Forecasting demand in Airplanes

Jackson Cates

8/29/2020

Libraries

```
library(readr)
library(tsibble)
library(dplyr)
library(tidyr)
library(ggplot2)
library(fable)
library(feasts)
library(urca)
library(gridExtra)
```

Reading in data

```
airplanes = read_csv('../Data/AirPassengers.csv')
```

```
## Parsed with column specification:
## cols(
##   monthdate = col_character(),
##   passengers = col_double()
## )
```

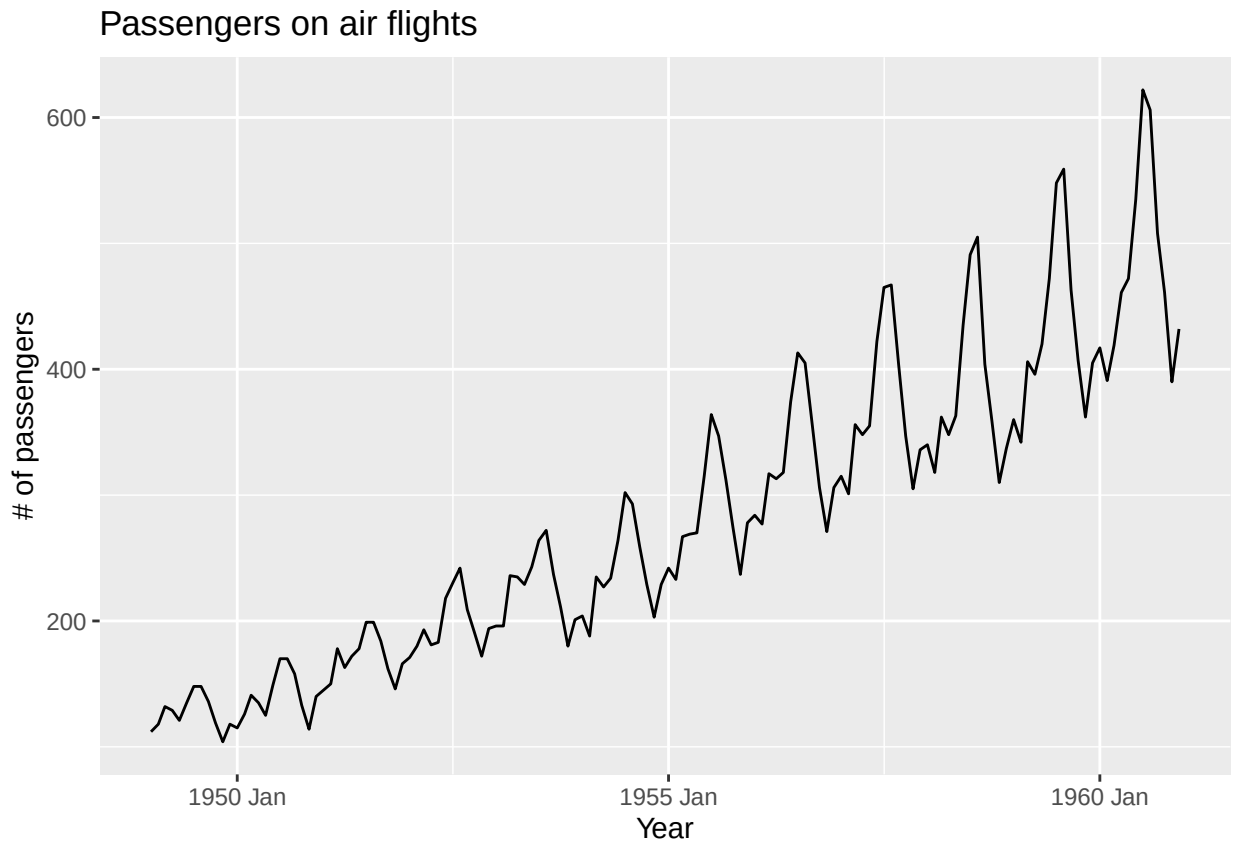
```
# Turns all of it into a tsibble
airplanes = airplanes %>%
  mutate(month = yearmonth(monthdate)) %>%
  select(-monthdate) %>%
  as_tsibble(index = month)
airplanes
```

```
## # A tsibble: 144 x 2 [1M]
##    passengers    month
##         <dbl>    <mth>
##  1        112 1949 Jan
##  2        118 1949 Feb
##  3        132 1949 Mar
##  4        129 1949 Apr
##  5        121 1949 May
##  6        135 1949 Jun
##  7        148 1949 Jul
##  8        148 1949 Aug
##  9        136 1949 Sep
## 10        119 1949 Oct
```

```
## # ... with 134 more rows
```

Time-plot

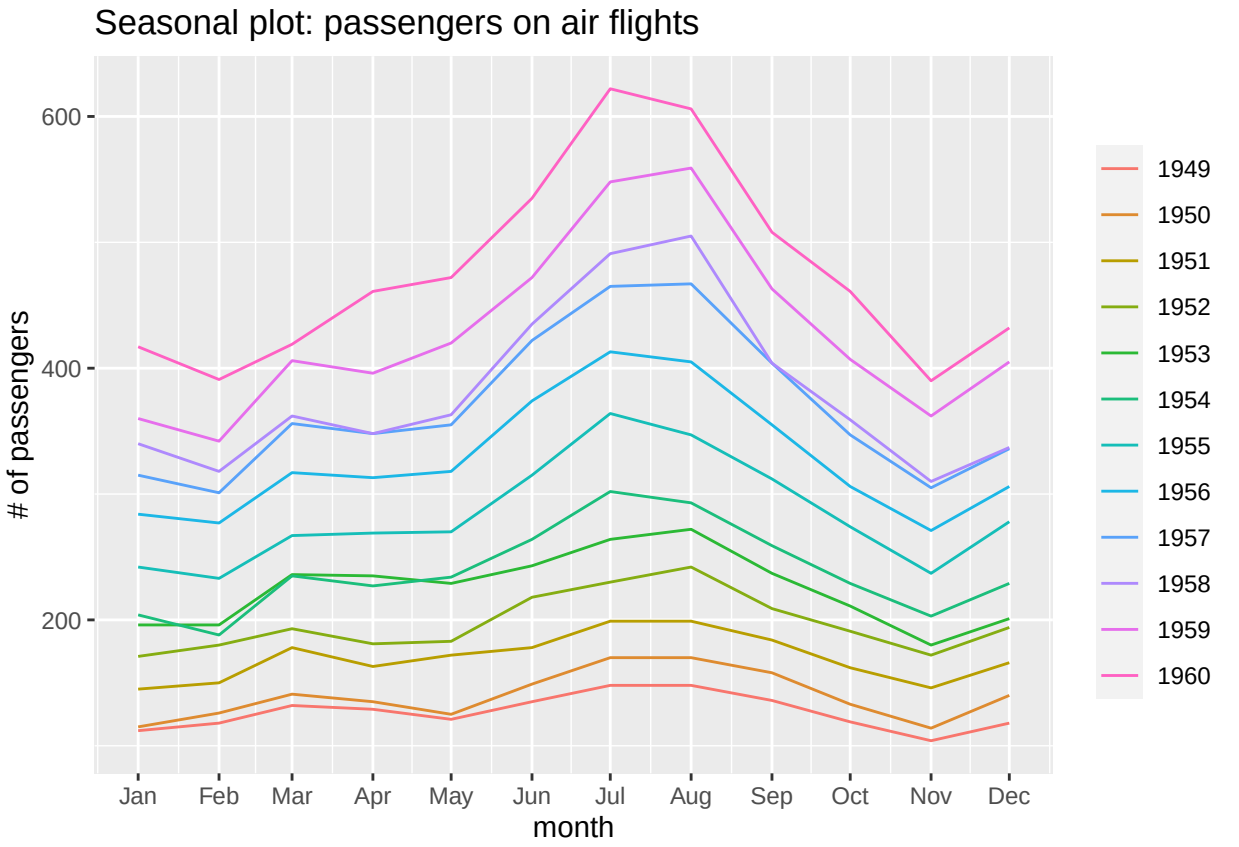
```
airplanes %>%
  autoplot(passengers) +
  ggtitle("Passengers on air flights") +
  ylab("# of passengers") +
  xlab("Year")
```



- It seems like the variance of the plot is increasing as the date gets farther

Seasonal Plot

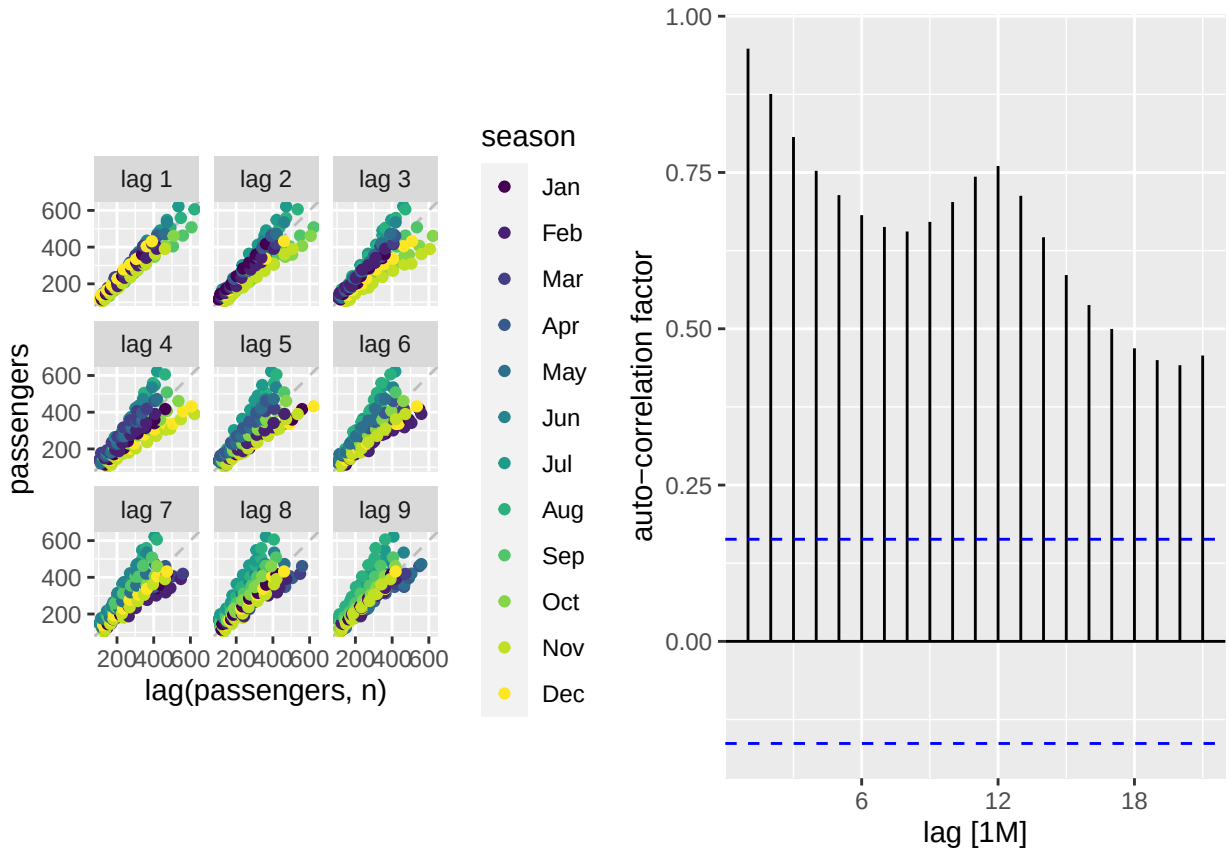
```
airplanes %>% gg_season(passengers) +
  ggtitle("Seasonal plot: passengers on air flights") + ylab("# of passengers")
```



- Notice that from above, as the year gets larger the amount of passengers get larger (makes sense from the time plot)
- Notice that the peak of the season gets more extreme as we get further in the year. Seems like we have peaks during July and August (summer) and at March. Also, we have troughs in February and November
- We may have seasonality

Lag plots

```
plot1 = airplanes %>% gg_lag(passengers, geom="point")
plot2 = airplanes %>% ACF(passengers) %>% autoplot() + ylab("auto-correlation factor")
grid.arrange(plot1, plot2, ncol=2)
```



- As seen in the plots above, we do have a high auto-correlation

Unit root test

Using the KPSS test, we have the following null hypothesis

H_O : The data is stationary

H_A : The data is not stationary

```
airplanes %>% features(passengers, unitroot_kpss)
```

```
## # A tibble: 1 x 2
##   kpss_stat kpss_pvalue
##       <dbl>       <dbl>
## 1      2.74        0.01
```

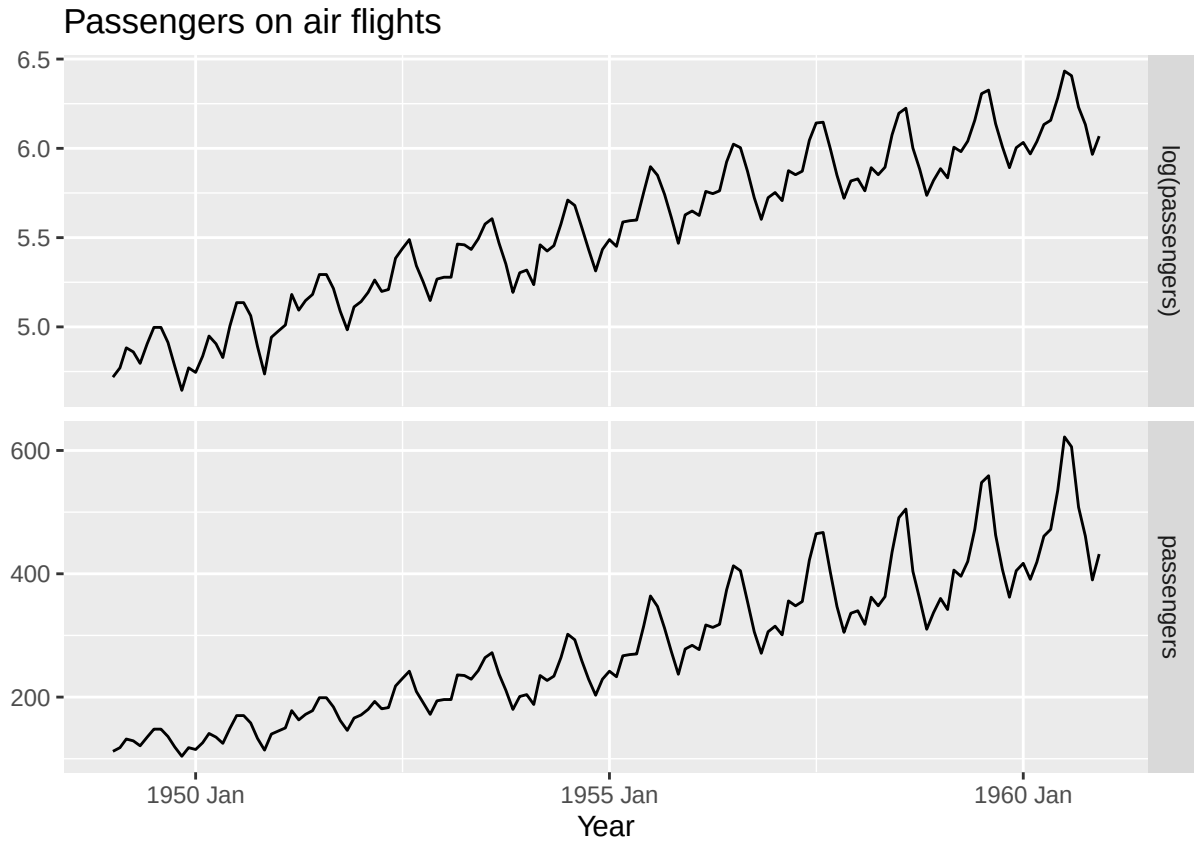
$p - value = 0.01$

We do indeed reject H_O and say our data is currently **non-stationary**. So we will need to do some difference. This will be a **ARIMA** model.

Differencing

We can first take a log transformation to stabilize the increased variance

```
airplanes %>%
  mutate(log(passengers)) %>%
  gather() %>%
  ggplot(aes(x = month, y = value)) +
  geom_line() +
  facet_grid(key ~ ., scales = "free_y") +
  xlab("Year") + ylab("") +
  ggtitle("Passengers on air flights")
```

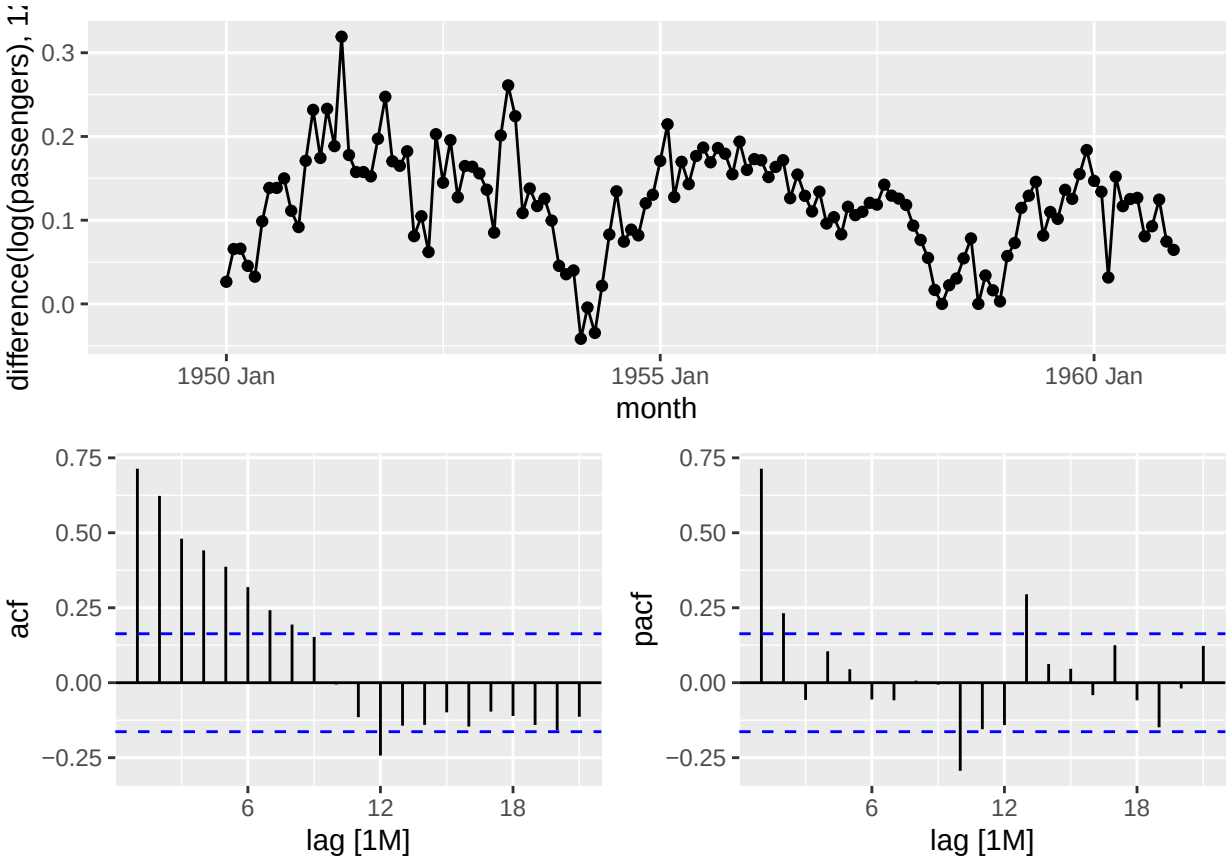


The variance do look more constant now. So we do have some seasonality within our model. Lets take the difference across seasons.

```
airplanes %>% gg_tsdisplay(difference(log(passengers), 12), plot_type = 'partial')
```

```
## Warning: Removed 12 row(s) containing missing values (geom_path).
```

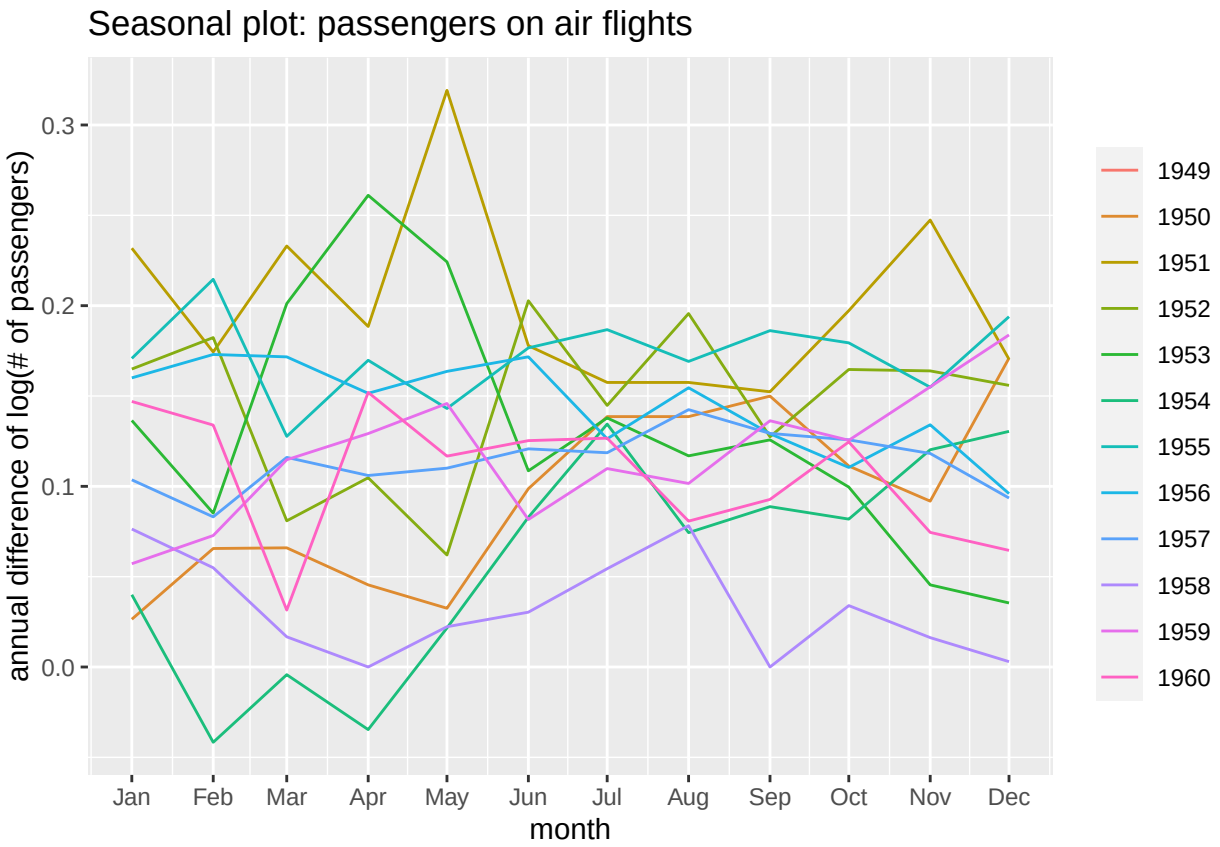
```
## Warning: Removed 12 rows containing missing values (geom_point).
```



Still looks a bit periodic. Also we do get a decay in the ACF graph. There is a lag spike at 12 in the ACF. In the PACF there are various lag spikes, but none at 12, 24, etc. There also seems to be no seasonal lags. Lets look at the seasonal plot.

```
airplanes %>% gg_season(difference(log(passengers), 12)) +
  ggtitle("Seasonal plot: passengers on air flights") + ylab("annual difference of log(# of passengers)")
```

```
## Warning: Removed 12 row(s) containing missing values (geom_path).
```



We do seem to lose the seasonality of the data. Lets look at the KPSS test.

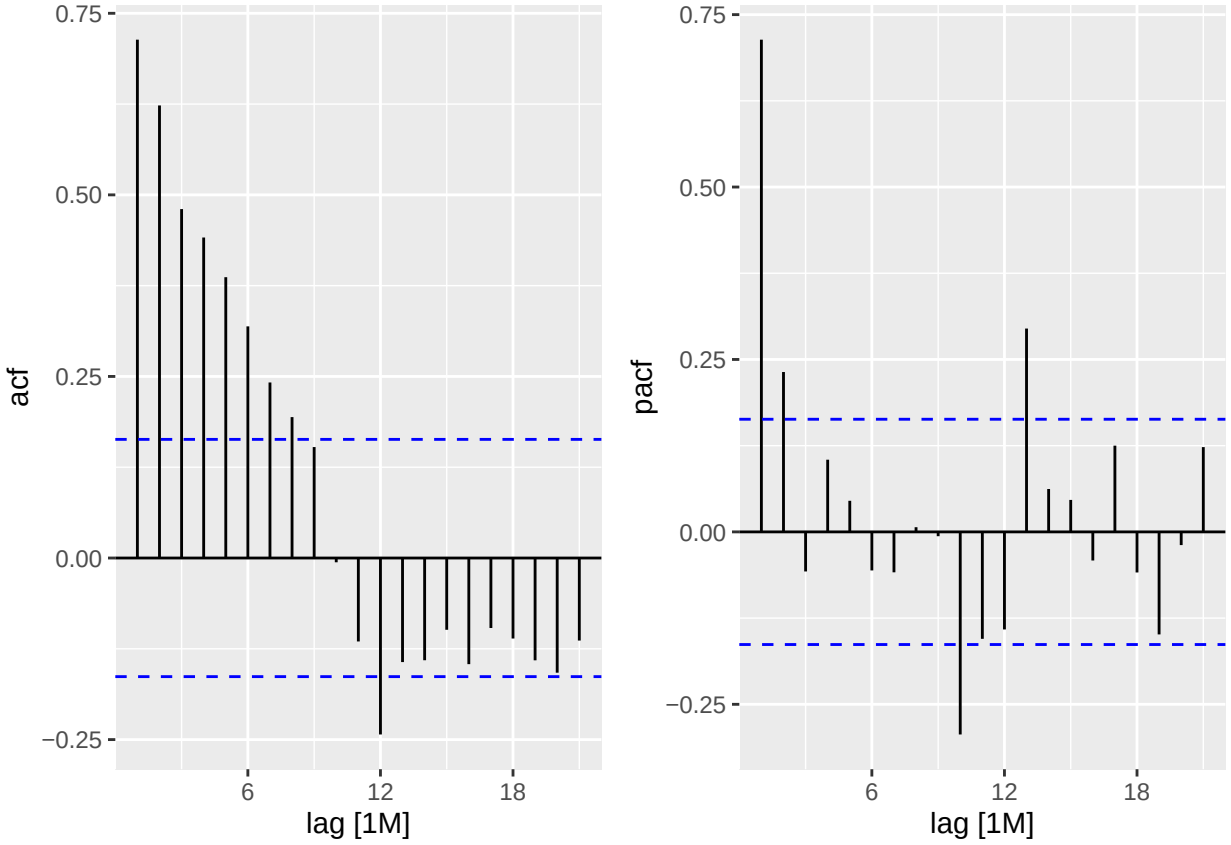
```
airplanes %>% features(difference(log(passengers), 12), unitroot_kpss)
```

```
## # A tibble: 1 x 2
##   kpss_stat kpss_pvalue
##       <dbl>       <dbl>
## 1     0.368      0.0909
```

We do in-fact have a more stationary data. We can say that it is "not bad."

Choosing a model

```
plot1 = airplanes %>% ACF(difference(log(passengers), 12)) %>% autoplot()
plot2 = airplanes %>% PACF(difference(log(passengers), 12)) %>% autoplot()
grid.arrange(plot1, plot2, ncol=2)
```



From the ACF and PACF plots above, it seems that a $ARIMA(3,0,0)(1,1,0)_{12}$ for the following reasons:

- The ACF is exponentially decaying for nonseasonal lags. The ACF seems to be exponentially decaying for seasonal lags as well.
- There are 3 significant lags in the PACF for nonseasonal lags. There is also 1 significant lag in the ACF for seasonal lags.
- The only difference I took was a seasonal difference.

Fitting the model

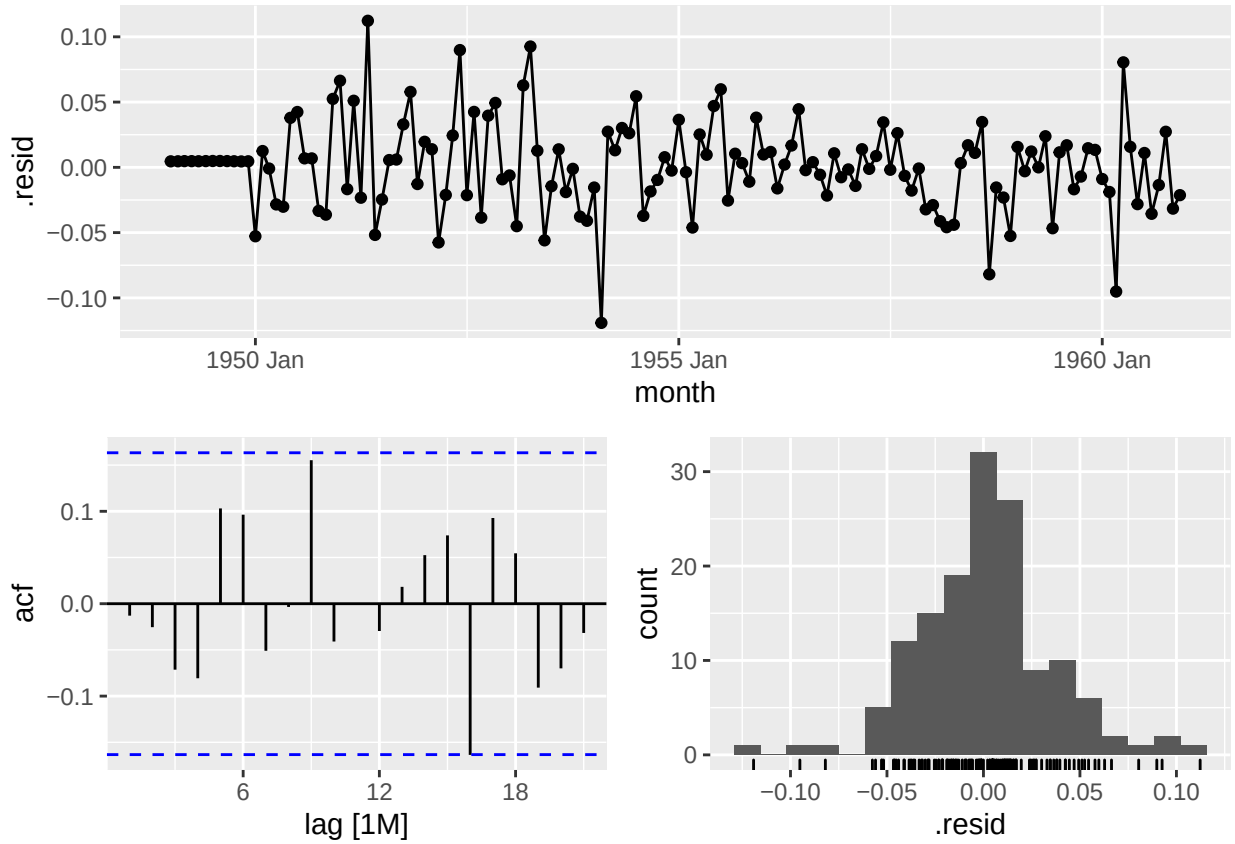
When fitting the original model, $ARIMA(3,0,0)(1,1,0)_{12}$, we have $AICc = -479.94$. I tried fitting other variations, and found that $ARIMA(2,0,0)(1,1,1)_{12}$ produces a smaller AICc, where $AICc = -486.74$.

```
fit = airplanes %>% model(ARIMA(log(passengers) ~ pdq(2, 0, 0) + PDQ(1, 1, 1)))
report(fit)
```

```
## Series: passengers
## Model: ARIMA(2,0,0)(1,1,1)[12] w/ drift
## Transformation: log(.x)
##
## Coefficients:
##          ar1     ar2     sar1     sma1  constant
##       0.5728  0.2663  -0.0502  -0.5205    0.0200
## s.e.  0.0846  0.0854   0.1602   0.1395    0.0016
##
## sigma^2 estimated as 0.001332:  log likelihood=249.7
## AIC=-487.39   AICc=-486.72   BIC=-470.09
```

Checking residuals

```
fit %>% gg_tsresiduals()
```




```
augment(fit) %>% features(.resid, ljung_box, lag = 12, dof = 4)
```

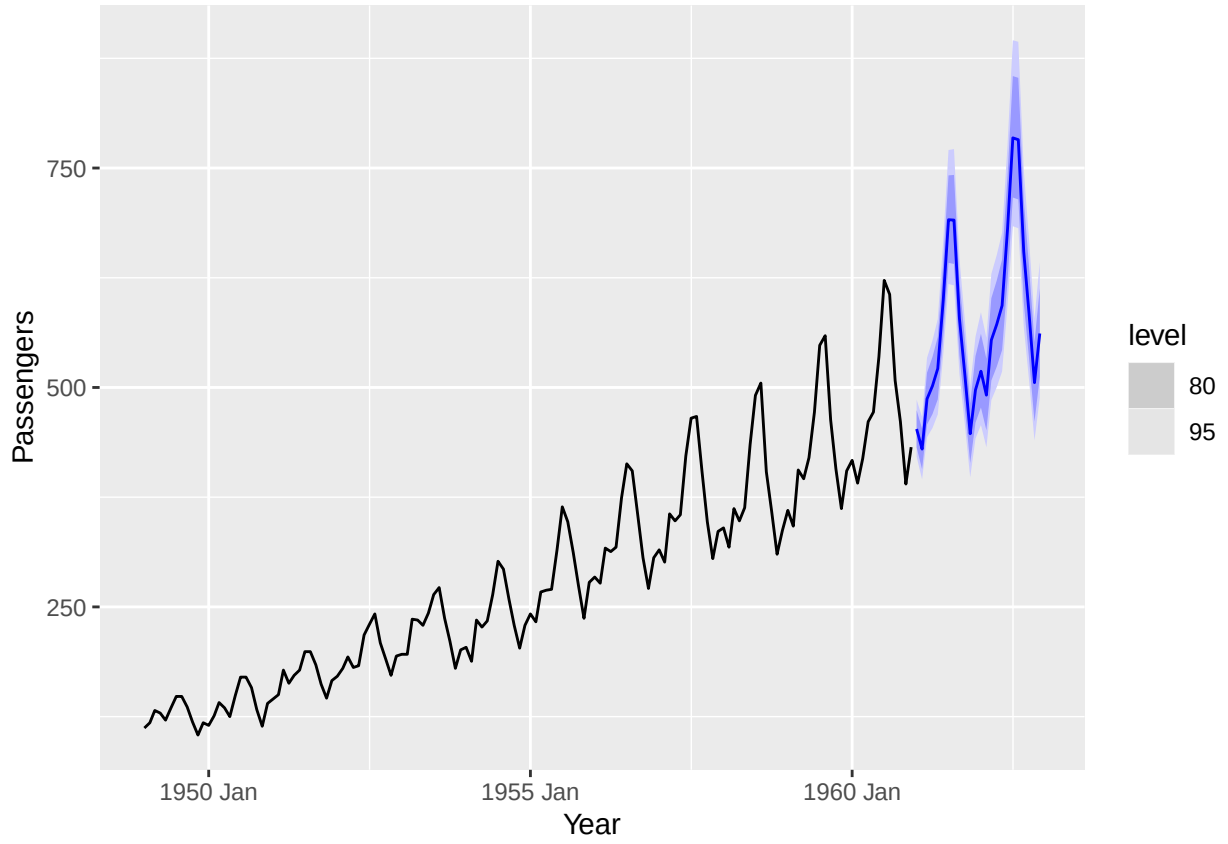
```
## # A tibble: 1 x 3
##   .model                                             lb_stat lb_pvalue
```

```
##   <chr>                                                  <dbl>     <dbl>
## 1 ARIMA(log(passengers) ~ pdq(2, 0, 0) + PDQ(1, 1, 1))    9.44     0.307
```

As seen above, our residuals do seem to be white noise.

Forecasting

```
airplanes %>%
  model(ARIMA(log(passengers) ~ pdq(2,0,0) + PDQ(1,1,1))) %>%
  forecast() %>%
  autoplot(airplanes) +
    ylab("Passengers") + xlab("Year")
```




```
accuracy(fit)
```

```
## # A tibble: 1 x 9
##   .model                    .type      ME  RMSE   MAE    MPE  MAPE  MASE    ACF1
##   <chr>                     <chr>   <dbl> <dbl> <dbl>  <dbl> <dbl> <dbl>   <dbl>
## 1 ARIMA(log(passengers) ~ ~ Train~ -0.124  9.95  7.00 0.0332  2.53 0.219 -0.0693
```