Nick Puglisi and Student B.
STAT 4280

A Time Series Analysis of COVID-19 Vaccine Uptake in Georgia

Introduction

Coronavirus disease 2019 (COVID-19) is a disease caused by the severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2). After being discovered in Wuhan, China in 2019, it quickly spread across the world and became a world-shifting pandemic. On January 20, 2020, the CDC confirmed that COVID-19 had made it to the United States[1], and on March 2, 2020, it was announcedthat the disease had reached the state of Georgia[2]. This recent pandemic givesepidemiological researchers a chance to work with real data concerning disease spread and human response to the novel disease.

One aspect of human response, a primary focus of this paper's investigation, is vaccine uptake. The development of the COVID-19 is a tale of dramatic speed. On December 11, 2020, the United States' Food and Drug Administration granted an emergency use authorization to the first COVID-19 vaccine[1], allowing its promise of curtailing the pandemic to be delivered nationwide.

Our data come from the CDC, which tracked the number of vaccines administered in each state from December 14, 2020, onward. We will specifically be looking at the trend of vaccine uptake in Georgia, a state in which political debates played a role in the decision making process of citizens.

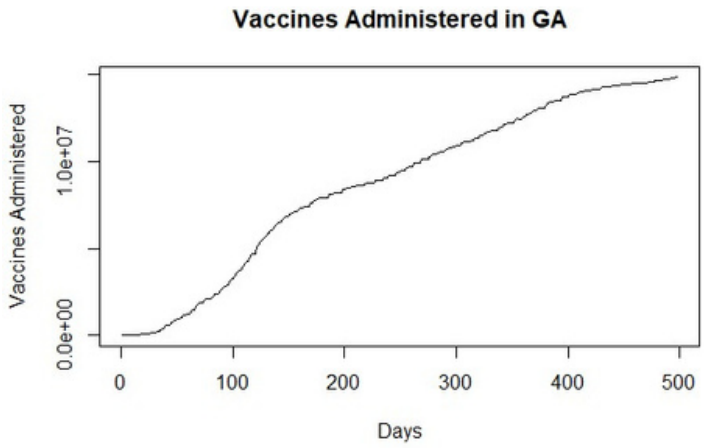
Exploratory Data Analysis

Figure 1 is a time series plot of the cumulative number of vaccines administered in the state of Georgia. The plot shows a fairly smooth trend for the majority of our

[1]https://www.cdc.gov/museum/timeline/covid19.html#:~:text=January%2020%2C%202020%20CDC,18%20in%20Washington%20state
[2] https://dph.georgia.gov/press-releases/2020-03-02/gov-kemp-officials-confirm-two-cases-covid-19-georgia

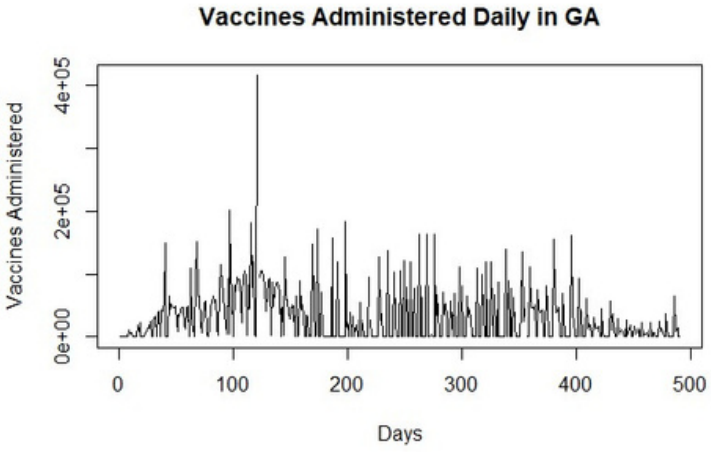
period of analysis, but there is a bit of a taper at both the beginning and the end of our data period. These periods indicate a potentially logarithmic character in our data.

Figure 1



While the overall number is important, our analysis will difference the cumulative counts in order to observe the number of new vaccines administered on each day during our period of study. Figure 2 shows a plot of this differenced data.

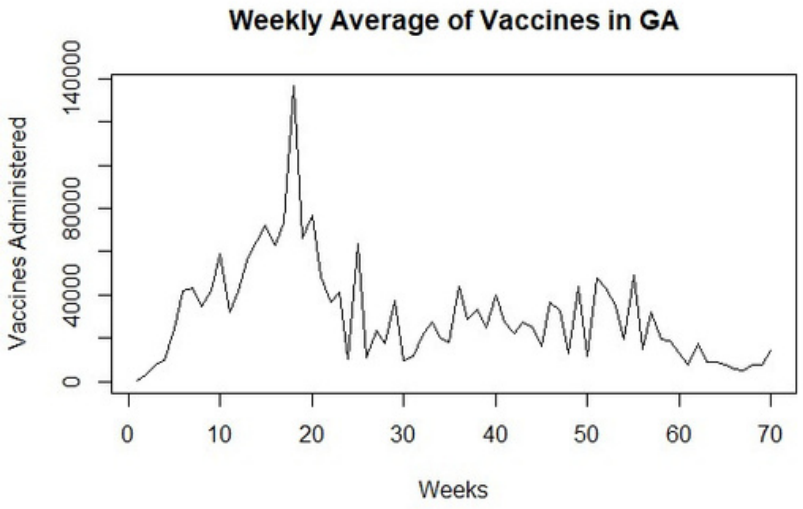
Figure 2



The figure shows a large degree of variation in our daily data (In the interest of transparency, our difference resulted in 3 days where the total number of vaccines

administered was negative. Since that is impossible, we removed those points). In order to focus more clearly on the overall trends in vaccine uptake, we wanted to make sure our analysis was not highly dependent on days where data collection might be skewed (holidays, weekends, and other days where data collection was just out of the ordinary), so we decided to work with weekly averages. Figure 3 displays these weekly averages

Figure 3



Cleaning the Data

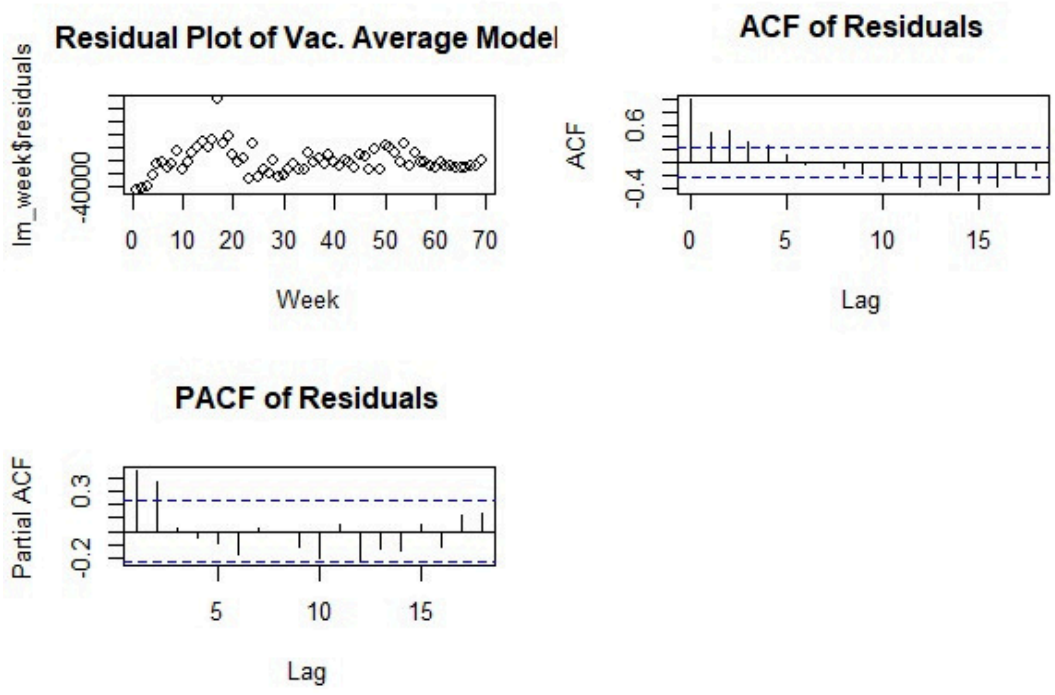
The weekly averaged data still exhibit the characteristic spike in administered vaccine count at week 17 - 18, which creates abnormal ACF and PACF plots. In order to remedy this, smoothing methods and some basic data cleaning will be applied to the time series data to allow for further analysis of residuals. The first smoothing technique applied to the data is a normal log transform. Log transforming the data preserves the general structure while allowing for easier analysis. In addition to smoothing techniques, both the first and last week of the vaccine administration data will be removed. The first week of data, ranging from 12/14/2020 to 12/21/2020, had few

observations recorded due to a lag in vaccine administration. The last week of data present also has few observations recorded, with a weekly average of only 108 administered vaccines. The team believes this unusual observation is caused by an incomplete week's worth of data when last pulled from the CDC's data repository. Removing these two weeks leaves a total of 69 weekly averages

Addressing Nonstationarity

To begin our analysis, we first fit a linear model over time to our data and then performed analysis on our residuals. The resulting analysis is shown in figure 4.

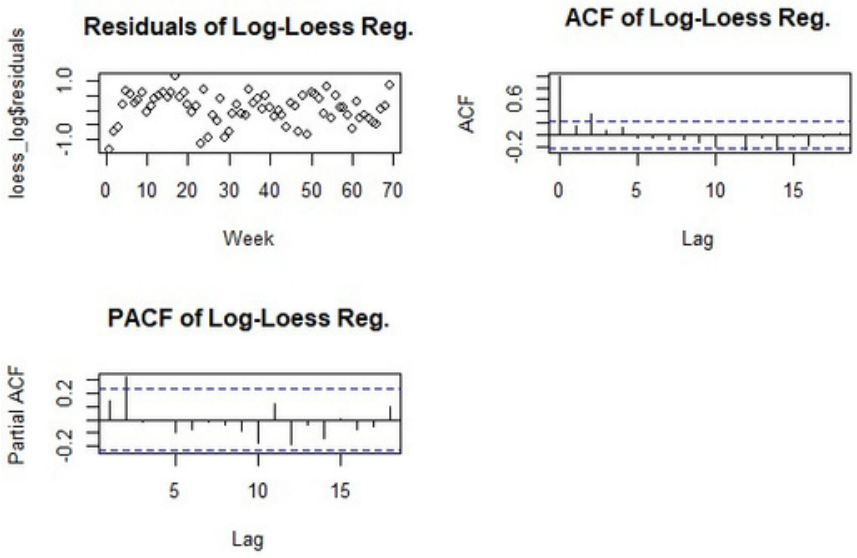
Figure 4



The resulting residual plots and (P)ACFs show that our data is lacking stationarity and that our linear model does not do a good job of accounting for the relationship we have. Therefore, we tested various transformations in an attempt to address this problem. Eventually, we decided that both a log transformation and the

nonparametric loess regression technique worked best to give us a fairly stationary data set. The resulting residual plots and (P)ACFs are shown below in figure 5.

Figure 5



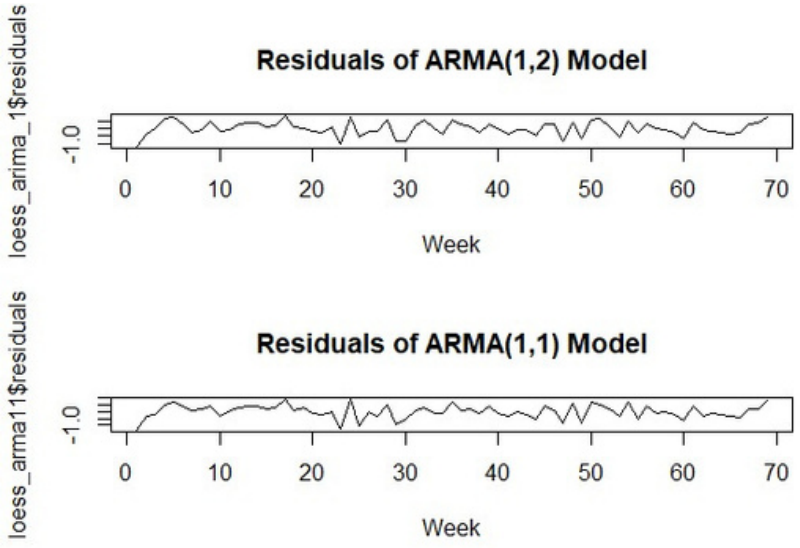
It is worth noting that, even after our best effort at transforming the data, the residual plots and (P)ACFs do not look perfect. There is still a spike at lag 3 of the ACF and at lag 2 of the PACF. Additionally, the residual plot does not look perfectly random. That said, these are the problems that happen when we work with real data as opposed to the much prettier theoretical data we worked with during class. We decided that our residuals looked good enough to proceed with attempting to fit a model.

Fitting the Models

To begin fitting the models we put the residuals from our log-loess regression into R's*auto.arima*function as a starting point inour model fitting process. The function suggested we use an ARMA(1,2) model, but in the interest of parsimony, we analyzed an ARMA(1,1) model to determine which had a better fit. The results of this

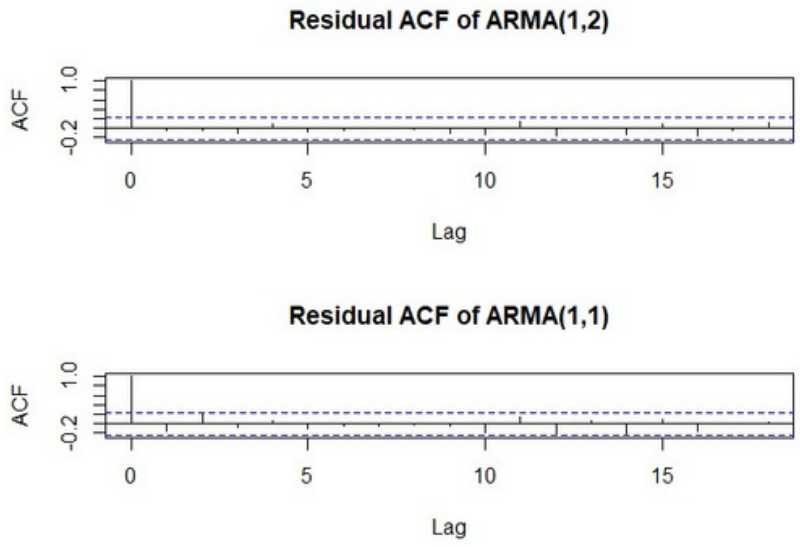
comparison are shown in the following figures, starting with a comparison of the residual plots of the fitted models in figure 6.

Figure 6



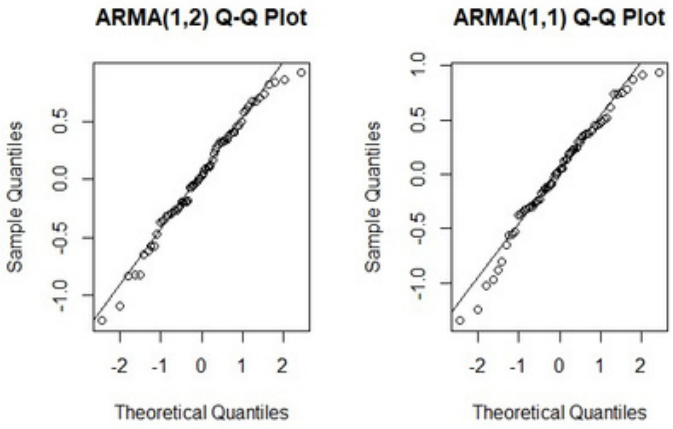
The residual plots of the models look pretty similar (both look fairly random, as they should), so we then compared the ACFs of the models in figure 7.

Figure 7



The ACF plot is where the ARMA(1,2) begins to show its strength. The ARMA(1,1) ACF has a fairly strong spike around lag 3. While this spike is not the most concerning thing, it weakens the model in comparison to the ARMA(1,2) model. Finally, in figure 8, we compare the Q-Q plots of the two models.

Figure 8



The Q-Q plot for the ARMA(1,1) has significantly more fanning than the Q-Q plot for the ARMA(1,2), suggesting that the ARMA(1,2)'s residuals better follow the assumption of normality.

While comparing these graphs was useful, we wanted to compare numbers before finally concluding that the ARMA(1,2) is a better model. Table 1 compares the model diagnostics from our two models.

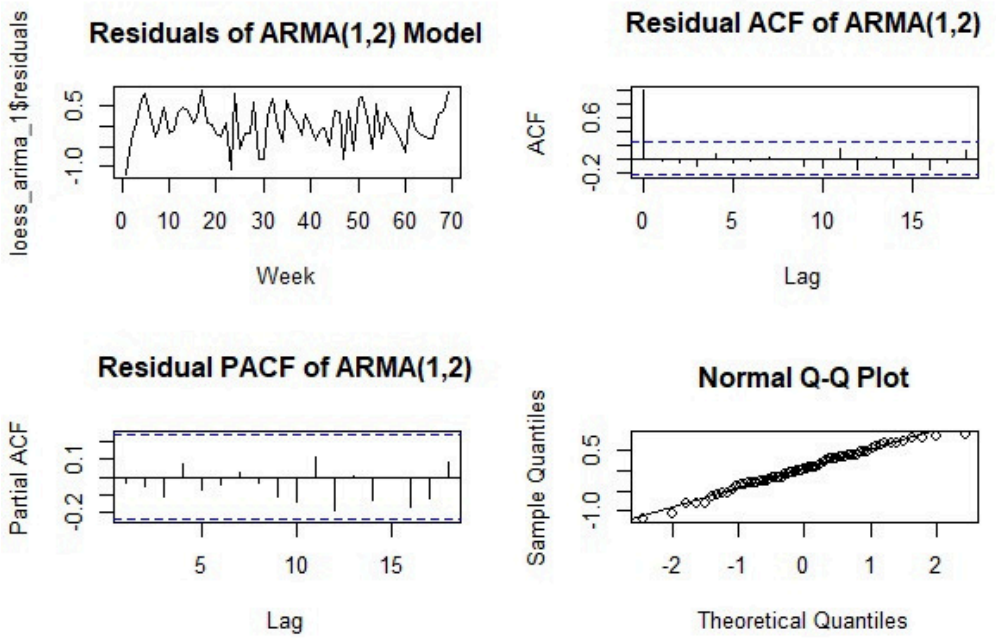
Table 1

	AIC	**sigma^2**	**Log Likelihood**
ARMA(1,2)	102.22	0.238	-47.11
ARMA(1,1)	108.89	0.252	-50.44

Our ARMA(1,2) has a lower AIC score, a lower variance, and a lower log likelihood, suggesting that our ARMA(1,2) model is an appropriately parsimonious model for our data.

Figure 9 summarizes the fit of our ARMA(1,2) model.

Figure 9



Model Applications and Future Analysis

A common application of time series models is to forecast future observations. However, we believe that using the ARMA(1,2) model to forecast the future number of vaccines administered in Georgia is not feasible. There is a "carrying capacity" to the number of people in the state who would be willing to or are able to get vaccinated. The initial daily count data suggests that there is a downward trend in the number of people who are getting it, and the team's hypothesis is that there will be a flatline in vaccine administration. Yet, if there is governmental approval for future vaccine booster doses

or the recommended age range for vaccine administration is expanded, leading to potential increases in administered doses.

COVID-19 is still a new disease, and much is still not understood about our future with it. One possibility is that COVID-19 becomes a “new” flu, where humans experience some seasonality with the disease. If this is the case, future analysis into the human uptake of vaccines would be possible and could provide great epidemiological insight.

APPENDIX -- R CODES

```
# STAT 4280 TIME SERIES FINAL
library(astsa)
library(xts)
library(knitr)
library(forecast)

## DATA IMPORTATION AND CLEANING
vaccines_us = read.csv("all_vaccines.csv")
attach(vaccines_us)
vaccines_ga = vaccines_us[Location == "GA",]
ts.plot(rev(vaccines_ga$Administered), gpars = list(xlab = "Days",
          ylab = "Vaccines Administered",main = "Vaccines Administered in GA"))
vaccines_ga = vaccines_ga[-c(1:8),]
#differencing data
admin_count = vaccines_ga$Administered
lag_count = admin_count[-1]
lag_count = c(lag_count,0)
daily_admin = admin_count - lag_count
daily_admin = replace(daily_admin, daily_admin < 0 , NA)
which(daily_admin == 'NA')
ts.plot(rev(daily_admin), gpars = list(xlab = "Days",
                                       ylab = "Vaccines Administered",main = "Vaccines Administered Daily in GA"))
acf(rev(daily_admin), na.action = na.pass, main = "ACF of Daily Vaccine Data")
pacf(rev(daily_admin), na.action = na.pass, main = "PACF of Daily Vaccine Data")

## GROUPING
index = rep(c(1:70), each = 7)
index = rev(index)
vaccines_ga = cbind(index,daily_admin, vaccines_ga)

weekly_admin = aggregate(vaccines_ga$daily_admin, list(vaccines_ga$index),
                         mean, na.rm = T)
weekly_avg = weekly_admin[,2] week_covid =
weekly_admin[,1] ts.plot(weekly_admin[,2], gpars =
list(xlab = "Weeks",
          ylab = "Vaccines Administered",main = "Weekly Average of Vaccines in GA"))
acf(weekly_avg)
pacf(weekly_avg)
log_week = log(weekly_avg)
ts.plot(log_week)
## REMOVING FIRST WEEK
#CHECKING RESIDUALS OF AVG ~ WEEK
weekly_avg = weekly_avg[-1]
week_covid = week_covid[-1]
lm_week = lm(weekly_avg~week_covid)
par(mfrow = c(2,2))
plot(lm_week$residuals, main = 'Residual Plot of Vac. Average Model', xlab = "Week")
acf(lm_week$residuals, main = "ACF of Residuals")
pacf(lm_week$residuals, main = "PACF of Residuals")
# LOG TRANSFORM
log_weekavg = log(weekly_avg)
log_lm = lm(log_weekavg~week_covid)
ts.plot(log_lm$residuals)
acf(log_lm$residuals)
```

```
pacf(log_lm$residuals)
## NON_LOG LOESS
loe_weekly = loess(weekly_avg~week_covid)
ts.plot(loe_weekly$residuals)
acf(loe_weekly$residuals)
pacf(loe_weekly$residuals)
## LOG LOESS
loe_log = loess(log_weekavg ~ week_covid)
ts.plot(loe_log$residuals)
acf(loe_log$residuals)
pacf(loe_log$residuals)
plot(loe_log$residuals)
## REMOVING FIRST WEEK
weekly_avg = weekly_avg[-1]
week_covid = week_covid[-1]
#log_transform
weekly_log = log(weekly_avg)
loess_log = loess(weekly_log ~ week_covid)
ts.plot(loess_log$residuals)
par(mfrow = c(2,2))
plot(loess_log$residuals, main = "Residuals of Log-Loess Reg.", xlab = "Week")
acf(loess_log$residuals, main = "ACF of Log-Loess Reg.")
pacf(loess_log$residuals, main = "PACF of Log-Loess Reg.")
## Model testing and fitting
# AUTO ARIMA MODEL
loess_arima_1 = auto.arima(loess_log$residuals)
par(mfrow = c(2,2))
plot(loess_arima_1$residuals, main = "Residuals of ARMA(1,2) Model", xlab = "Week")
acf(loess_arima$residuals, main = "Residual ACF of ARMA(1,2)")
pacf(loess_arima$residuals, main = "Residual PACF of ARMA(1,2)")
qqnorm(loess_arima$residuals)
qqline(loess_arima$residuals)
# ARMA(1,1)
loess_arma11 = arima(loess_log$residuals, order = c(1,0,1))
par(mfrow = c(2,2))
plot(loess_arma11$residuals, main = "Residuals of ARMA(1,1) Model", xlab = "Week")
acf(loess_arma11$residuals, main = "Residual ACF of ARMA(1,1)")
pacf(loess_arma11$residuals, main = "Residual PACF of ARMA(1,1)")
qqnorm(loess_arma11$residuals)
qqline(loess_arma11$residuals)

## COMPARING AMRA(1,1) and ARMA(1,2)
# Residual comparison
par(mfrow = c(2,1))
plot(loess_arima_1$residuals, main = "Residuals of ARMA(1,2) Model", xlab = "Week")
plot(loess_arma11$residuals, main = "Residuals of ARMA(1,1) Model", xlab = "Week")
# ACF Comparison
par(mfrow = c(2,1))
acf(loess_arima$residuals, main = "Residual ACF of ARMA(1,2)")
acf(loess_arma11$residuals, main = "Residual ACF of ARMA(1,1)")
# QQ_comparison
par(mfrow = c(1,2))
qqnorm(loess_arima_1$residuals, main = "ARMA(1,2) Q-Q Plot")
qqline(loess_arima_1$residuals)
qqnorm(loess_arma11$residuals, main = "ARMA(1,1) Q-Q Plot")
qqline(loess_arma11$residuals)
```