Untitled

```r
library(data.table)
```

```
## Warning: package 'data.table' was built under R version 3.6.3
```

```r
library(forecast)
```

```
## Warning: package 'forecast' was built under R version 3.6.3
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```r
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 3.6.3
```

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

```
## The following objects are masked from 'package:data.table':
##
##     between, first, last
```

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

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

```r
library(magrittr)
```

```
## Warning: package 'magrittr' was built under R version 3.6.3
```

```r
library(tseries)
```

```
## Warning: package 'tseries' was built under R version 3.6.3
```

```
library(gridExtra)
```

```
## Warning: package 'gridExtra' was built under R version 3.6.3

##
## Attaching package: 'gridExtra'

## The following object is masked from 'package:dplyr':
##
##     combine
```

```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 3.6.3
```

```
options(repr.plot.width=18, repr.plot.height=12)
```

```
temp = read.csv("global_surface_temperature_anomalies.csv", header=T)
TS_temp = ts(temp$Avg_Anomaly_deg_C, frequency = 12)
df_train = temp[1:1560,1:2]
df_test = temp[1561:1680,1:2]
TS_train = ts(df_train$Avg_Anomaly_deg_C, frequency = 12)
TS_test = ts(df_test$Avg_Anomaly_deg_C, frequency = 12)
TS_test
```

```
##     Jan  Feb  Mar  Apr  May  Jun  Jul  Aug  Sep  Oct  Nov  Dec
## 1  0.73 0.79 0.92 0.85 0.73 0.62 0.59 0.63 0.59 0.69 0.77 0.46
## 2  0.49 0.51 0.62 0.62 0.51 0.58 0.71 0.71 0.54 0.63 0.55 0.53
## 3  0.45 0.47 0.56 0.68 0.74 0.63 0.53 0.61 0.72 0.75 0.73 0.52
## 4  0.66 0.55 0.66 0.53 0.58 0.65 0.58 0.66 0.77 0.67 0.78 0.65
## 5  0.73 0.51 0.76 0.77 0.85 0.66 0.56 0.81 0.88 0.81 0.66 0.77
## 6  0.81 0.87 0.90 0.74 0.75 0.79 0.71 0.78 0.82 1.07 1.03 1.10
## 7  1.15 1.34 1.31 1.07 0.91 0.77 0.82 1.00 0.87 0.89 0.90 0.83
## 8  0.98 1.12 1.12 0.92 0.89 0.70 0.82 0.87 0.76 0.88 0.85 0.89
## 9  0.77 0.84 0.91 0.87 0.81 0.75 0.78 0.73 0.76 0.99 0.78 0.89
## 10 0.87 0.92 0.92 0.92 0.92 0.92 0.92 0.92 0.92 0.92 0.92 0.92
```

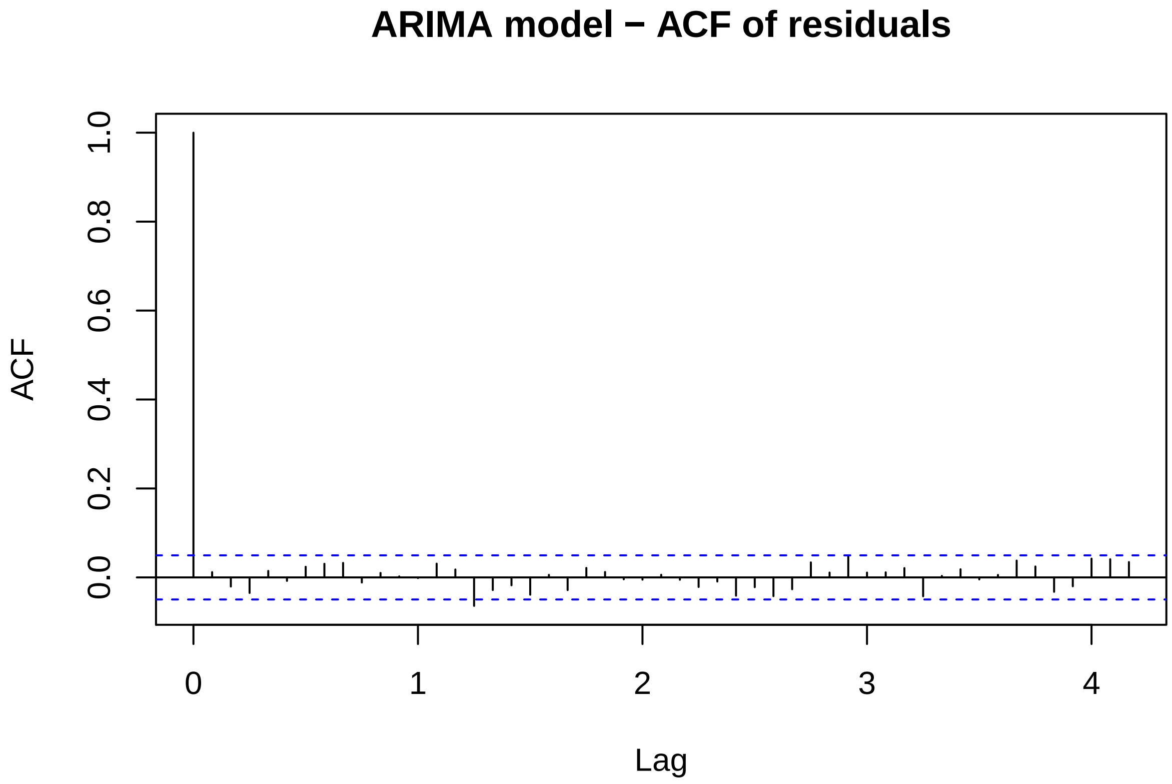
```
par(mfrow=c(1,2),mar=c(2.5,2.5,3.5,1.5))
```

Using auto ARIMA to find the best possible model to describe the time series.

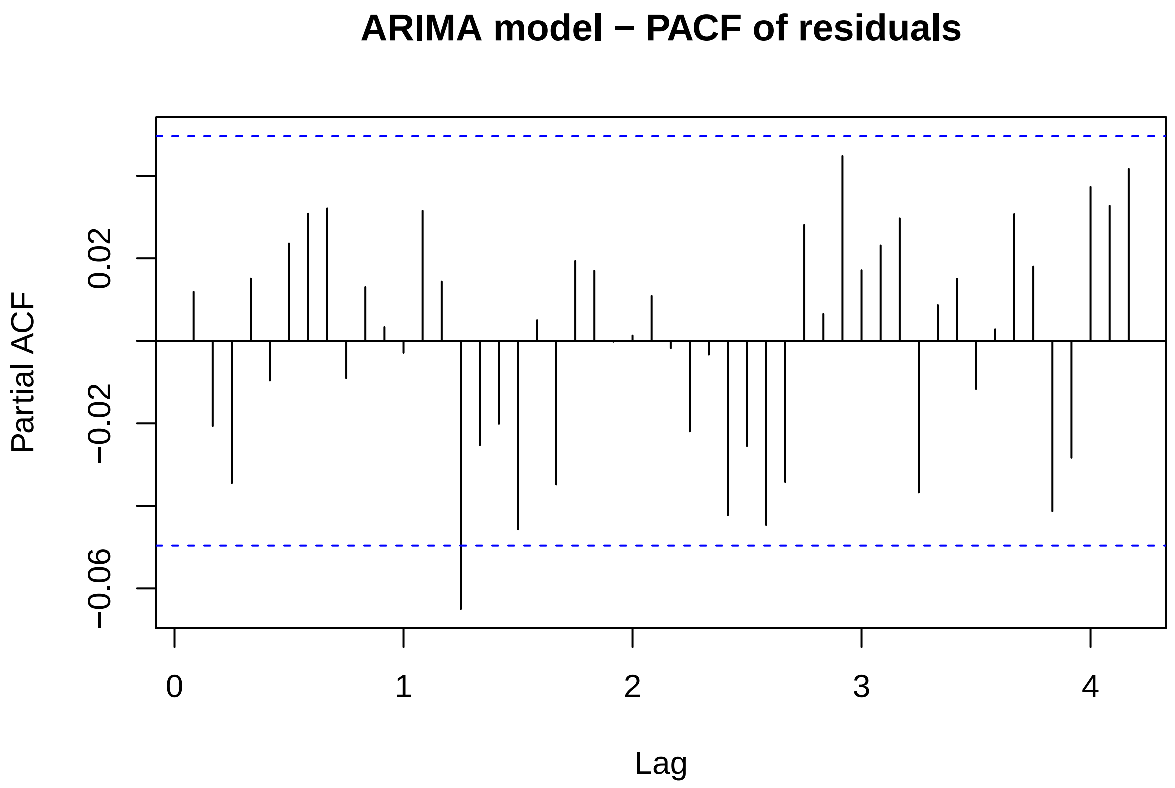
```
arima_temp = auto.arima(TS_train, stepwise=FALSE, approximation=FALSE)
arima_temp
```

```
## Series: TS_train
## ARIMA(1,1,2)(2,0,0)[12] with drift
##
## Coefficients:
##          ar1      ma1     ma2    sar1     sar2  drift
##        0.8600  -1.3734  0.3833  0.0505  0.1194  5e-04
## s.e.  0.0226   0.0381  0.0364  0.0254  0.0257  2e-04
##
## sigma^2 estimated as 0.01182:  log likelihood=1249.6
## AIC=-2485.2   AICc=-2485.13   BIC=-2447.74
```

```
acf(arima_temp$res,main="ARIMA model - ACF of residuals", lag.max=50)
```



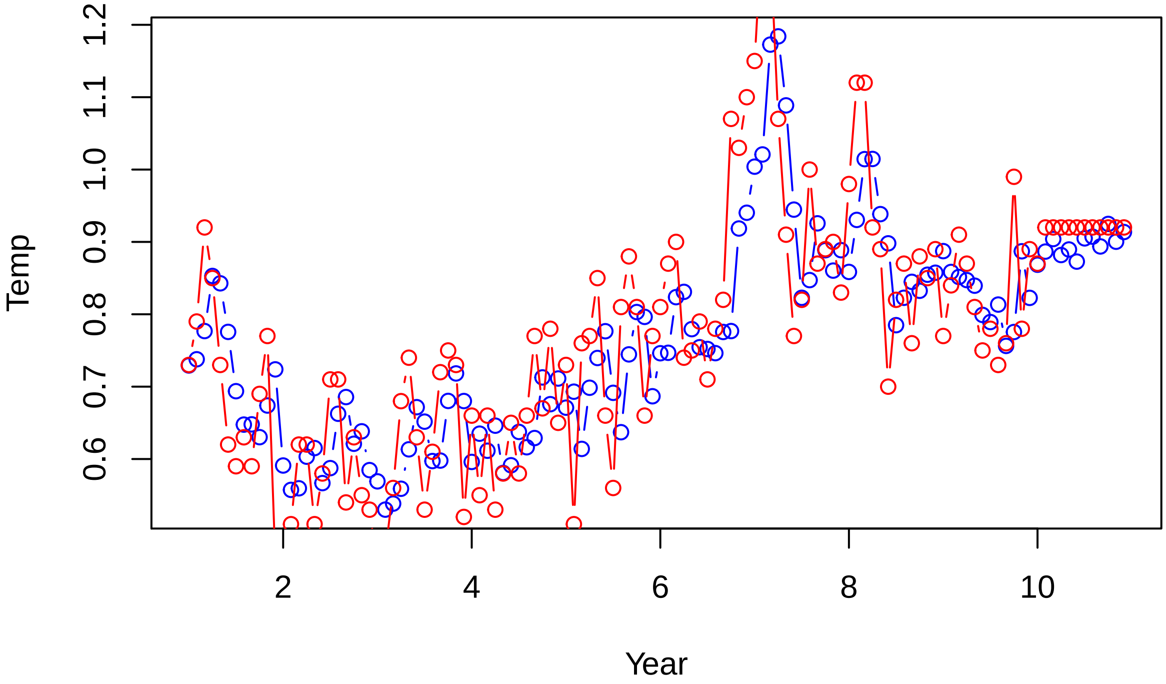
```
pacf(arima_temp$res,main="ARIMA model - PACF of residuals", lag.max=50)
```



```
forecast = Arima(TS_test, model = arima_temp)
accuracy(forecast)
```

```
##                       ME      RMSE        MAE        MPE     MAPE     MASE
## Training set 0.008302171 0.1014526 0.07974729 -0.7495301 10.78369 0.611698
##                   ACF1
## Training set 0.1854138
```

```
plot(forecast$fitted,type="b",col="blue", ylab="Temp",xlab="Year")
lines(TS_test,col="red",type="b")
```


Temp
0.6
0.7
0.8
0.9
1.0
1.1
1.2
2
4
6
8
10
Year