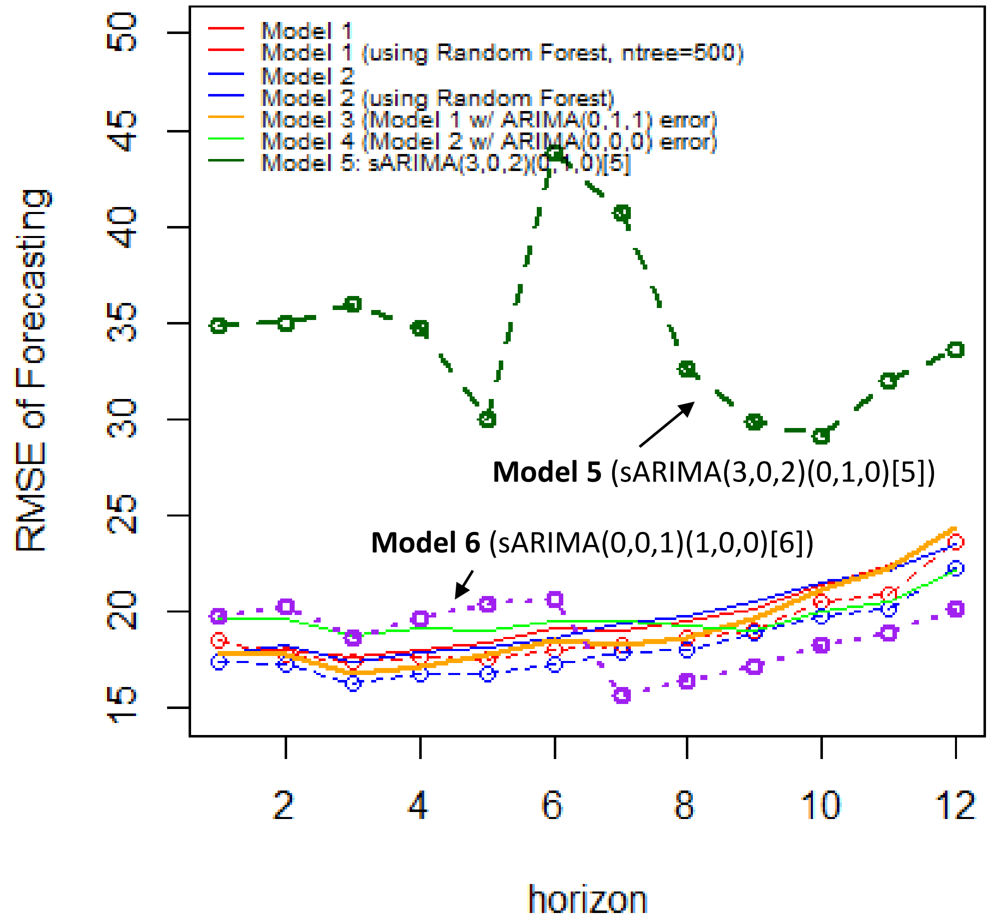




Models 5 & 6: Seasonal ARIMA (sARIMA)

RMSE for Forecast of Sales





Model 7: SVM model

Select Modeling Techniques

Build Model Parameter Settings

Residual Diagnosis

Model Evaluation

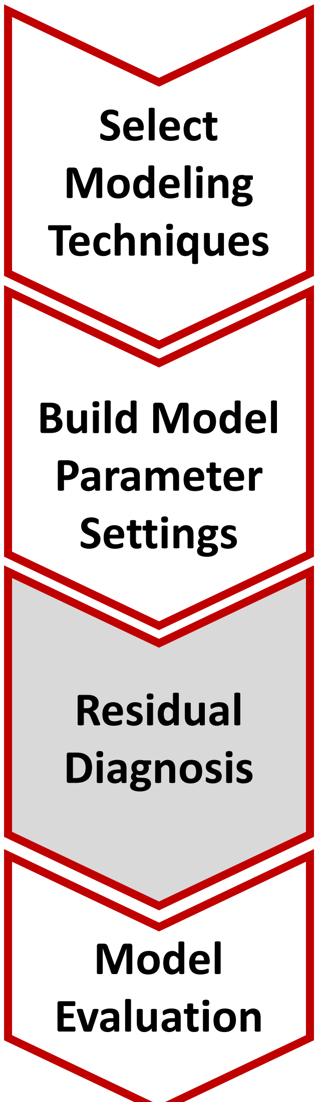
- Support vector machine (SVM)

Model 7: SVM model

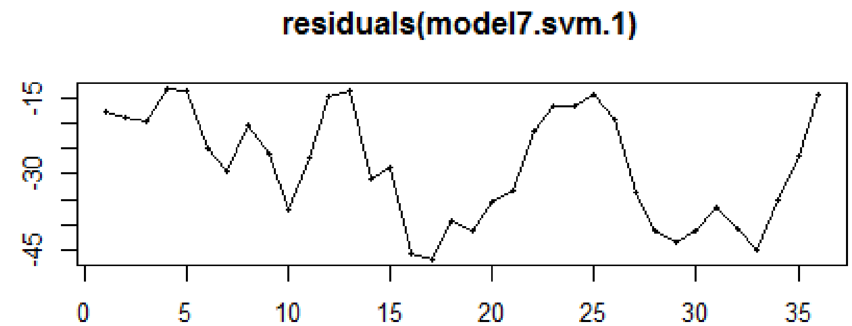
- Select Modeling Techniques
- Build Model Parameter Settings
- Residual Diagnosis
- Model Evaluation

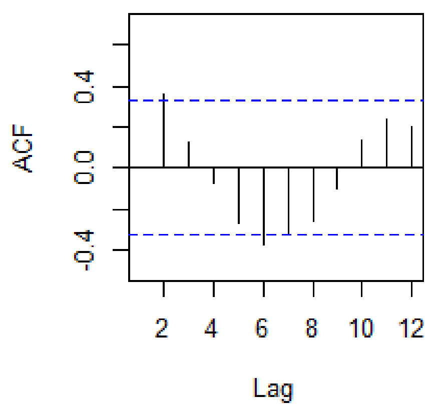
- Support vector machine (SVM):
  model7.svm.1: Sales ~ Advertising

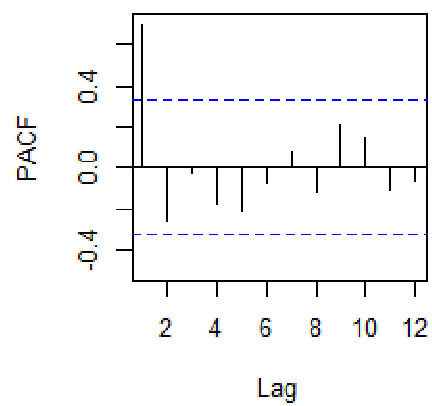
Model 7: SVM model



- Support vector machine (SVM):
  model7.svm.1: Sales ~ Advertising



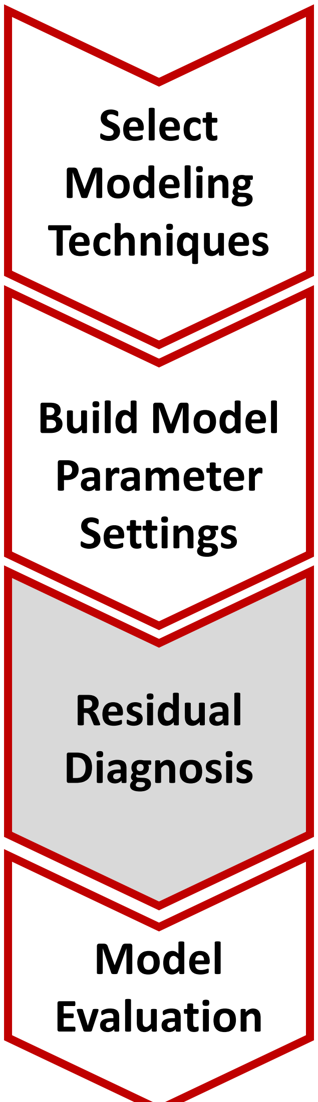




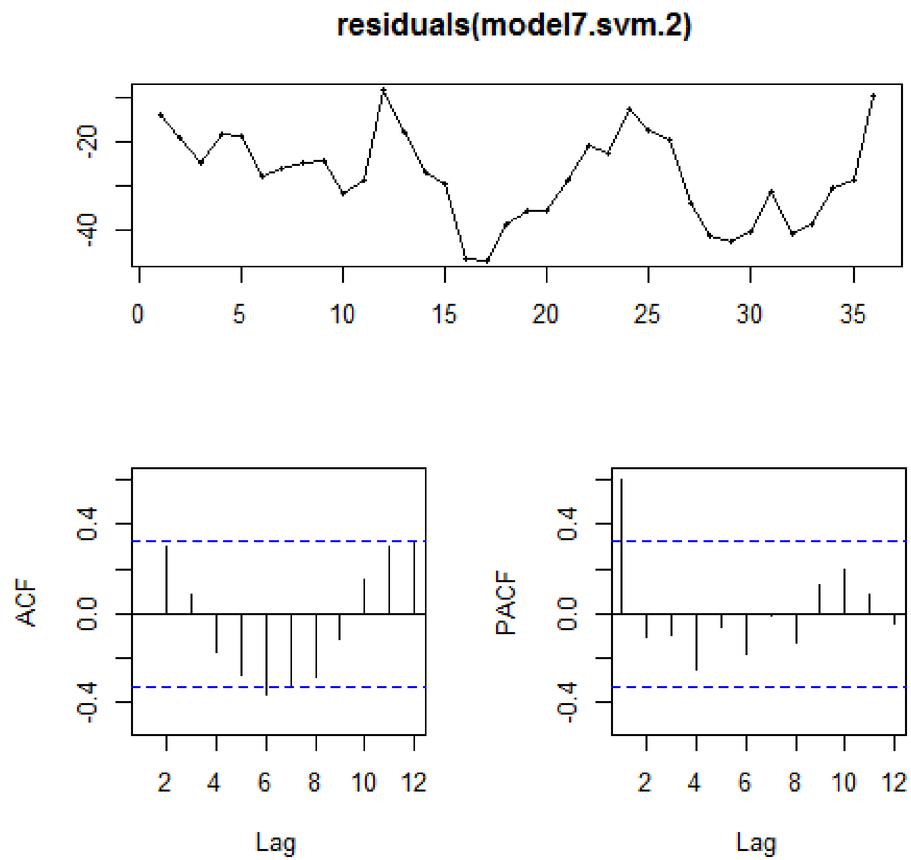
Model 7: SVM model

- Support vector machine (SVM):
  model7.svm.2: Sales ~ Advertising + Month

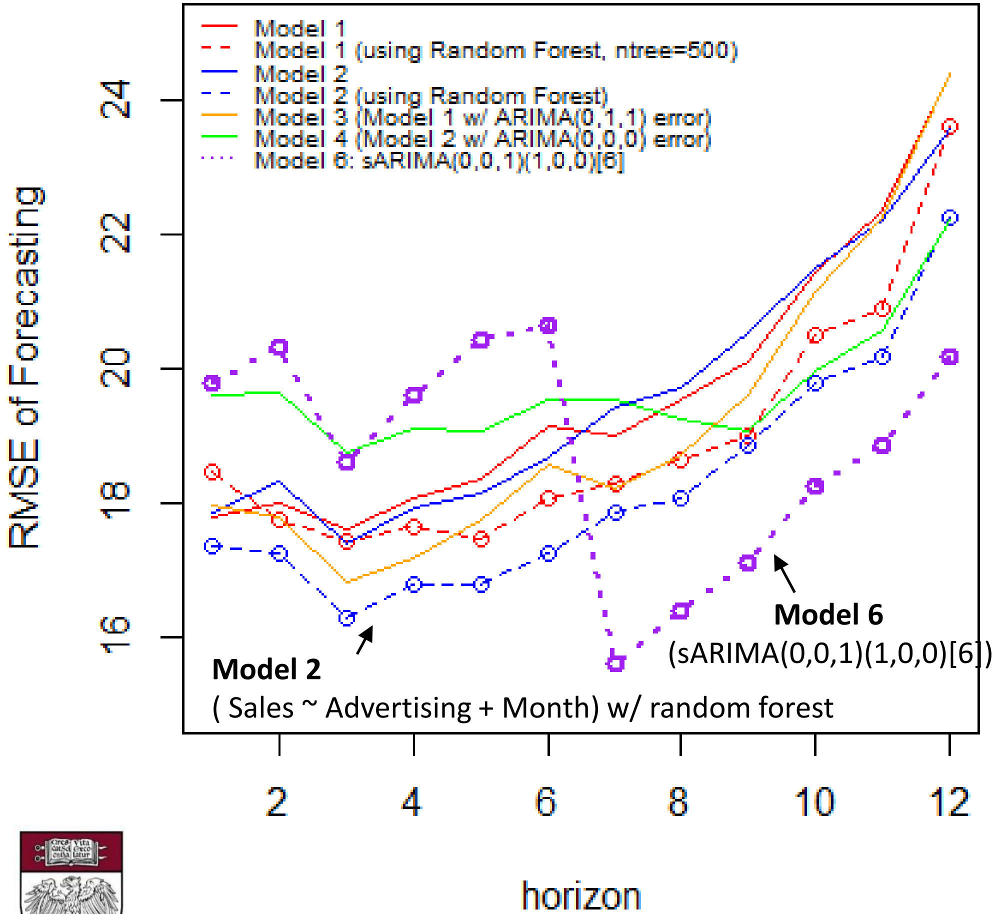
Model 7: SVM model



- Support vector machine (SVM):
  model7.svm.2: Sales ~ Advertising + Month



Summary of models: model evaluation & forecast performance



	p-values for Box-Ljung test	DW	AIC	BIC
Model 1: (Sales ~ Advertising)	0.4196	1.95	299.99	304.74
Model 2: (Sales ~ Advertising + Month)	0.0029	1.93	303.50	325.64
Model 3: Model 1 with ARIMA(0,1,1) errors	0.1479	2.19	**286.36**	**291.03**
Model4.202: Model2 with ARIMA(2,0,2) error	0.0406	2.21	301.78	312.86
Model4.000: Model2 with ARIMA(2,0,2) error	0.1817	1.98	298.35	303.10
Model 5: sARIMA(3,0,2)(0,1,0)[5]	0.0285	1.95	296.70	305.30
Model 6: sARIMA(0,0,1)(1,0,0)[6]	**0.5239**	1.91	313.86	320.19

Ensemble Modeling

- Ensemble methods train multiple predictive models and then combine the predictions to achieve a higher overall performance and stability.

- **Weighted Linear Stacking**:
  Seek a blended prediction function to compute the estimated prediction, $\hat{y}(x)$, for datapoint *x*:

$$\hat{y}(x) = \sum_i w_i \, g_i(x)$$

w_i: model weight, $\sum_i w_i = 1$

g_i: the learned prediction functions of *L* learning models

One way to determine w_i is to satisfy the optimization problem as follows:

$$\min \sum_i [w_i g_i(x) - y(x)]^2$$

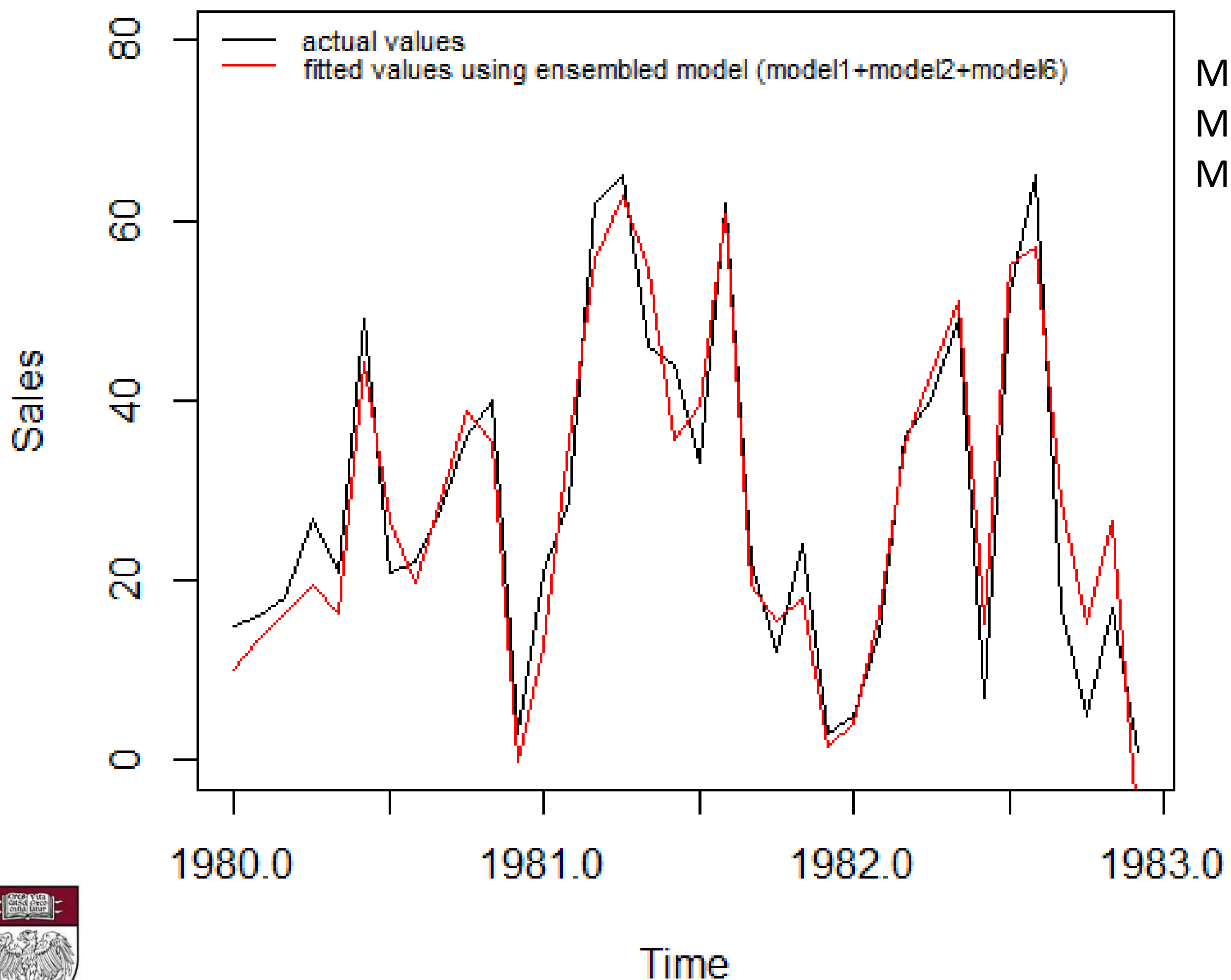
y(*x*) is the target prediction for datapoint *x*

Therefore,

$$\min Var[\hat{y}(x)] = \sum_i w_i^2 \, Var[g_i(x)] \quad \text{where} \quad w_i = \frac{\sqrt{\frac{1}{Var[g_i(x)]}}}{\sum_i \sqrt{\frac{1}{Var[g_i(x)]}}}$$

Ensembled Model:

Performance of Ensembled Model



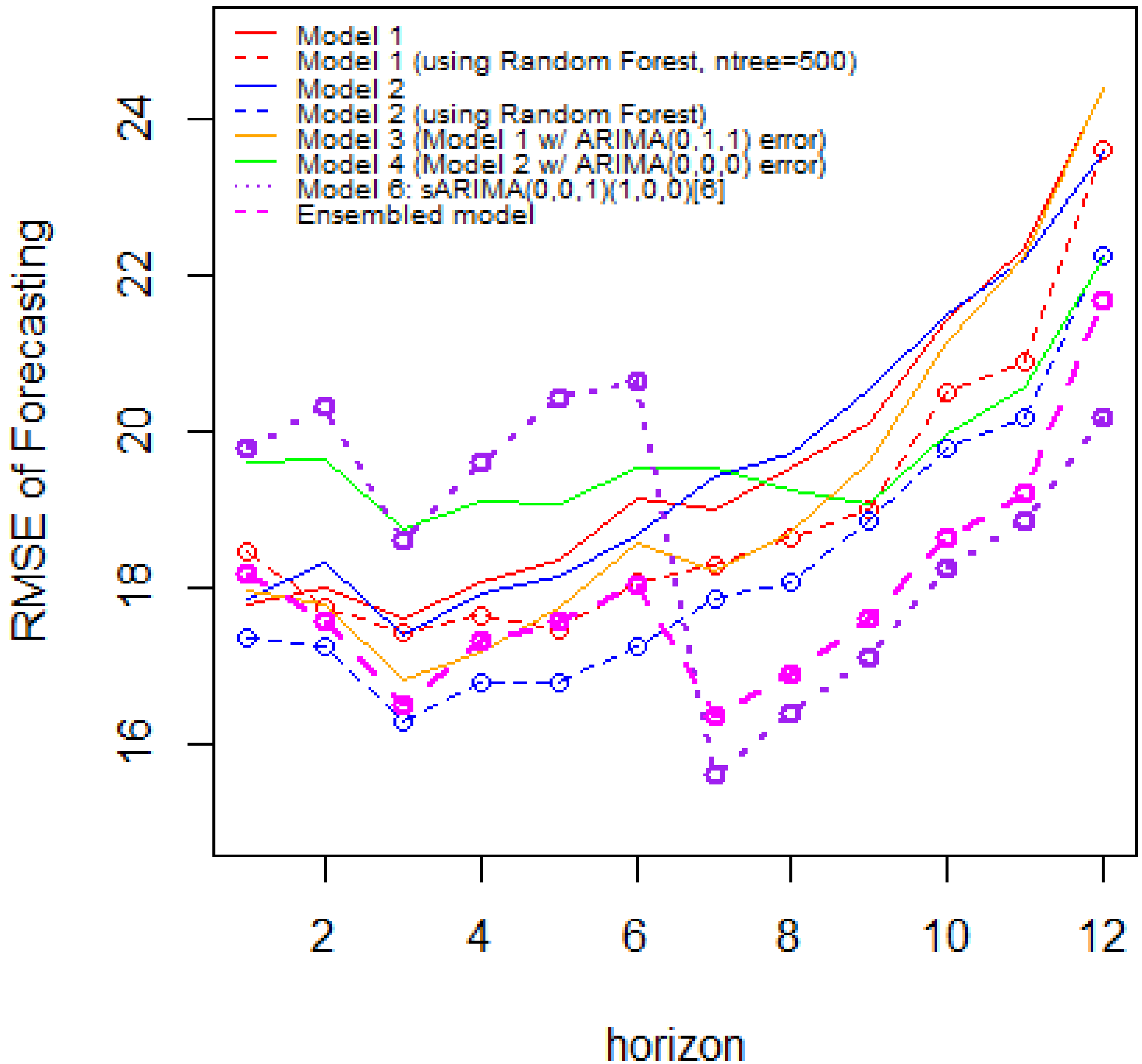
Model 1: Sales ~ Advertising

Model 2: Sales ~ Advertising + Month

Model 6: sARIMA(0,0,1)(1,0,0)[6]

Ensembled Model:

RMSE for Forecast of Sales



Summary

- Linear regression with random forest algorism, regression with ARIMA error, seasonal ARIMA, SVM, and ensemble models were constructed to forecast monthly sales of a dietary weight control product.
- Monthly advertising expenditures were applied as an independent variable for constructing models to fit the sales data.
- Including seasonality information is important while constructing models for forecasting the monthly sales.
- Time series cross validation was performed for prediction horizon.
- Ensembles model enhanced the overall performance and stability of forecasting.

Future Work

- Improve ensembling algorism.
- Bass and Clarke [1] proposed distributed lag models to describe simultaneously non-linear and delayed effects between predictors and an outcome. Gasparrini[2] has implemented the distributed lag non-linear methodology in R (dlnm package).

[1] Bass F.M. and Clarke D.G. *Journal of Marketing Research*, vol. IX (August 1972). 298-308
[2] Gasparrini A. Distributed lag linear and non-linear models in R: the package dlnm. Journal of Statistical Software, 43(8):1{20, 2011. URL http://www.jstatsoft.org/v43/i08/.