Climate Data Forecasting -

Atmospheric CO_2 Concentration / Temperature / Precipitation

Wolfgang Vollmer

2026-01-08

Contents

1 Forecasting of Potsdam - Temperature and Precipitation Climate Analysis

1.1 Stationarity and differencing

Stationary time series is one whose properties do not depend on the time at which the series is observed. Thus, time series with trends, or with seasonality, are not stationary — the trend and seasonality will affect the value of the time series at different times. On the other hand, a white noise series is stationary — it does not matter when you observe it, it should look much the same at any point in time.

Stationary time series will have no predictable patterns in the long-term. Time plots will show the series to be roughly horizontal (although some cyclic behaviour is possible), with constant variance.

If y_t is a *stationary* time series, then for all s, the distribution of
$(y_t, ..., y_{t+s})$ does not depend on t.

If Time Series data with seasonality are non-stationary

- => first take a seasonal difference
- if seasonally differenced data appear are still non-stationary
- => take an additional first seasonal difference

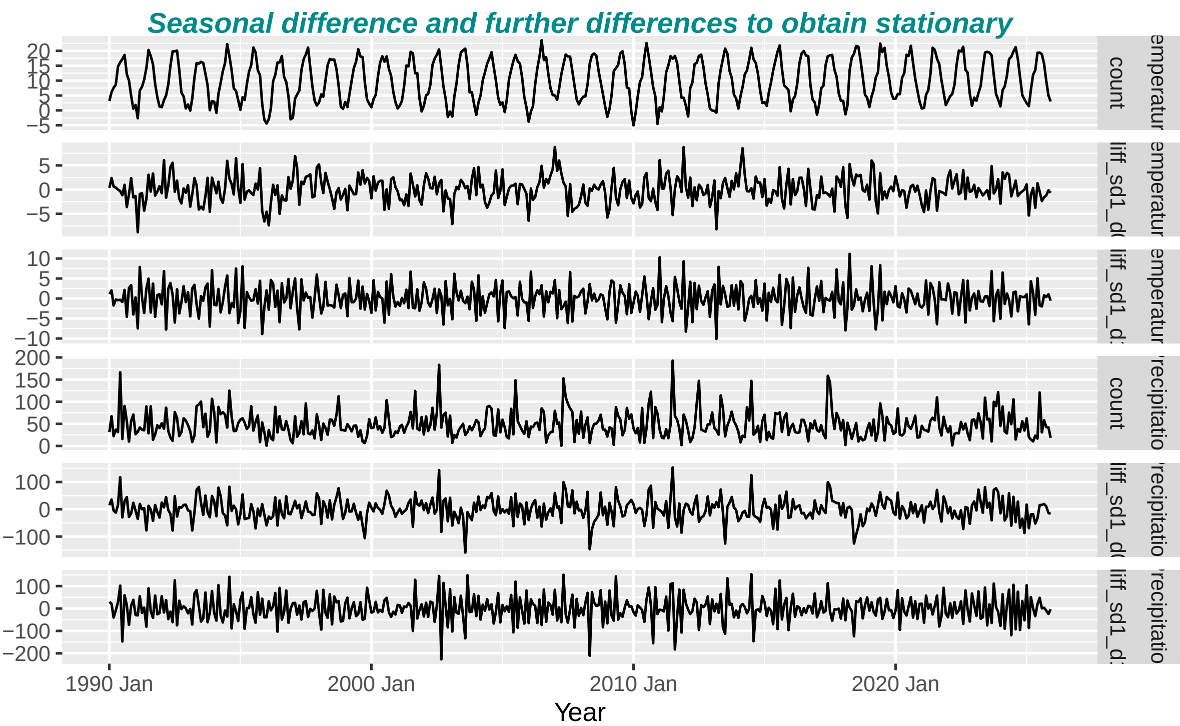
The model fit residuals have to be stationary. For good forecasting this has to be verified with residual diagnostics.

Essential:

- Residuals are uncorrelated
- The residuals have zero mean

Useful (but not necessary):

- The residuals have constant variance.
- The residuals are normally distributed.



1.1.1 Unitroot KPSS Test - fix number of seasonal differences/differences required

- For **ARIMA model** use Unitroot KPSS Test to fix number of differences
  - unitroot_nsdiffs() to determine D (the number of seasonal differences to use)
  - unitroot_ndiffs() to determine d (the number of ordinary differences to use)
  - The selection of the other model parameters (p,q,P and Q) are all determined by minimizing the AICc
- kpss test of stationary of the differentiated data (with seasonal and ordinary differences) used in the **ARIMA model**
  - stationary times series: the distribution of $(y_t, ..., y_{t-s})$ does not depend on t.

  - *Null Hypothesis* H_0: stationary is given in the time series: data are stationary and non seasonal
    - for $p < \alpha = 0.05$: reject H_0
    - for $p >> \alpha = 0.05$: conclude: differentiated data are stationary and non seasonal
- kpss test of stationary w/ unitroot_nsdiffs & unitroot_ndiff provides
  - minimum number of seasonal & ordinariel differences required for a stationary series
  - first fix required seasonal differences and then apply ndiffs to the seasonally differenced data
  - returns 1 => for stationarity one seasonal difference rsp. difference is required

```
#>   ndiffs gives the number of differences required and
#>  nsdiffs gives the number of seasonal differences required
#>  - to make a series stationary (test is based on the KPSS test
#> unitroot_kpss test to define seasonal (nsdiffs) and ordinary (ndiffs) differences
#> # A tibble: 2 x 5
#>   Measure       kpss_stat kpss_pvalue nsdiffs ndiffs
#>   <fct>             <dbl>       <dbl>   <int>  <int>
#> 1 Temperature      0.783         0.01       1      1
#> 2 Precipitation    0.0286        0.1        0      0
#> unitroot_kpss test on (difference(count, 12) + difference())
#> # A tibble: 2 x 3
#>   Measure       kpss_stat kpss_pvalue
#>   <fct>             <dbl>       <dbl>
#> 1 Temperature     0.00720         0.1
#> 2 Precipitation   0.00276         0.1
```

1.1.2 Ljung-Box Test - independence/white noise of the time series

The Ljung-Box Test becomes important when checking independence/white noise of the forecasts residuals of the fitted ETS rsp. ARIMA models. There we have to check whether the forecast errors are normally distributed with mean zero

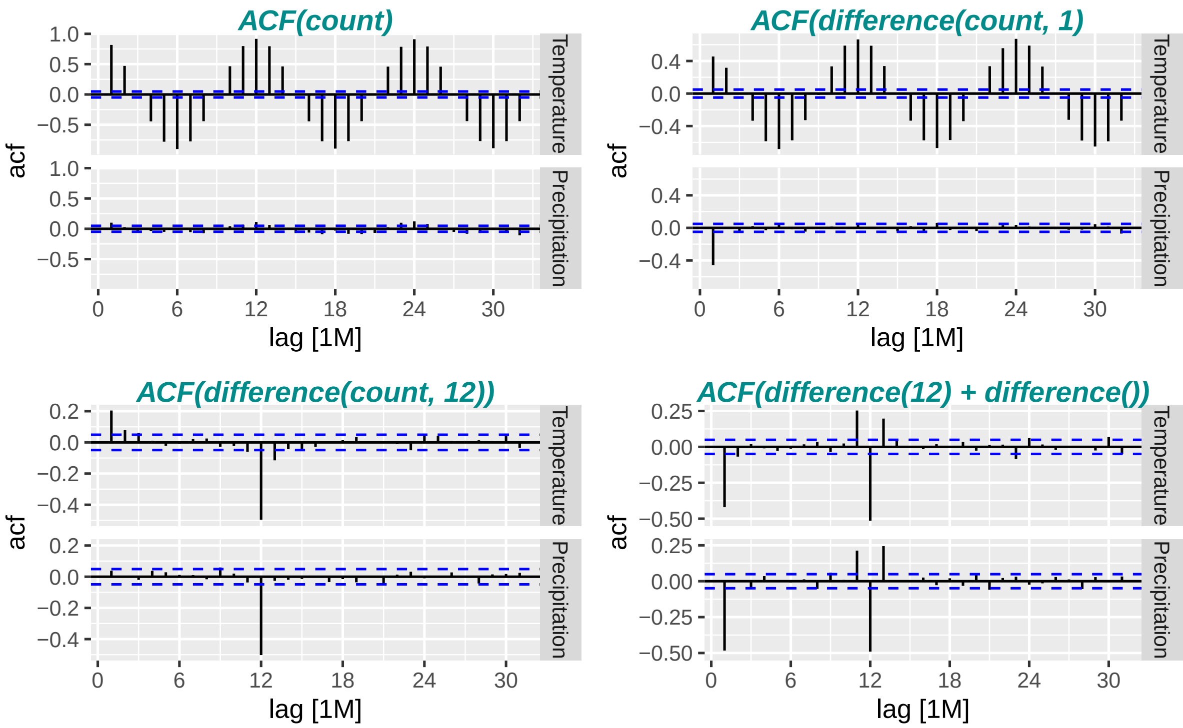
- augment(fit) |> features(.innov, ljung_box, lag=x, dof=y) (Ch. 5.4 Residaul diagnostics)
  - portmanteau test suggesting that the residuals are white noise
  - *Null Hypothesis* H_0: independence/white noise for residuals, i.e. each autocorrelation in the time series residuals of the model for lag l is close to zero.
    - for $p < \alpha = 0.05$: reject H_0
    - for $p >> \alpha = 0.05$: conclude: the residuals are not distinguishable from a white noise series
  - lag = 2*m (period of season, e.g. m=12 for monthly season) | no season: lag=10
  - dof = p + q + P + Q (for ARIMA models only, degree of freedom)

Time series with trend and/or seasonality is not stationary. Tests are to be taken on the residuals of the fitted models.

```
#> Ljung-Box test with (count), w/o differences
#> # A tibble: 2 x 3
#>   Measure       lb_stat  lb_pvalue
#>   <fct>           <dbl>      <dbl>
#> 1 Temperature    5636.  0
#> 2 Precipitation    45.6 0.00000170
#> Ljung-Box test on (difference(count, 12))
```

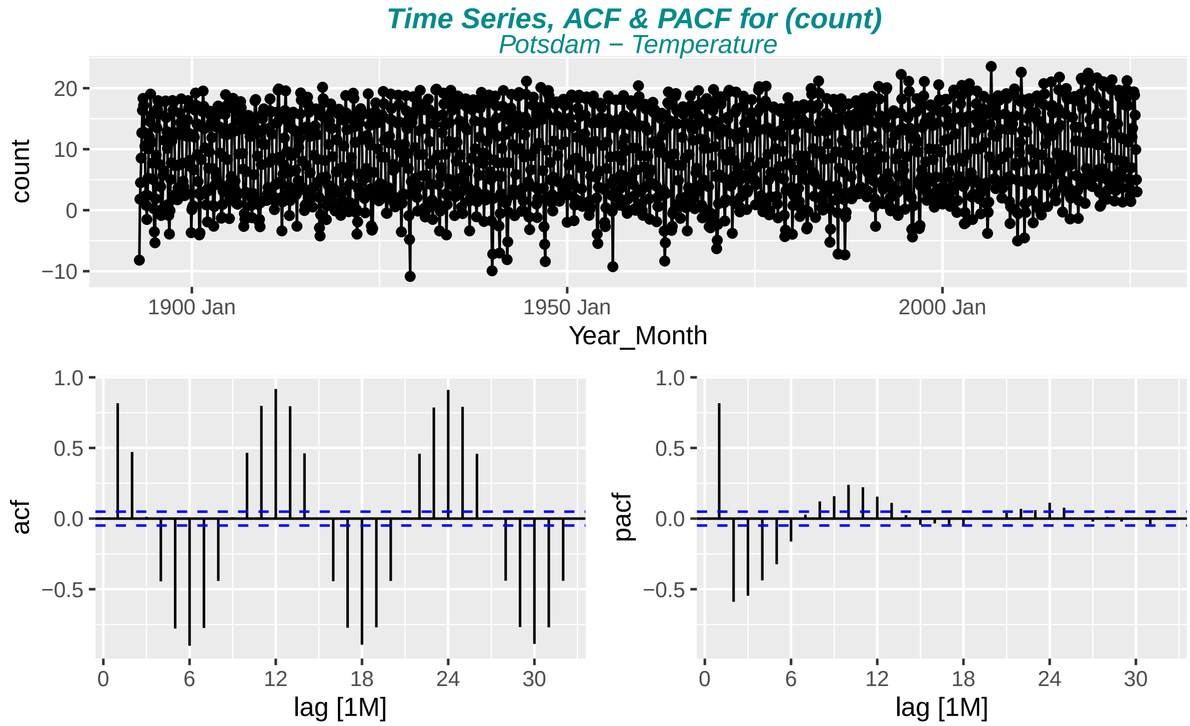
```
#> # A tibble: 2 x 3
#>   Measure       lb_stat lb_pvalue
#>   <fct>           <dbl>     <dbl>
#> 1 Temperature      86.8  2.30e-14
#> 2 Precipitation    13.8  1.82e- 1
#> Ljung-Box test on (difference(count, 12) + difference())
#> # A tibble: 2 x 3
#>   Measure       lb_stat lb_pvalue
#>   <fct>           <dbl>     <dbl>
#> 1 Temperature      295.         0
#> 2 Precipitation    386.         0
```

1.1.3 ACF (Autocorrelation Function) Plots of Differences

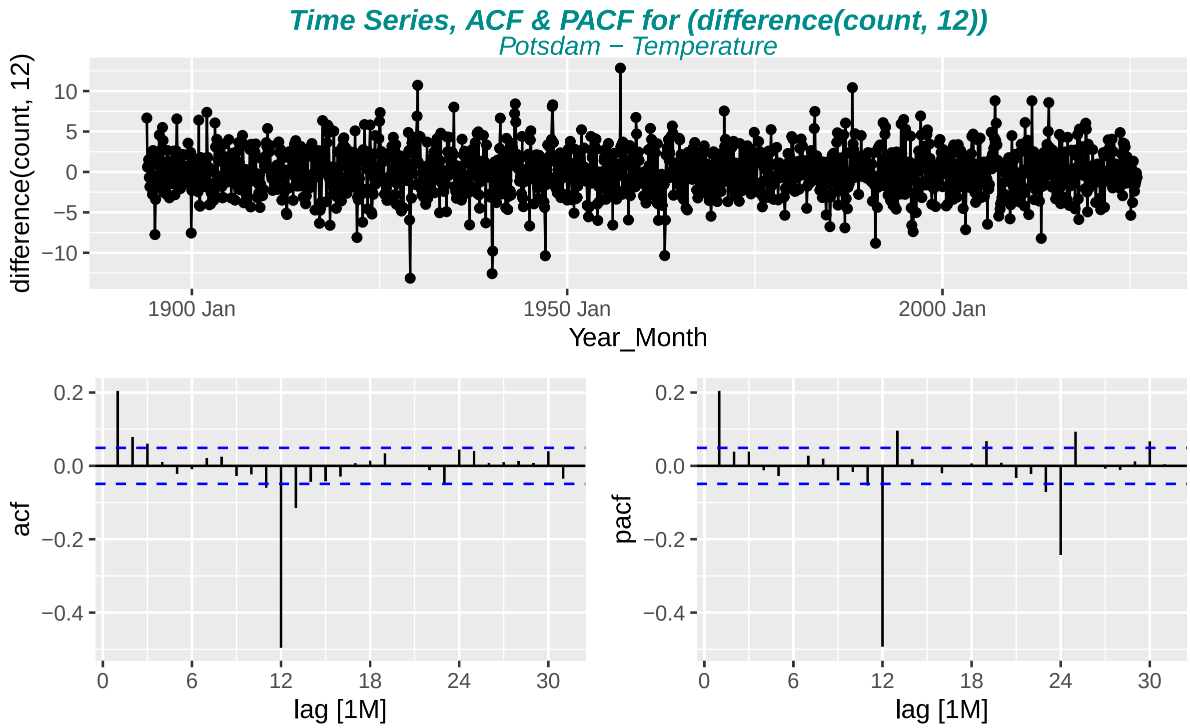


1.1.4 Time Series, ACF and PACF (Partial) Plots of Differences - for ARIMA p, q check

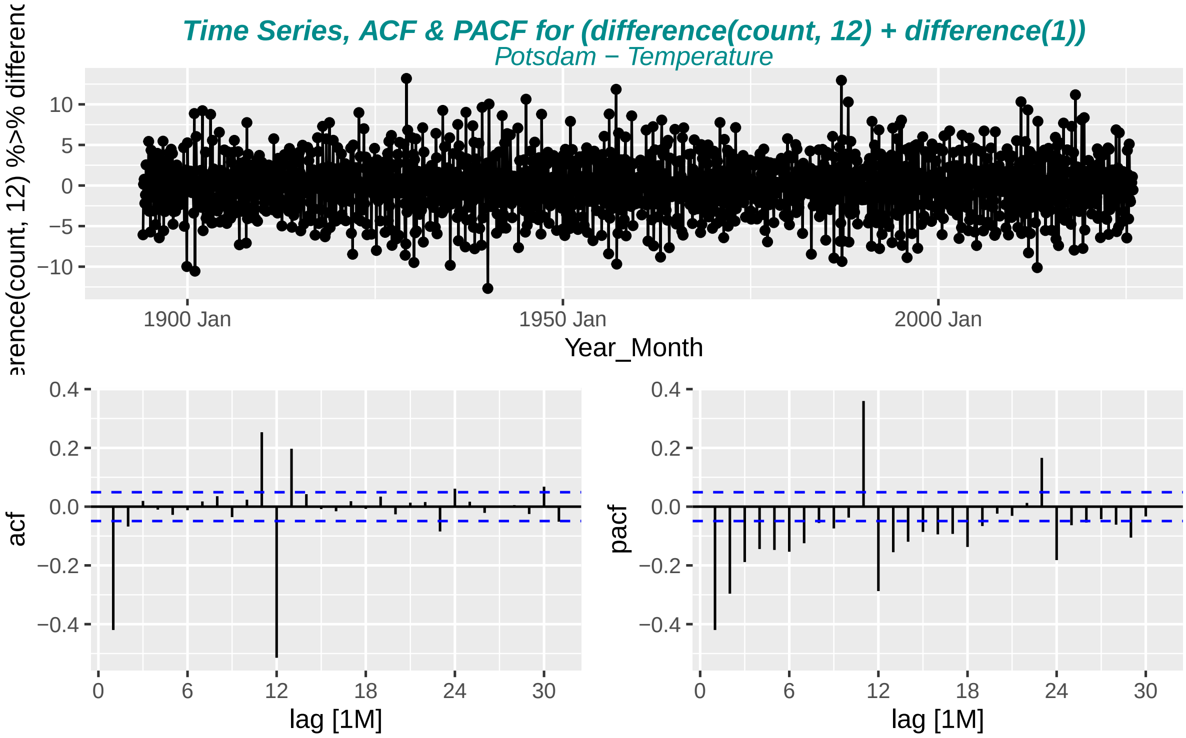
```
#> # A tibble: 1 x 4
#> # Groups:   City [1]
#>   City    Measure       Sum  Mean
#>   <chr>   <fct>       <dbl> <dbl>
#> 1 Potsdam Temperature 14269.  8.94
```



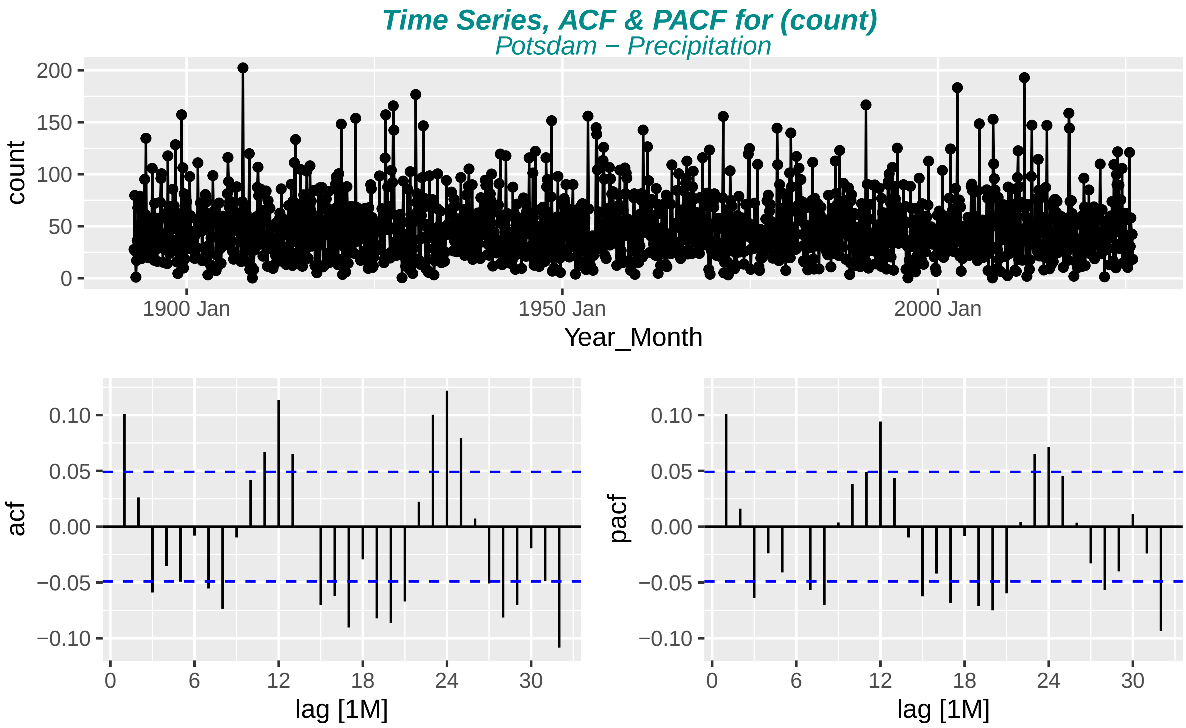

```
#> # A tibble: 1 x 4
#> # Groups:   City [1]
#>   City    Measure       Sum   Mean
#>   <chr>   <fct>       <dbl>  <dbl>
#> 1 Potsdam Temperature  29.8 0.0188
```



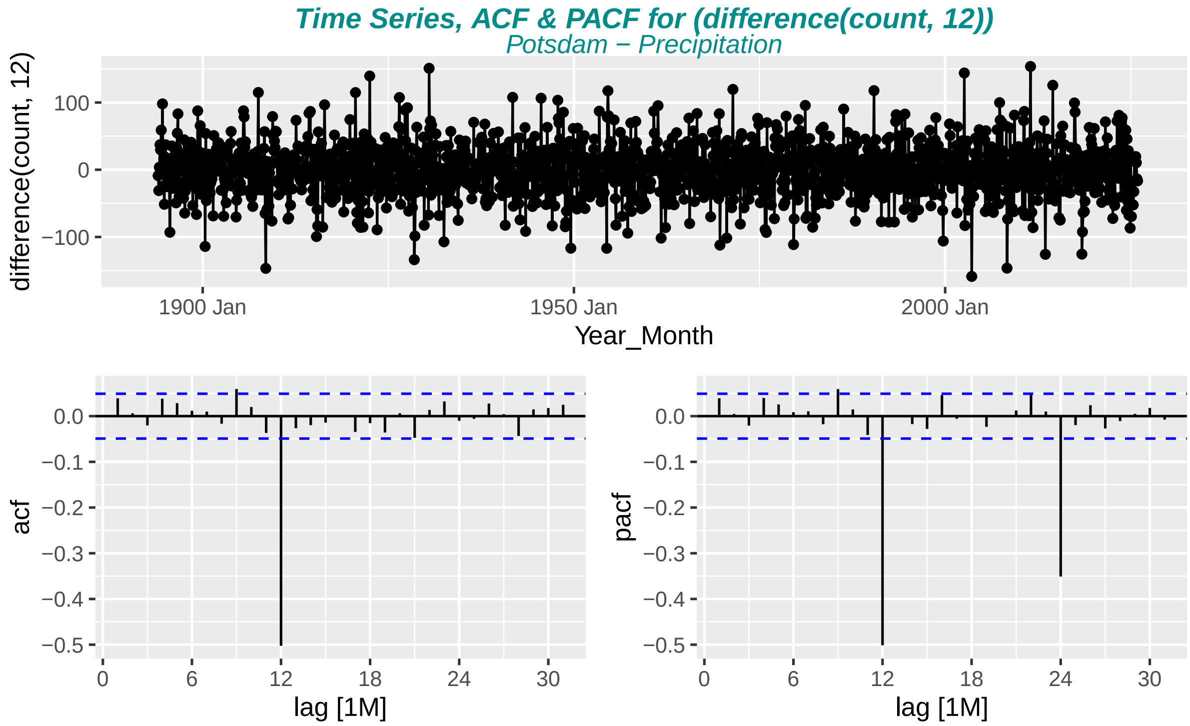

```
#> # A tibble: 1 x 4
#> # Groups:   City [1]
#>   City    Measure       Sum     Mean
#>   <chr>   <fct>       <dbl>    <dbl>
#> 1 Potsdam Temperature -7.34 -0.00464
```



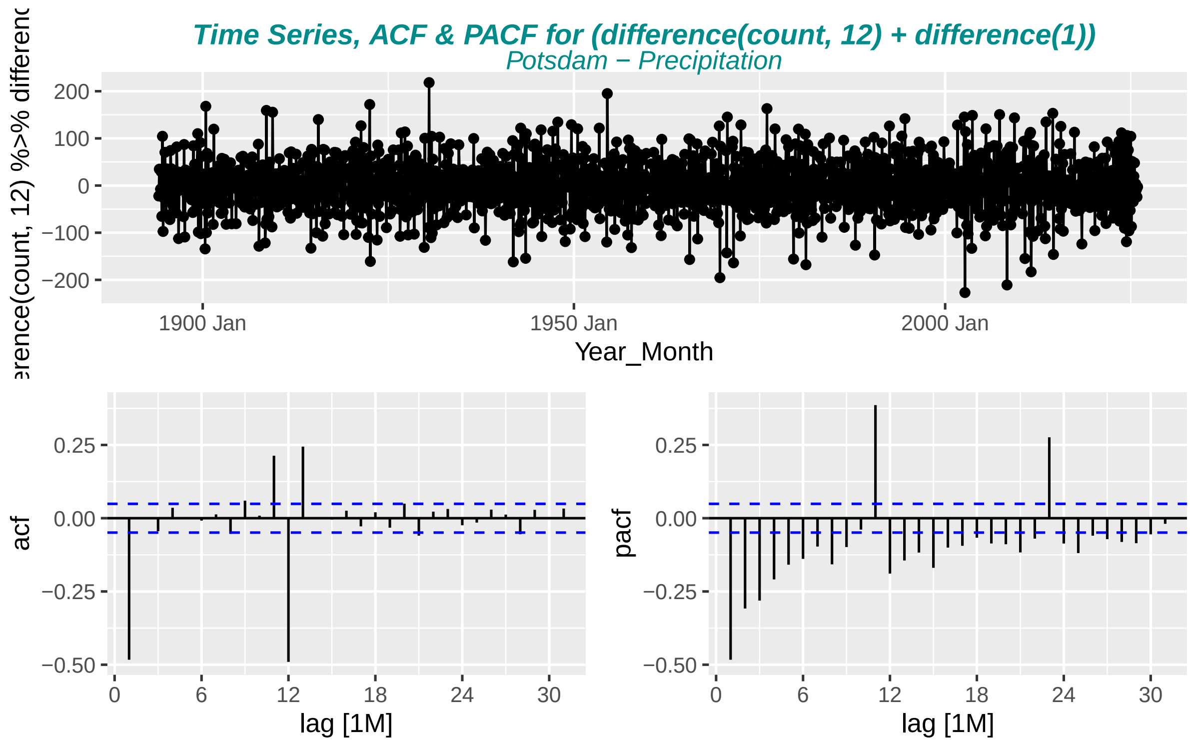

```
#> # A tibble: 1 x 4
#> # Groups:   City [1]
#>   City    Measure           Sum  Mean
#>   <chr>   <fct>           <dbl> <dbl>
#> 1 Potsdam Precipitation 77771.   48.7
```




```
#> # A tibble: 1 x 4
#> # Groups:   City [1]
#>   City    Measure          Sum    Mean
#>   <chr>   <fct>          <dbl>   <dbl>
#> 1 Potsdam Precipitation -39.4 -0.0249
```




```
#> # A tibble: 1 x 4
#> # Groups:   City [1]
#>   City    Measure          Sum     Mean
#>   <chr>   <fct>          <dbl>    <dbl>
#> 1 Potsdam Precipitation -7.80 -0.00493
```



2 ExponenTial Smoothing (ETS) Forecasting Models

Forecasts produced using exponential smoothing methods are weighted averages of past observations, with the weights decaying exponentially as the observations get older.

The parameters are estimated by maximising the "likelihood". The likelihood is the probability of the data arising from the specified model. AIC, AICc and BIC can be used here to determine which of the ETS models is most appropriate for a given time series (see output glance(fit_ets)).

The model selection is based on recognising key components of the time series (trend and seasonal) and the way in which these enter the smoothing method (e.g., in an additive, damped or multiplicative manner).

- Mauna Loa CO_2 data best Models: ETS(M,A,A) & ETS(A,A,A)
- Basel Temperature data best Models: ETS(A,N,A), ETS(A,A,A), ETS(A,Ad,A) (close togehter). Best Forecast accuracy is with ETS(A,A,A), ETS(A,Ad,A).
- Basel Precipitation data best Models: ETS(A,N,A), ETS(A,Ad,A), ETS(A,A,A) (close togehter). Best Forecast accuracy is with ETS(A,A,A), ETS(A,Ad,A), ETS(A,N,A),

Trend term "N" for Basel Temperature/Precipitation correspondends to a "pure" exponential smooothing which results in a slope $\beta = 0$. This results in a forecast predicting a constant level. This does not fit to the result of the STL decomposition. Therefore best model choice is **ETS(A,A,A)**.

Method Selection

Error term: either additive ("A") or multiplicative ("M").

Both methods provide identical point forecasts, but different prediction intervals and different likelihoods. AIC & BIC are able to select between the error types because they are based on likelihood.

Nevertheless, difference is for

- Mauna Loa CO_2 not relevant and AIC/AICc/BIC values are only a little bit smaller for multiplicative errors. The prediction intervall plots are fully overlapping.
- Basel Temperature AIC/AICc/BIC of additive error types are much better than the multiplicative ones.
- Basel Precipitation AIC/AICc/BIC of additive error types are much better than the multiplicative ones.

Note: For Basel Temperature and Precipitation Forecast plots the models ETS_MAdA, ETS_MMA, ETS_MMA, ETS_MNA are to be taken out since forecasts with multiplicative errors are exploding (forecast > 3 years impossible !!)

Therefore finally **Error term = "A"** is chosen in general.

Trend term: either none ("N"), additive ("A"), multiplicative ("M") or damped variants ("Ad", "Md").

Note: Mauna Loa CO_2 model ETS(A,Ad,A) fit plot shows to strong damping. For Basel Temperature model ETS(A,N,A) and ETS(A,Ad,A) are providing more or less the same forecast. This means that forecast remains on constant level since Trend "N" means "pure" exponentiell smoothing without trend (see above).

Therefore finally **Trend term = "A"** is chosen in general.

Seasonal term: either none ("N"), additive ("A") or multiplicative ("M").

For CO2 and Temperature Data we have a clear seasonal pattern and seasonal term adds always a (more or less) fix amount on level and trend component. Therefore "A" additve term is chosen. For Precipitation the seasonal patttern is only slight. Indead, a multiplicative seasonal term results in "exploding" forecasts.

Since monthly data are strongly seasonal **seasonal term "A"** is chosen.

2.1 ETS Models and their componentes

ETS model with automatically selected $ETS(A|M, N|A|M, N|A|M)$

```
#> [1] "model(ETS(count)) => provides default / best automatically chosen model"
#> # A mable: 2 x 3
#> # Key:     City, Measure [2]
#>   City    Measure                 ETS
#>   <chr>   <fct>               <model>
#> 1 Potsdam Temperature   <ETS(A,N,A)>
#> 2 Potsdam Precipitation <ETS(M,N,A)>
#> [1] "Potsdam Temperature"
#> Series: count
#> Model: ETS(A,N,A)
#>   Smoothing parameters:
#>     alpha = 0.0633865
#>     gamma = 0.0001000048
#>
#>   Initial states:
#>      l[0]      s[0]     s[-1]     s[-2]    s[-3] s[-4]    s[-5]    s[-6]
#>  9.536028 -7.972627 -4.925633 0.1500849 4.925383 9.017 9.447066 7.644086
#>     s[-7]      s[-8]     s[-9]    s[-10]    s[-11]
#>  4.418703 -0.3625626 -4.910643 -8.242478 -9.188379
#>
#>   sigma^2:  3.5378
#>
#>      AIC     AICc      BIC
#> 5662.654 5663.336 5731.343
#> [1] "Potsdam Precipitation"
#> Series: count
#> Model: ETS(M,N,A)
#>   Smoothing parameters:
#>     alpha = 0.0001002221
#>     gamma = 0.0001000348
#>
#>   Initial states:
#>      l[0]      s[0]     s[-1]     s[-2]     s[-3]    s[-4]    s[-5]    s[-6]
#>  48.90008 0.4824555 -6.308517 -4.489432 -2.175042 10.06408 18.57598 13.29243
#>     s[-7]     s[-8]     s[-9]    s[-10]    s[-11]
#>  9.992399 -13.15976 -10.77413 -8.28397 -7.216502
#>
#>   sigma^2:  0.3075
#>
#>      AIC     AICc      BIC
#> 9475.841 9476.523 9544.530
#> # A tibble: 2 x 8
#>   City    Measure       .model   AIC  AICc   BIC    MSE   MAE
#>   <chr>   <fct>         <chr>  <dbl> <dbl> <dbl>  <dbl> <dbl>
#> 1 Potsdam Temperature   ETS    5663. 5663. 5731.   3.47 1.45
#> 2 Potsdam Precipitation ETS    9476. 9477. 9545. 762.   0.430
```

Fit of different pre-defined $ETS(A|M, N|A|M, N|A|M)$ models

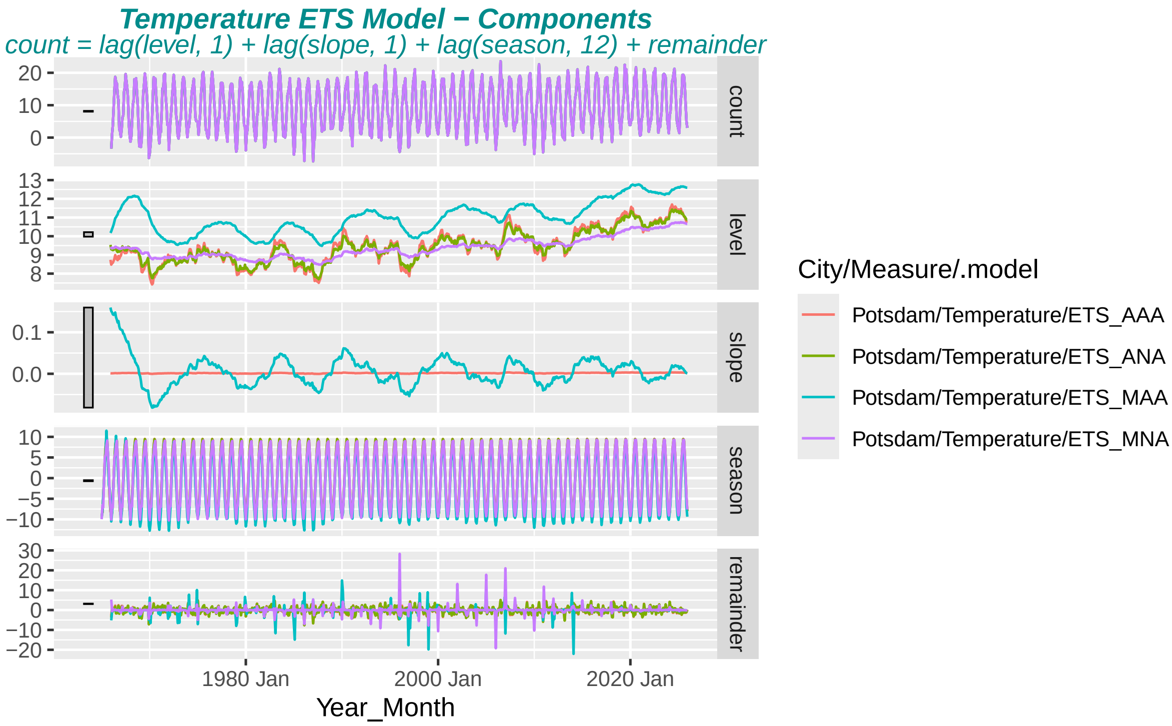
Model Selection by Information Criterion - lowest AIC, AICc, BIC

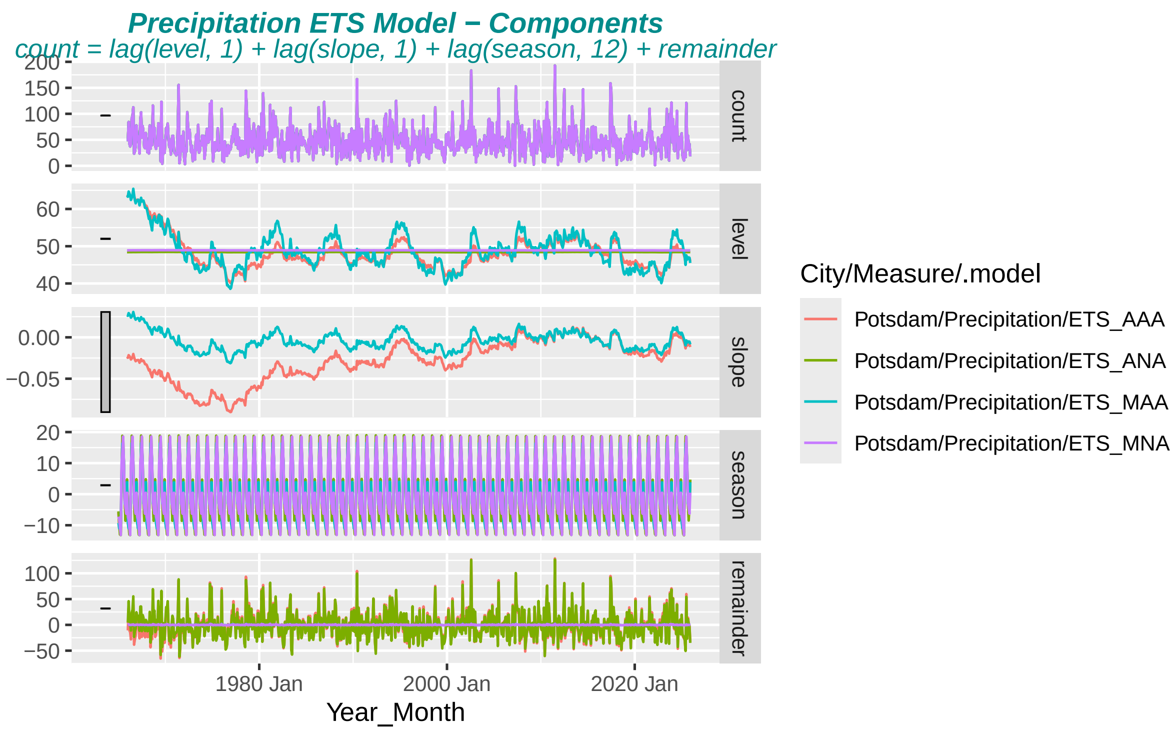
Best model fit with

- CV, AIC, AICc and BIC with the lowest values
- Adjusted R^2 the model with the highest value.

```
#> # A tibble: 16 x 9
#>    City    Measure       .model     AIC  AICc   BIC    MSE   AMSE    MAE
```

```
#>    <chr>   <fct>         <chr>    <dbl> <dbl> <dbl>  <dbl>  <dbl>  <dbl>
#>  1 Potsdam Temperature   ETS_ANA  5663. 5663. 5731.   3.47   3.52  1.45
#>  2 Potsdam Temperature   ETS_AAA  5668. 5669. 5746.   3.47   3.56  1.47
#>  3 Potsdam Temperature   ETS_AAdA 5668. 5669. 5751.   3.47   3.54  1.46
#>  4 Potsdam Temperature   ETS_AMA  5671. 5672. 5749.   3.49   3.58  1.47
#>  5 Potsdam Temperature   ETS_MNA  8543. 8544. 8612.   4.04   4.06  0.748
#>  6 Potsdam Temperature   ETS_MMA  8573. 8574. 8651.   4.47   4.99  0.753
#>  7 Potsdam Temperature   ETS_MAdA 8634. 8635. 8716.   3.88   3.94  0.810
#>  8 Potsdam Temperature   ETS_MAA  8656. 8657. 8734.   4.74   4.78  0.824
#>  9 Potsdam Precipitation ETS_MNA  9476. 9477. 9545. 762.   762.    0.430
#> 10 Potsdam Precipitation ETS_MAdA 9488. 9489. 9570. 757.   758.    0.432
#> 11 Potsdam Precipitation ETS_MMA  9517. 9518. 9595. 779.   780.    0.433
#> 12 Potsdam Precipitation ETS_MAA  9520. 9521. 9598. 779.   779.    0.438
#> 13 Potsdam Precipitation ETS_ANA  9540. 9541. 9609. 757.   757.   20.9
#> 14 Potsdam Precipitation ETS_AAdA 9544. 9545. 9627. 755.   756.   21.0
#> 15 Potsdam Precipitation ETS_AMA  9556. 9557. 9634. 770.   770.   21.0
#> 16 Potsdam Precipitation ETS_AAA  9557. 9558. 9635. 771.   771.   21.1
```





2.1.1 Residual Accuracy with one-step-ahead fitted residuals - check RMSE, MAE

Residual accuracy can be computed directly from models as the one-step-ahead fitted residuals are available. Select forecast models that minimises for lowest

- MAE (Mean absolute error, will lead to forecasts of the median) and
- RMSE (Root mean squared error, lead to forecasts of the mean)

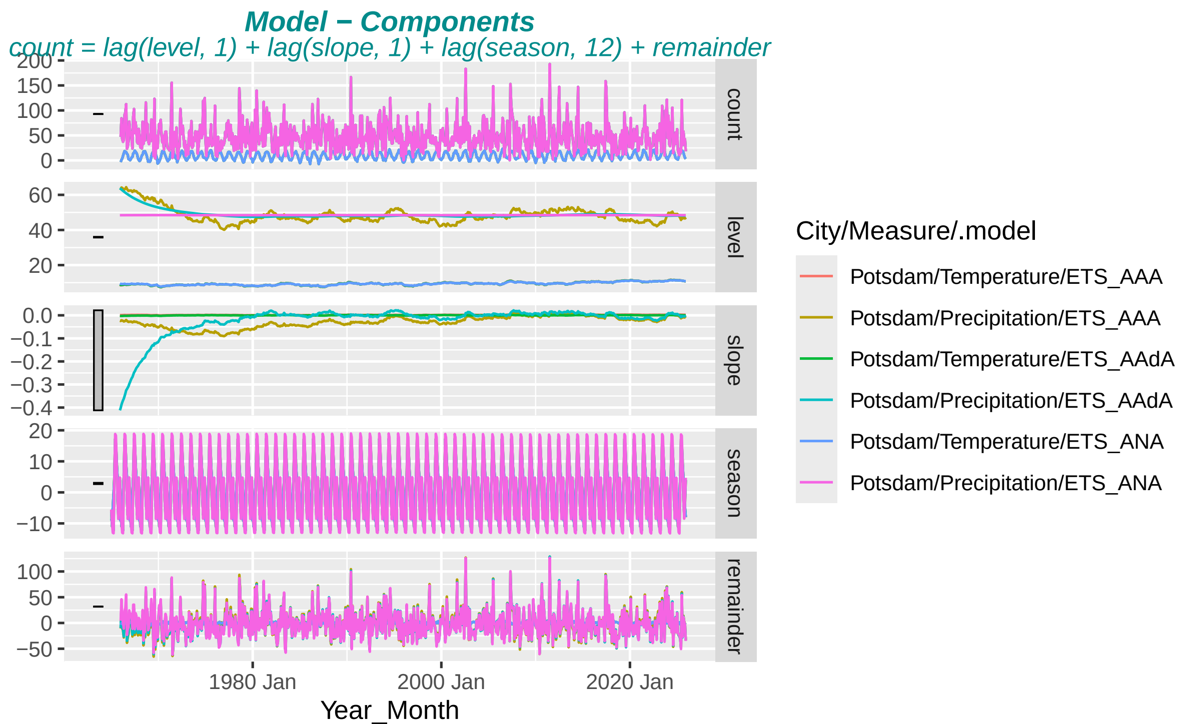
```
#> # A tibble: 16 x 7
#>    City    Measure       .model   .type           ME  RMSE   MAE
#>    <chr>   <fct>         <chr>    <chr>        <dbl> <dbl> <dbl>
#>  1 Potsdam Temperature   ETS_AAdA Training  0.0315    1.86  1.46
#>  2 Potsdam Temperature   ETS_ANA  Training  0.0299    1.86  1.45
#>  3 Potsdam Temperature   ETS_AAA  Training  0.0141    1.86  1.47
#>  4 Potsdam Temperature   ETS_AMA  Training -0.000109  1.87  1.47
#>  5 Potsdam Temperature   ETS_MAdA Training -0.0454    1.97  1.51
#>  6 Potsdam Temperature   ETS_MNA  Training  0.101     2.01  1.55
#>  7 Potsdam Temperature   ETS_MMA  Training -0.112     2.11  1.66
#>  8 Potsdam Temperature   ETS_MAA  Training -0.106     2.18  1.67
#>  9 Potsdam Precipitation ETS_AAdA Training -0.482    27.5  21.0
#> 10 Potsdam Precipitation ETS_ANA  Training -0.00109  27.5  20.9
#> 11 Potsdam Precipitation ETS_MAdA Training -0.165    27.5  21.0
#> 12 Potsdam Precipitation ETS_MNA  Training -0.453    27.6  21.1
#> 13 Potsdam Precipitation ETS_AMA  Training  0.347    27.7  21.0
#> 14 Potsdam Precipitation ETS_AAA  Training  0.166    27.8  21.1
#> 15 Potsdam Precipitation ETS_MAA  Training -0.485    27.9  21.3
#> 16 Potsdam Precipitation ETS_MMA  Training -1.14     27.9  21.4
```

2.1.2 Ljung-Box Test - independence/white noise of the forecasts residuals

```
#> Null Hypothesis of independence/white noise for residuals - for p < 0.05: reject H_0
#> # A tibble: 16 x 5
#>    City    Measure       .model   lb_stat    lb_pvalue
#>    <chr>   <fct>         <chr>      <dbl>        <dbl>
#>  1 Potsdam Precipitation ETS_MNA     12.9 0.913
```

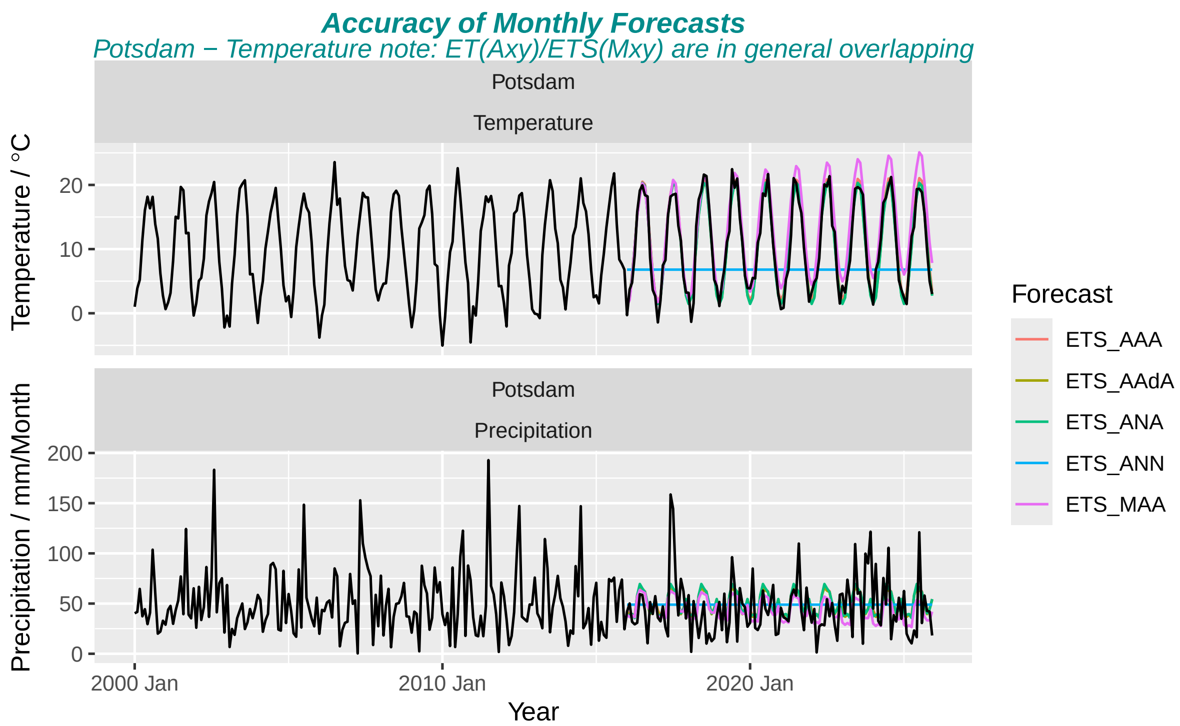
```
#>   2 Potsdam Precipitation ETS_AAA     13.6 0.886
#>   3 Potsdam Precipitation ETS_AMA     13.6 0.885
#>   4 Potsdam Precipitation ETS_ANA     14.0 0.868
#>   5 Potsdam Precipitation ETS_AAdA    14.1 0.865
#>   6 Potsdam Precipitation ETS_MMA     14.2 0.860
#>   7 Potsdam Precipitation ETS_MAdA    14.3 0.858
#>   8 Potsdam Precipitation ETS_MAA     15.1 0.818
#>   9 Potsdam Temperature   ETS_AMA     35.4 0.0257
#>  10 Potsdam Temperature   ETS_AAA     36.7 0.0180
#>  11 Potsdam Temperature   ETS_AAdA    36.8 0.0179
#>  12 Potsdam Temperature   ETS_ANA     38.2 0.0123
#>  13 Potsdam Temperature   ETS_MMA     51.8 0.000205
#>  14 Potsdam Temperature   ETS_MNA     60.0 0.0000129
#>  15 Potsdam Temperature   ETS_MAdA    83.0 0.00000000249
#>  16 Potsdam Temperature   ETS_MAA    219.   0
```

2.1.3 ETS Models - components of ETS(A,N,A), ETS(A,A,A), ETS(A,Ad,A), models



2.1.4 Forecast Accuracy with Training/Test Data

```
#> # A tibble: 10 x 7
#>    .model   City    Measure       .type       ME  RMSE   MAE
#>    <chr>    <chr>   <fct>         <chr>    <dbl> <dbl> <dbl>
#>  1 ETS_AAdA Potsdam Temperature   Test   0.00217  1.64  1.30
#>  2 ETS_ANA  Potsdam Temperature   Test   0.0872   1.65  1.31
#>  3 ETS_AAA  Potsdam Temperature   Test  -0.455    1.69  1.37
#>  4 ETS_MAA  Potsdam Temperature   Test  -2.25     3.10  2.58
#>  5 ETS_ANN  Potsdam Temperature   Test   4.03     7.94  6.52
#>  6 ETS_AAdA Potsdam Precipitation Test  -3.16    27.6  21.2
#>  7 ETS_ANA  Potsdam Precipitation Test  -3.68    27.7  21.5
#>  8 ETS_AAA  Potsdam Precipitation Test  -4.07    27.7  21.4
#>  9 ETS_MAA  Potsdam Precipitation Test   2.71    27.8  20.6
#> 10 ETS_ANN  Potsdam Precipitation Test  -3.01    28.6  21.8
```

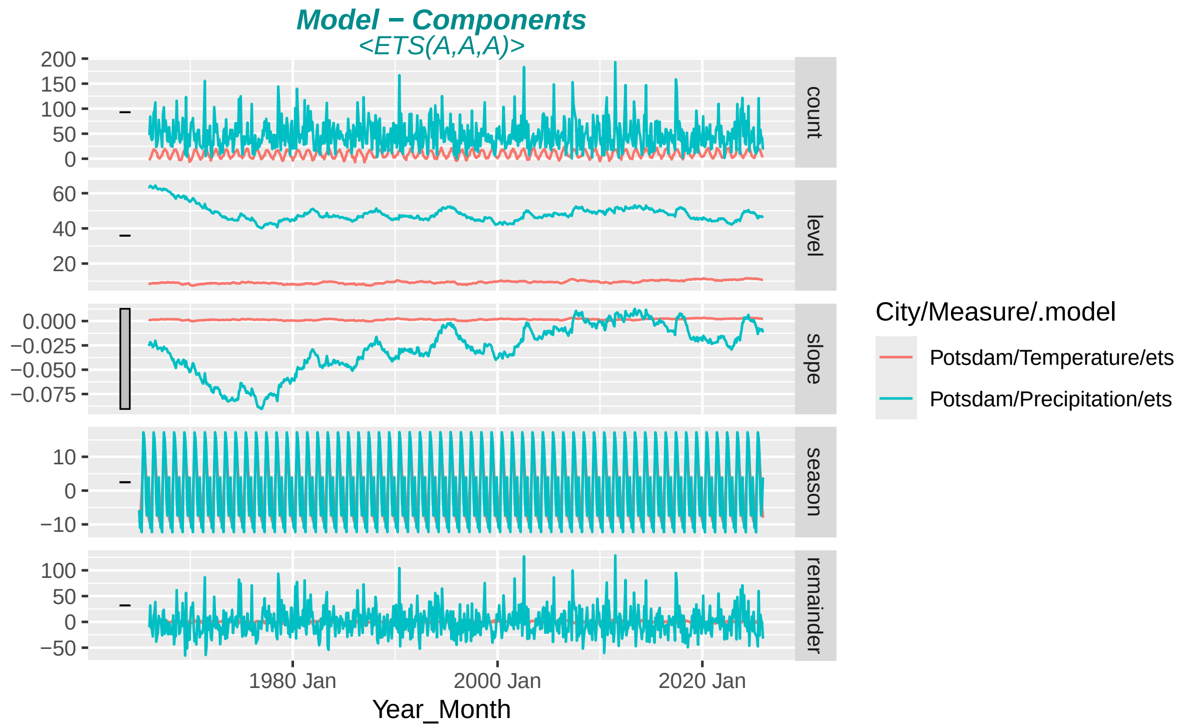


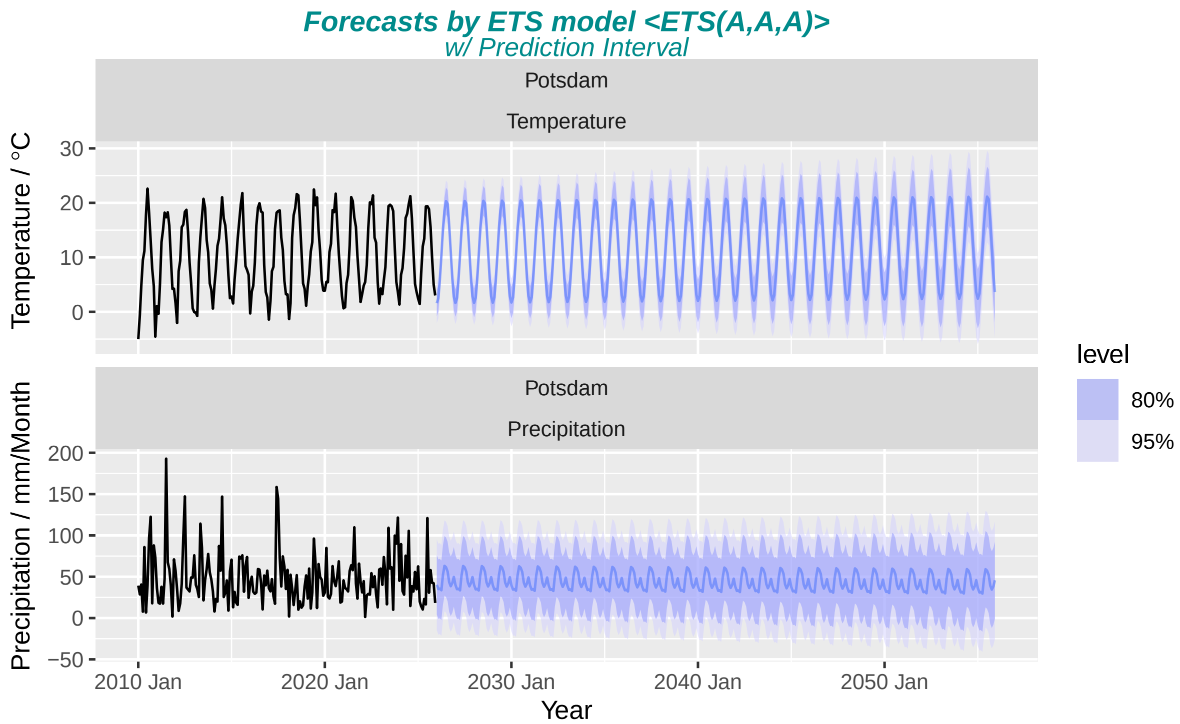
Accuracy of Monthly Forecasts
Potsdam − Temperature note: ET(Axy)/ETS(Mxy) are in general overlapping
Potsdam
Temperature
Temperature / °C
20
10
0
Potsdam
Precipitation
Precipitation / mm/Month
200
150
100
50
0
2000 Jan
2010 Jan
2020 Jan
Year
Forecast
ETS_AAA
ETS_AAdA
ETS_ANA
ETS_ANN
ETS_MAA

2.2 Forecasting with selected ETS model <ETS(A,A,A)>, <ETS(A,A,A)>

2.2.1 Forecast Plot of selected ETS model

```
#> Provide model coefficients by report(fit_model)
#> # A tibble: 2 x 11
#>   City    Measure    .model sigma2 log_lik   AIC  AICc   BIC   MSE  AMSE   MAE
#>   <chr>   <fct>      <chr>   <dbl>   <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
#> 1 Potsdam Temperatu~ ets      3.55  -2817. 5668. 5669. 5746.  3.47  3.56  1.47
#> 2 Potsdam Precipita~ ets    788.    -4762. 9557. 9558. 9635. 771.   771.   21.1
```

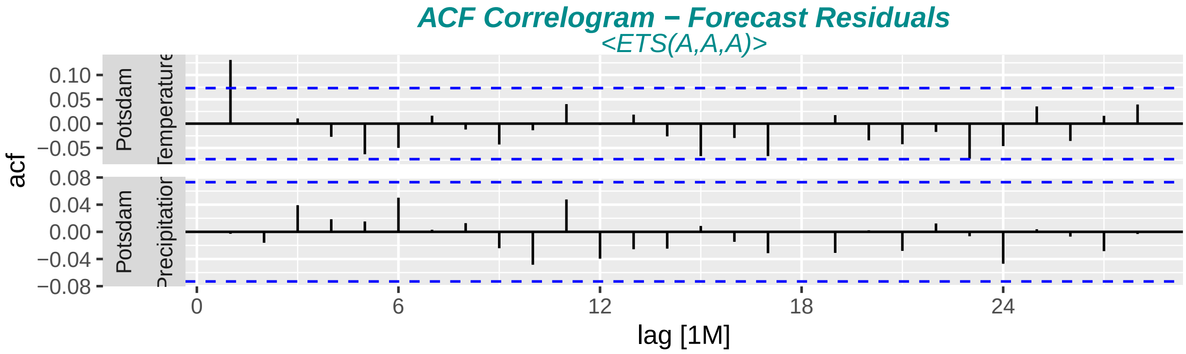


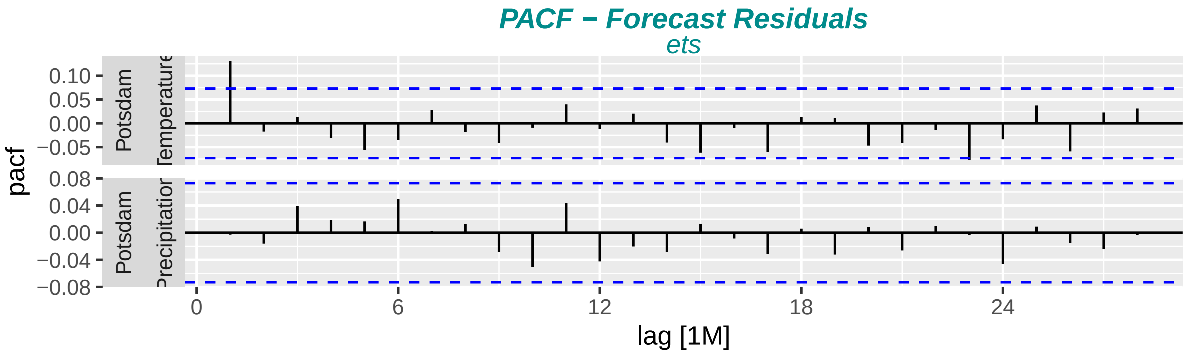


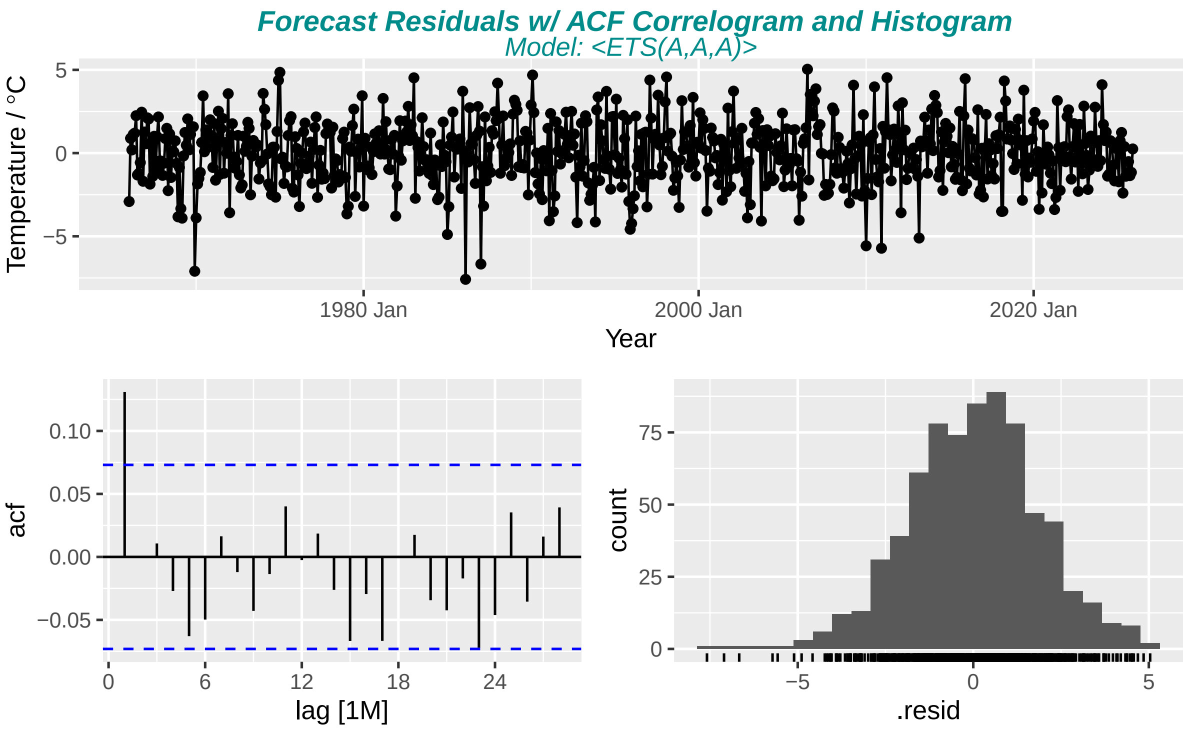
2.2.2 Residual Stationarity

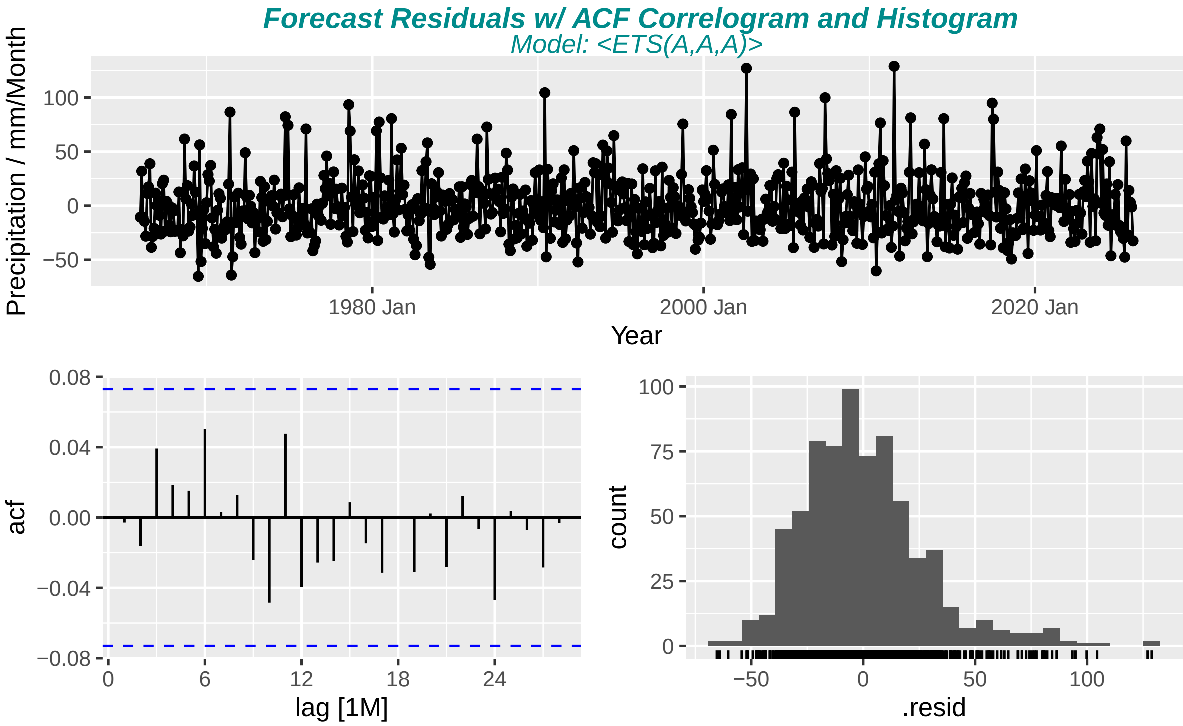
Required checks to be ready for forecasting:

- ACF Forecast Residual: all spikes are within the significance limits, so the residuals appear to be white noise
- The Ljung-Box test also shows that the residuals have no remaining autocorrelations
- Forecast Residuals are more or less normally distributed with roughly centred on zero



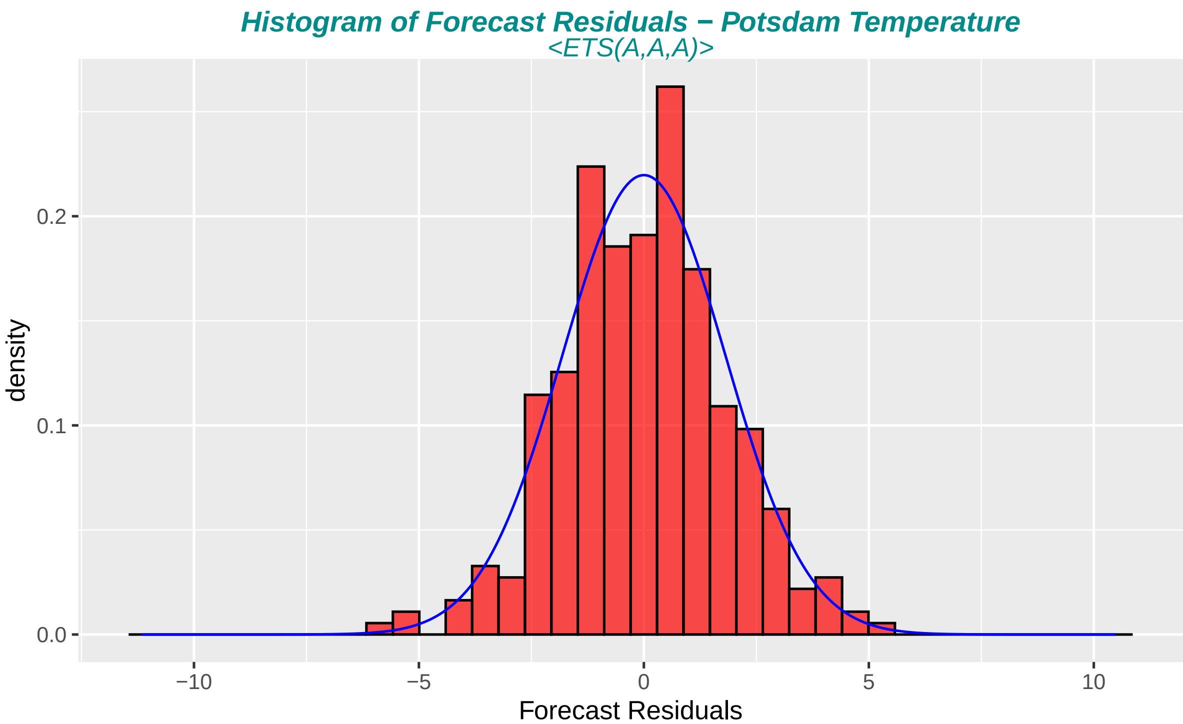


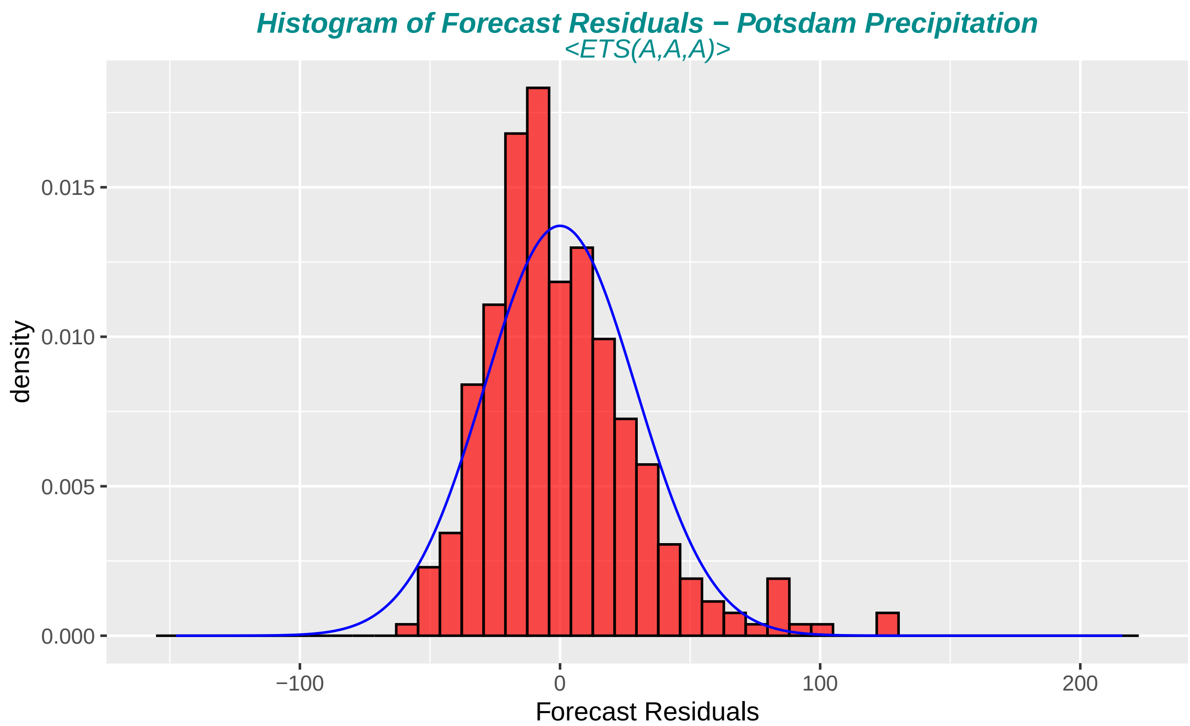




2.2.3 Ljung-Box Test and Histogram of forecast residuals with overlaid normal curve

```
#> Null Hypothesis of independence/white noise for residuals - for p < 0.05: reject H_0
#> # A tibble: 2 x 5
#>   City    Measure       .model lb_stat lb_pvalue
#>   <chr>   <fct>         <chr>    <dbl>     <dbl>
#> 1 Potsdam Temperature   ets       21.1     0.635
#> 2 Potsdam Precipitation ets       19.0     0.752
```




Histogram of Forecast Residuals − Potsdam Precipitation
<ETS(A,A,A)>
density
0.015
0.010
0.005
0.000
−100
0
100
200
Forecast Residuals

3 ARIMA Forecasting Models - AutoRegressive-Integrated Moving Average

Exponential smoothing and ARIMA (AutoRegressive-Integrated Moving Average)models are the two most widely used approaches to time series forecasting, and provide complementary approaches to the problem.

While exponential smoothing models are based on a description of the trend and seasonality in the data, ARIMA models aim to describe the autocorrelations in the data.

3.1 Seasonal ARIMA models

Non-seasonal ARIMA models are generally denoted ARIMA(p,d,q) where parameters p, d, and q are non-negative integers, * p is the order (number of time lags) of the autoregressive model * d is the degree of differencing (number of times the data have had past values subtracted) * q is the order of the moving-average model of past forecast errors .

The value of d has an effect on the prediction intervals — the higher the value of d, the more rapidly the prediction intervals increase in size. For d=0, the point forecasts are equal to the mean of the data and the long-term forecast standard deviation will go to the standard deviation of the historical data, so the prediction intervals will all be essentially the same.

Seasonal ARIMA models are usually denoted ARIMA(p,d,q)(P,D,Q)m, where m refers to the number of periods in each season, and the uppercase P,D,Q refer to the autoregressive, differencing, and moving average terms for the seasonal part of the ARIMA model.

$ARIMA(pdq)(PDQ)$ **model with automatically selected** $(pdq)(PDQ)$ **values**

Fit of different pre-defined $ARIMA(pdq)(PDQ)$ **models**

```
#> Model Selection by Information Criterion - lowest AIC, AICc, BIC
#>  choose p, q parameter accordingly - but only for same d, D values
#> # A tibble: 8 x 8
#>   City    Measure     .model          sigma2 log_lik   AIC  AICc   BIC
#>   <chr>   <fct>       <chr>            <dbl>   <dbl> <dbl> <dbl> <dbl>
#> 1 Potsdam Temperature arima_111_011     3.42  -1464. 2936. 2936. 2954.
#> 2 Potsdam Temperature arima_211_011     3.42  -1463. 2937. 2937. 2960.
#> 3 Potsdam Temperature arima_111_012     3.42  -1464. 2937. 2938. 2960.
#> 4 Potsdam Temperature arima_012_011     3.43  -1465. 2938. 2938. 2956.
#> 5 Potsdam Temperature arima_012_112     3.43  -1464. 2940. 2941. 2968.
#> 6 Potsdam Temperature arima_100_210     4.71  -1554. 3116. 3116. 3134.
#> 7 Potsdam Temperature arima_200_011     5.13  -1584. 3175. 3175. 3193.
#> 8 Potsdam Temperature arima_100_110_c   5.14  -1583. 3177. 3177. 3200.
#> # A tibble: 8 x 8
#>   City    Measure       .model          sigma2 log_lik   AIC  AICc   BIC
#>   <chr>   <fct>         <chr>            <dbl>   <dbl> <dbl> <dbl> <dbl>
#> 1 Potsdam Precipitation arima_111_012     789.  -3379. 6768. 6768. 6791.
#> 2 Potsdam Precipitation arima_012_112     789.  -3378. 6768. 6768. 6795.
#> 3 Potsdam Precipitation arima_012_011     790.  -3380. 6768. 6768. 6786.
#> 4 Potsdam Precipitation arima_111_011     790.  -3380. 6768. 6768. 6786.
#> 5 Potsdam Precipitation arima_211_011     791.  -3380. 6770. 6770. 6793.
#> 6 Potsdam Precipitation arima_001_002     848.  -3447. 6904. 6904. 6927.
#> 7 Potsdam Precipitation arima_200_011    1189.  -3511. 7031. 7031. 7049.
#> 8 Potsdam Precipitation arima_100_110_c  1190.  -3511. 7032. 7033. 7055.
```

Good models are obtained by minimising the AIC, AICc or BIC (see glance(fit_arima) output). The preference is to use the AICc to selec p and q.

These information criteria tend not to be good guides to selecting the appropriate order of differencing (d) of a model, but only for selecting the values of p and q. This is because the differencing changes the data on which the likelihood is computed, making the AIC values between models with different orders of differencing not comparable.

3.1.1 Residual Accuracy with one-step-ahead fitted residuals - check RMSE, MAE

Residual accuracy can be computed directly from models as the one-step-ahead fitted residuals are available. Select forecast models that minimises for lowest

- MAE (Mean absolute error, will lead to forecasts of the median) and
- RMSE (Root mean squared error, lead to forecasts of the mean)

```
#> # A tibble: 8 x 7
#>   City    Measure     .model          .type         ME  RMSE   MAE
#>   <chr>   <fct>       <chr>           <chr>      <dbl> <dbl> <dbl>
#> 1 Potsdam Temperature arima_211_011   Training 0.0901   1.83  1.41
#> 2 Potsdam Temperature arima_111_011   Training 0.0883   1.83  1.41
#> 3 Potsdam Temperature arima_111_012   Training 0.0886   1.83  1.41
#> 4 Potsdam Temperature arima_012_112   Training 0.0825   1.83  1.41
#> 5 Potsdam Temperature arima_012_011   Training 0.0824   1.83  1.41
#> 6 Potsdam Temperature arima_100_210   Training 0.0545   2.15  1.67
#> 7 Potsdam Temperature arima_100_110_c Training 0.00234  2.24  1.73
#> 8 Potsdam Temperature arima_200_110_c Training 0.00234  2.24  1.73
#> # A tibble: 8 x 7
#>   City    Measure       .model          .type          ME  RMSE   MAE
#>   <chr>   <fct>         <chr>           <chr>       <dbl> <dbl> <dbl>
#> 1 Potsdam Precipitation arima_012_112   Training  1.40     27.7  20.6
#> 2 Potsdam Precipitation arima_111_012   Training  1.41     27.8  20.6
#> 3 Potsdam Precipitation arima_211_011   Training  1.37     27.8  20.7
#> 4 Potsdam Precipitation arima_111_011   Training  1.36     27.8  20.7
#> 5 Potsdam Precipitation arima_012_011   Training  1.36     27.8  20.7
#> 6 Potsdam Precipitation arima_001_002   Training -0.0192   29.0  22.1
#> 7 Potsdam Precipitation arima_100_110_c Training -0.00954  34.1  25.9
#> 8 Potsdam Precipitation arima_200_110_c Training -0.00954  34.1  25.9
```

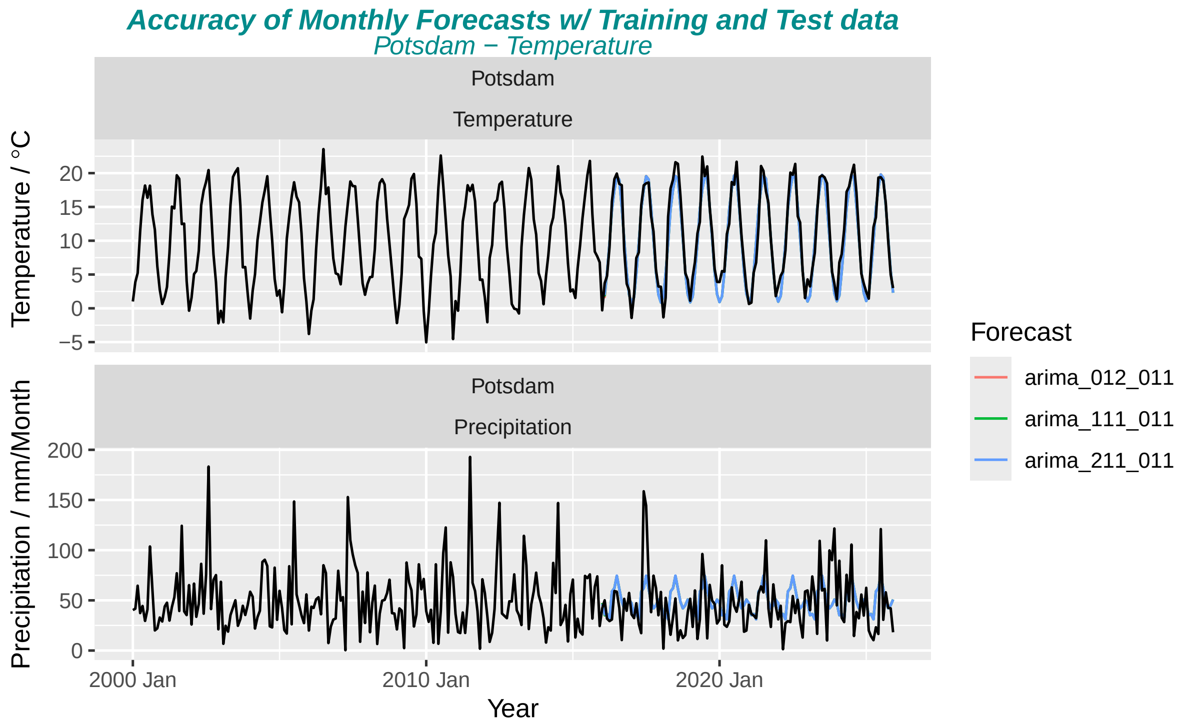
3.1.2 Ljung-Box Test - independence/white noise of the forecasts residuals

```
#> Null Hypothesis of independence/white noise for residuals - for p < 0.05: reject H_0
#> # A tibble: 8 x 5
#>   City    Measure     .model          lb_stat lb_pvalue
#>   <chr>   <fct>       <chr>             <dbl>     <dbl>
#> 1 Potsdam Temperature arima_211_011      18.7  6.02e- 1
#> 2 Potsdam Temperature arima_111_011      19.1  5.77e- 1
#> 3 Potsdam Temperature arima_111_012      19.3  5.66e- 1
#> 4 Potsdam Temperature arima_012_011      20.8  4.73e- 1
#> 5 Potsdam Temperature arima_012_112      20.8  4.73e- 1
#> 6 Potsdam Temperature arima_100_210      49.5  4.23e- 4
#> 7 Potsdam Temperature arima_200_011     106.   2.45e-13
#> 8 Potsdam Temperature arima_100_110_c   106.   2.37e-13
#> # A tibble: 8 x 5
#>   City    Measure       .model        lb_stat lb_pvalue
#>   <chr>   <fct>         <chr>           <dbl>     <dbl>
#> 1 Potsdam Precipitation arima_012_112    13.1     0.907
#> 2 Potsdam Precipitation arima_111_012    15.0     0.821
#> 3 Potsdam Precipitation arima_111_011    16.6     0.737
#> 4 Potsdam Precipitation arima_012_011    16.6     0.737
#> 5 Potsdam Precipitation arima_211_011    16.6     0.735
#> 6 Potsdam Precipitation arima_001_002    29.2     0.110
#> 7 Potsdam Precipitation arima_010_110   346.      0
#> 8 Potsdam Precipitation arima_012_010   195.      0
```

3.1.3 Forecast Accuracy with Training/Test Data

```
#> # A tibble: 6 x 7
```

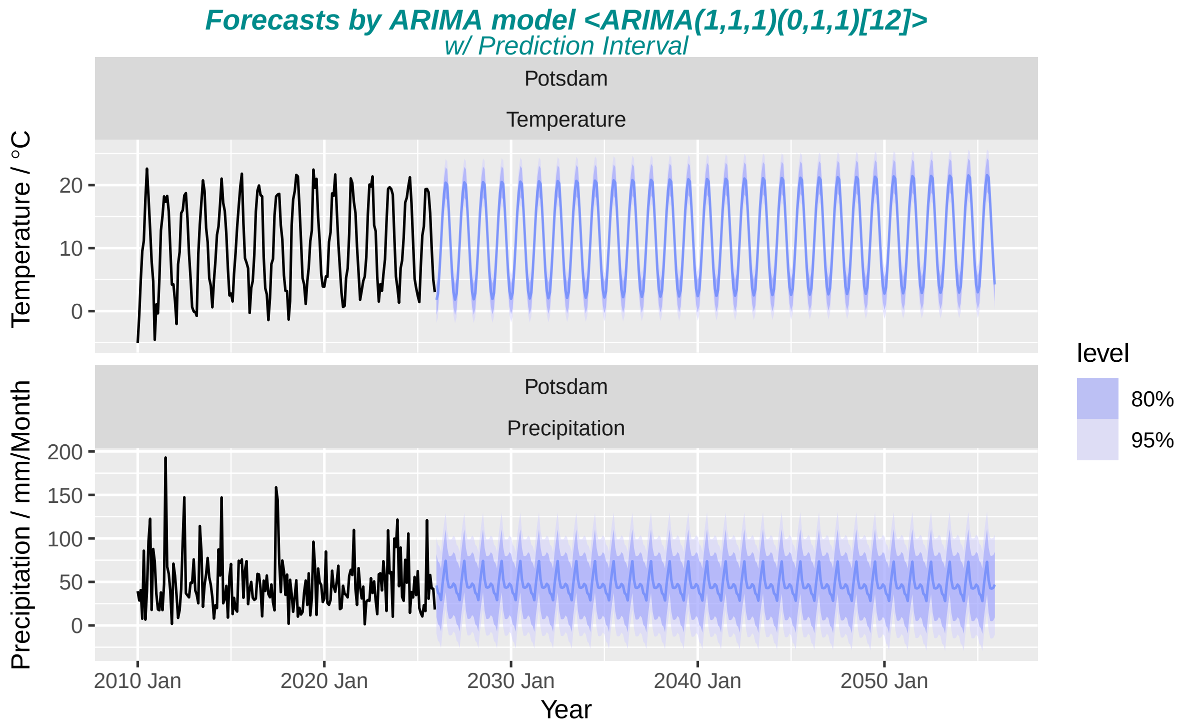
```
#>    .model          City    Measure       .type     ME  RMSE   MAE
#>    <chr>           <chr>   <fct>         <chr>  <dbl> <dbl> <dbl>
#> 1 arima_012_011 Potsdam Temperature   Test   0.636  1.77  1.40
#> 2 arima_211_011 Potsdam Temperature   Test   0.654  1.77  1.41
#> 3 arima_111_011 Potsdam Temperature   Test   0.661  1.78  1.41
#> 4 arima_211_011 Potsdam Precipitation Test  -3.53  27.3  20.9
#> 5 arima_111_011 Potsdam Precipitation Test  -3.52  27.3  20.9
#> 6 arima_012_011 Potsdam Precipitation Test  -3.52  27.3  20.9
```



3.2 Temperature, Precipitation - Forecasting with selected ARIMA model <ARIMA(1,1,1)(0,1,1)[12]>, <ARIMA(1,1,1)(0,1,1)[12]>

3.2.1 Forecast Plot of selected ARIMA model

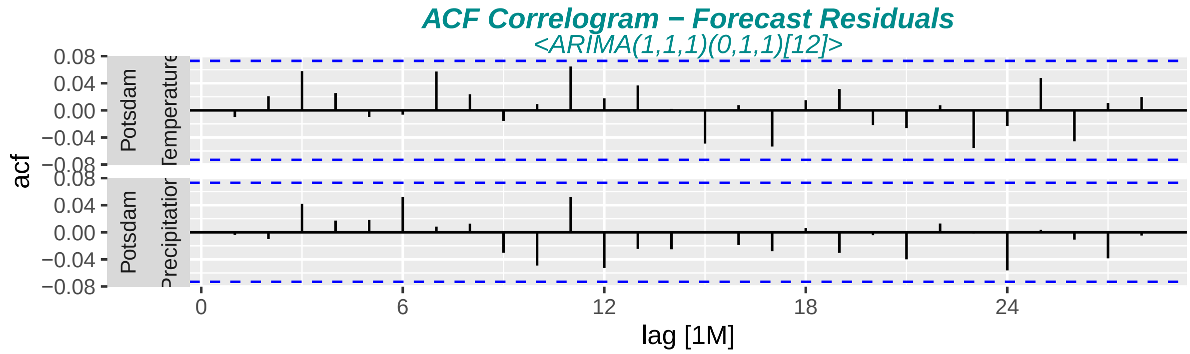
```
#> Provide model coefficients by report(fit_model)
#> # A tibble: 2 x 10
#>   City    Measure      .model sigma2 log_lik   AIC  AICc   BIC ar_roots ma_roots
#>   <chr>   <fct>        <chr>   <dbl>   <dbl> <dbl> <dbl> <dbl> <list>   <list>
#> 1 Potsdam Temperature  arima    3.42  -1464. 2936. 2936. 2954. <cpl>    <cpl>
#> 2 Potsdam Precipitati~ arima  790.    -3380. 6768. 6768. 6786. <cpl>    <cpl>
```

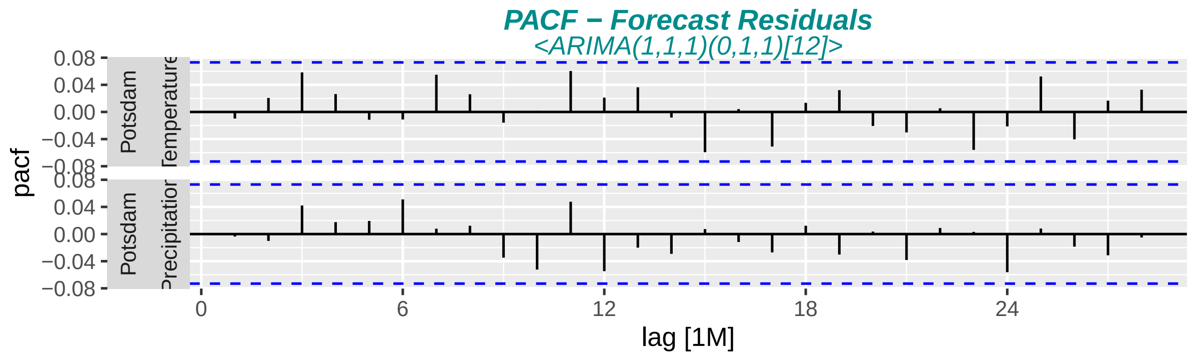


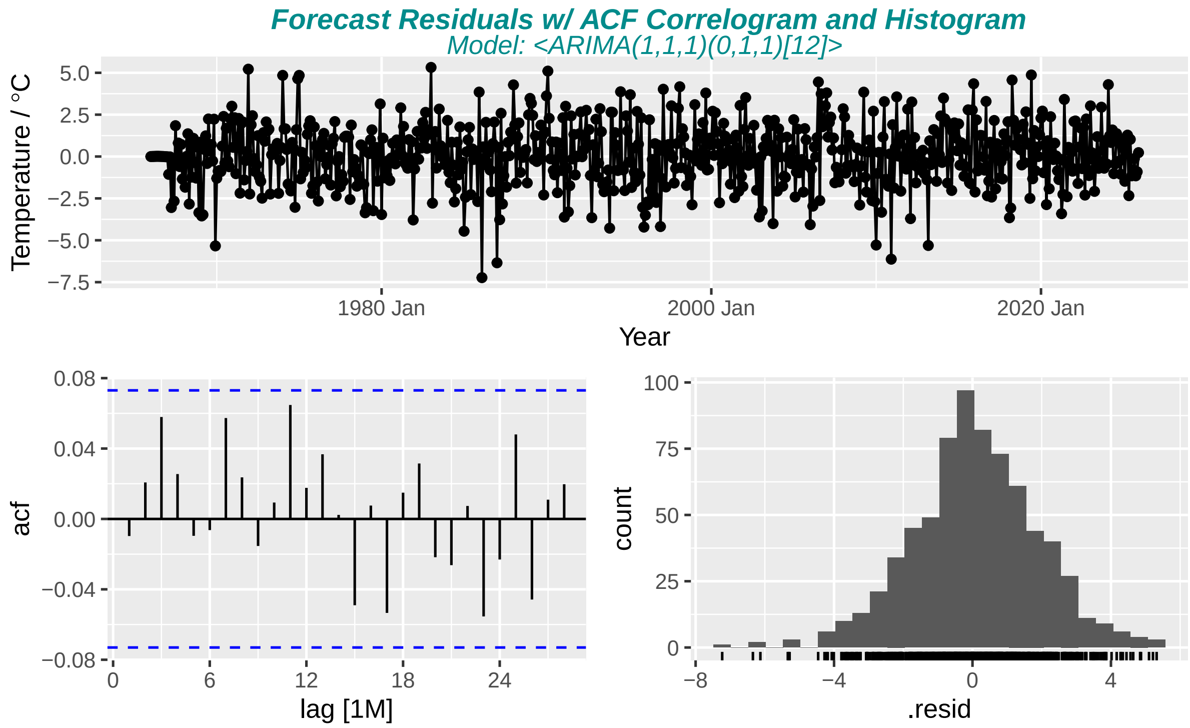
3.2.2 Residual Stationarity

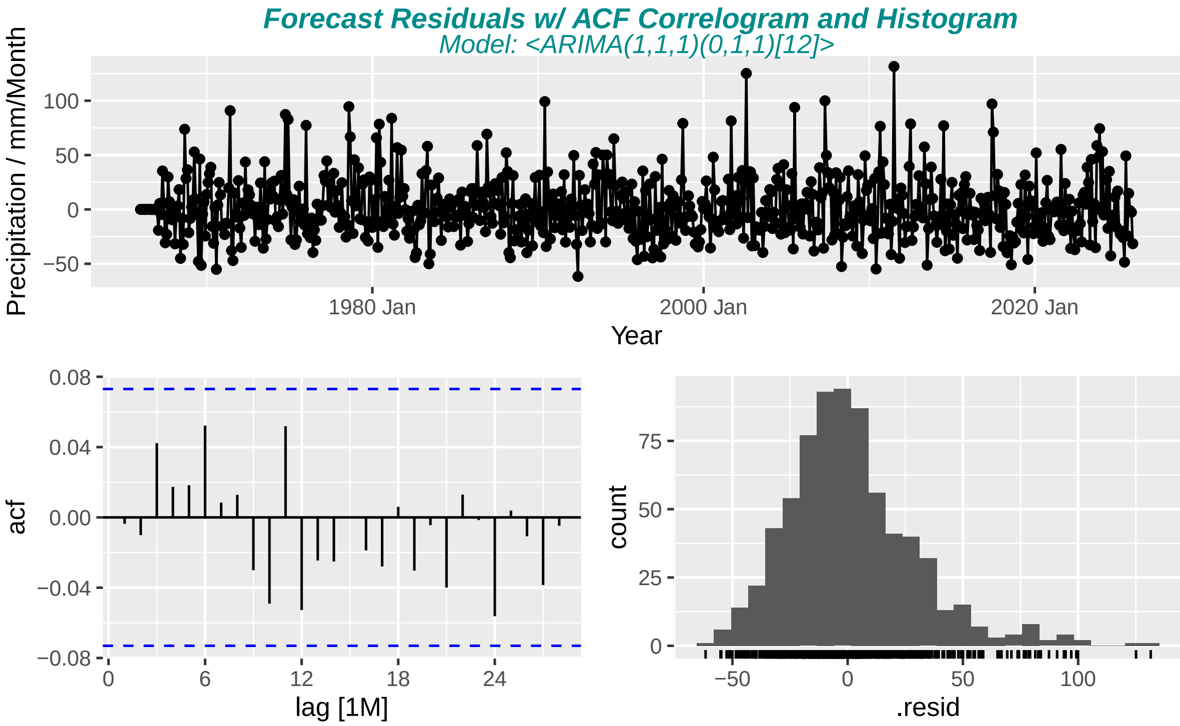
Required checks to be ready for forecasting:

- ACF Forecast Residual: all spikes are within the significance limits, so the residuals appear to be white noise
- The Ljung-Box test also shows that the residuals have no remaining autocorrelations
- Forecast Residuals are more or less normally distributed with roughly centred on zero


ACF Correlogram − Forecast Residuals
<ARIMA(1,1,1)(0,1,1)[12]>
acf
Potsdam
Temperature
Potsdam
Precipitation
0.08
0.04
0.00
−0.04
−0.08
0
6
12
18
24
lag [1M]

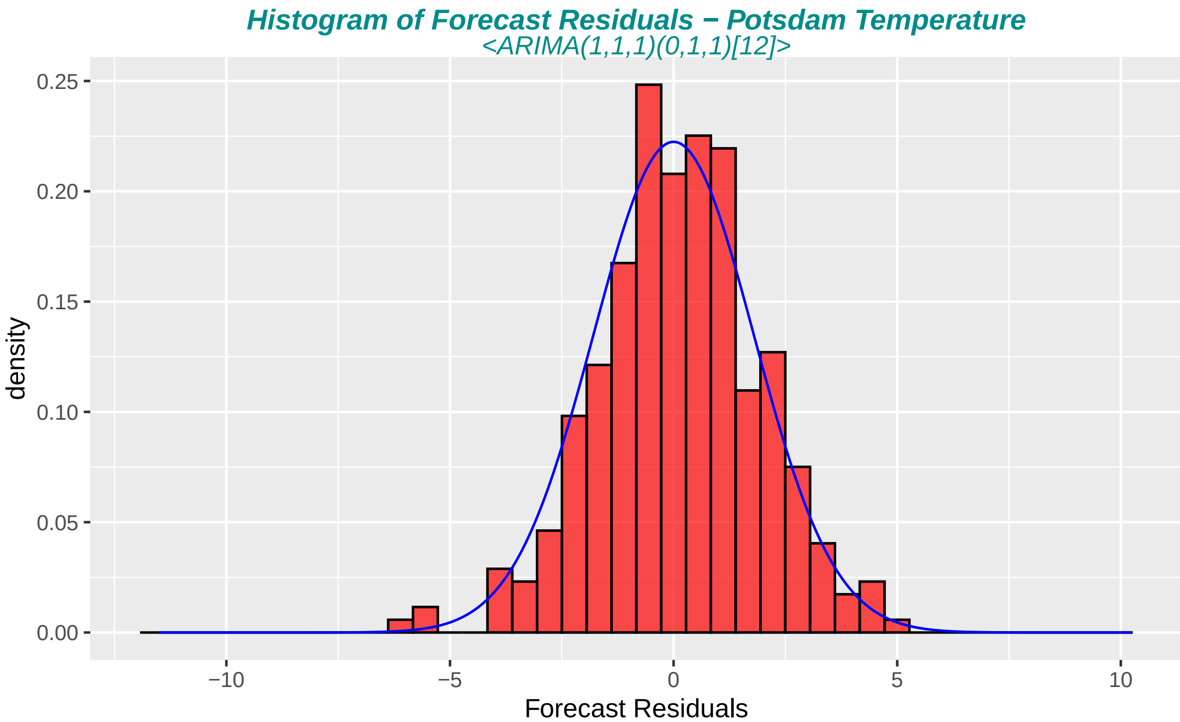

PACF − Forecast Residuals
<ARIMA(1,1,1)(0,1,1)[12]>
pacf
Potsdam
Temperature
Potsdam
Precipitation
0.08
0.04
0.00
−0.04
−0.08
0
6
12
18
24
lag [1M]

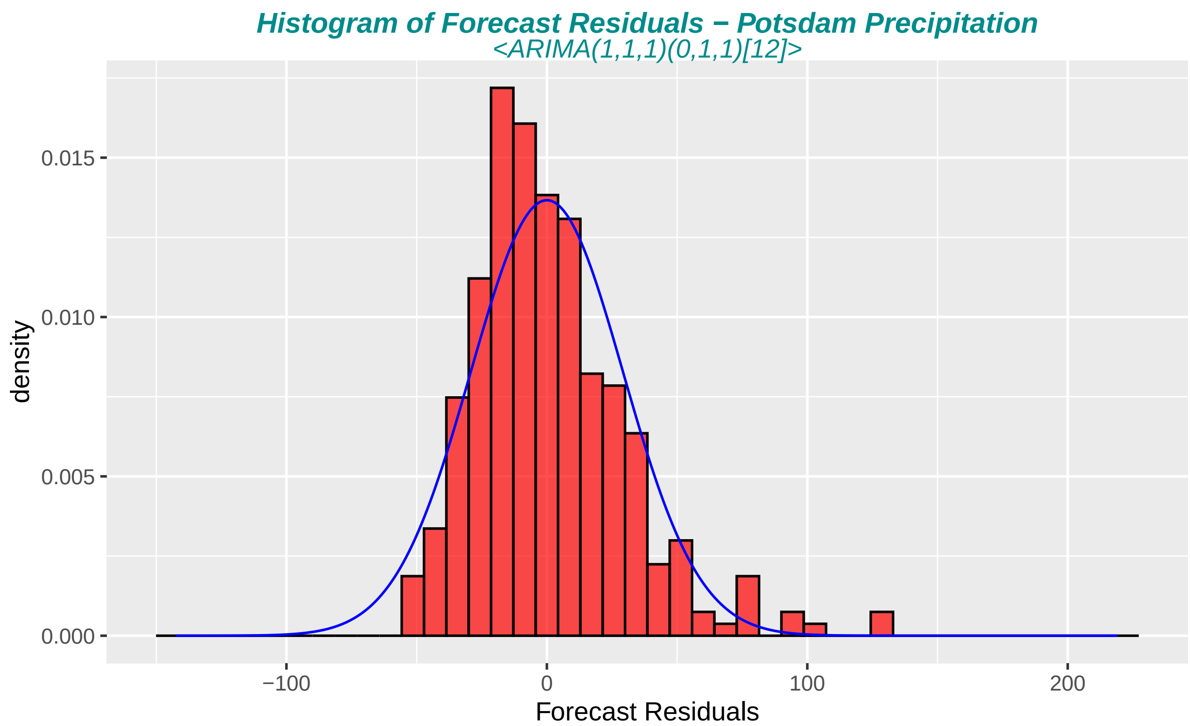

Forecast Residuals w/ ACF Correlogram and Histogram
Model: <ARIMA(1,1,1)(0,1,1)[12]>
Temperature / °C
5.0
2.5
0.0
−2.5
−5.0
−7.5
1980 Jan
2000 Jan
2020 Jan
Year
acf
0.08
0.04
0.00
−0.04
−0.08
0
6
12
18
24
lag [1M]
count
100
75
50
25
0
−8
−4
0
4
.resid



3.2.3 Ljung-Box Test and Histogram of forecast residuals with overlaid normal curve

```
#> Null Hypothesis of independence/white noise for residuals - for p < 0.05: reject H_0
#> # A tibble: 2 x 5
#>   City    Measure       .model lb_stat lb_pvalue
#>   <chr>   <fct>         <chr>    <dbl>     <dbl>
#> 1 Potsdam Temperature   arima     18.1     0.644
#> 2 Potsdam Precipitation arima     19.9     0.527
```




Histogram of Forecast Residuals − Potsdam Precipitation
<ARIMA(1,1,1)(0,1,1)[12]>
density
0.015
0.010
0.005
0.000
−100
0
100
200
Forecast Residuals

4 ARIMA vs ETS

In particular, all ETS models are non-stationary, while some ARIMA models are stationary.

The ETS models with seasonality or non-damped trend or both have two unit roots (i.e., they need two levels of differencing to make them stationary). All other ETS models have one unit root (they need one level of differencing to make them stationary).

We compare for the chosen ETS rsp. ARIMA model the RMSE / MAE values. Lower values indicate a more accurate model based on the test set RMSE, . . . , MASE.

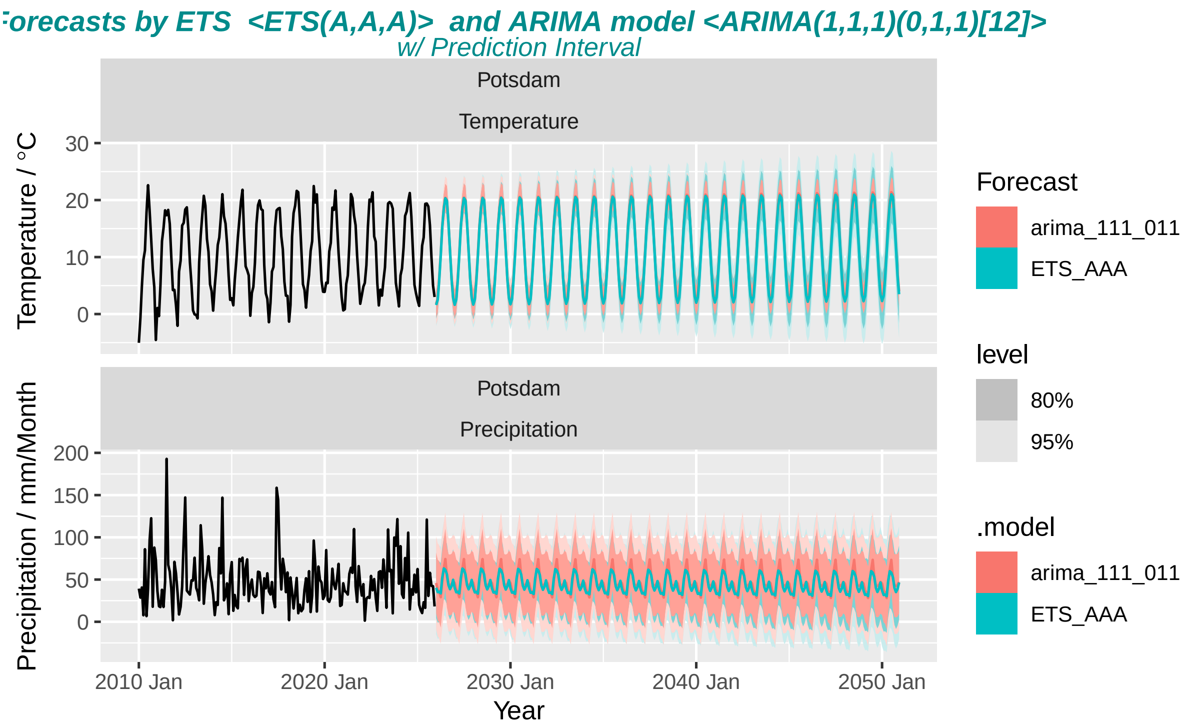
- Residual Accuracy with one-step-ahead fitted residuals
- Forecast Accuracy with Training/Test Data

Note: a good fit to training data is never an indication that the model will forecast well. Therefore the values of the Forecast Accuracy are the more relevant one.

4.0.1 Comparing Residual and Forecast Accuracy of selected ETS and ARIMA model

```
#> # A tibble: 8 x 9
#>   City    Measure       .model        .type     RMSE   MAE  MAPE  MASE RMSSE
#>   <chr>   <fct>         <chr>         <chr>    <dbl> <dbl> <dbl> <dbl> <dbl>
#> 1 Potsdam Temperature   ETS_AAA       Test      1.69  1.37  34.2 0.681 0.643
#> 2 Potsdam Temperature   arima_111_011 Test      1.78  1.41  29.9 0.701 0.676
#> 3 Potsdam Temperature   arima         Training  1.83  1.41  62.8 0.707 0.704
#> 4 Potsdam Temperature   ets           Training  1.86  1.47  63.0 0.737 0.717
#> 5 Potsdam Precipitation arima_111_011 Test     27.3  20.9  106.  0.688 0.686
#> 6 Potsdam Precipitation ETS_AAA       Test     27.7  21.4  112.  0.701 0.698
#> 7 Potsdam Precipitation ets           Training 27.8  21.1  122.  0.692 0.701
#> 8 Potsdam Precipitation arima         Training 27.8  20.7  119.  0.680 0.701
```

4.0.2 Forecast Plot of selected ETS and ARIMA model



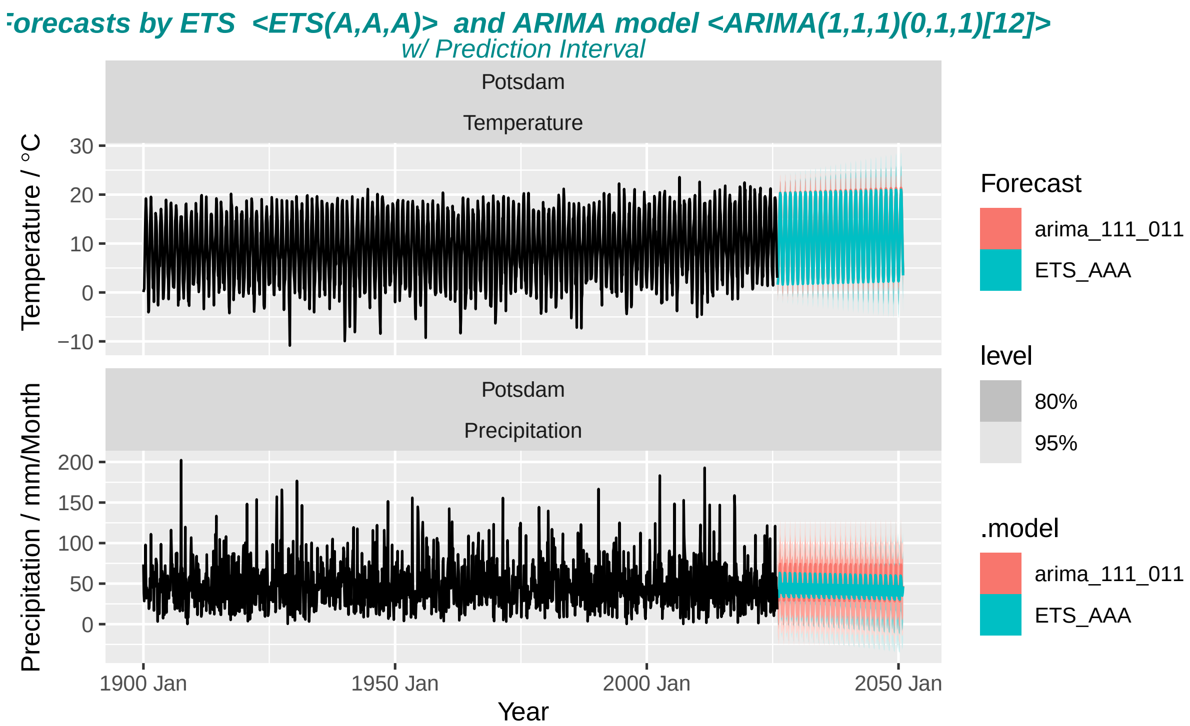

Forecasts by ETS <ETS(A,A,A)> and ARIMA model <ARIMA(1,1,1)(0,1,1)[12]>
w/ Prediction Interval
Potsdam
Temperature
Temperature / °C
30
20
10
0
−10
Potsdam
Precipitation
Precipitation / mm/Month
200
150
100
50
0
1900 Jan
1950 Jan
2000 Jan
2050 Jan
Year
Forecast
arima_111_011
ETS_AAA
level
80%
95%
.model
arima_111_011
ETS_AAA

Table 1: Mean values for the given time periods; Units: Temperature (degree C), Precipitation (mm/Month), CO2 (ppm)

Period_Time	Temperature	Precipitation
1871-1900	8.3	48.3
1901-1930	8.4	49.2
1931-1960	8.7	49.1
1961-1990	8.7	48.8
1991-2020	9.8	48.1
2021-2025	10.9	47.9

Table 2: Mean Yearly ARIMA and ETS Forecast values (next 25 years); Units: Temperature (degree C), Precipitation (mm/Month), CO2 (ppm)

City	Measure	Year	ETS_AAA	arima_111_011
Potsdam	Temperature	2026	10.82	11.00
Potsdam	Temperature	2030	10.94	11.16
Potsdam	Temperature	2035	11.08	11.37
Potsdam	Temperature	2040	11.22	11.57
Potsdam	Temperature	2045	11.36	11.78
Potsdam	Temperature	2050	11.51	11.98
Potsdam	Precipitation	2026	45.96	47.20
Potsdam	Precipitation	2030	45.39	47.04
Potsdam	Precipitation	2035	44.69	46.83
Potsdam	Precipitation	2040	43.98	46.62
Potsdam	Precipitation	2045	43.28	46.42
Potsdam	Precipitation	2050	42.58	46.21

Table 3: Forecast increase/decrease over the next 25 years; Units: Temperature (degree C), Precipitation (mm/Month), CO2 (ppm)

Measure	Year.x	Year.y	ETS.x	ARIMA.x	ETS.y	ARIMA.y	Delta_ETS	Delta_ARIMA
Temperature	2026	2050	10.82	11.0	11.51	11.98	0.68	0.99
Precipitation	2026	2050	45.96	47.2	42.58	46.21	-3.38	-0.99

Table 4: Forecast increase/decrease over the next 25 years; Units: Temperature (degree C), Precipitation (mm/Month), CO2 (ppm)

Measure	Month	Year.x	Year.y	ETS.x	ARIMA.x	ETS.y	ARIMA.y	Delta_ETS	Delta_ARIMA
Temperature	Jan	2026	2050	1.59	1.79	2.27	2.78	0.68	0.99
Temperature	Feb	2026	2050	2.53	2.74	3.21	3.73	0.68	0.99
Temperature	Mrz	2026	2050	5.76	6.07	6.44	7.06	0.68	0.99
Temperature	Apr	2026	2050	10.23	10.54	10.91	11.52	0.68	0.99
Temperature	Mai	2026	2050	15.20	15.43	15.88	16.42	0.68	0.99
Temperature	Jun	2026	2050	18.52	18.70	19.20	19.69	0.68	0.99
Temperature	Jul	2026	2050	20.32	20.39	21.00	21.38	0.68	0.99
Temperature	Aug	2026	2050	19.86	20.02	20.54	21.01	0.68	0.99
Temperature	Sep	2026	2050	15.86	15.92	16.54	16.91	0.68	0.99
Temperature	Okt	2026	2050	11.12	11.16	11.80	12.15	0.68	0.99
Temperature	Nov	2026	2050	6.11	6.17	6.79	7.15	0.68	0.99
Temperature	Dez	2026	2050	2.80	3.04	3.48	4.02	0.68	0.99

Measure	Month	Year.x	Year.y	ETS.x	ARIMA.x	ETS.y	ARIMA.y	Delta_ETS	Delta_ARIMA
Precipitation	Jan	2026	2050	40.29	45.91	36.91	45.05	-3.38	-0.86
Precipitation	Feb	2026	2050	35.01	37.60	31.63	36.60	-3.38	-1.00
Precipitation	Mrz	2026	2050	35.64	35.75	32.26	34.75	-3.38	-1.00
Precipitation	Apr	2026	2050	33.63	29.15	30.26	28.15	-3.38	-1.00
Precipitation	Mai	2026	2050	51.92	47.75	48.54	46.75	-3.38	-1.00
Precipitation	Jun	2026	2050	63.19	62.63	59.81	61.63	-3.38	-1.00
Precipitation	Jul	2026	2050	61.45	74.30	58.07	73.30	-3.38	-1.00
Precipitation	Aug	2026	2050	56.29	53.21	52.91	52.21	-3.38	-1.00
Precipitation	Sep	2026	2050	43.40	43.98	40.02	42.98	-3.38	-1.00
Precipitation	Okt	2026	2050	38.51	43.52	35.13	42.52	-3.38	-1.00
Precipitation	Nov	2026	2050	42.36	44.46	38.98	43.46	-3.38	-1.00
Precipitation	Dez	2026	2050	49.79	48.11	46.41	47.11	-3.38	-1.00

5 Yearly Data Forecasts with ARIMA and ETS

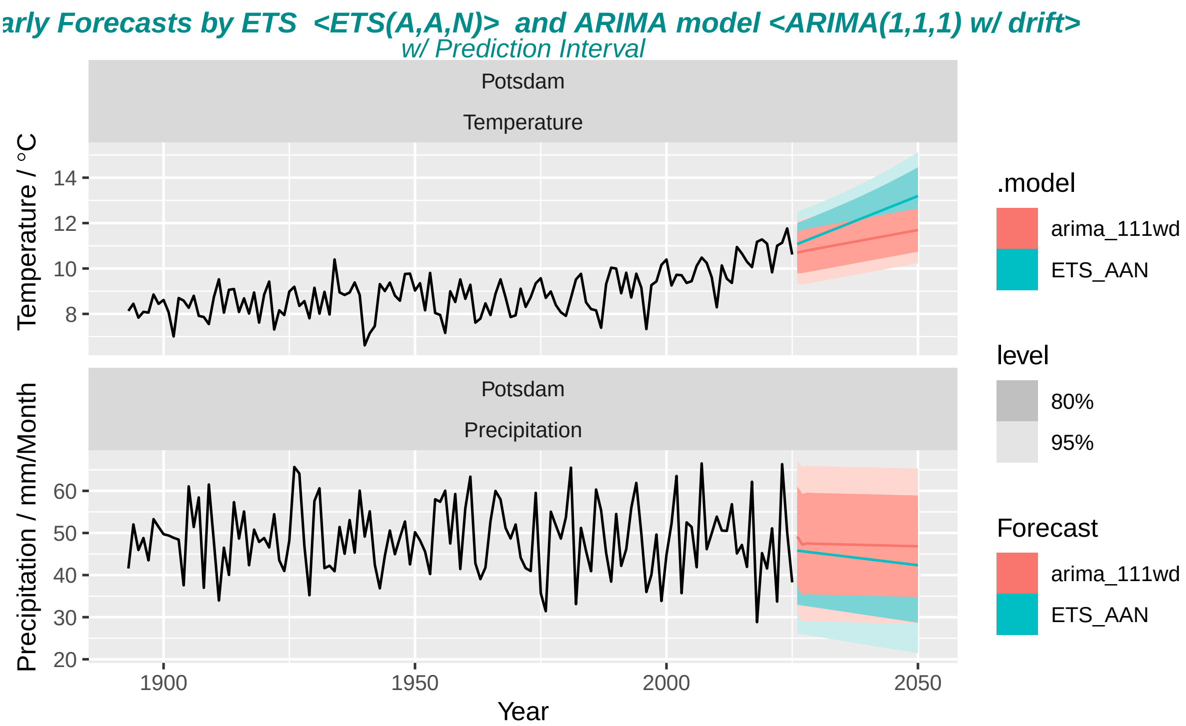
For yearly data the seasonal monthly data are replaced by the yearly average data. Therefore the seasonal component of the ETS and ARIMA model are to be taken out.

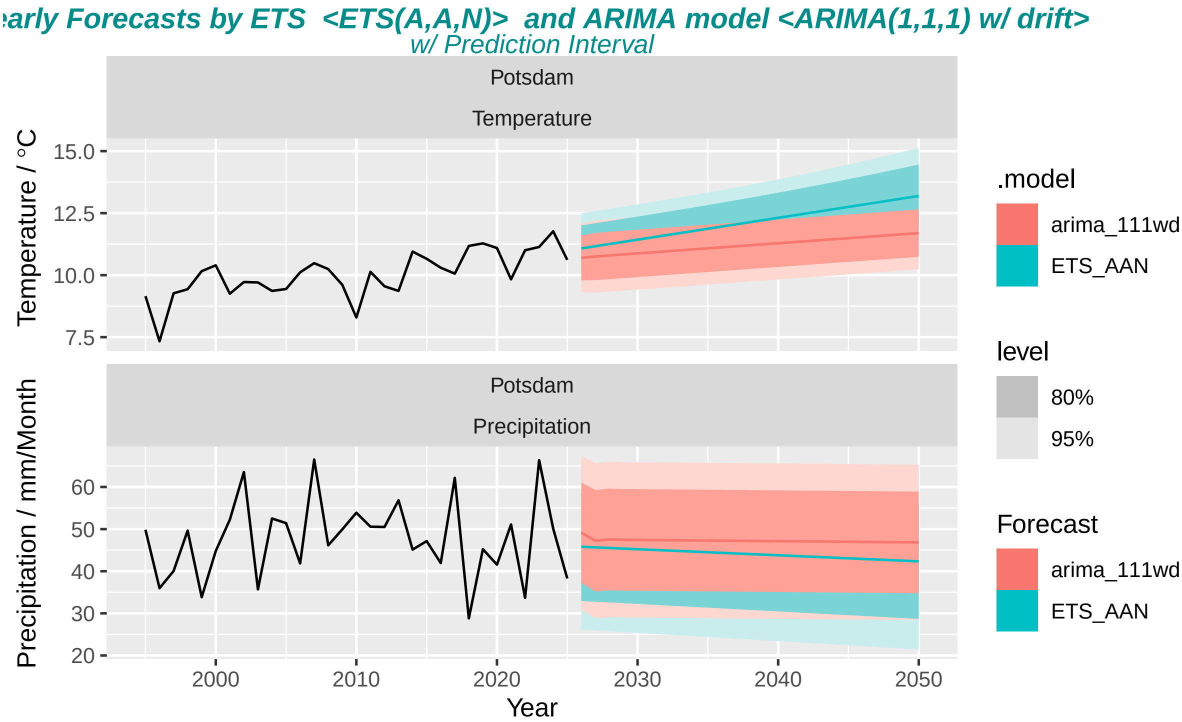
The ETS model $< ETS(A, A, N) >$ with seasonal term change "A" -> "N" is chosen. For ARIMA models the seasonal term (P,D,Q)m has to be taken out and an optimal ARIMA(p,1,q) with one differencing (d=1) is selected. However, for Mauna Loa two times differncing had to be selected $CO_2 <ARIMA(0,2,1) w/ poly>. For Temperature and Precipitation the same model as for monthly data can be taken by leaving out the seasonal term $< ARIMA(0, 1, 2)w/drift >$.

5.0.1 Comparing Residual and Forecast Accuracy of selected ETS and ARIMA model

5.0.2 Forecast Plot of selected ETS and ARIMA model

```
#> selected: ETS_AAN and arima_111 w/ drift
```






```
#> # A tibble: 4 x 13
#>   City    Measure  .model  sigma2 log_lik   AIC  AICc   BIC    MSE   AMSE    MAE
#>   <chr>   <fct>    <chr>    <dbl>   <dbl> <dbl> <dbl> <dbl>  <dbl>  <dbl>  <dbl>
#> 1 Potsdam Tempera~ arima~   0.493   -63.0  134.  135.  142. NA     NA     NA
#> 2 Potsdam Tempera~ ETS_A~   0.523  -101.   213.  214.  223.  0.489  0.493  0.571
#> 3 Potsdam Precipi~ arima~  85.0    -215.   439.  440.  447. NA     NA     NA
#> 4 Potsdam Precipi~ ETS_A~ 101.     -259.   528.  529.  539. 93.9   95.0    7.81
#> # i 2 more variables: ar_roots <list>, ma_roots <list>
```

Ljung-Box Test - independence/white noise of the forecasts residuals

```
#> # A tibble: 2 x 5
#>   City    Measure       .model  lb_stat lb_pvalue
#>   <chr>   <fct>         <chr>     <dbl>     <dbl>
#> 1 Potsdam Temperature   ETS_AAN    52.5  0.000674
#> 2 Potsdam Precipitation ETS_AAN    20.7  0.658
#> # A tibble: 2 x 5
#>   City    Measure       .model      lb_stat lb_pvalue
#>   <chr>   <fct>         <chr>         <dbl>     <dbl>
#> 1 Potsdam Temperature   arima_111wd    26.5     0.189
#> 2 Potsdam Precipitation arima_111wd    22.0     0.401
```

Table 5: Mean Yearly ARIMA and ETS Forecast values of mean yearly data (next 25 years); Units: Temperature (degree C), Precipitation (mm/Month), CO2 (ppm)

City	Measure	Year	ETS_AAN	arima_111wd
Potsdam	Temperature	2026	11.08	10.70
Potsdam	Temperature	2030	11.43	10.88
Potsdam	Temperature	2035	11.87	11.08
Potsdam	Temperature	2040	12.31	11.29
Potsdam	Temperature	2045	12.75	11.49
Potsdam	Temperature	2050	13.19	11.69
Potsdam	Precipitation	2026	45.81	49.14
Potsdam	Precipitation	2030	45.23	47.43
Potsdam	Precipitation	2035	44.51	47.28
Potsdam	Precipitation	2040	43.79	47.14
Potsdam	Precipitation	2045	43.07	46.99
Potsdam	Precipitation	2050	42.34	46.84

Table 6: Forecast increase/decrease over the next 25 years; Units: Temperature (degree C), Precipitation (mm/Month), CO2 (ppm)

Measure	Year.x	Year.y	ETS.x	ARIMA.x	ETS.y	ARIMA.y	Delta_ETS	Delta_ARIMA
Temperature	2026	2050	11.08	10.70	13.19	11.69	2.11	1.0
Precipitation	2026	2050	45.81	49.14	42.34	46.84	-3.46	-2.3