PREDICTION OF CASH WITHDRAWAL DATA AT ATM MACHINES USING THE *SEASONAL AUTOREGRESSIVE INTEGRATED MOVING AVERAGE METHOD* (SARIMA)

Fitrinadi[1], Irwan Budiman[2], Andi Farmadi[3], Dodon Turianto Nugrahadi[4], M. Itqan Mazdadi[5]
[1,2,3,4,5]Computer Science Study Program FMIPA ULM
A. Yani St. Km 36 Banjarbaru, South Kalimantan
Email: J1F114021@mhs.ulm.ac.id[1], irwan.budiman@ulm.ac.id[2], andifarmadi@ulm.ac.id[3], dodonturianto@ulm.ac.id[4], mazdadi@ulm.ac.id[5]

Abstract
Data mining is a series of processes to explore the added value of knowledge that has been unknown from a data set. Many algorithms can be used in solving a problem related to prediction or forecasting a new data value for the future based on pre-existing data. Sarima model is a model in time series analysis. The performance of the Seasonal Autoregressive Integrated Moving Average (SARIMA) method produces a suitable or good model used to predict cash withdrawal data at ATM machines. The data used in the study is a dataset of ATM transactions originating from Finhacks. The result of error using MAPE (Mean Absolute Percenttage Error) on the predicted result of cash withdrawal data at atm machines is K1 16.75%, K2 18.09%, K3 7.85%, K4 5.67%, and K5 11.80%. So it can be concluded that the data matches using the SARIMA model that has been selected because the MAPE value is smaller than 20%.

Keywords *: Data Mining, Finhacks, SARIMA, MAPE, ATM*

1. INTRODUCTION

Data mining is a process that uses statistical techniques, mathematics, artificial intelligence and machine learning to extract and identify useful information and related knowledge from a variety of large databases. Data mining is the process of discovering interesting knowledge, patterns, and information from a large set of data through descriptive, comprehension and prediction processes using a model or algorithm. [5].

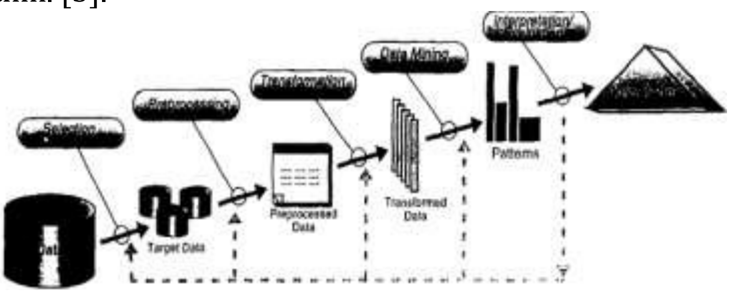

Figure 1 Data Mining Process (Source: Fayyad, 1996)

Prediction is the science to predict future events. This can be done by retrieving historical data or past data and projecting into the future with a mathematical model form [1].

Many algorithms can be used in solving a problem related to prediction or forecasting a new data value for the future based on pre-existing data. One of the methods that can be used is Autoregressive Integrated Moving Average (ARIMA) or Seasonal Autoregressive Integrated Moving Average (SARIMA). However, the Autoregressive Integrated Moving Average (ARIMA) method can only predict data that does not contain iterations (seasonal) and the Seasonal Autoregressive Integrated Moving Average (SARIMA) method itself can make predictions with data containing seasonality [2].

2. RESEARCH METHOD

To obtain the best Seasonal Autoregressive Integrated Moving Average (SARIMA) model that can be used to predict the amount of cash withdrawals at future ATMs:

a) Tabulate the data of cash withdrawal amount at atm machines from January 2018 – March 2018.
b) Plot time series, ACF and PACF for original data.
c) Identify the data whether it is stationary or not. If the data has not been stationary in average then differentiation is done and if the data has not been stationary in its variance then transformation is carried out.
d) Plot the timeline, ACF and PACF of the differentiation and transformation result data as well as determine the model. If it is stationary, immediately specify the model.
e) Estimate the parameters of the obtained model.
f) Test the model's suitability. If the model is not sufficient then identification of the new model is carried out.
g) Select the best model by looking at the smallest Mean Square Error (MSE) or Standard Error Estimated (SEE) value.

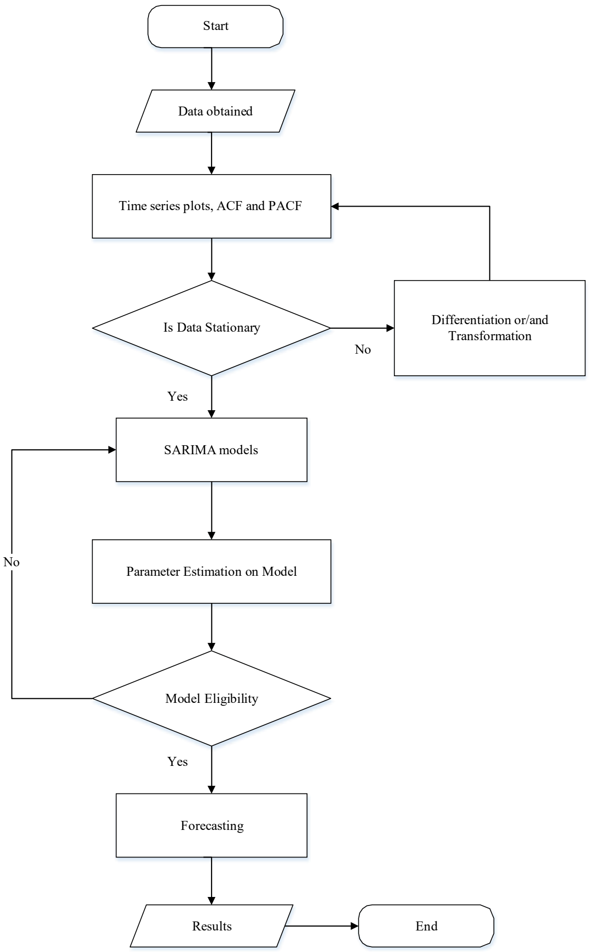

Figure 2. Research Flow

3. RESULT AND DISCUSSION

3.1. Data Collection

The data used in this study is atm transaction dataset obtained from Finhacks. This dataset was obtained from Finhacks supported by BCA. The data used is daily withdrawal data with a period of three months, namely from January to March 2018 with the total data used is 415 data obtained from 5 samples of ATM numbers and has 4 attributes namely numbers, ATM numbers, dates, and withdrawals.

3.2. Preprocessing Data

3.2.1 Data Filtering

Data filtering aims to select the data to be used. Here there are 5 sample data that will be used, namely 5 ATM numbers with ATM numbers symbolized with K, namely K1, K2, K3, K4, and K5.

3.2.2 Data Sharing Training dan Testing

The process after the data is filtered is then done the sharing of training data and testing data. Where data sharing is done by using excel with data training sharing as much as 76 data and data testing as much as 7 data per K or ATM number. Data testing is divided by 1 week because it will be done forecasting in the next 1 week and will be compared with the actual data and forecasting data or predictions.

3.2.3 Data Mining and Evaluation

Data mining process is done with 5 experiments with 5 different ATM or K numbers using Seasonal Autoregressive Integrated Moving Average (SARIMA) method. The design of this study can be seen in Table 5 as follows.

Tabel 1. Research Desain

Experiment	Dataset	Method
1	K1	SARIMA
2	K2	SARIMA
3	K3	SARIMA
4	K4	SARIMA
5	K5	SARIMA

For syntax diagram of time series (original data plot) data on cash withdrawal amount at ATM machine K1 – K5 in RStudio can be seen as follows:

```
title("Penarikan Uang di ATM")
plot(1:76,y=dataobjek,type="l",xaxt="n")
axis(1, at=1:83, labels=mydata$Tanggal[1:83],las=3)
```

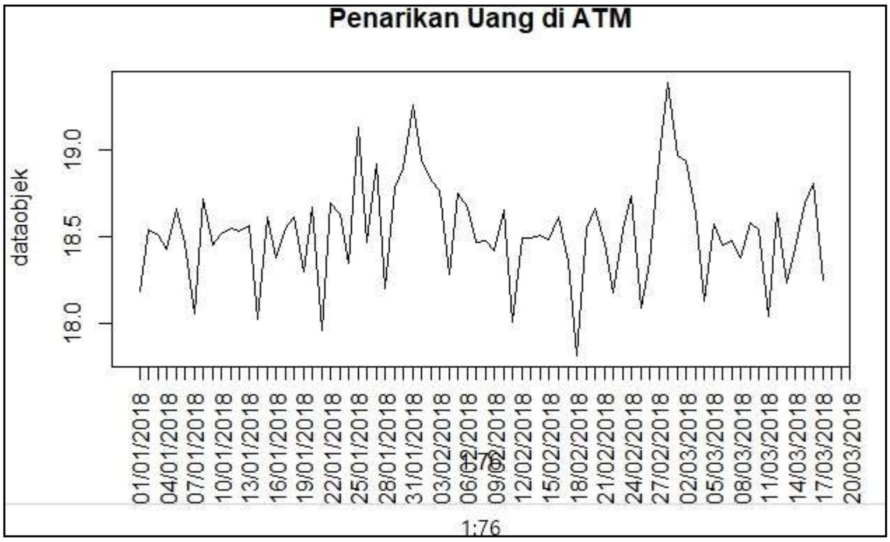

Figure 3. Plot Time Series Money Withdrawal Data at ATM machine K1

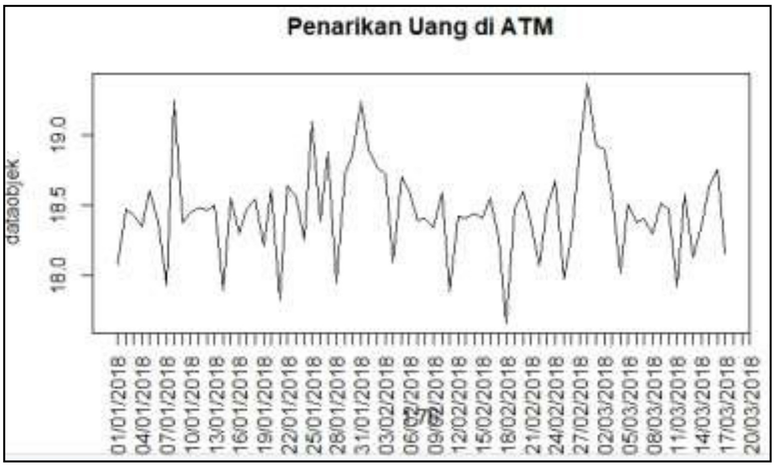

Figure 4. Plot Time Series Money Withdrawal Data at K2 ATM machines

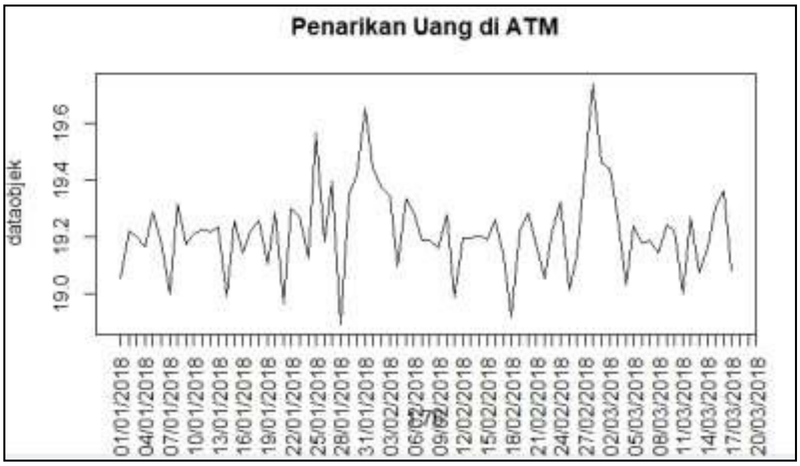

Figure 5 Plot Time Series Money Withdrawal Data at ATM machine K3

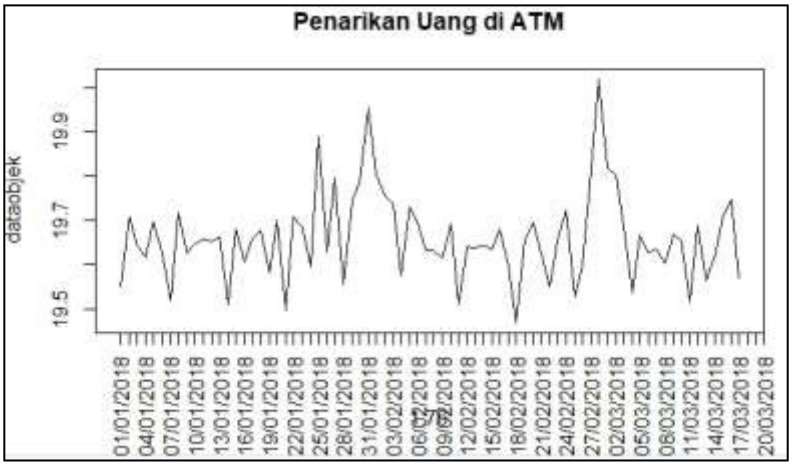

Figure 6. Plot Time Series Money Withdrawal Data at K4 ATM machines

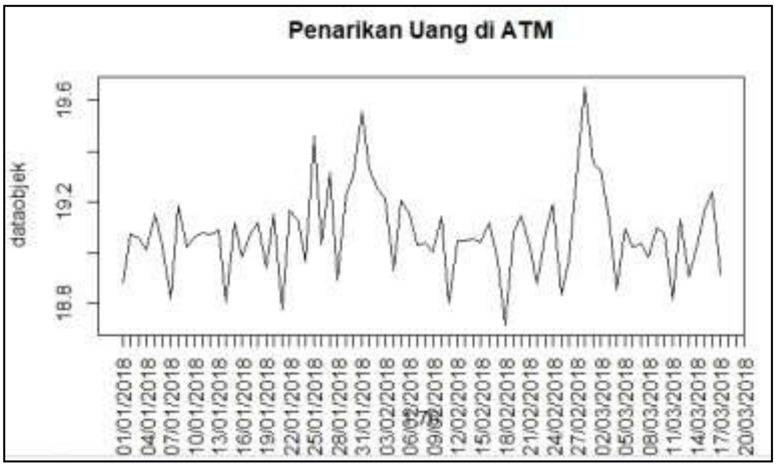

Figure 7. Plot Time Series Money Withdrawal Data at K5 ATM machines

3.2.4 ACF and PACF

ACF is an autocorrelation function that indicates the magnitude of the correlation that explains the relationship between t-time data and subsequent t+1 data by looking at the ACF plot formed does not have significant changes between lags. ACF serves to identify the kestasioneran data and to identify the time series model to be used.

PACF is a partial autocorrelation coefficient that measures the level of tightness of the partial relationship between the embed at the t-time and the equalization at the time of t+1 when the lag used is 1,2,3, and so on with notation (økk= k = 1.2.3...). PACF is used in the formation of models and orders of ARIMA. For syntak looking for ACF and PACF in RStudio can be seen as follows:

```
acfku<-acf(transformasidata, lag.max=20)
acfku
pacfku<-pacf(transformasidata, lag.max=20)
pacfku
```

And here are the ACF and PACF obtained from running using RStudio.

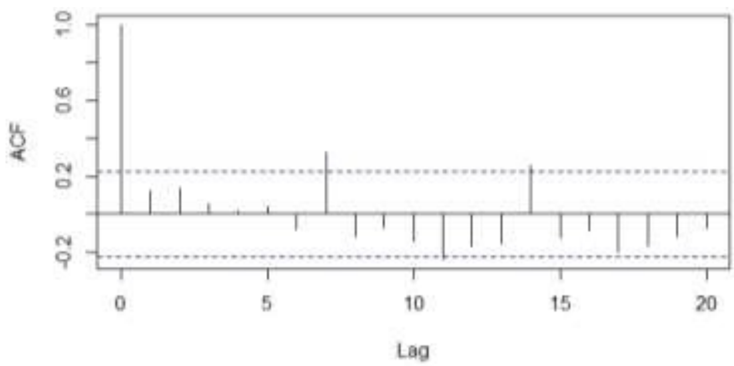

Figure 8. ACF Money Withdrawal Data at K1 ATM Machine

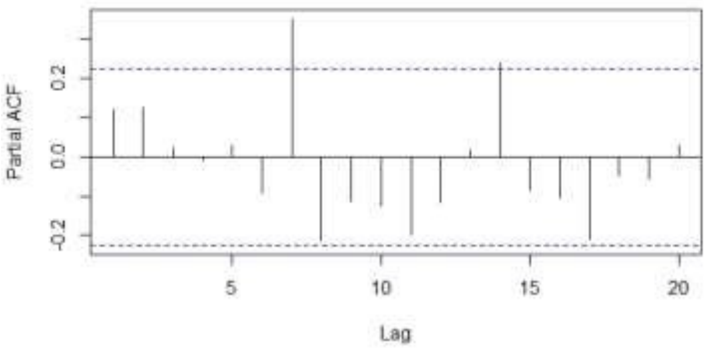

Figure 9. PACF Data on Withdrawal of Money at K1 ATM Machine

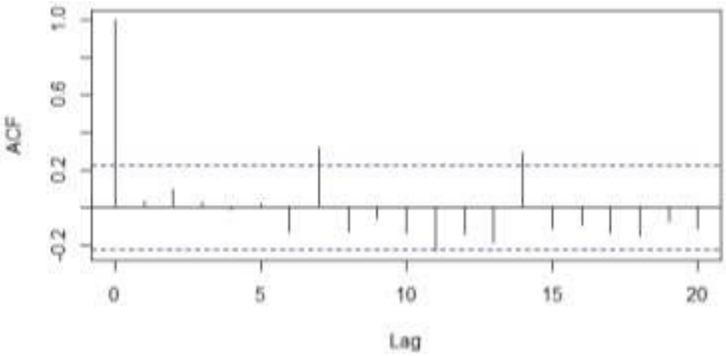

Figure 10. ACF Data on Withdrawal of Money at K2 ATM Machine

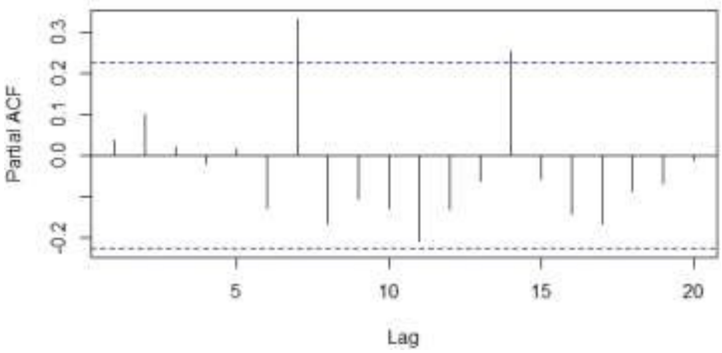

Figure 11. PACF Data on Withdrawal of Money at K2 ATM Machine

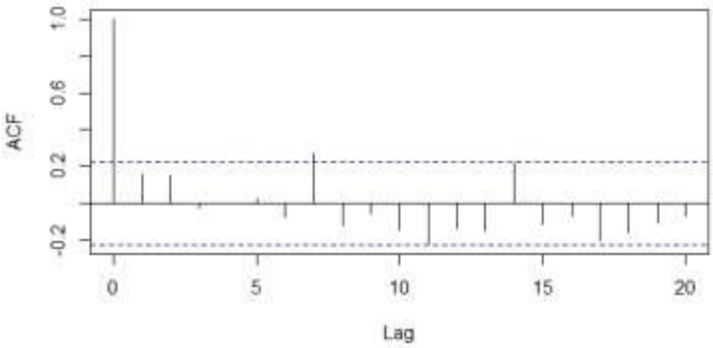

Figure 12. ACF Money Withdrawal Data at K3 ATM Machine

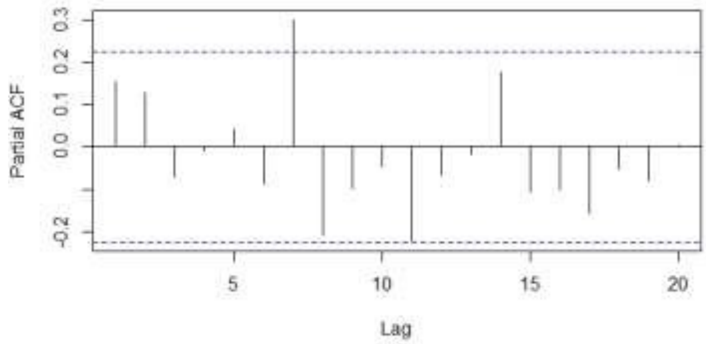

Figure 13. PACF Data on Withdrawal of Money at K3 ATM Machine

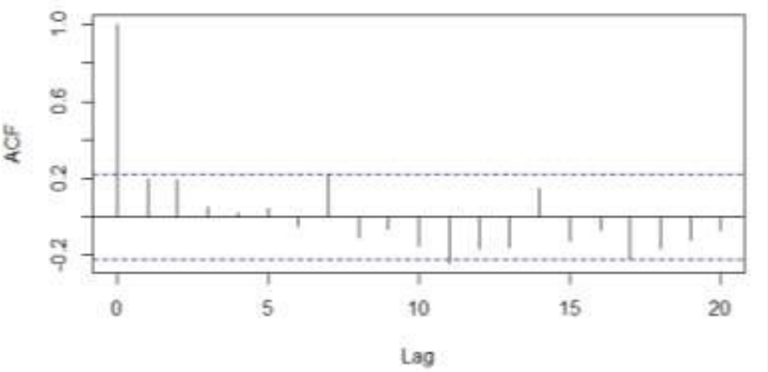

Figure 14 ACF Money Withdrawal Data at K4 ATM Machine

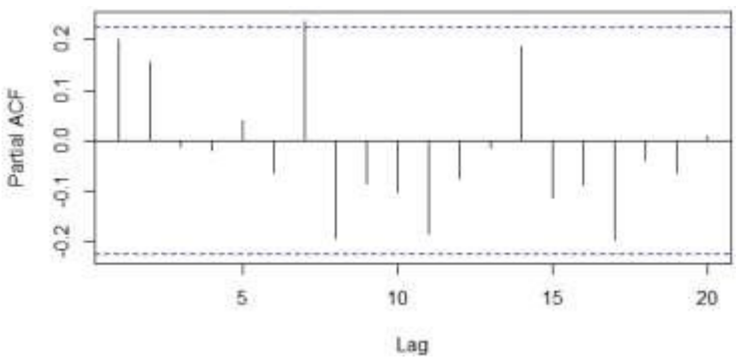

Figure 15 PACF Money Withdrawal Data at K4 ATM Machine

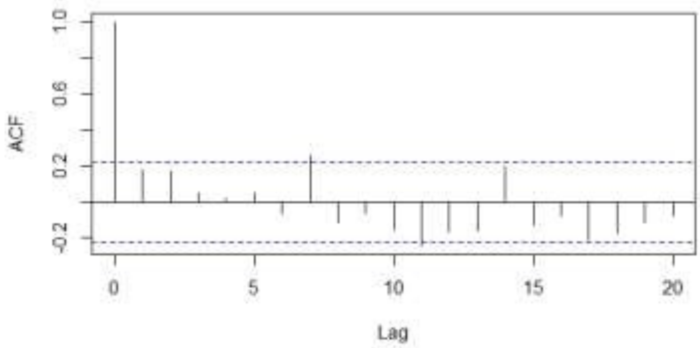

Figure 16. ACF Money Withdrawal Data at K5 ATM Machine

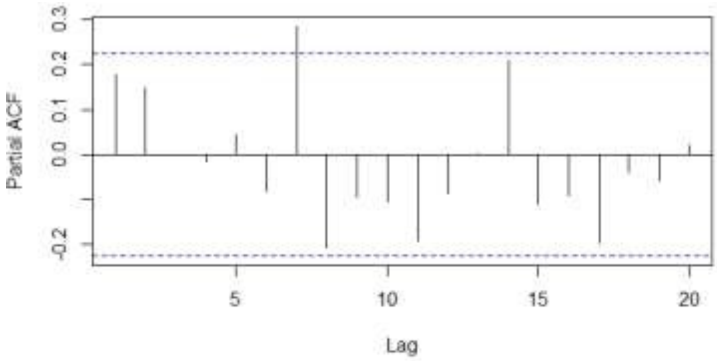

Figure 17. PACF Data on Withdrawal of Money at K5 ATM Machine

3.3 Data Transformation

Syntax data transformation for K1:

```
transformasidata=log(mydata$K1[1:76])
transformasidata
```

```
 [1] 18.18115 18.54201 18.50410 18.42766 18.66559 18.45990 18.05034
 [8] 18.72041 18.45072 18.51735 18.55039 18.53267 18.56435 18.01796
[15] 18.61295 18.38142 18.54467 18.61047 18.30077 18.67105 17.95307
[22] 18.69414 18.63297 18.34326 19.13265 18.46470 18.92206 18.20377
[29] 18.78359 18.89971 19.26440 18.94393 18.82481 18.76958 18.28142
[36] 18.75358 18.66715 18.46899 18.47327 18.41667 18.64981 18.00819
[43] 18.49300 18.48787 18.50640 18.48225 18.61666 18.35133 17.81279
[50] 18.54599 18.65970 18.45557 18.17478 18.54113 18.73366 18.08381
[57] 18.35294 18.95776 19.39051 18.97082 18.93679 18.63620 18.12429
[64] 18.57297 18.45218 18.47659 18.37830 18.58365 18.54290 18.04088
[71] 18.63901 18.22952 18.43261 18.69071 18.80424 18.25167
```

Figure 18. Data Transformation Results In K1

Syntax data transformation for K2:

```
transformasidata=log(mydata$K2[1:76])
transformasidata
```

```
 [1] 18.08100 18.47327 18.43261 18.35026 18.60508 18.38505 17.93536
 [8] 19.24244 18.37516 18.44684 18.48225 18.46326 18.49718 17.89896
[15] 18.54907 18.30021 18.47612 18.54643 18.21243 18.61089 17.82557
[22] 18.63539 18.57039 18.25875 19.09515 18.39022 18.87557 17.94747
[29] 18.73000 18.85214 19.23161 18.89847 18.77345 18.71521 18.09910
[36] 18.69831 18.60675 18.39485 18.39946 18.33839 18.58831 17.88795
[43] 18.42068 18.41517 18.43508 18.40911 18.55302 18.26753 17.66459
[50] 18.47753 18.59883 18.38038 18.07396 18.47232 18.67726 17.97283
[57] 18.26928 18.91293 19.36166 18.92659 18.89100 18.57383 18.01796
[64] 18.50640 18.37673 18.40303 18.29682 18.51781 18.47422 17.92474
[71] 18.57683 18.13433 18.35561 18.63175 18.75178 18.15867
```

Figure 19. Data Transformation Results in K2

Syntax data transformation for K3:

```
transformasidata=log(mydata$K3[1:76])
transformasidata
```

```
 [1] 19.05620 19.22268 19.20367 19.16642 19.28715 19.18195 19.00373
 [8] 19.31697 19.17751 19.21027 19.22693 19.21796 19.23405 18.99138
[15] 19.25923 19.14460 19.22403 19.25793 19.10781 19.29009 18.96736
[22] 19.30259 19.26976 19.12699 19.56459 19.18429 19.39599 18.89443
[29] 19.35225 19.41966 19.65207 19.44618 19.37581 19.34434 19.09922
[36] 19.33537 19.28799 19.18638 19.18847 19.16119 19.27871 18.98770
[43] 19.19817 19.19564 19.20481 19.19287 19.26117 19.13069 18.91873
[50] 19.22470 19.28399 19.17985 19.05355 19.22223 19.32429 19.01677
[57] 19.13142 19.45458 19.73937 19.46255 19.44187 19.27147 19.03289
[64] 19.23848 19.17821 19.19009 19.14314 19.24398 19.22313 19.00010
[71] 19.27297 19.07665 19.16879 19.30072 19.36400 19.08620
```

Figure 20. Data Transformation Results In K3

Syntax data transformation for K4:

```
transformasidata=log(mydata$K4[1:76])
transformasidata
```

```
 [1] 19.55015 19.70771 19.64269 19.61884 19.69730 19.62875 19.51846
 [8] 19.71719 19.62590 19.64695 19.65775 19.65193 19.66238 19.51109
[15] 19.67886 19.60501 19.65586 19.67800 19.58195 19.69925 19.49688
[22] 19.70757 19.68580 19.59394 19.88971 19.63024 19.79642 19.55595
[29] 19.74097 19.78715 19.95367 19.80560 19.75700 19.73561 19.57662
[36] 19.72955 19.69786 19.63158 19.63292 19.61551 19.69170 19.50891
[43] 19.63915 19.63752 19.64343 19.63574 19.68014 19.59625 19.46853
[50] 19.65630 19.69521 19.62740 19.54853 19.65470 19.72210 19.52627
[57] 19.59672 19.81147 20.01896 19.81705 19.80259 19.68692 19.53599
[64] 19.66527 19.62635 19.63396 19.60409 19.66886 19.65528 19.51629
[71] 19.68791 19.56267 19.62034 19.70633 19.74895 19.56856
```

Figure 21. Data Transformation Results in K4

Syntax data transformation for K5:

```
transformasidata=log(mydata$K5[1:76])
transformasidata
```

```
 [1] 18.88127 19.07665 19.05461 19.01124 19.15088 19.02936 18.81845
 [8] 19.18499 19.02418 19.06227 19.08156 19.07118 19.08979 18.80356
[15] 19.11882 18.98571 19.07820 19.11732 18.94245 19.15425 18.77450
[22] 19.16855 19.13093 18.96504 19.46308 19.03208 19.31636 18.89256
[29] 19.22515 19.30134 19.55947 19.33115 19.25185 19.21616 18.93231
[36] 19.20595 19.15185 19.03451 19.03695 19.00513 19.14120 18.79912
[43] 19.04822 19.04528 19.05593 19.04206 19.12105 18.96938 18.71521
[50] 19.07898 19.14726 19.02691 18.87811 19.07613 19.19333 18.83411
[57] 18.97025 19.34057 19.65484 19.34949 19.32631 19.13289 18.85344
[64] 19.09490 19.02500 19.03883 18.98400 19.10125 19.07716 18.81407
[71] 19.13461 18.90557 19.01401 19.16642 19.23848 18.91690
```

Figure 22. Data Transformation Results In K5

After the data transformation, then performed a search for the best model. Transforming data by using the Box-Cox transformation method if the data used is not stationary in variants. Differencing data when the data used is not stationary in average. Furthermore, check the stationary data that has been transformed or differentiated (differencing). This stage is done up to stationary data.

3.4 Best Model And Forecasting

Sarima model is a development of ARIMA model by adding seasonal or seasonal influences. Next, the process of searching for the best model and predictive results using the best model. Here are the best model search results and prediction results from the best models by using RStudio.

The syntax used in RStudio to find the best model on K1 – K5 is as follows:

```
dataseasonal<-auto.arima(dataobjek, trace=TRUE, test="kpss", ic="bic")
plot(1:76,y=dataobjek,type="l",xaxt="n")
axis(1, at=1:83, labels=mydata$Tanggal[1:83],las=3)
```

3.4.1 Best Model K1-K5

Table 2. Best Model K1-K5

Best Model K1 – K5				
Model K1	**Model K2**	**Model K3**	**Model K4**	**Model K5**
ARIMA(2,0,2)(1,0,1)[7]	ARIMA(2,0,2)(1,0,1)[7]	ARIMA(2,0,2)(1,0,1)[7]	ARIMA(2,0,2)(1,0,1)[7]	ARIMA(2,0,2)(1,0,1)[7]
ARIMA(0,0,0)	ARIMA(0,0,0)	ARIMA(0,0,0)	ARIMA(0,0,0)	ARIMA(0,0,0)
ARIMA(1,0,0)(1,0,0)[7]	ARIMA(1,0,0)(1,0,0)[7]	ARIMA(1,0,0)(1,0,0)[7]	ARIMA(1,0,0)(1,0,0)[7]	ARIMA(1,0,0)(1,0,0)[7]
ARIMA(0,0,1)(0,0,1)[7]	ARIMA(0,0,1)(0,0,1)[7]	ARIMA(0,0,1)(0,0,1)[7]	ARIMA(0,0,1)(0,0,1)[7]	ARIMA(0,0,1)(0,0,1)[7]
ARIMA(0,0,0)	ARIMA(0,0,0)	ARIMA(0,0,0)	ARIMA(0,0,0)	ARIMA(0,0,0)
ARIMA(1,0,0)	ARIMA(1,0,0)	ARIMA(1,0,0)	ARIMA(1,0,0)	ARIMA(1,0,0)
ARIMA(1,0,0)(2,0,0)[7]	ARIMA(1,0,0)(2,0,0)[7]	ARIMA(1,0,0)(2,0,0)[7]	ARIMA(1,0,0)(2,0,0)[7]	ARIMA(1,0,0)(2,0,0)[7]
ARIMA(1,0,0)(2,0,1)[7]	ARIMA(1,0,0)(2,0,1)[7]	ARIMA(1,0,0)(2,0,1)[7]	ARIMA(1,0,0)(1,0,1)[7]	ARIMA(1,0,0)(2,0,1)[7]
ARIMA(1,0,0)(1,0,1)[7]	ARIMA(1,0,0)(1,0,1)[7]	ARIMA(1,0,0)(1,0,1)[7]	ARIMA(1,0,0)(0,0,1)[7]	ARIMA(1,0,0)(1,0,1)[7]
ARIMA(0,0,0)(2,0,0)[7]	ARIMA(0,0,0)(2,0,0)[7]	ARIMA(0,0,0)(2,0,0)[7]	ARIMA(1,0,0)(2,0,1)[7]	ARIMA(0,0,0)(2,0,0)[7]

ARIMA(2,0,0)(2,0,0)[7]	ARIMA(2,0,0)(2,0,0)[7]	ARIMA(2,0,0)(2,0,0)[7]	ARIMA(0,0,0)(1,0,0)[7]	ARIMA(2,0,0)(2,0,0)[7]
ARIMA(1,0,1)(2,0,0)[7]	ARIMA(1,0,1)(2,0,0)[7]	ARIMA(1,0,1)(2,0,0)[7]	ARIMA(2,0,0)(1,0,0)[7]	ARIMA(1,0,1)(2,0,0)[7]
ARIMA(0,0,1)(2,0,0)[7]	ARIMA(0,0,1)(2,0,0)[7]	ARIMA(0,0,1)(2,0,0)[7]	ARIMA(1,0,1)(1,0,0)[7]	ARIMA(0,0,1)(2,0,0)[7]
ARIMA(2,0,1)(2,0,0)[7]	ARIMA(2,0,1)(2,0,0)[7]	ARIMA(2,0,1)(2,0,0)[7]	ARIMA(0,0,1)(1,0,0)[7]	ARIMA(2,0,1)(2,0,0)[7]
ARIMA(1,0,0)(2,0,0)[7]	ARIMA(1,0,0)(2,0,0)[7]	ARIMA(1,0,0)(2,0,0)[7]	ARIMA(2,0,1)(1,0,0)[7]	ARIMA(1,0,0)(2,0,0)[7]
			ARIMA(1,0,0)(1,0,0)[7]	
Best model: ARIMA(1,0,0)(2,0,0)[7	Best model: ARIMA(1,0,0)(2,0,0)[7]	Best model: ARIMA(1,0,0)(2,0,0)[7]	Best model: ARIMA(1,0,0)(1,0,0)[7]	Best model: ARIMA(1,0,0)(2,0,0)[7]

3.4.2 Predicted Result

After knowing the best model results obtained from the SARIMA model, it can be known the data of prediction results in the next 7 periods in K1 - K5. For synnot prediction on ATM machine number K1 – K5 on RStudio and the result of prediction can be seen as follows:

Predictive syntax for each K1 K1 – K5

```
dataseasonal<-auto.arima(dataobjek, trace=TRUE, test="kpss", ic="bic")
```

Table 3. Predicted Result

Predicted Result				
Model K1	**Model K2**	**Model K3**	**Model K4**	**Model K5**
73462838	68071879	187074918	319802322	155850576
110572300	104411015	221560505	344700600	190560301
95109530	89662122	208071306	332775225	177245201
103903567	97247609	215538697	339739910	184988006
110974055	102047952	221875643	349645649	191429879
122694805	113958992	232346697	354625757	202020352
100039622	94251545	212593821	334289044	182118182

3.5 *Mean Absolute Percentage Error* (MAPE)

MAPE is calculated using absolute errors in each period divided by real observation values for that period. Then, the absolute percentage error averages. MAPE is an error measurement that calculates the percentage size of deviations between actual data and forecasting data [4].

$$MAPE = \left(\frac{100\%}{n}\right)\sum_{t=1}^{n} \frac{|Xt-Ft}{Xt}$$

For:

Xt = actual data in period t
Ft = forecast value in period t
n = amount of data

Forecasting capability is very good if it has a MAPE value of less than 10% and has good forecasting capability if the MAPE value is less than 20% [3]. For syntax looking mape (Mean Absolute Percent Error) K1 in RStudio can be seen as follows:

```
df<-data.frame(mydata$K1[77:83],finalforecastvalues)
col_headings<-c("Data Aktual","Data Prediksi")
names(df)<-col_headings
attach(df)
percentage_error=(abs(df$`Data Aktual`-df$`Data Prediksi`)/(df$`Data
Aktual`))
nilaier<-
data.frame(mydata$K1[77:83],finalforecastvalues,percentage_error)
col_headings<-c("Data Aktual","Data Prediksi","Error")
names(nilaier)<-col_headings
nilaier
MAPE<-mean(percentage_error)
MAPE
```

Table 4. MAPE Calculation Results

Mape 1st Calculation Result			
Number	Actual Data	Prediction Data	Error
1	62750000	73462838	0.17072252
2	118150000	110572300	0.06413627
3	110800000	95109530	0.14161074
4	96200000	103903567	0.08007865
5	78750000	110974055	0.40919435
6	152750000	122694805	0.19676069
7	112450000	100039622	0.11036352
2nd MAPE Calculation Result			
Number	Actual Data	Prediction Data	Error
1	55250000	68071879	0.23207021
2	110650000	104411015	0.05638486
3	103300000	89662122	0.13202206
4	88700000	97247609	0.09636538
5	71250000	102047952	0.43225195
6	145250000	113958992	0.21542863
7	104950000	94251545	0.10193859
3nd MAPE Calculation Result			
Number	172850000	187074918	0.08229631

1	228250000	221560505	0.02930775
2	220900000	208071306	0.05807467
3	206300000	215538697	0.04478282
4	188850000	221875643	0.17487764
5	262850000	232346697	0.11604833
6	222550000	212593821	0.04473682
7	172850000	187074918	0.08229631
4nd MAPE Calculation Result			
Number	Actual Data	Prediction Data	Error
1	293450000	319802322	0.08980175
2	348850000	344700600	0.01189451
3	341500000	332775225	0.02554839
4	326900000	339739910	0.03927779
5	309450000	349645649	0.12989384
6	383450000	354625757	0.07517080
7	343150000	334289044	0.02582240
5nd MAPE Calculation Result			
Number	Actual Data	Prediction Data	Error
1	142550000	155850576	0.09330464
2	197950000	190560301	0.03733114
3	190600000	177245201	0.07006715
4	176000000	184988006	0.05106822
5	158550000	191429879	0.20737861
6	294670000	202020352	0.31441833
7	192250000	182118182	0.05270127

3.6 Result

Based on the results of mining data and evaluation above, table can be created that contains prediction results using seasonal autoregressive integrated moving average (SARIMA) method.

Table 5. Predictive results using SARIMA Method

Nomor ATM	Model Terbaik	MAPE
K1	ARIMA(1,0,0)(2,0,0)[7]	16.75%
K2	ARIMA(1,0,0)(2,0,0)[7]	18.09%
K3	ARIMA(1,0,0)(2,0,0)[7]	7.85%
K4	ARIMA(1,0,0)(1,0,0)[7]	5.67%
K5	ARIMA(1,0,0)(2,0,0)[7]	11.80%

This study aims to find out the similarity of seasonal autoregressive integrated moving average (SARIMA) prediction model that is suitable to predict the amount of cash withdrawal at atm machines. Then, to obtain the error level comparison data predicted the amount of cash withdrawals at atm machines with seasonal autoregressive integrated moving average (SARIMA) method. The dataset used is an ATM transaction dataset that has 17 attributes and which is used only 4 attributes. The data used as much as 415 data is taken from 5 ATM machine numbers. Before the data mining process and evaluation of data first in filterization then in the sharing of data testing and data training where for data testing has a total of 76 data per each ATM machine number while for data testing has a total of 7 data per each ATM machine number with a total of every 83 data for each ATM machine number. Atm machine number is symbolised with the letters K, namely K1, K2, K3, K4, and K5.

Researchers made predictions with 415 data using seasonal autoregressive integrated moving average (SARIMA) method where the data is a combination of data from several ATM machine numbers. Then the data mining process is carried out in each ATM number until it gets the best model on the Seasonal Autoregressive Integrated Moving Average (SARIMA) method. Furthermore, the best model is predicted to get the results of prediction data in the future. Then to lookup the MAPE value the actual data and the predictive data are calculated using a formula so that in can mape values. From the implementation using ATM transaction dataset by using 5 ATM machine numbers, mape K1 with MAPE value of 16.75%, MAPE K2 of 18.09%, MAPE K3 of 7.85%, MAPE K4 of 5.67%, and MAPE K5 of 11.80%.

4. CONCLUSION

The performance of the Seasonal Autoregressive Integrated Moving Average (SARIMA) method results in a suitable or good model used to predict cash withdrawal data at ATM machines with the best model K1 being ARIMA(1,0,0)(2,0,0)[7], for the K2 the best model is ARIMA(1,0,0)(2,0,0)[7], the best K3 model ARIMA(1,0,0)(2,0,0)[7], the K4 best model ARIMA(1,0,0)(1,0,0)[7] , and the K5 model is ARIMA(1,0,0)(2,0,0)[7].

The result of error using MAPE (Mean Absolute Percenttage Error) on the predicted result of cash withdrawal data at atm machines is K1 16.75%, K2 18.09%, K3 7.85%, K4 5.67%, and K5 11.80%. So it can be concluded that the data matches using the SARIMA model that has been selected because the MAPE value is smaller than 20%.

5. SUGGESTIONS

It is expected that the next research can try to implement the Seasonal Autoregressive Integrated Moving Average (SARIMA) method by using other data that has more time.

LIBRARY LIST

[1] Abdullah, Dahlan DKK "**Implementasi Metode Rough Set Untuk Menentukan Data Nasabah Potensial Mendapatkan Pinjaman**", *Universitas Putra Indonesia YPTK Padang. ISSN : 2460 – 4690. Vol. 1. 2015.*

[2] Aprilla, C., D. A. Baskoro, L. Ambarwati, & I W. S. Wicaksana., "**Belajar Data mining dengan RapidMiner",** Jakarta, 2013.

[3] Margi, Kristien. "**Analisa dan Penerapan Metode Single Exponential Smoothing Untuk Prediksi Penjualan Pada Periode Tertentu**". https://jurnal.umk.ac.id (diakses pada tanggal 7 November 2020).

[4] Purwanto, Hari & Agustina, Rini, "**Sistem Informasi Peramalan Transaksi Tunai ATM Menggunakan Triple Exponential Smoothing**", Universitas Kanjuruhan Malang, *ISSN : 2615-8388,* 2019.

[5] Zaki, Mohammed J., Meira Jr, Wagner "**Kernel Data Mining and Analysis**", Cambridge University Press: Cambridge. 2014.