Assignment 2

Applied Forecasting in Complex Systems 2021

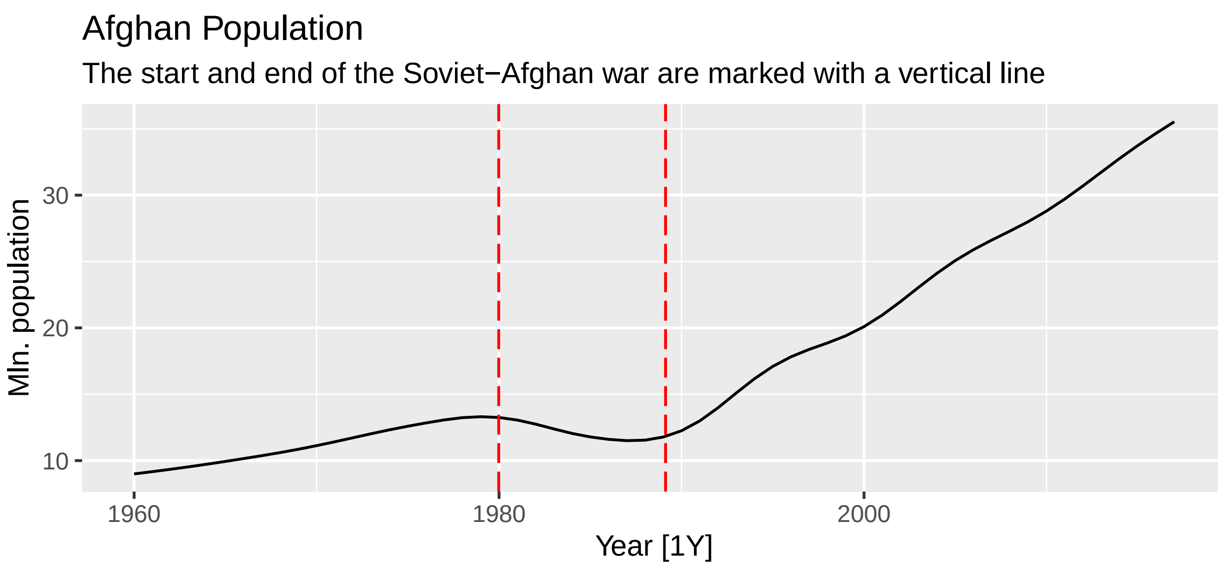
Emiel Steegh (14002558)

University of Amsterdam
November, 21, 2021

Note to the reader:
All of the code can be found in the appendix, starting on page 10

Exercise 1

1.1)

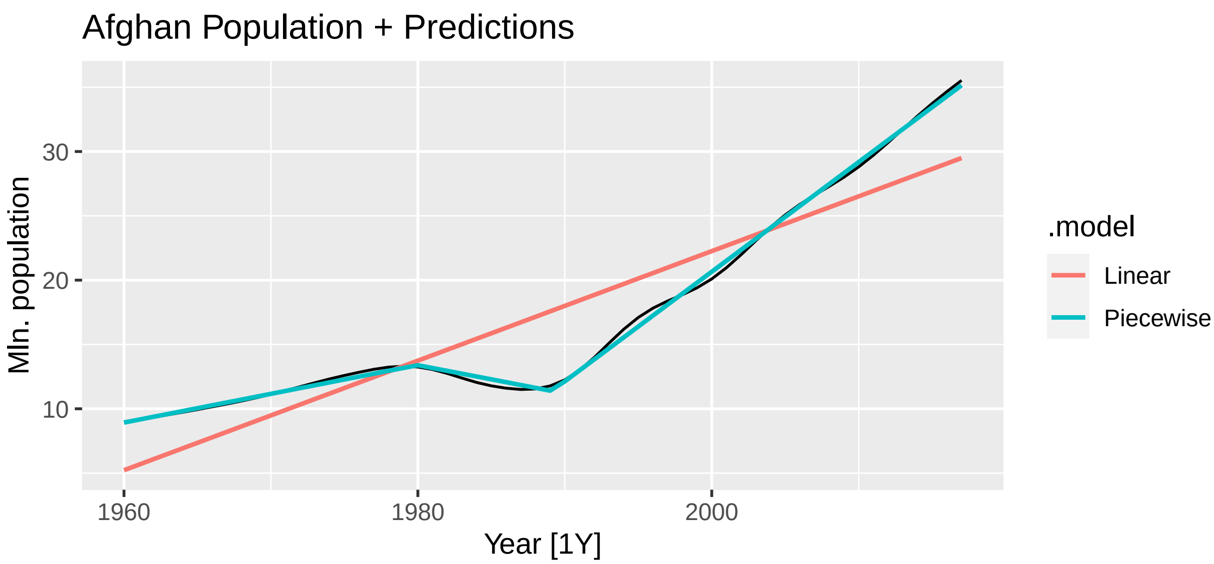


In the plot of Afghanistan's population we can clearly see a correlation between the period of the Soviet-Afghan war and the countries population growth. The trend can be divided into three sections:
1. 1960:1979 an upward trend until the year before the start of the war
2. 1980:1988 a downward trend which ends a little over a year before the end of the war
3. 1989:2017 an upward trend steeper than the pre-war period.

War frequently contributes to a higher death-rate and a lower birth-rate in a country, an effect that can be observed in section two. In the third section we can see a steeper growth, which might be explained by a post-war (economic) boom. It also looks like this section includes light cyclic behaviour. Other than that no cyclicity or seasonality can be observer

To make the plot easier to read, population has been scaled to millions.

1.2)



Forecasting model accuracies

.model	r_squared	adj_r_squared	AICc	CV
Linear	0.838	0.835	139	10.569
Piecewise	0.999	0.998	-130	0.099

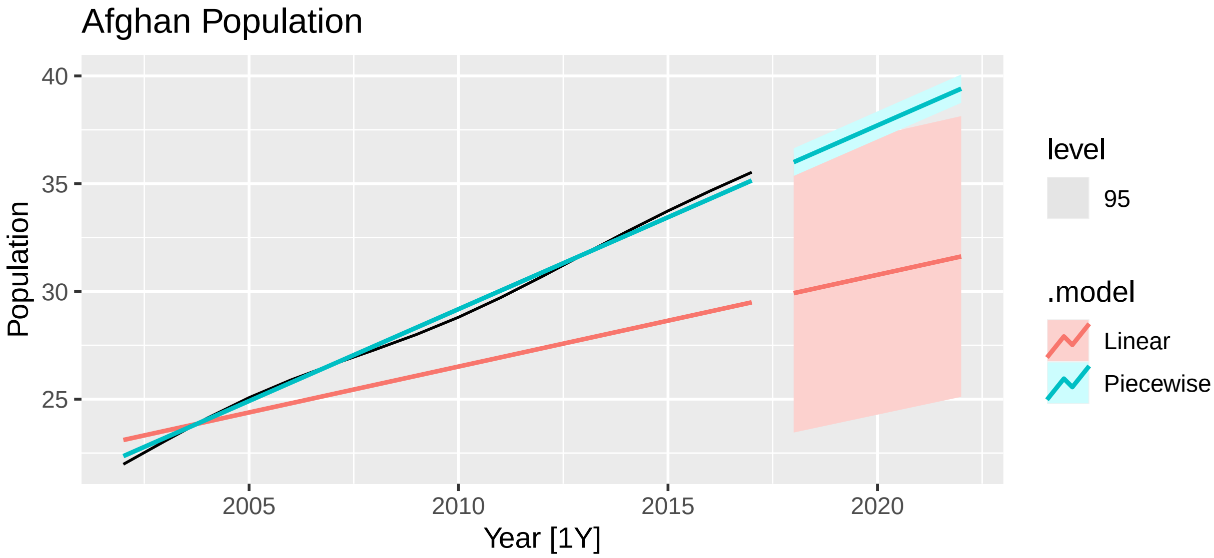
Clearly the *linear model* is a poor fit to the actual data. The *piecewise model*, however looks like a decent estimate. A plot of both models' residuals (found in Appendix 1.2) reveals that like expected the *linear model*'s residuals look nothing like white noise. There is a strong upward trend in the residuals from 1989 onwards introducing an increasing bias.

Although the *piecewise model*'s residuals look better, they still do no behave like white noise.

A Box-Ljung test with 2 DoF for the *linear model* (β_0 and β_1) and another with 6 DoF for the *piecewise model* (β_0 and β_1 for all three sections) result in p-values < 0.05 which means that we can reject the accompanying null hypothesis that the residuals are zero. Simply put neither model is able to describe all the information in the timeseries.

The accuracies table shows us that the *piecewise model* does indeed do much better than the simpler *linear model*, because it has a higher R^2 (almost 1) and much lower AICc and CV scores.

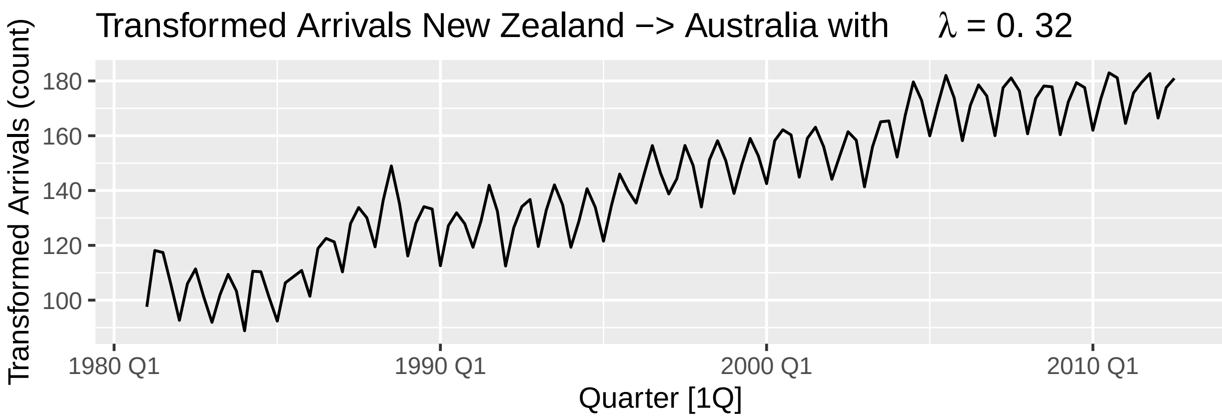
1.3)

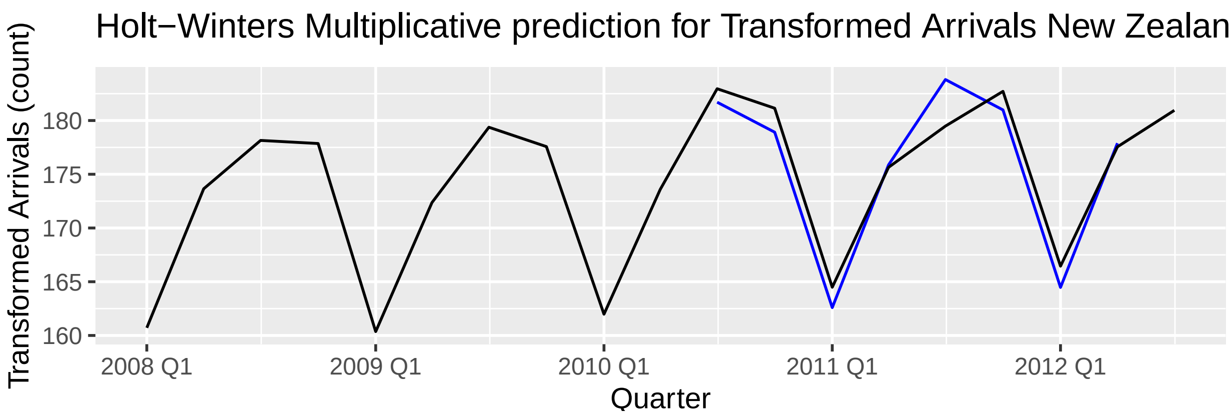


The prediction of the *piecewise model* looks decent, when eyeballing the graph, it seems to follow the latest trend and it has a small confidence interval. The *linear model* produces a terrible forecast, it starts a little over 5 million below the last known value, a very unlikely jump for the data to make and additionally has a confidence interval of over 10 million. The *piecewise model* definitely outperforms the *linear model*, but from the information obtained in *(1.2)* I would suggest neither.

Exercise 2

2.1)





We observe an obvious seasonal pattern, The ACF (in appendix 2.1) helps to establish this. The seasonal period is a year long, starting low in Q1 then making a jump for Q2, Q3 & Q4, with Q3 usually being the highest out of them all. This is likely explained by the timing of holidays and nicer weather.

Furthermore there is a positive trend and growing variance over levels. Because this growing variance makes predicting harder a Box-Cox transformation was performed with $\lambda = 0.32$, this helps stabilize the variance a lot. The data contains cyclicity over levels, 4 of which can be seen in the sample.

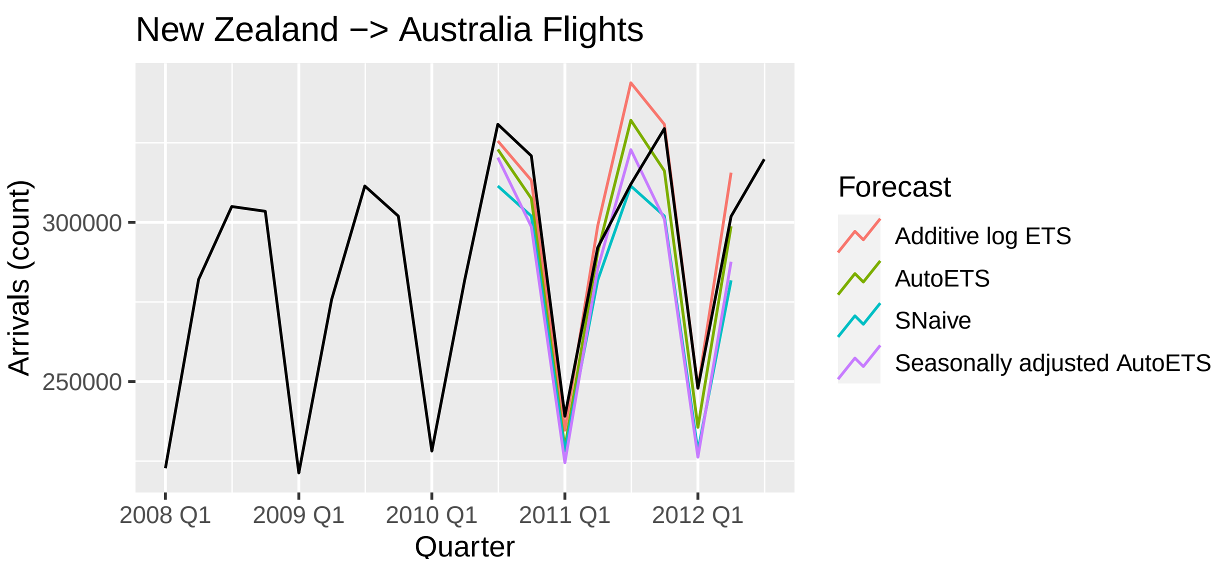
A multiplicative method for Holt-Winters model makes sense because the seasonal variations are changing proportionally to the level of the series, i.e. we see the variance grow as the trendline rises, even with the transformation. Because we have quarterly data we set $m = 4$.

A model report (Appendix 2.1) reveal the following smoothing parameters:
- $\alpha = 0.636$: level smoothing is applied so there is an update in the levels
- $\beta \approx 0$: no trend smoothing required so we have a stable linear trend
- $\gamma = 0.221$: a small amount of seasonal smoothing means that the seasonal component hardly changes.

2.2)

Next we fit the following four models to the data, without prior transformation this time.

Model name	Description
AutoETS	ETS model where all the parameters are automatically chosen
SNaive	Seasonal Naïve model with a 1 year lag
Additive log ETS	ETS model that is fixed to additive applied to log transformed data
Seasonally adjusted AutoETS	An AutoETS model applied to the seasonally adjusted component of an STL decomposition of the data



Model *fit* accuracies

as fitted to the *New Zealand -> Australia* flight arrivals (RMSE sorted)

.model	RMSE	MAE	MPE	MAPE
Seasonally adjusted AutoETS	9929	7783	1.314	5.56
AutoETS	11169	8891	-1.110	6.70
Additive log ETS	11184	8973	-0.322	6.68
SNaive	19354	14839	3.900	10.64

Model *forecasting* accuracies

as fitted to the *New Zealand -> Australia* flight arrivals (RMSE sorted)

.model	RMSE	MAE	MPE	MAPE
AutoETS	11640	10179	1.85	3.46
Additive log ETS	13024	8881	-1.46	2.91
Seasonally adjusted AutoETS	17552	16154	4.64	5.51
SNaive	17707	15926	5.37	5.37

From the fit and forecast accuracy tables we can draw some inferences, they are ordered by RMSE. *Seasonally adjusted AutoETS* comes out on top for model fit, however it comes in third with forecasting accuracy. *Auto ETS* does pretty well in both, getting first place for forecasts and second for fit.

When reporting the parameters of these two best models (Appendix 2.2) you should note that the *AutoETS* model picked near identical to the parameters of Holt-Winters multiplicative model. The residuals plots for the top ranking forecasting model (almost) resembles white noise. Although they have decreasing variance over time and a somewhat skewed histogram, all the correlations of the ACF are within the confidence interval. A Box-Ljung test (Appendix 2.2) with DoF = 6 for all estimated parameters, and 10 lags results in $p \approx 0.168 > \alpha$ where $\alpha = 0.05$. This allows us to not reject the weak null-hypothesis that all residuals are zero. This means that the *AutoETS* model captures most of available information. Because the *AutoETS model* is essentially equivalent to the *Holt-Winters Multiplicative model* from *(2.1)* we conclude the same for both.

2.3)

**Forecasting model accuracies: Cross Validated (CV) vs the models from 2.2 **
as fitted to the *New Zealand -> Australia* flight arrivals (RMSE sorted)

.model	RMSE	MAE	MPE	MAPE
AutoETS	11640	10179	1.85	3.46
Additive log ETS	13024	8881	-1.46	2.91
Seasonally adjusted AutoETS	17552	16154	4.64	5.51
SNaive	17707	15926	5.37	5.37
CV Seasonally adjusted autoETS	20765	15261	3.50	7.67
CV AutoETS	21737	15489	3.86	7.74
CV Additive log ETS	22451	17496	-2.42	8.74
CV SNaive	24245	17826	6.33	8.81

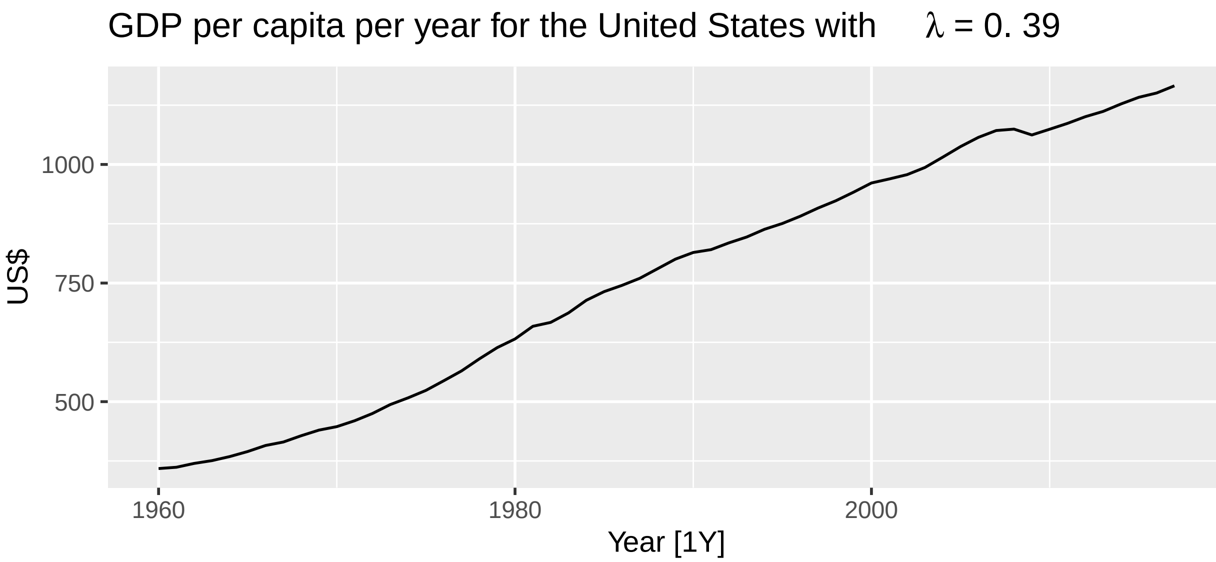
the initial amount of data-points for cross validation is selected as 40, 30% of the data rounded up to the next full seasonal period.

Error went up for each model when using cross validation, this is not surprising as the calculated accuracies are aggregates from the same model also trained on way fewer data, taking down the average.

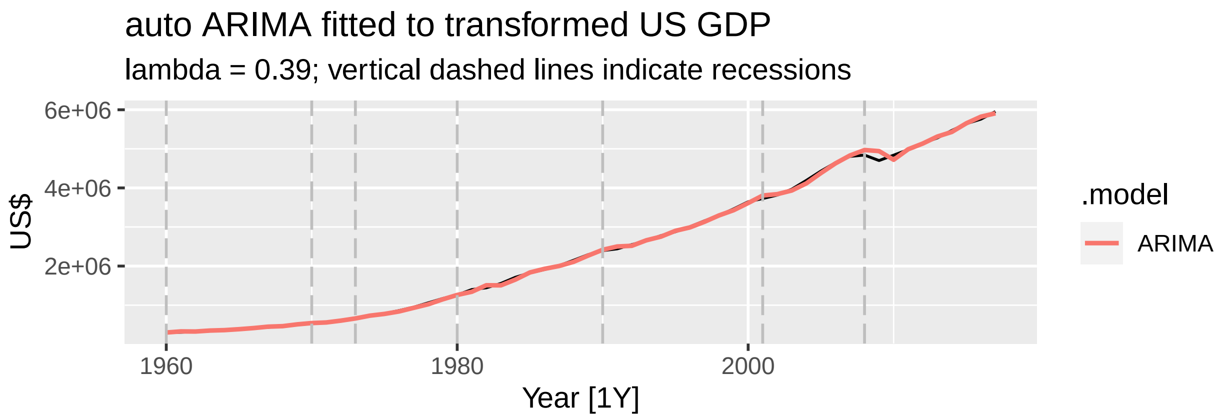
Seasonally adjusted auto ETS would probably make the best forecasting model because it has the best accuracy on the multi-step forecast environment, meaning that it is likely to generalize the best out of all four.

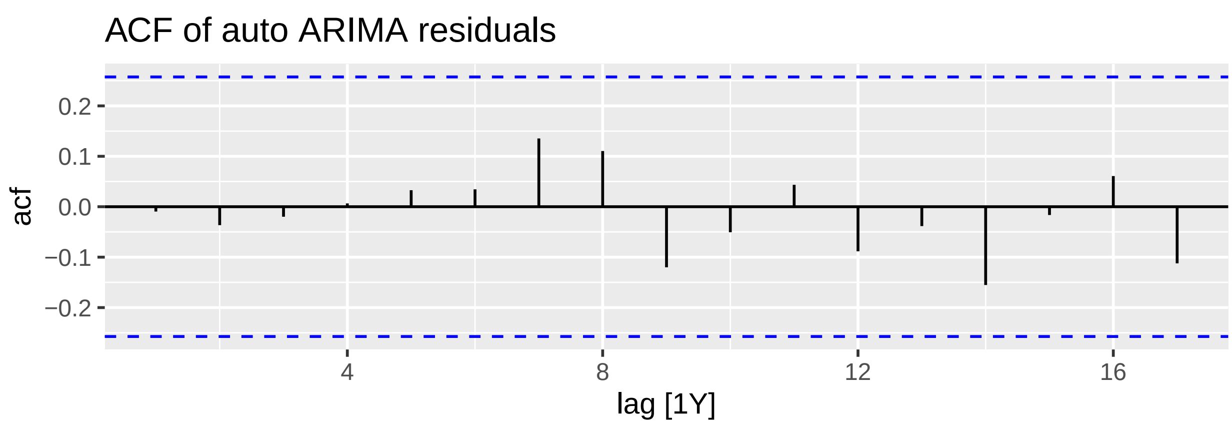
Exercise 3

3.1)



The original is transformed to per-capita, to account for population growth, an inflation transformation is left aside because that seems out of the scope of the question. After the per capita transformation, the data still has a non-linear (lightly exponential) trend, so a Box-Cox transformation is applied. Using the guerrero method we find $\lambda = 0.39$ which approximates a root transformation. The data now displays a more linear upward trend with a drop in 2008, likely correlated with the 2008 financial crisis. There is no seasonality present nor large enough variation in trend to speak of cyclicity.

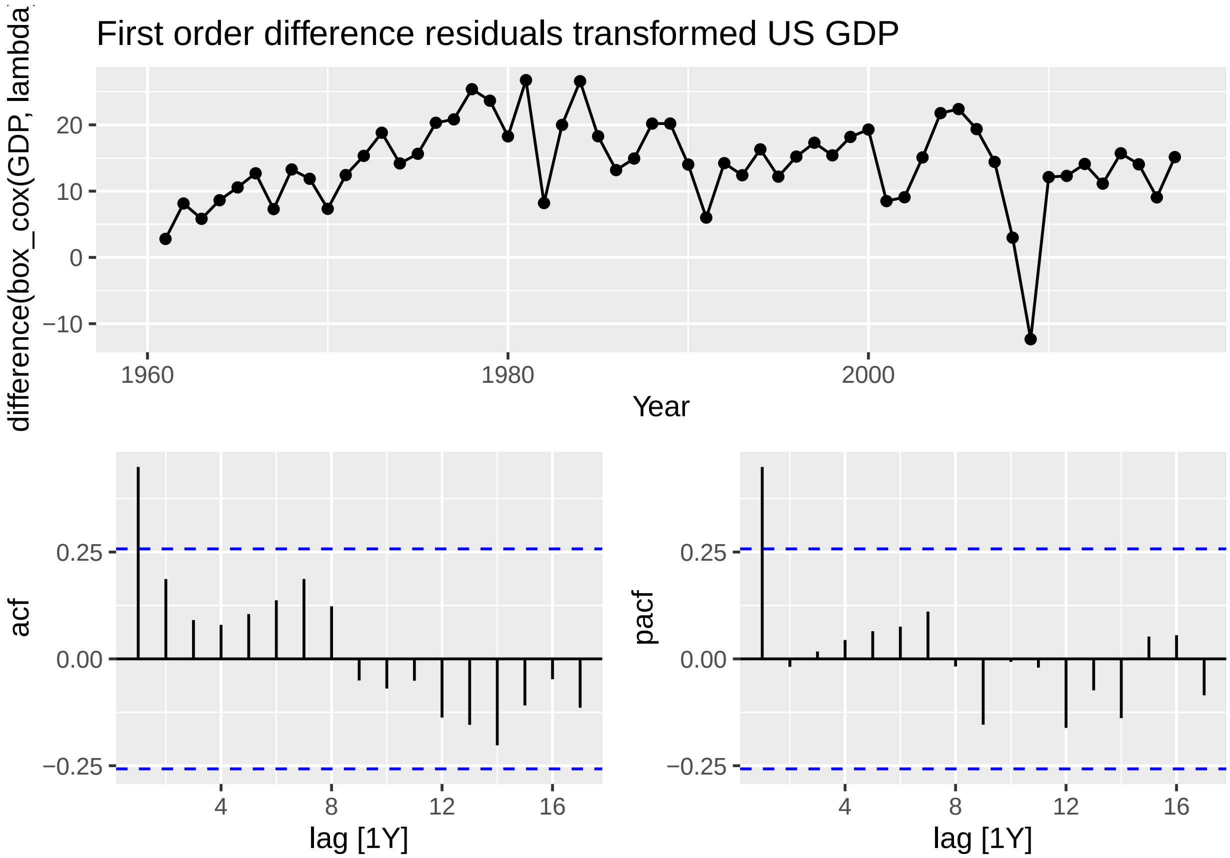




We fit an auto ARIMA model to the transformed data which results in an `ARIMA(1,1,0) with dirft` model. It has first order Auto-Regressive and first order differencing part, to reach a stationary result. The model follows the data closely but struggles a little around the hitches that coincide around the recessions (marked by vertical dashed lines) likely because they are unpredictable events (in the scope of this time series). But, the residuals of the model behave nicely like white noise.

3.2)

The *General process for forecasting using an ARIMA model* flowchart in Chapter 9.7 from the Forecasting: Principles and Practice (3rd edition) will be used to identify a plausible ARIMA model. Steps one and two have been conducted in *(3.1)*: The data was plotted, inspected and transfomed with a Box-Cox because it was appropriate.



Unitroot KPSS test

kpss_stat	kpss_pvalue
0.197	0.1

The next step is to inspect the pACF and ACF of the first order difference. We immedeatly see a positive spike for the first lag, which hints that the data is non-stationary. However, the unitroot KPSS test results in $p = 0.1 > \alpha$ which does not let us reject the null hypothesis that the time series is stationary. Since the data is non-stationary we should try $d = 1$, and because of the large positive r_1 we should either pick $p = 1 \vee q = 1$. So we try both `ARIMA(1,1,0)` and `ARIMA(0,1,1)`. Notice that the first on is the model obtained from running the auto ARIMA in *(3.3)*

Manual ARIMA model comparison

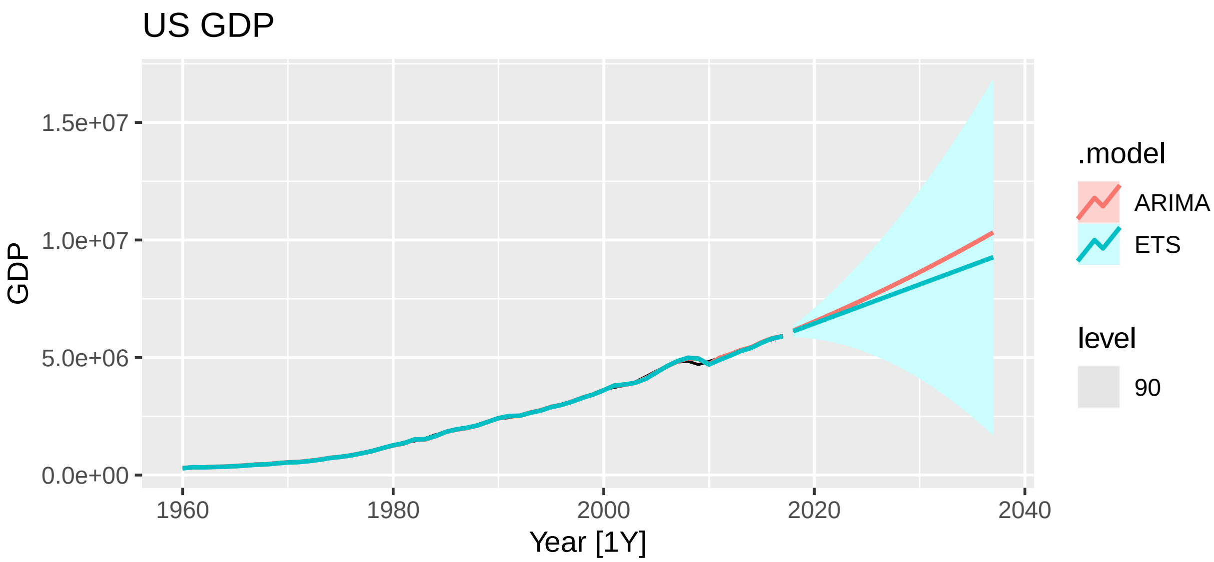
.model	sigma2	log_lik	AIC	AICc	BIC
ARIMA(1,1,0)	34.6	-181	368	368	374
ARIMA(0,1,1)	35.8	-182	370	370	376

Box-Ljung for the `ARIMA(1,1,0)`

.model	lb_stat	lb_pvalue
ARIMA(1,1,0)	3.58	0.893

From the model comparison we can conclude that `ARIMA(1,1,0)` outperforms `ARIMA(0,1,1)` in all four categories. Considering that the better model is exactly the same as the model from *(3,2)* the same conclusion about it's residuals apply: they look like white noise. A Ljung-Box test with DoF=2 and 10 lags gives us $p = 0.89 > \alpha$ and we cannot *reject* the null hypothesis that the residuals are 0. In other words, the residuals behave like white noise and the model captures all (or enough) information in the time series.

3.3)



Comparing the model derived in *(3.3)* to an automatically generated ETS shows the ARIMA fitting the true data better than the ETS, even if only slightly. The bigger difference comes in the 20 year forecasts. The ETS makes a linear prediction, while the ARIMA model continues the exponential trend observed in the data. Next to that the confidence interval of the ETS is much larger and grows faster.

Appendix

Note: Due to an error in Knitr that I have not been able to fix, ggplot titles and labels are not being wrapped properly. However, you are able to read the contents in the graphs just fine.

Setup code

```
options(digits = 3)
library(fpp3)
library(latex2exp)
library(forecast)
library(formatR)
library(gridExtra)
library(gt)
```

```r
knitr::opts_chunk$set(echo = FALSE, message = FALSE, warning = FALSE,
    cache = TRUE, dev.args = list(pointsize = 11))
knitr::opts_chunk$set(tidy.opts = list(width.cutoff = 60), tidy = TRUE)
```

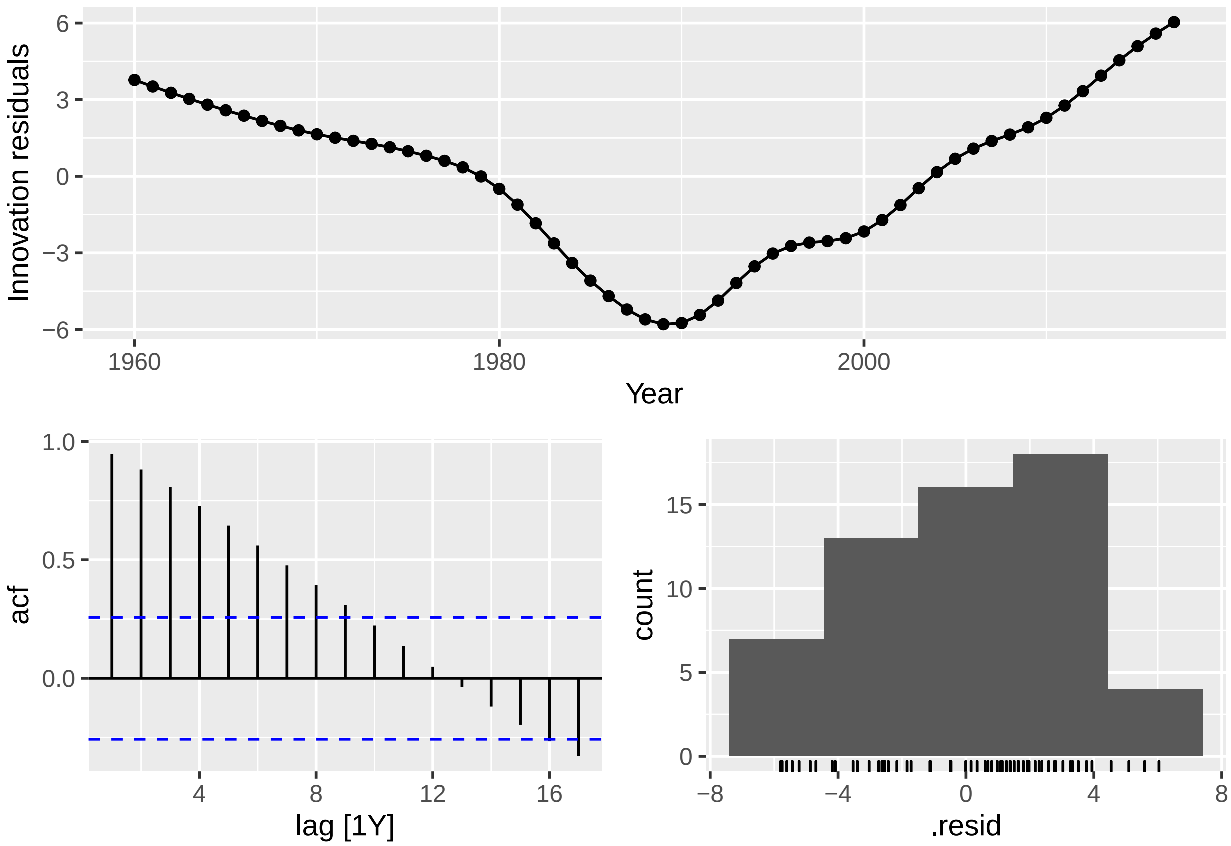
1.1 code

```r
afg_pop <- global_economy %>%
    filter(Country == "Afghanistan") %>%
    mutate(Population = Population/1e+06) %>%
    select(Year, Population)
sov_afg_war <- c(1979 + ((1/12) * (11 + (24/31))), 1989 + ((1/12) *
    (1 + (15/31))))  # https://en.wikipedia.org/wiki/Soviet%E2%80%93Afghan_War

afg_pop %>%
    autoplot(Population) + labs(y = "Mln. population", title = "Afghan Population",
    subtitle = "The start and end of the Soviet-Afghan war are marked with a vertical line") +
    geom_vline(xintercept = sov_afg_war, colour = "red", linetype = "longdash")
```

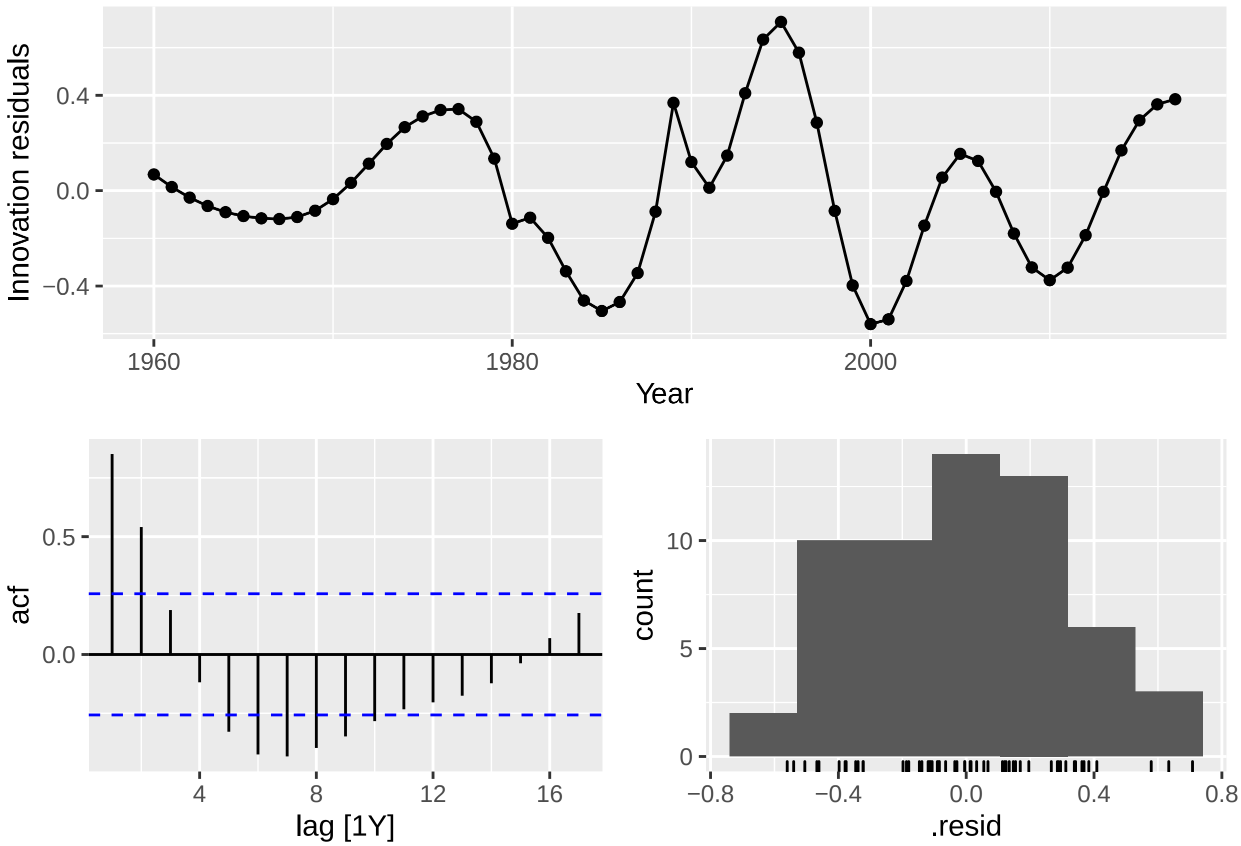
1.2 code

```r
afg_pop_fit <- afg_pop %>%
        model(
          Linear = TSLM(Population ~ trend()),
          Piecewise = TSLM(Population ~ trend(knots=sov_afg_war))
        )

afg_pop %>% autoplot(Population) + #plot
        labs(y = "Mln. population", title = "Afghan Population + Predictions") +
        geom_line(data = fitted(afg_pop_fit),
                    aes(y = .fitted, colour = .model), size = 0.8)
afg_pop_fit %>% glance() %>% select(.model , r_squared , adj_r_squared , AICc, CV) %>% gt %>%
        tab_header(
                title = md("**Forecasting model accuracies**")
        ) %>%
        opt_align_table_header(align = "center")#table
```

```r
gg_tsresiduals(select(afg_pop_fit, Linear))
```




```
gg_tsresiduals(select(afg_pop_fit, Piecewise))
```




```
rbind((augment(afg_pop_fit %>%
    select(Linear)) %>%
    features(.innov, ljung_box, lag = 10, dof = 2)), (augment(afg_pop_fit %>%
    select(Piecewise)) %>%
    features(.innov, ljung_box, lag = 10, dof = 6))) %>%
    gt() %>%
    tab_header(title = md("**Box-Ljung for the linear and piecewise model**"),
        subtitle = md("Linear with 2 DoF, Piecewise with 6 DoF")) %>%
    opt_align_table_header(align = "center")
```

Box-Ljung for the linear and piecewise model

Linear with 2 DoF, Piecewise with 6 DoF

.model	lb_stat	lb_pvalue
Linear	263	0
Piecewise	123	0

1.3 code

```
afg_pop_fc <- forecast(afg_pop_fit, h = 5)
afg_pop %>%
    filter(Year >= 2002) %>%
    autoplot(Population) + geom_line(data = fitted(afg_pop_fit) %>%
```

```
    filter(Year >= 2002), aes(y = .fitted, colour = .model),
    size = 0.8) + autolayer(afg_pop_fc, level = 95, size = 0.8) +
    labs(title = "Afghan Population", ylab = "Mln. population")
```

2.1 code

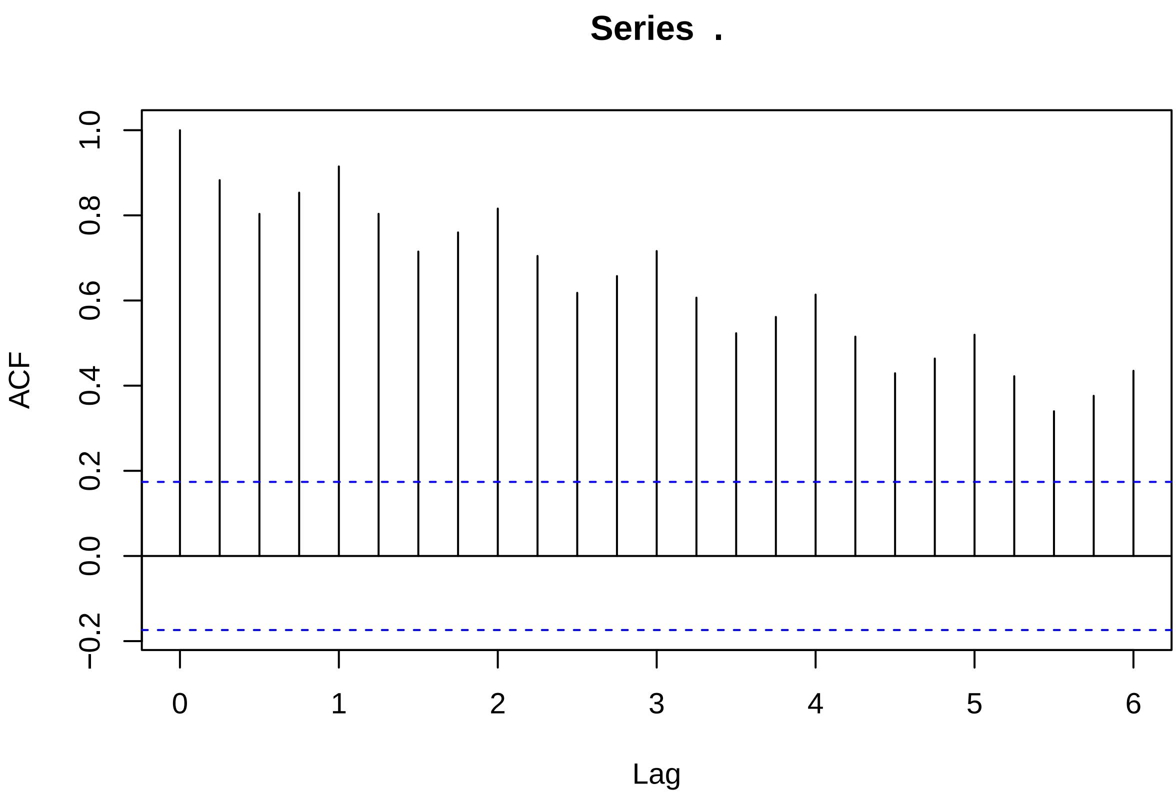
```
# get data
data <- aus_arrivals %>%
    filter(Origin == "NZ", Quarter >= yearquarter("1981 Q1"),
        Quarter <= yearquarter("2012 Q3")) %>%
    select(!Origin)

# transform to stabilize variance
lambda <- data %>%
    features(Arrivals, features = guerrero) %>%
    pull(lambda_guerrero)
data <- data %>%
    mutate(Arrivals_transformed = BoxCox(Arrivals, lambda))

# plot
p1 <- data %>%
    autoplot(Arrivals_transformed) + labs(y = "Transformed Arrivals (count)",
    title = latex2exp::TeX(paste0("Transformed Arrivals New Zealand -> Australia with $\\;\\;\
        round(lambda, 2))))

# test,train split
test <- data %>%
    filter(Quarter >= yearquarter("2010 Q3"))
train <- data %>%
    filter(Quarter < yearquarter("2010 Q3"))

# train model
fit <- train %>%
    model(multiplicative = ETS(Arrivals_transformed ~ error("M") +
        trend("A") + season("M")))
# fit model
fc <- fit %>%
    forecast(h = "2 years")
# plot fit
p2 <- fc %>%
    autoplot((data %>%
        filter(Quarter >= yearquarter("2008 Q1"))), level = NULL) +
    labs(y = "Transformed Arrivals (count)", title = latex2exp::TeX(paste0("Holt-Winters Multi
        round(lambda, 2)))) + guides(colour = guide_legend(title = "Forecast"))
```

```
grid.arrange(p1, p2, ncol = 1)
```

```
fit %>%
    report()
```

```
## # A tibble: 2 x 11
##   .model    sigma2 log_lik   AIC  AICc   BIC ar_roots  ma_roots     MSE     AMSE
##   <chr>      <dbl>   <dbl> <dbl> <dbl> <dbl> <list>    <list>     <dbl>    <dbl>
## 1 ARIMA  34.6        -181.  368.  368.  374. <cpl [1]> <cpl [0~ NA       NA
## 2 ETS     0.000614   -734. 1477. 1479. 1488. <NULL>    <NULL>    3.21e9  1.33e10
## # ... with 1 more variable: MAE <dbl>
```

```
data %>%
    select(Arrivals_transformed) %>%
    acf(lag.max = 4 * 6)
```



2.2 code

```
fit_tr <- train %>%
    model(AutoETS = ETS(Arrivals), SNaive = SNAIVE(Arrivals ~
        lag("year")), `Additive log ETS` = ETS(log(Arrivals) ~
        error("A") + trend("A") + season("A")), `Seasonally adjusted AutoETS` = decomposition_m
        ETS(season_adjust)))

fc_tr <- fit_tr %>%
```

```
    forecast(h = "2 years")

fc_tr %>%
    autoplot((data %>%
        filter(Quarter >= yearquarter("2008 Q1"))), level = NULL) +
    labs(title = "New Zealand -> Australia Flights", y = "Arrivals (count)") +
    guides(colour = guide_legend(title = "Forecast"))

acc_fc_tr <- accuracy(fc_tr, test) %>%
    select(c(".model", "RMSE", "MAE", "MPE", "MAPE")) %>%
    arrange(RMSE)
acc_fit_tr <- accuracy(fit_tr) %>%
    select(c(".model", "RMSE", "MAE", "MPE", "MAPE")) %>%
    arrange(RMSE)
acc_fit_tr %>%
    gt() %>%
    tab_header(title = md("**Model _fit_ accuracies**"), subtitle = md("as fitted to the _New Z
    opt_align_table_header(align = "center")
acc_fc_tr %>%
    gt() %>%
    tab_header(title = md("**Model _forecasting_ accuracies**"),
        subtitle = md("as fitted to the _New Zealand -> Australia_ flight arrivals (RMSE sorted
    opt_align_table_header(align = "center")
```

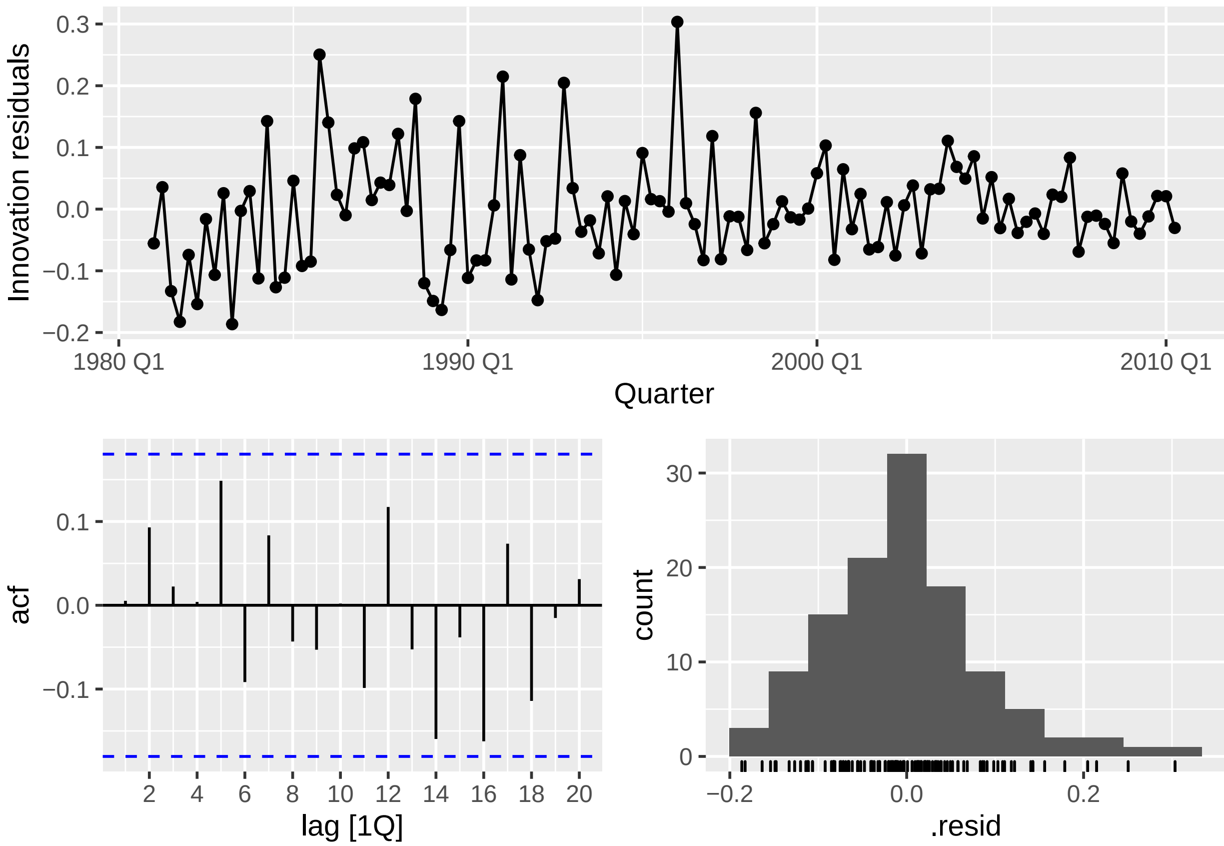
```
fit_tr %>%
    select(AutoETS) %>%
    report()
```

```
## Series: Arrivals
## Model: ETS(M,A,M)
##   Smoothing parameters:
##     alpha = 0.656
##     beta  = 1e-04
##     gamma = 0.222
##
##   Initial states:
##   l[0] b[0] s[0] s[-1] s[-2] s[-3]
##  75567 2028    1  1.23   1.1 0.671
##
##   sigma^2:  0.0083
##
##  AIC AICc  BIC
## 2806 2807 2831
```

```
fit_tr %>%
    select("Seasonally adjusted AutoETS") %>%
    report()
```

```
## Series: Arrivals
## Model: STL decomposition model
## Transformation: log(Arrivals)
## Combination: season_adjust + season_year
##
## ===========================================
##
## Series: season_adjust
## Model: ETS(A,N,N)
##   Smoothing parameters:
##     alpha = 0.832
##
##   Initial states:
##  l[0]
##  11.2
##
##   sigma^2:  0.0056
##
##   AIC  AICc   BIC
## -44.2 -44.0 -35.9
##
## Series: season_year
## Model: SNAIVE
##
## sigma^2: 0
```

```
gg_tsresiduals(fit_tr %>%
    select("AutoETS"))
```




```
augment(fit_tr %>%
    select(AutoETS)) %>%
    features(.innov, ljung_box, lag = 10, dof = 6)
```

```
## # A tibble: 1 x 3
##   .model  lb_stat lb_pvalue
##   <chr>     <dbl>     <dbl>
## 1 AutoETS    6.45     0.168
```

2.3 code

```
# Creating the sets for cross-validation
train_cv <- data %>%
    stretch_tsibble(.init = 4 * 10, .step = 1) %>%
    relocate(Quarter, Arrivals, .id)

# CV accuracy
fit_cv <- train_cv %>%
    model(`CV AutoETS` = ETS(Arrivals), `CV SNaive` = SNAIVE(Arrivals),
        `CV Additive log ETS` = ETS(log(Arrivals) ~ error("A") +
            trend("A") + season("A")), `CV Seasonally adjusted autoETS` = decomposition_model(S
            trend() + season(window = "periodic"), robust = T),
            ETS(season_adjust)))
fc_cv <- fit_cv %>%
```

```
    forecast(h = "2 years")
acc_cv <- accuracy(fc_cv, data) %>%
    select(c(".model", "RMSE", "MAE", "MPE", "MAPE"))
rbind(acc_cv, acc_fc_tr) %>%
    arrange(RMSE) %>%
    gt() %>%
    tab_header(title = md("**Forecasting model accuracies: Cross Validated (CV) vs the models i
        subtitle = md("as fitted to the _New Zealand -> Australia_ flight arrivals (RMSE sorted
    opt_align_table_header(align = "center")
```

3.1 code

```
usgdp <- global_economy %>%
    filter(Country == "United States") %>%
    mutate(GDP = (GDP/Population) * 100) %>%
    select(Year, GDP)

lambda <- usgdp %>%
    features(GDP, features = guerrero) %>%
    pull(lambda_guerrero)

usgdp %>%
    autoplot(BoxCox(GDP, lambda)) + labs(y = "US$", title = latex2exp::TeX(paste0("GDP per capi
    round(lambda, 2))))
```

```
usgdp_fit <- usgdp %>% model ("ARIMA" = ARIMA(box_cox(GDP, lambda)))
p2 <- usgdp_fit %>% residuals() %>% ACF() %>% autoplot() + labs(title = "ACF of auto ARIMA resi

us_recessions <- c(1960, 1970,1973,1980,1990,2001,2008) #https://www.thebalance.com/the-history

p1 <- usgdp %>% autoplot(GDP) + #plot
        labs(y = "US$", title = "auto ARIMA fitted to transformed US GDP", subtitle = "lambda =
        geom_line(data = fitted(usgdp_fit),
                  aes(y = .fitted, colour = .model), size = 0.8) +
        geom_vline(xintercept = us_recessions, colour = "gray", linetype = "longdash")


grid.arrange(p1, p2, ncol = 1)
```

3.2 code

```
usgdp %>%
    gg_tsdisplay(difference(box_cox(GDP, lambda)), plot_type = "partial") +
    labs(title = "First order difference residuals transformed US GDP")
usgdp %>%
    features(difference(box_cox(GDP, lambda)), unitroot_kpss) %>%
    gt() %>%
```

```
    tab_header(title = md("**Unitroot KPSS test**")) %>%
    opt_align_table_header(align = "center")
```

```
usgdp_arima_fit <- usgdp %>%
    model(`ARIMA(1,1,0)` = ARIMA(box_cox(GDP, lambda) ~ pdq(1,
        1, 0)), `ARIMA(0,1,1)` = ARIMA(box_cox(GDP, lambda) ~
        pdq(0, 1, 1)))
usgdp_arima_fit %>%
    report() %>%
    select(-c(ar_roots, ma_roots)) %>%
    gt() %>%
    tab_header(title = md("**Manual ARIMA model comparison**")) %>%
    opt_align_table_header(align = "center")

augment(usgdp_arima_fit %>%
    select("ARIMA(1,1,0)")) %>%
    features(.innov, ljung_box, lag = 10, dof = 2) %>%
    gt() %>%
    tab_header(title = md("**Box-Ljung for the `ARIMA(1,1,0)`**")) %>%
    opt_align_table_header(align = "center")
```

3.3 code

```
fit <- usgdp %>%
    model(ARIMA = ARIMA(box_cox(GDP, lambda) ~ pdq(p = 1, d = 1,
        q = 0)), ETS = ETS(GDP))
fc <- fit %>%
    forecast(h = 20)

usgdp %>%
    autoplot(GDP) + geom_line(data = fitted(fit), aes(y = .fitted,
    colour = .model), size = 0.8) + autolayer(fc, level = 90,
    size = 0.8) + labs(title = "US GDP", ylab = "$US")
```