Assignment 5

Feiyu Zheng (fz114)

2022/2/21

Problem 1

1. Get the data in a single data frame. Create 3 data frames (or tibbles) from three files and combine the 3 data frames into one.

```
file1_t <- read.csv("https://raw.githubusercontent.com/jennybc/lotr-tidy/master/data/The_Fellowship_Of_T
  as_tibble()
file1_t
```

```
## # A tibble: 3 x 4
##   Film                       Race   Female  Male
##   <chr>                      <chr>   <int> <int>
## 1 The Fellowship Of The Ring Elf      1229   971
## 2 The Fellowship Of The Ring Hobbit     14  3644
## 3 The Fellowship Of The Ring Man         0  1995
```

```
file2_t <- read.csv("https://raw.githubusercontent.com/jennybc/lotr-tidy/master/data/The_Two_Towers.csv
  as_tibble()
file2_t
```

```
## # A tibble: 3 x 4
##   Film           Race   Female  Male
##   <chr>          <chr>   <int> <int>
## 1 The Two Towers Elf       331   513
## 2 The Two Towers Hobbit      0  2463
## 3 The Two Towers Man       401  3589
```

```
file3_t <- read.csv("https://raw.githubusercontent.com/jennybc/lotr-tidy/master/data/The_Return_Of_The_K
  as_tibble()
file3_t
```

```
## # A tibble: 3 x 4
##   Film                   Race   Female  Male
##   <chr>                  <chr>   <int> <int>
## 1 The Return Of The King Elf       183   510
## 2 The Return Of The King Hobbit      2  2673
## 3 The Return Of The King Man       268  2459
```

```
# bind three tibbles to one
untidy_combined_t <- rbind(file1_t, file2_t, file3_t)
untidy_combined_t
```

```
## # A tibble: 9 x 4
##   Film                       Race   Female  Male
```

```
##   <chr>                        <chr>   <int> <int>
## 1 The Fellowship Of The Ring Elf      1229   971
## 2 The Fellowship Of The Ring Hobbit     14  3644
## 3 The Fellowship Of The Ring Man         0  1995
## 4 The Two Towers             Elf       331   513
## 5 The Two Towers             Hobbit      0  2463
## 6 The Two Towers             Man       401  3589
## 7 The Return Of The King     Elf       183   510
## 8 The Return Of The King     Hobbit      2  2673
## 9 The Return Of The King     Man       268  2459
```

2. Tidy the combined data frame by creating new variables "Gender" and "Words"

```
tidy_combined_t <- untidy_combined_t %>%
  pivot_longer(Female:Male, names_to = "Gender", values_to = "Words")
tidy_combined_t
```

```
## # A tibble: 18 x 4
##    Film                       Race   Gender Words
##    <chr>                      <chr>  <chr>  <int>
##  1 The Fellowship Of The Ring Elf    Female  1229
##  2 The Fellowship Of The Ring Elf    Male     971
##  3 The Fellowship Of The Ring Hobbit Female    14
##  4 The Fellowship Of The Ring Hobbit Male    3644
##  5 The Fellowship Of The Ring Man    Female     0
##  6 The Fellowship Of The Ring Man    Male    1995
##  7 The Two Towers             Elf    Female   331
##  8 The Two Towers             Elf    Male     513
##  9 The Two Towers             Hobbit Female     0
## 10 The Two Towers             Hobbit Male    2463
## 11 The Two Towers             Man    Female   401
## 12 The Two Towers             Man    Male    3589
## 13 The Return Of The King     Elf    Female   183
## 14 The Return Of The King     Elf    Male     510
## 15 The Return Of The King     Hobbit Female     2
## 16 The Return Of The King     Hobbit Male    2673
## 17 The Return Of The King     Man    Female   268
## 18 The Return Of The King     Man    Male    2459
```

3. Use the combined data frame to answer the following questions

How many words were spoken in each movie?

```
tidy_combined_t %>%
  group_by(Film) %>%
  summarise(Words = sum(Words))
```

```
## # A tibble: 3 x 2
##   Film                       Words
##   <chr>                      <int>
## 1 The Fellowship Of The Ring  7853
## 2 The Return Of The King      6095
## 3 The Two Towers              7297
```

How many words were spoken by each gender in total?

```
tidy_combined_t %>%
  group_by(Gender) %>%
  summarise(Words = sum(Words))
```

```
## # A tibble: 2 x 2
##   Gender Words
##   <chr>  <int>
## 1 Female  2428
## 2 Male   18817
```

How many words were spoken by each race in total?

```
tidy_combined_t %>%
  group_by(Race) %>%
  summarise(Words = sum(Words))
```

```
## # A tibble: 3 x 2
##   Race   Words
##   <chr>  <int>
## 1 Elf     3737
## 2 Hobbit  8796
## 3 Man     8712
```

4. Create a data frame with totals by race and movie, calling it *by_race_film*.

```
by_race_film <- tidy_combined_t %>%
  group_by(Film, Race) %>%
  summarise(Words = sum(Words)) %>%
  ungroup()
by_race_film
```

```
## # A tibble: 9 x 3
##   Film                       Race   Words
##   <chr>                      <chr>  <int>
## 1 The Fellowship Of The Ring Elf     2200
## 2 The Fellowship Of The Ring Hobbit  3658
## 3 The Fellowship Of The Ring Man     1995
## 4 The Return Of The King     Elf      693
## 5 The Return Of The King     Hobbit  2675
## 6 The Return Of The King     Man     2727
## 7 The Two Towers             Elf      844
## 8 The Two Towers             Hobbit  2463
## 9 The Two Towers             Man     3990
```

Problem 2

1. Split/group the gapminder data by country. For each country, fit an ARIMA(0, 0, 1) or MA(1) model to *lifeExp*, and produce a tibble that contains the country-wise values of AIC and BIC, two measures of goodness of model fit. Obtain a scatter plot of AIC versus BIC and comment.

Import gapminder package

```
library(gapminder)
```

Split the gapminder data by country and fit ARIMA(0, 0, 1) to *lifeExp*.

```
gapminder_split <- gapminder %>%
  split(.$country)
gapminder_split_arima1 <- gapminder_split %>%
  map(., ~arima(.$lifeExp, order = c(0, 0, 1)))
```

Produce a tibble that contains the country-wise values of AIC and BIC

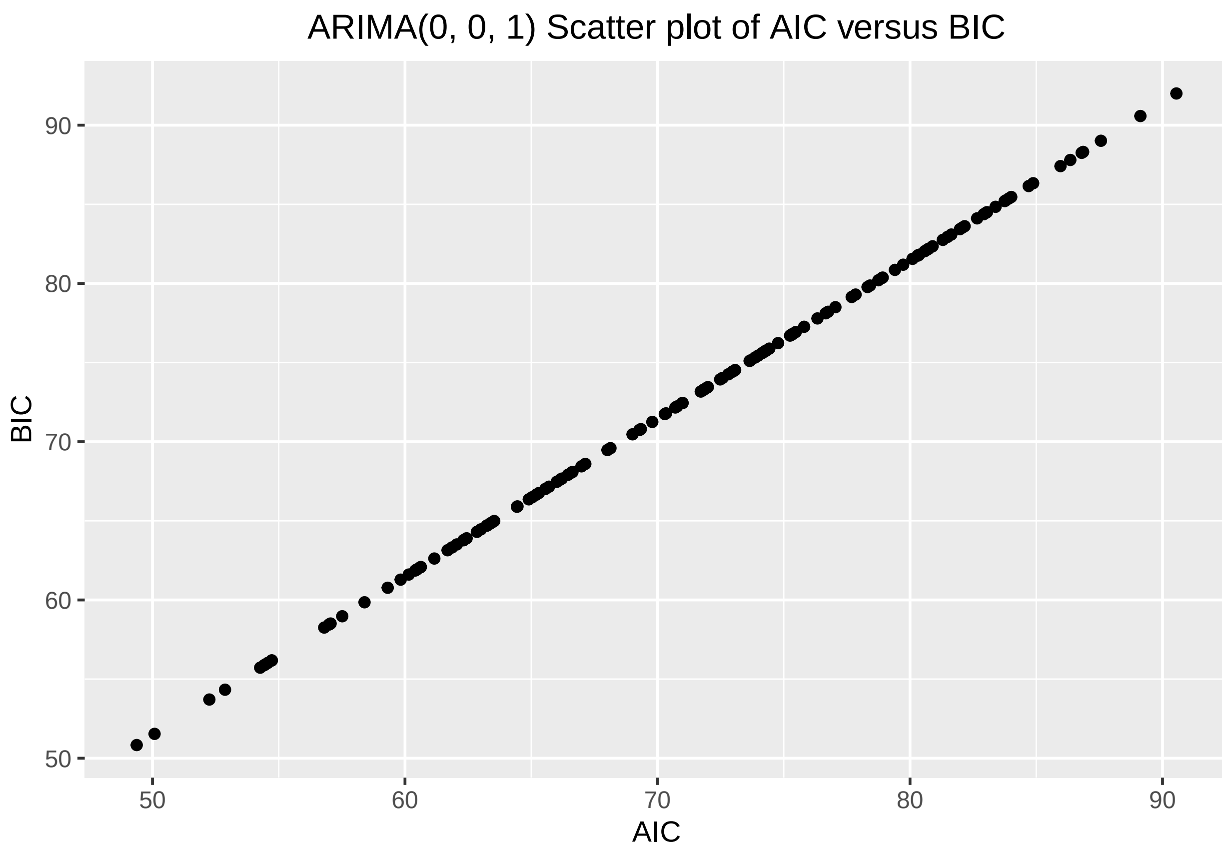
```
# create a function to get AIC and BIC from arima object
getAIC_BIC <- function(obj){
  aic <- obj %>%
    map_dbl(AIC) %>%
    tibble(country = names(.), AIC = .)
  bic <- obj %>%
    map_dbl(BIC) %>%
    tibble(country = names(.), BIC = .)
  result <- merge(aic, bic, all = T) %>%
    as_tibble()
  result
}

aic_bic_1_t <- getAIC_BIC(gapminder_split_arima1)
aic_bic_1_t
```

```
## # A tibble: 142 x 3
##    country       AIC   BIC
##    <chr>       <dbl> <dbl>
##  1 Afghanistan  67.1  68.6
##  2 Albania      72.6  74.0
##  3 Algeria      83.8  85.2
##  4 Angola       62.1  63.5
##  5 Argentina    62.3  63.8
##  6 Australia    61.9  63.3
##  7 Austria      63.2  64.7
##  8 Bahrain      78.9  80.4
##  9 Bangladesh   80.3  81.8
## 10 Belgium      60.4  61.9
## # ... with 132 more rows
```

Draw the scatter plot of AIC versus BIC.

```
aic_bic_1_t %>% ggplot(aes(x = AIC, y = BIC)) +
  geom_point() +
  labs(title = "ARIMA(0, 0, 1) Scatter plot of AIC versus BIC") +
  theme(plot.title = element_text(hjust = 0.5))
```



2. Now repeat the previous step for four other models: ARIMA(0, 0, 1), ARIMA(0, 0, 2), ARIMA(0, 0, 3), ARIMA(0, 1, 0), ARIMA(0, 1, 1), and in a single plot, show boxplots of AIC values for the five models. Based on the boxplot, which of these five models do you think fits the data best for most countries?

ARIMA(0, 0, 2)

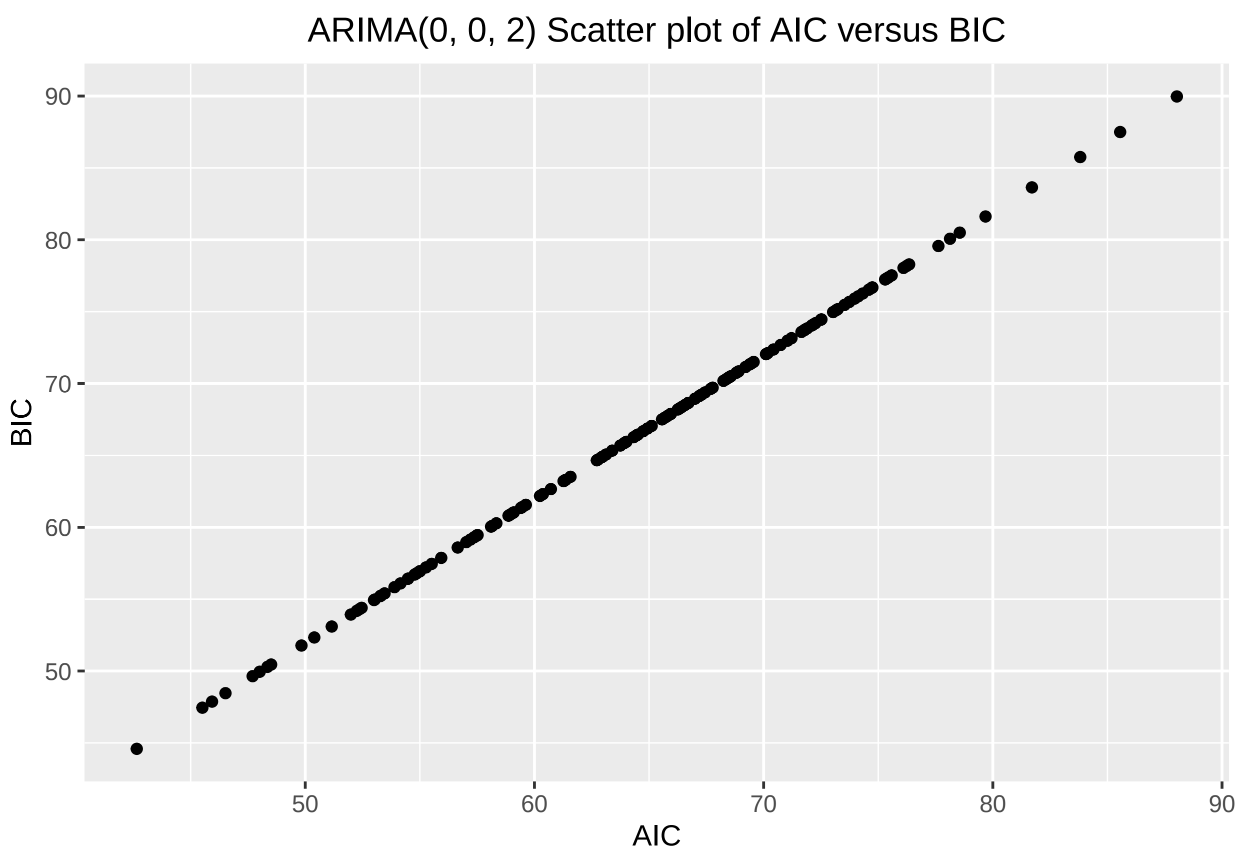
```
gapminder_split_arima2 <- gapminder_split %>%
  map(., ~arima(.$lifeExp, order = c(0, 0, 2)))

aic_bic_2_t <- getAIC_BIC(gapminder_split_arima2)
aic_bic_2_t
```

```
## # A tibble: 142 x 3
##    country       AIC   BIC
##    <chr>       <dbl> <dbl>
##  1 Afghanistan  59.4  61.4
##  2 Albania      65.6  67.5
```

```
## 3 Algeria      75.3  77.2
## 4 Angola       55.0  56.9
## 5 Argentina    54.8  56.7
## 6 Australia    53.5  55.4
## 7 Austria      55.5  57.5
## 8 Bahrain      73.5  75.5
## 9 Bangladesh   72.1  74.0
## 10 Belgium     52.5  54.4
## # ... with 132 more rows
```

```
aic_bic_2_t %>% ggplot(aes(x = AIC, y = BIC)) +
  geom_point() +
  labs(title = "ARIMA(0, 0, 2) Scatter plot of AIC versus BIC") +
  theme(plot.title = element_text(hjust = 0.5))
```



ARIMA(0, 0, 3)

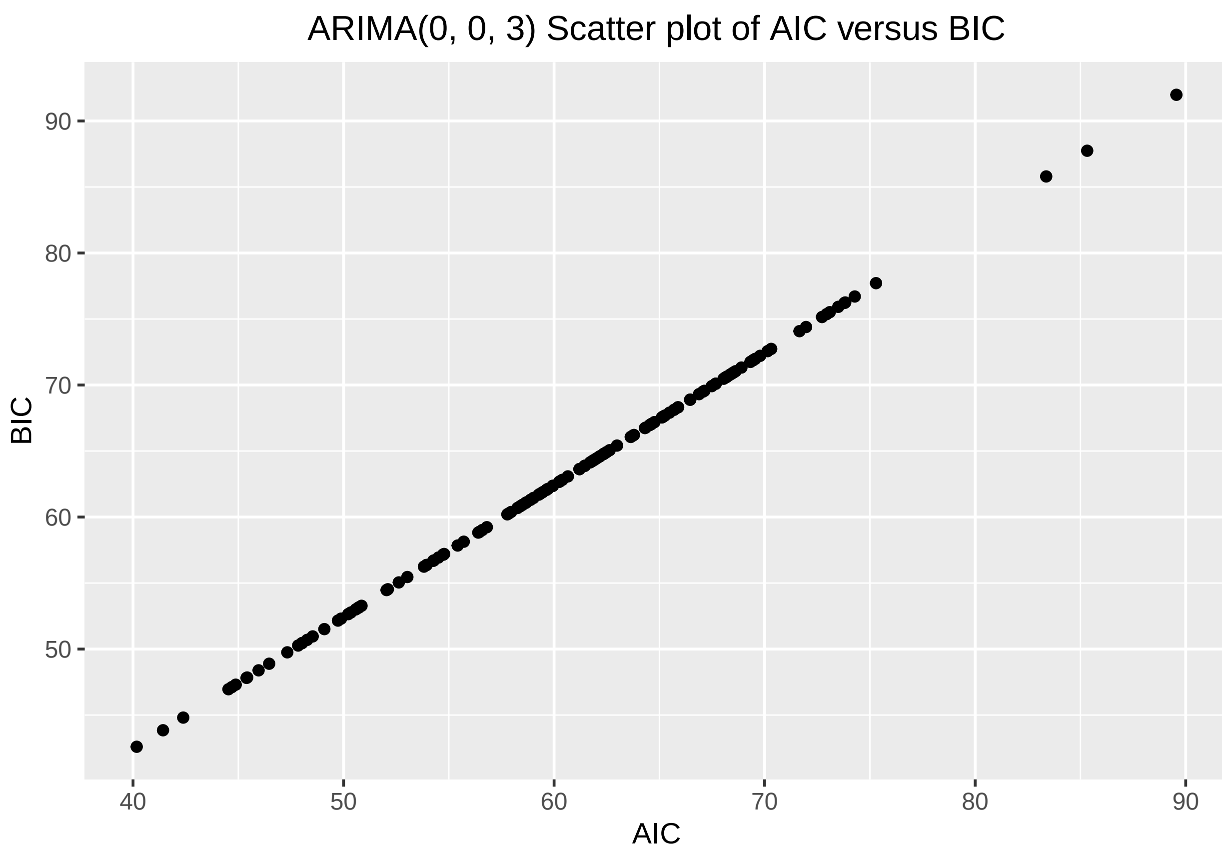
```
gapminder_split_arima3 <- gapminder_split %>%
  map(., ~arima(.$lifeExp, order = c(0, 0, 3)))

aic_bic_3_t <- getAIC_BIC(gapminder_split_arima3)
aic_bic_3_t
```

```
## # A tibble: 142 x 3
##   country       AIC   BIC
##   <chr>       <dbl> <dbl>
## 1 Afghanistan  54.8  57.2
```

```
##   2 Albania      64.6  67.0
##   3 Algeria      70.3  72.7
##   4 Angola       49.9  52.3
##   5 Argentina    50.7  53.1
##   6 Australia    47.8  50.3
##   7 Austria      52.1  54.5
##   8 Bahrain      66.5  68.9
##   9 Bangladesh   67.1  69.6
##  10 Belgium      48.5  51.0
## # ... with 132 more rows
```

```
aic_bic_3_t %>% ggplot(aes(x = AIC, y = BIC)) +
  geom_point() +
  labs(title = "ARIMA(0, 0, 3) Scatter plot of AIC versus BIC") +
  theme(plot.title = element_text(hjust = 0.5))
```



ARIMA(0, 1, 0)

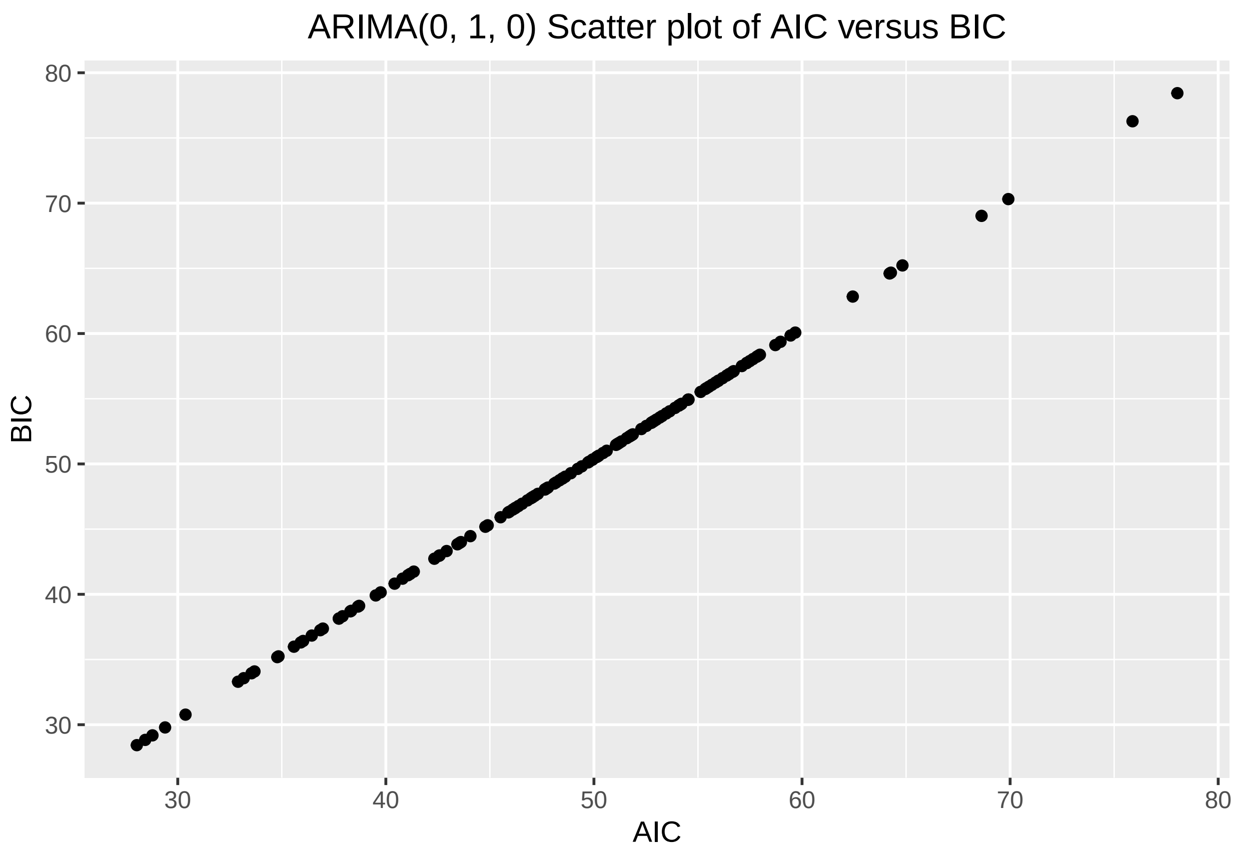
```
gapminder_split_arima4 <- gapminder_split %>%
  map(., ~arima(.$lifeExp, order = c(0, 1, 0)))

aic_bic_4_t <- getAIC_BIC(gapminder_split_arima4)
aic_bic_4_t
```

```
## # A tibble: 142 x 3
##    country       AIC   BIC
##    <chr>       <dbl> <dbl>
```

```
##   1 Afghanistan  42.6  43.0
##   2 Albania      53.2  53.6
##   3 Algeria      56.0  56.4
##   4 Angola       40.8  41.2
##   5 Argentina    37.7  38.1
##   6 Australia    36.8  37.2
##   7 Austria      38.7  39.1
##   8 Bahrain      52.8  53.2
##   9 Bangladesh   53.3  53.7
##  10 Belgium      34.8  35.2
## # ... with 132 more rows
```

```
aic_bic_4_t %>% ggplot(aes(x = AIC, y = BIC)) +
  geom_point() +
  labs(title = "ARIMA(0, 1, 0) Scatter plot of AIC versus BIC") +
  theme(plot.title = element_text(hjust = 0.5))
```



ARIMA(0, 1, 1)

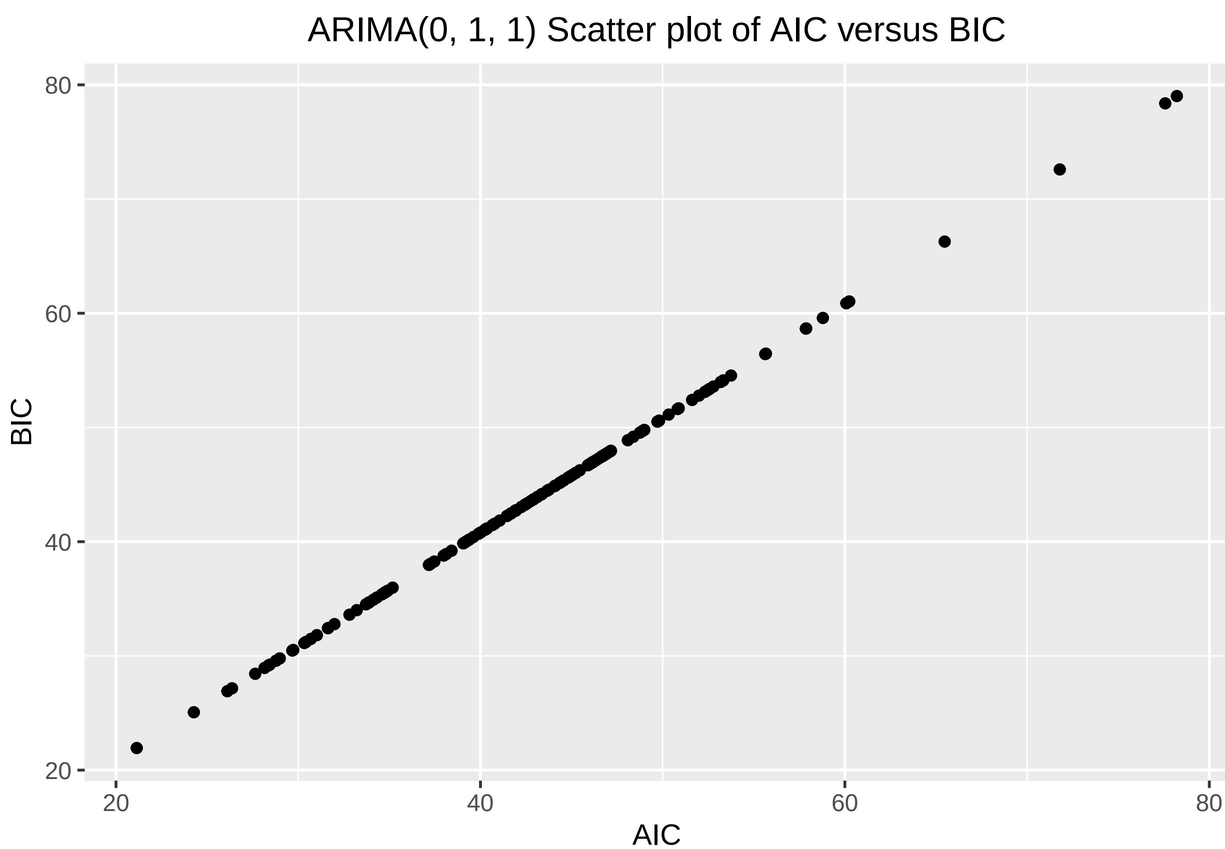
```
gapminder_split_arima5 <- gapminder_split %>%
  map(., ~arima(.$lifeExp, order = c(0, 1, 1)))

aic_bic_5_t <- getAIC_BIC(gapminder_split_arima5)
aic_bic_5_t
```

```
## # A tibble: 142 x 3
##    country       AIC   BIC
```

```
##     <chr>        <dbl> <dbl>
##   1 Afghanistan  34.9  35.7
##   2 Albania      48.1  48.9
##   3 Algeria      49.0  49.8
##   4 Angola       33.8  34.6
##   5 Argentina    30.4  31.2
##   6 Australia    29.0  29.8
##   7 Austria      34.6  35.4
##   8 Bahrain      47.0  47.8
##   9 Bangladesh   43.8  44.5
##  10 Belgium      29.7  30.5
## # ... with 132 more rows
```

```
aic_bic_5_t %>% ggplot(aes(x = AIC, y = BIC)) +
  geom_point() +
  labs(title = "ARIMA(0, 1, 1) Scatter plot of AIC versus BIC") +
  theme(plot.title = element_text(hjust = 0.5))
```



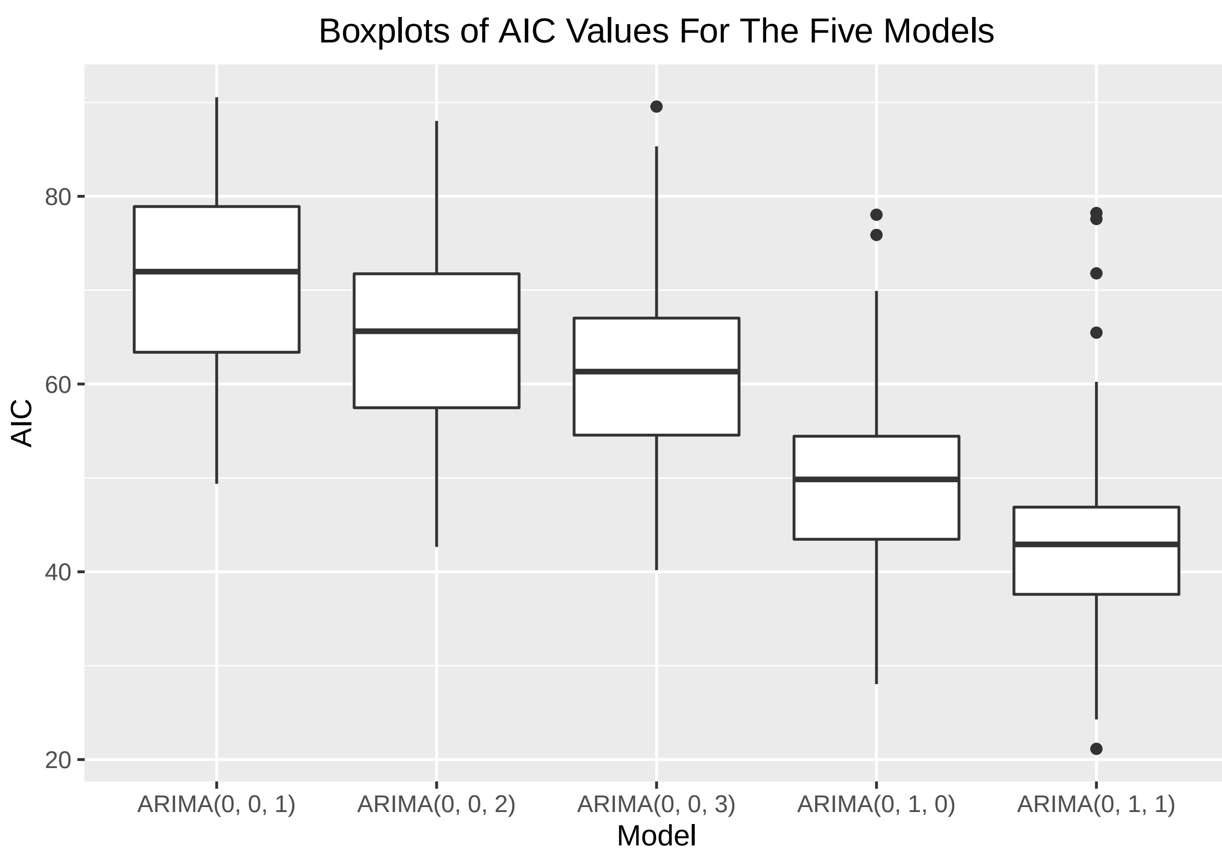
Boxplot

```
# add a variable called Model to identify each model
aic_bic_1_t <- aic_bic_1_t %>%
  mutate(Model = "ARIMA(0, 0, 1)")
aic_bic_2_t <- aic_bic_2_t %>%
  mutate(Model = "ARIMA(0, 0, 2)")
aic_bic_3_t <- aic_bic_3_t %>%
```

```
  mutate(Model = "ARIMA(0, 0, 3)")
aic_bic_4_t <- aic_bic_4_t %>%
  mutate(Model = "ARIMA(0, 1, 0)")
aic_bic_5_t <- aic_bic_5_t %>%
  mutate(Model = "ARIMA(0, 1, 1)")

# merge five tibbles containing aic data into one
merged_aic_bic_t <- aic_bic_1_t %>%
  merge(aic_bic_2_t, all = T) %>%
  merge(aic_bic_3_t, all = T) %>%
  merge(aic_bic_4_t, all = T) %>%
  merge(aic_bic_5_t, all = T)

merged_aic_bic_t %>%
  ggplot(aes(x = Model, y = AIC)) +
  geom_boxplot() +
  labs(title = "Boxplots of AIC Values For The Five Models") +
  theme(plot.title = element_text(hjust = 0.5))
```



Based on the boxplot, we can see that AIC values of most countries fitted using ARIMA(0, 1, 1) are much lower than other models. This indicates the model ARIMA(0, 1, 1) fits the data best.

3. Filter the data only for continent Europe. For the best model identified in step 2, create a tibble showing the country-wise model parameters (moving average coefficients) and their standard errors using the broom package.

Filter the data only for continent Europe

```
gapminder_Europe <- gapminder %>%
  filter(continent == "Europe")
```

Fit the data with ARIMA(0, 1, 1)

```
gapminder_Europe_split <- gapminder_Europe %>%
  split(.$country, drop = T)
gapminder_Europe_split_arima <- gapminder_Europe_split %>%
  map(., ~arima(.$lifeExp, order = c(0, 1, 1)))
```

Create the tibble

```
tidyList <- gapminder_Europe_split_arima %>%
  map(broom::tidy)
paraTibble <- tidyList %>%
  tibble(country = names(.))
paraTibble <- paraTibble$. %>%
  mutate(.data = paraTibble,
         term = bind_rows(.)$term,
         estimate = bind_rows(.)$estimate,
         std.error = bind_rows(.)$std.error) %>%
  select(country, term, estimate, std.error)
paraTibble
```

```
## # A tibble: 30 x 4
##    country                term  estimate std.error
##    <chr>                  <chr>    <dbl>     <dbl>
##  1 Albania                ma1      1.00      0.353
##  2 Austria                ma1      0.708     0.263
##  3 Belgium                ma1      0.645     0.183
##  4 Bosnia and Herzegovina ma1      1.00      0.353
##  5 Bulgaria               ma1      1.00      0.411
##  6 Croatia                ma1      0.676     0.199
##  7 Czech Republic         ma1      0.606     0.203
##  8 Denmark                ma1      0.494     0.204
##  9 Finland                ma1      0.778     0.227
## 10 France                 ma1      0.706     0.189
## # ... with 20 more rows
```