

Climate Data Forecasting - Atmospheric CO_2 Concentration / Temperature / Precipitation

Wolfgang Vollmer

2023-02-20

Contents

1	Forecasting of Giessen - Temperature Climate Analysis	2
1.1	Stationarity and differencing	2
1.1.1	Ljung-Box Test - independence/white noise of the time series	3
1.1.2	Unitroot KPSS Test - fix number of seasonal differences/differences required . . .	3
1.1.3	ACF Plots of Differences	4
1.1.4	Time Series, ACF and PACF Plots of Differences - for ARIMA p, q check	5
2	ExponenTial Smoothing (ETS) Forecasting Models	6
2.1	ETS Models and their componentes	7
2.1.1	Residual Accuracy with one-step-ahead fitted residuals - check RMSE, MAE . . .	8
2.1.2	Ljung-Box Test - independence/white noise of the forecasts residuals	9
2.1.3	ETS Models - components of ETS(A,N,A), ETS(A,A,A), ETS(A,Ad,A), models .	9
2.1.4	Forecast Accuracy with Training/Test Data	9
2.2	Forecasting with selected ETS model <ETS(A,A,A)>	10
2.2.1	Forecast Plot of selected ETS model	10
2.2.2	Residual Stationarity	11
2.2.3	Histogram of forecast residuals with overlaid normal curve	12
3	ARIMA Forecasting Models - AutoRegressive-Integrated Moving Average	13
3.1	Seasonal ARIMA models	13
3.1.1	Residual Accuracy with one-step-ahead fitted residuals - check RMSE, MAE . . .	15
3.1.2	Ljung-Box Test - independence/white noise of the forecasts residuals	15
3.1.3	Forecast Accuracy with Training/Test Data	15
3.2	Temperature - Forecasting with selected ARIMA model <ARIMA(0,1,2)(0,1,2)[12]> . . .	16
3.2.1	Forecast Plot of selected ARIMA model	16
3.2.2	Residual Stationarity	17
3.2.3	Histogram of forecast residuals with overlaid normal curve	18

4	ARIMA vs ETS	19
4.0.1	Comparing Residual and Forecast Accuracy of selected ETS and ARIMA model	19
4.0.2	Forecast Plot of selected ETS and ARIMA model	19
4.0.3	Ljung-Box Test - independence/white noise of the forecasts residuals	21
5	Yearly Data Forecasts with ARIMA and ETS	21
5.0.1	Comparing Residual and Forecast Accuracy of selected ETS and ARIMA model	21
5.0.2	Forecast Plot of selected ETS and ARIMA model	22
5.0.3	Ljung-Box Test - independence/white noise of the forecasts residuals	23
6	Backup	23

1 Forecasting of Giessen - Temperature 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 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

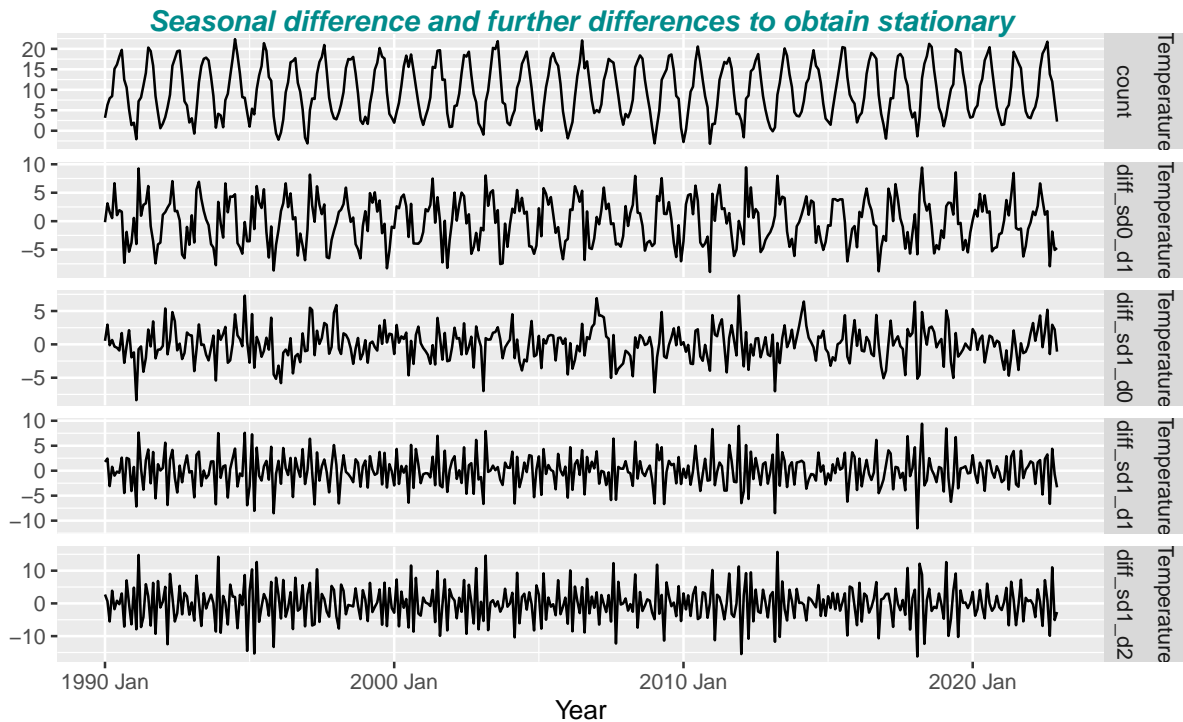
The model fit residuals have to be stationary. For good forecasting this has to be verified with residual diagnostics.

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 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 resp. ARIMA models. There we have to check whether the forecast errors are normally distributed with mean zero

Null Hypothesis of independence/white noise in a given time series

=> H_0 to be rejected for $p < \alpha = 0.05$

=> data in the given time series are dependent

=> even differenced data are dependent if $p < \alpha = 0.05$

=> independence/white noise of residuals of fitted models to be verified

```
#> Ljung-Box test with (count), w/o differences
#> # A tibble: 1 x 3
#>   Measure    lb_stat lb_pvalue
#>   <fct>      <dbl>    <dbl>
#> 1 Temperature 5232.      0
#> Ljung-Box test on (difference(count, 12))
#> # A tibble: 1 x 3
#>   Measure    lb_stat lb_pvalue
#>   <fct>      <dbl>    <dbl>
#> 1 Temperature 74.8 5.29e-12
#> Ljung-Box test on (difference(count, 12) + difference())
#> # A tibble: 1 x 3
#>   Measure    lb_stat lb_pvalue
#>   <fct>      <dbl>    <dbl>
#> 1 Temperature 256.      0
```

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

kpss test of stationary

Null Hypothesis of stationary in a given time series

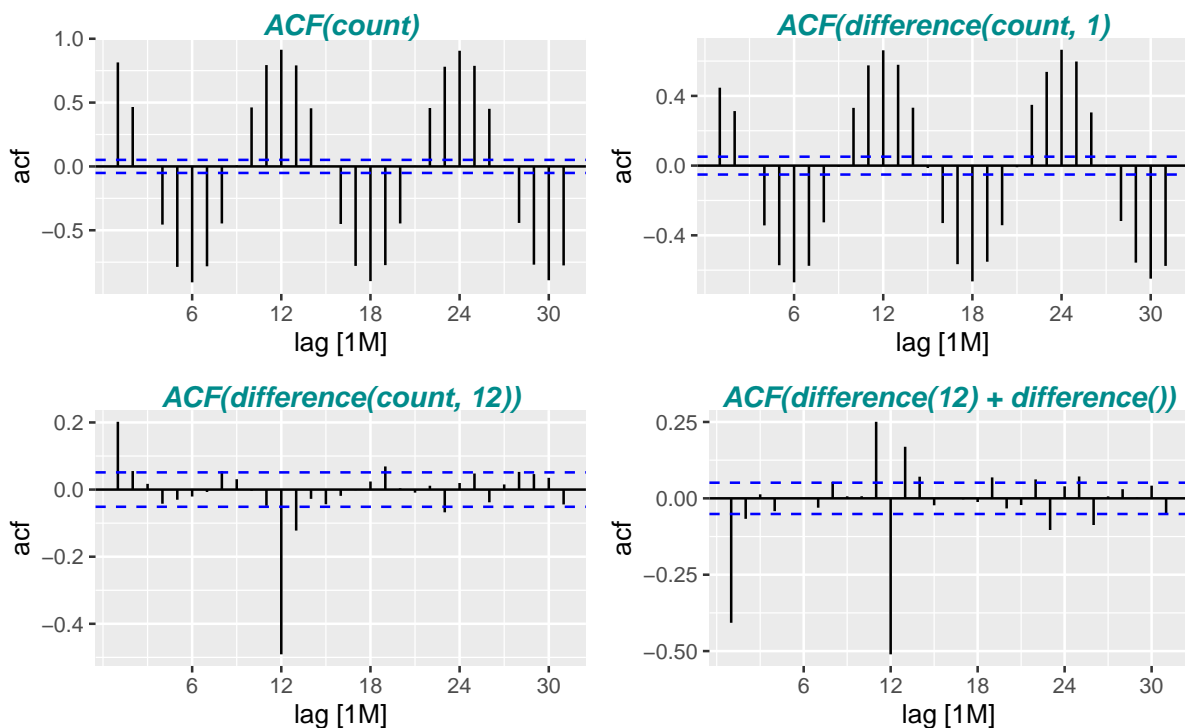
=> H_0 to be rejected for $p < \alpha = 0.05$

unitroot_nsdiffs/ndiff provides minimum number of seasonal differences/differences required for a stationary series. First fix required seasonal differences and then apply ndiffs to the seasonally differenced data.

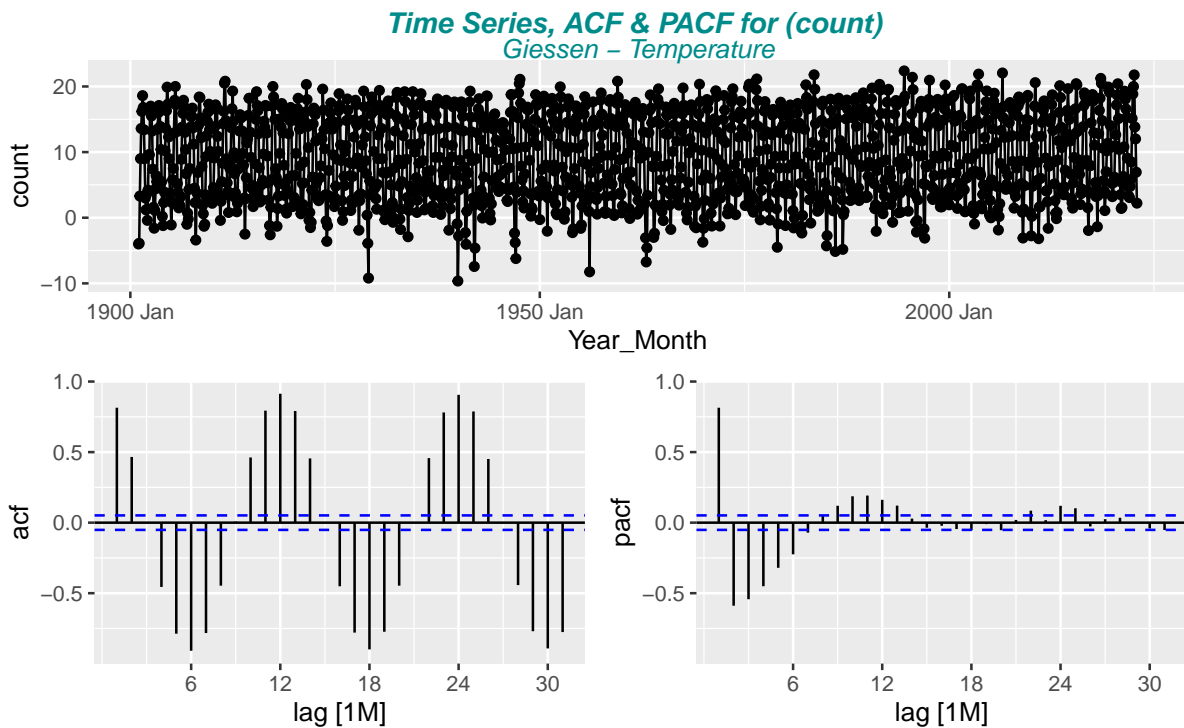
- returns 1 => for stationarity one seasonal difference resp. difference is required

```
#> ndiffs gives the number of differences required resp.
#> nsdiffs gives the number of seasonal differences required to make
#> a series stationary (test is based on the KPSS test)
#> kpss test, nsdiffs & ndiffs on (count), w/o differences
#> # A tibble: 1 x 5
#>   Measure      kpss_stat kpss_pvalue nsdiffs ndiffs
#>   <fct>         <dbl>         <dbl>   <int> <int>
#> 1 Temperature    0.361         0.0939     1     0
#> kpss test, nsdiffs & ndiffs on (difference(count, 12))
#> # A tibble: 1 x 5
#>   Measure      kpss_stat kpss_pvalue nsdiffs ndiffs
#>   <fct>         <dbl>         <dbl>   <int> <int>
#> 1 Temperature    0.00920         0.1       0     0
#> kpss test, nsdiffs & ndiffs on (difference(count, 12) %>% difference(1))
#> # A tibble: 1 x 5
#>   Measure      kpss_stat kpss_pvalue nsdiffs ndiffs
#>   <fct>         <dbl>         <dbl>   <int> <int>
#> 1 Temperature    0.0101         0.1       0     0
```

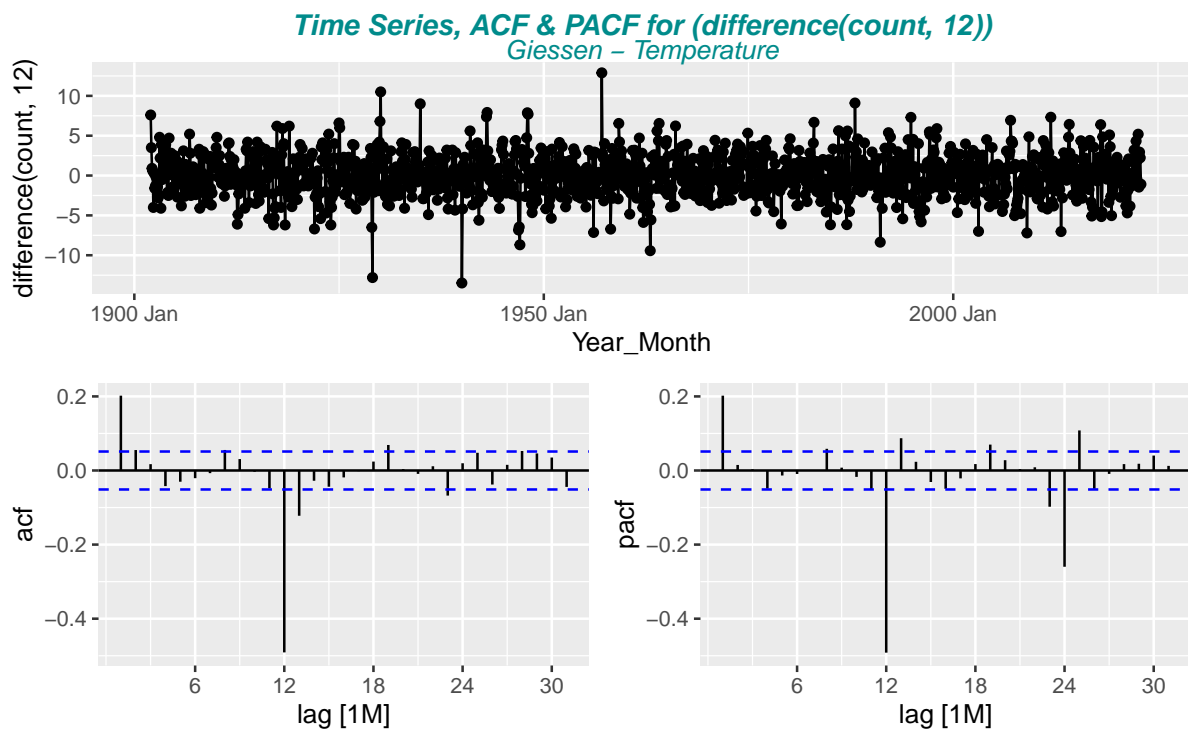
1.1.3 ACF Plots of Differences



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

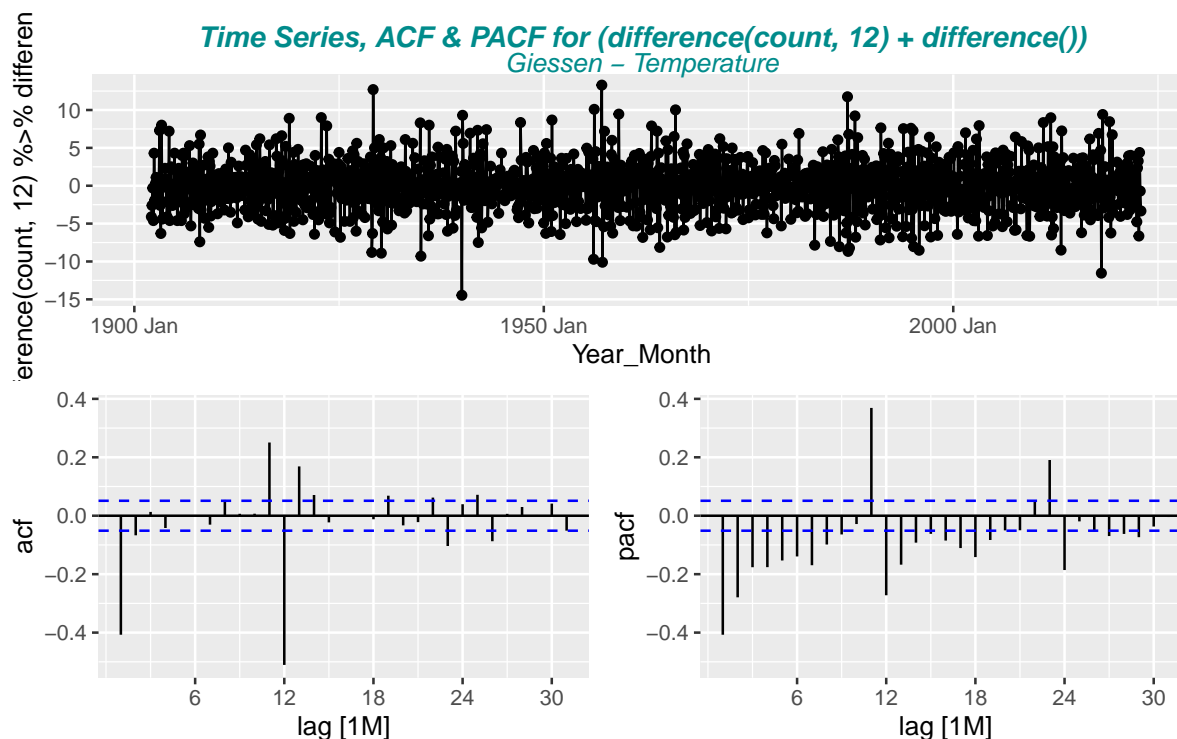


```
#> # A tibble: 1 x 2
#>   Sum   Mean
#>   <dbl> <dbl>
#> 1  36.7  0.0253
```



```
#> # A tibble: 1 x 2
#>   Sum   Mean
```

```
#> <dbl> <dbl>
#> 1 36.7 0.0253
```



```
#> # A tibble: 1 x 2
#>   Sum      Mean
#>   <dbl>   <dbl>
#> 1 -8.69 -0.00599
```

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 together). 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 together). Best Forecast accuracy is with ETS(A,A,A), ETS(A,Ad,A), ETS(A,N,A),

Trend term “N” for Basel Temperature/Precipitation corresponds to a “pure” exponential smoothing 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 interval 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”).

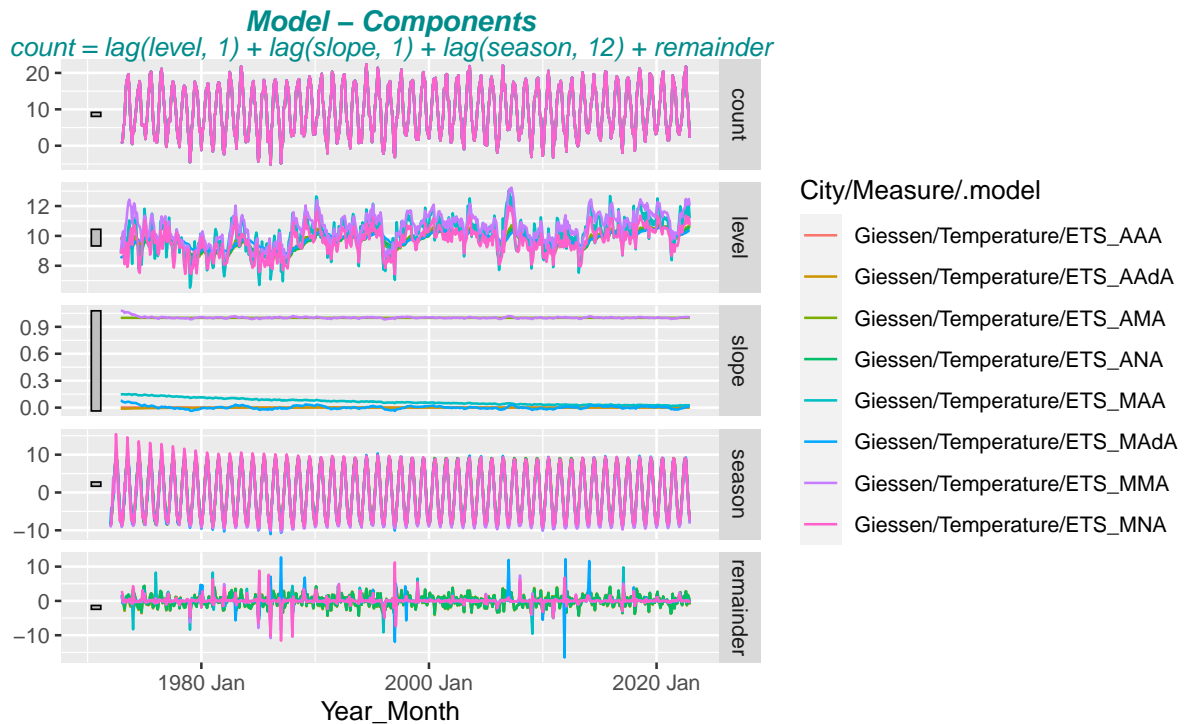
For CO2 and Temperature Data we have a clear seasonal pattern and seasonal term adds always a (more or less) fix amount on level and trend component. Therefore “A” additve term is chosen. For Precipitation the seasonal patttern is only slight. Instead, a multiplicative seasonal term results in “exploding”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

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

2.1 ETS Models and their componentes

```
#> [1] "model(ETS(count)) => provides best automatically chosen model"
#> # A tibble: 1 x 11
#>   City      Measure      .model sigma2 log_lik  AIC  AICc  BIC  MSE  AMSE  MAE
#>   <chr>    <fct>      <chr>   <dbl>  <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
#> 1 Giessen Temperature ETS(co~    3.10  -2252. 4533. 4534. 4599.  3.03  3.07  1.38
#> Series: count
#> Model: ETS(A,N,A)
#> Smoothing parameters:
#>   alpha = 0.0525464
#>   gamma = 0.0001008331
#>
#> Initial states:
#>   l[0]      s[0]      s[-1]      s[-2]      s[-3]      s[-4]      s[-5]      s[-6]
#> 9.051672 -7.554722 -4.648354 -0.1551301 4.439662 8.641045 9.070664 7.15734
#>   s[-7]      s[-8]      s[-9]      s[-10]     s[-11]
#> 3.936461 -0.458371 -4.193559 -7.801169 -8.433866
#>
#> sigma^2: 3.1033
#>
#>   AIC      AICc      BIC
#> 4533.469 4534.291 4599.423
#> Model Selection by Information Criterion - lowest AIC, AICc, BIC
#> # A tibble: 8 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 Giessen Temperature ETS_ANA    3.10  -2252. 4533. 4534. 4599.  3.03  3.07 1.38
#> 2 Giessen Temperature ETS_AAA    3.12  -2252. 4539. 4540. 4613.  3.04  3.08 1.38
#> 3 Giessen Temperature ETS_AMA    3.12  -2252. 4539. 4540. 4614.  3.04  3.09 1.39
#> 4 Giessen Temperature ETS_AA~    3.12  -2252. 4539. 4540. 4618.  3.03  3.07 1.38
#> 5 Giessen Temperature ETS_MAA    1.75  -3247. 6527. 6528. 6602.  3.91  4.16 0.511
#> 6 Giessen Temperature ETS_MMA    1.79  -3248. 6531. 6532. 6606.  4.31  4.62 0.522
#> 7 Giessen Temperature ETS_MNA    2.23  -3290. 6609. 6610. 6675.  4.24  4.40 0.581
#> 8 Giessen Temperature ETS_MA~    2.83  -3379. 6794. 6796. 6874.  3.76  3.78 0.572
```



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

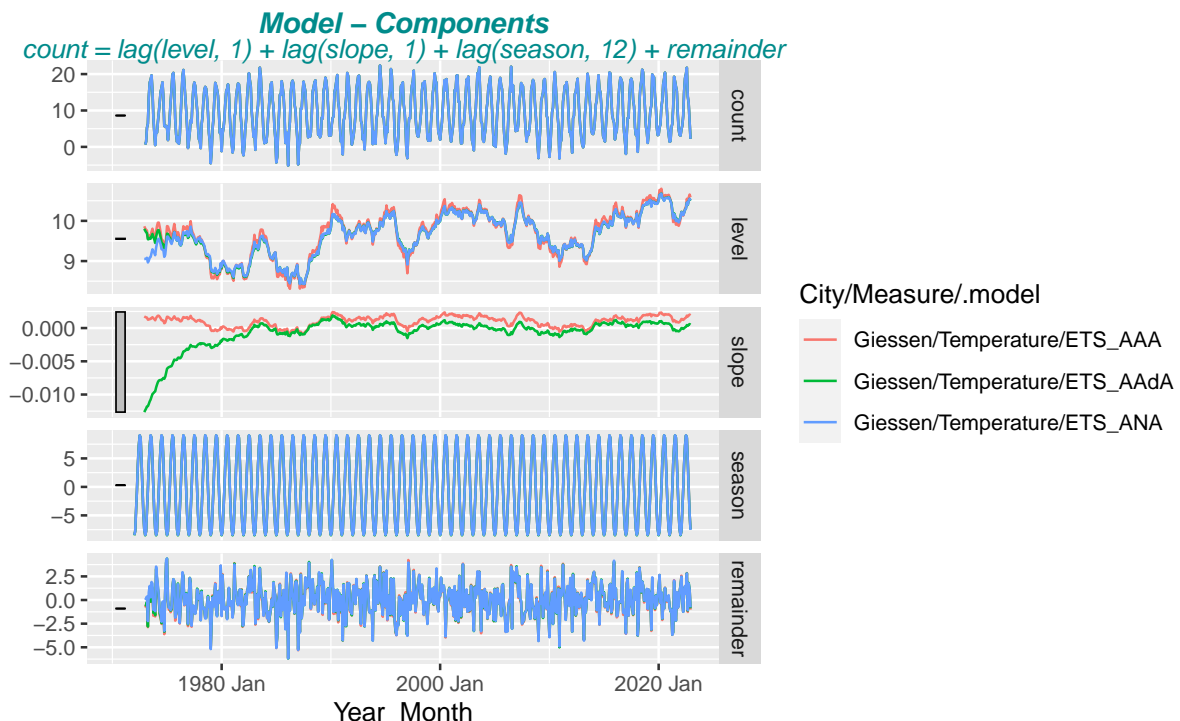
- MAE (Mean absolute error, will lead to forecasts of the median) and
- RMSE (Root mean squared error, lead to forecasts of the mean)

```
#> # A tibble: 8 x 12
#>   City      Measure      .model  .type      ME  RMSE  MAE  MPE  MAPE  MASE  RMSSE
#>   <chr>    <fct>      <chr>  <chr>    <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
#> 1 Giessen Temperature ETS_AA~ Trai~  0.0363  1.74  1.38  46.2  80.7  0.717  0.699
#> 2 Giessen Temperature ETS_ANA Trai~  0.0463  1.74  1.38  46.3  79.6  0.717  0.700
#> 3 Giessen Temperature ETS_AAA Trai~  0.00261  1.74  1.38  43.6  78.2  0.718  0.700
#> 4 Giessen Temperature ETS_AMA Trai~  0.0134  1.74  1.39  43.8  78.0  0.720  0.701
#> 5 Giessen Temperature ETS_MA~ Trai~  0.0104  1.94  1.53  57.5  96.0  0.794  0.779
#> 6 Giessen Temperature ETS_MAA Trai~ -0.149  1.98  1.58  51.3  91.8  0.820  0.795
#> 7 Giessen Temperature ETS_MNA Trai~  0.0127  2.06  1.63  56.8  93.7  0.848  0.828
#> 8 Giessen Temperature ETS_MMA Trai~ -0.131  2.08  1.63  52.4  95.9  0.845  0.835
#> # ... with 1 more variable: ACF1 <dbl>
```


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

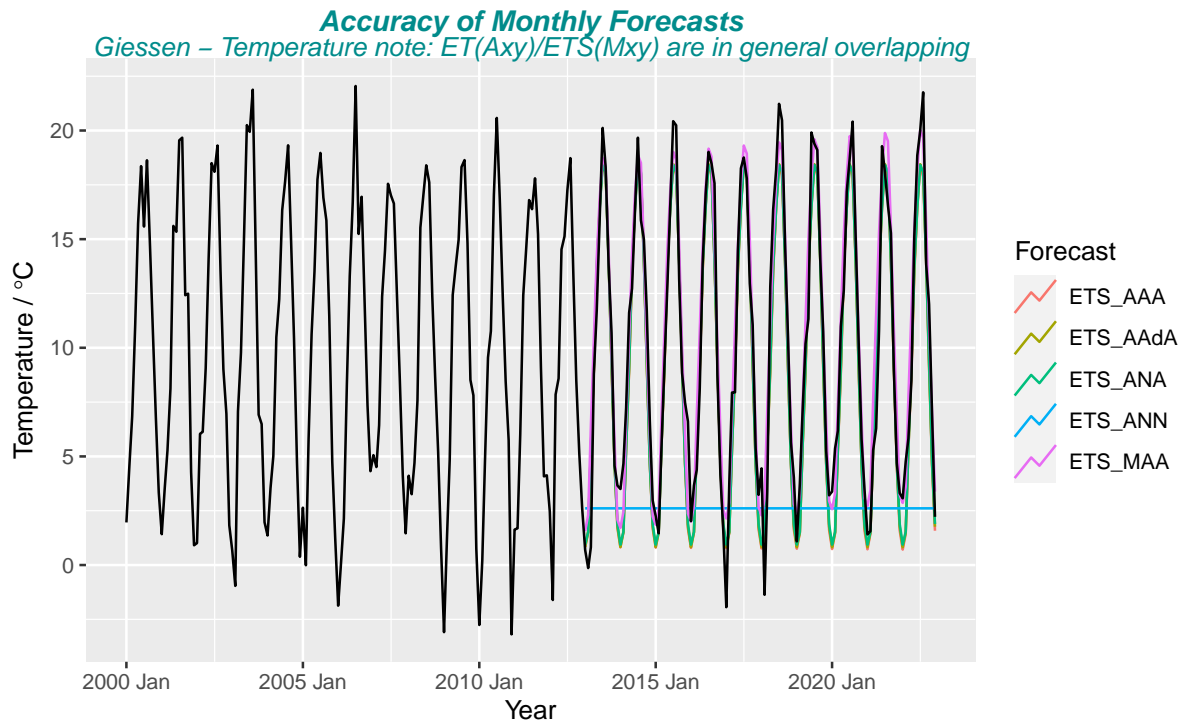
```
#> Null Hypothesis of independence/white noise for residuals - for p < 0.05: reject H_0
#> # A tibble: 8 x 5
#>   City      Measure      .model lb_stat lb_pvalue
#>   <chr>    <fct>      <chr>    <dbl>   <dbl>
#> 1 Giessen Temperature ETS_MMA      36.0  0.210
#> 2 Giessen Temperature ETS_AAA      36.0  0.208
#> 3 Giessen Temperature ETS_ANA      36.3  0.200
#> 4 Giessen Temperature ETS_AMA      36.3  0.199
#> 5 Giessen Temperature ETS_AAdA     36.9  0.181
#> 6 Giessen Temperature ETS_MAdA     66.4  0.000147
#> 7 Giessen Temperature ETS_MAA      67.1  0.000117
#> 8 Giessen Temperature ETS_MNA      74.3  0.0000126
```

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

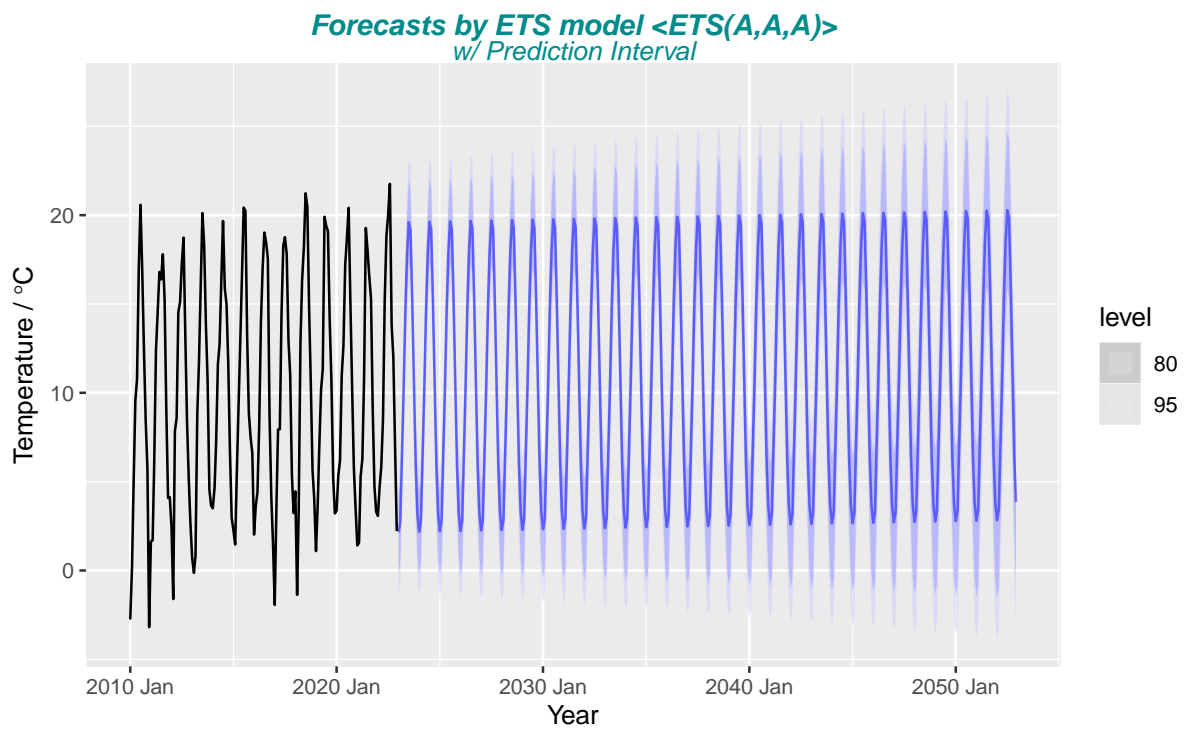
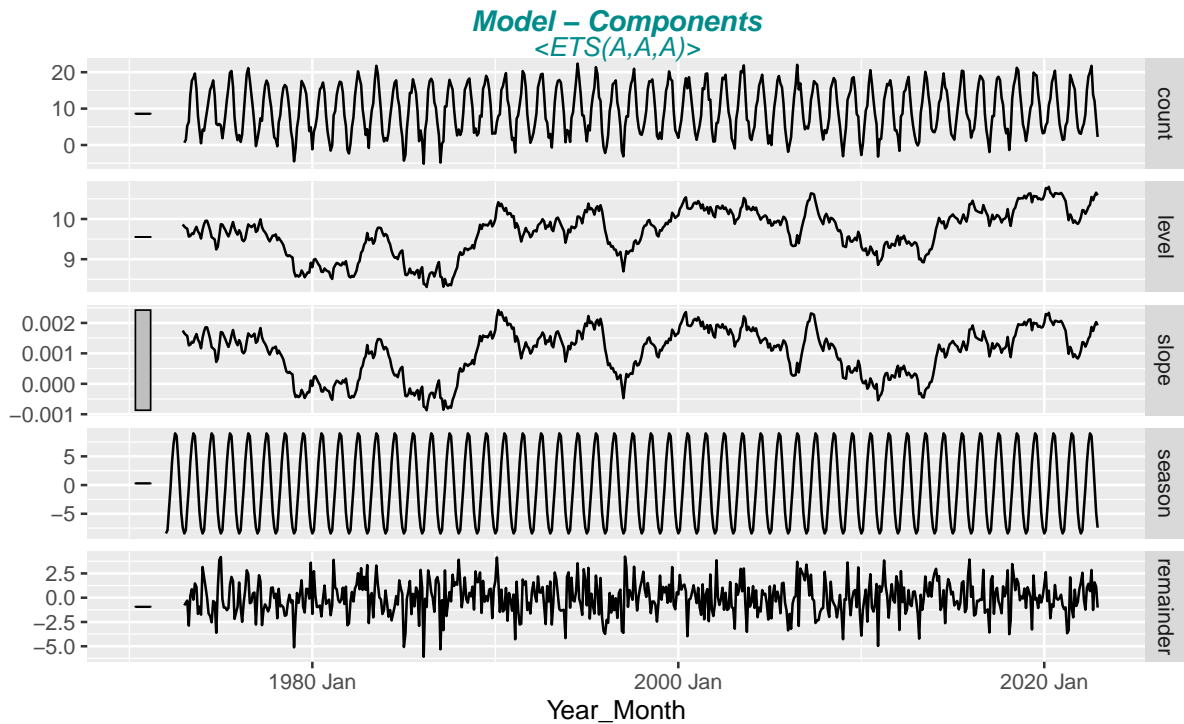
```
#> # A tibble: 5 x 12
#>   .model City      Measure .type      ME  RMSE  MAE  MPE  MAPE  MASE  RMSSE  ACF1
#>   <chr>   <chr>    <fct>   <chr>   <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
#> 1 ETS_MAA Giess~ Temper~ Test  -0.474  1.78  1.39  5.10  45.4  0.725  0.719  0.203
#> 2 ETS_ANA Giess~ Temper~ Test   0.773  1.82  1.47  21.1  37.1  0.766  0.736  0.182
#> 3 ETS_AAdA Giess~ Temper~ Test   0.838  1.85  1.48  22.3  37.4  0.774  0.746  0.179
#> 4 ETS_AAA Giess~ Temper~ Test   0.886  1.87  1.51  23.4  38.2  0.787  0.756  0.185
#> 5 ETS_ANN Giess~ Temper~ Test   7.63  9.96  8.01  72.5  89.1  4.18  4.02  0.806
```



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

2.2.1 Forecast Plot of selected ETS model

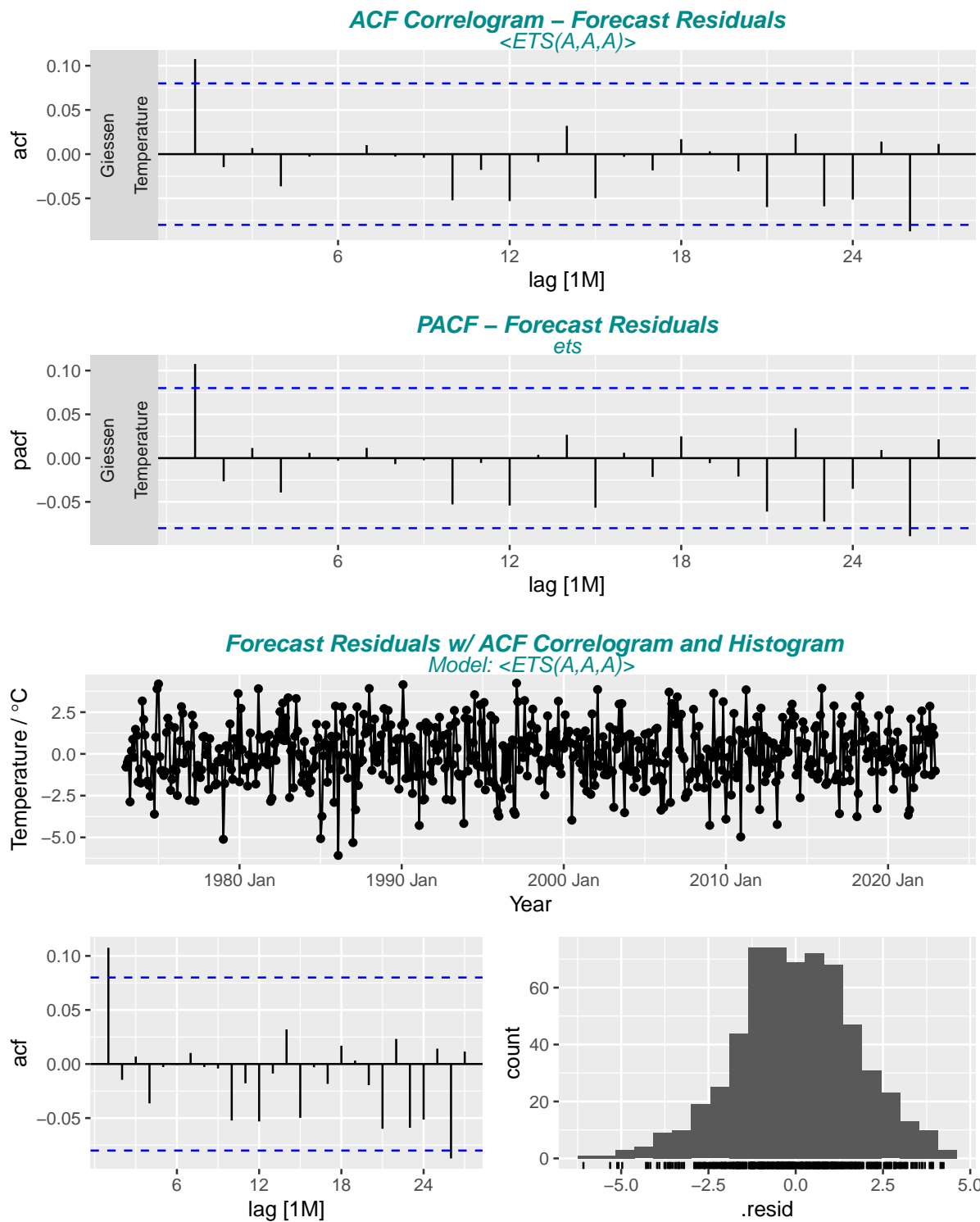
```
#> Provide model coefficients by report(fit_model)
#> Series: count
#> Model: ETS(A,A,A)
#> Smoothing parameters:
#>   alpha = 0.06878278
#>   beta  = 0.0001083632
#>   gamma = 0.0001114003
#>
#> Initial states:
#>   l[0]      b[0]      s[0]      s[-1]      s[-2]      s[-3]      s[-4]
#> 9.864865 0.001749682 -7.418715 -4.781709 -0.03983853 4.543051 8.549955
#>   s[-5]      s[-6]      s[-7]      s[-8]      s[-9]      s[-10]      s[-11]
#> 9.007734 7.208669 3.963142 -0.4976057 -4.241446 -7.859552 -8.433685
#>
#> sigma^2: 3.1193
#>
#>   AIC   AICc   BIC
#> 4538.509 4539.561 4613.257
```



2.2.2 Residual Stationarity

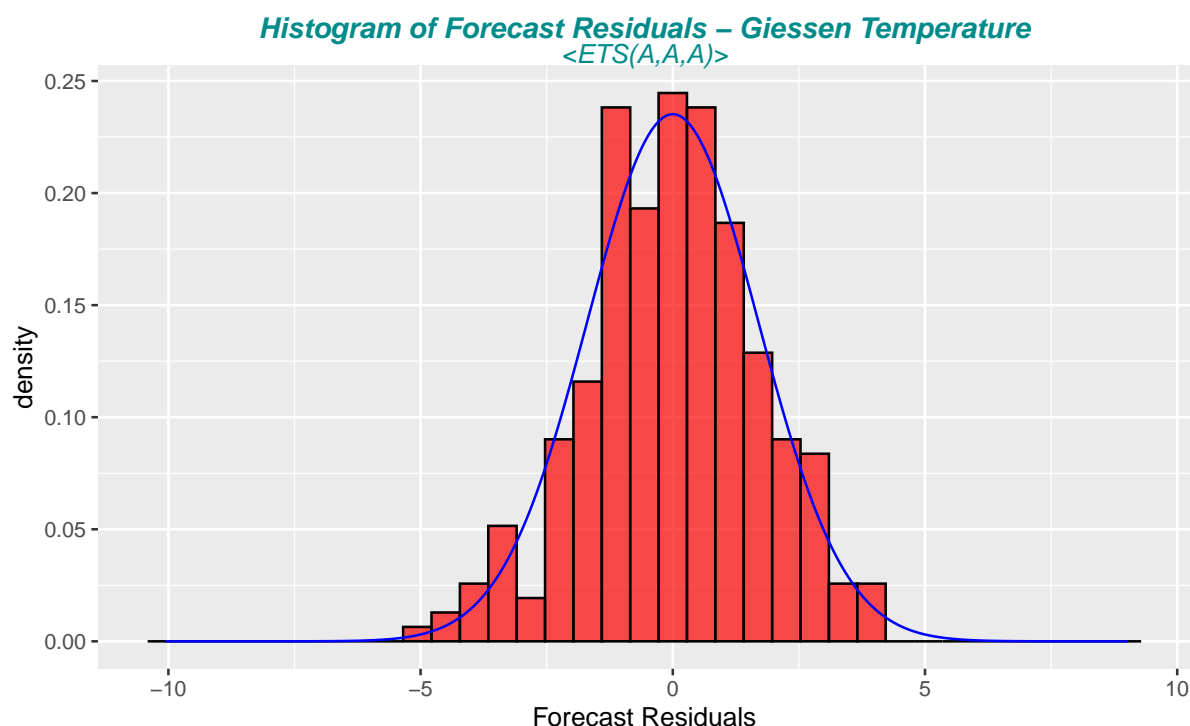
Required checks to be ready for forecasting:

- 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 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: 1 x 5
#>   City      Measure    .model lb_stat lb_pvalue
#>   <chr>    <fct>      <chr>   <dbl>   <dbl>
#> 1 Giessen Temperature ets      32.9    0.328
```



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

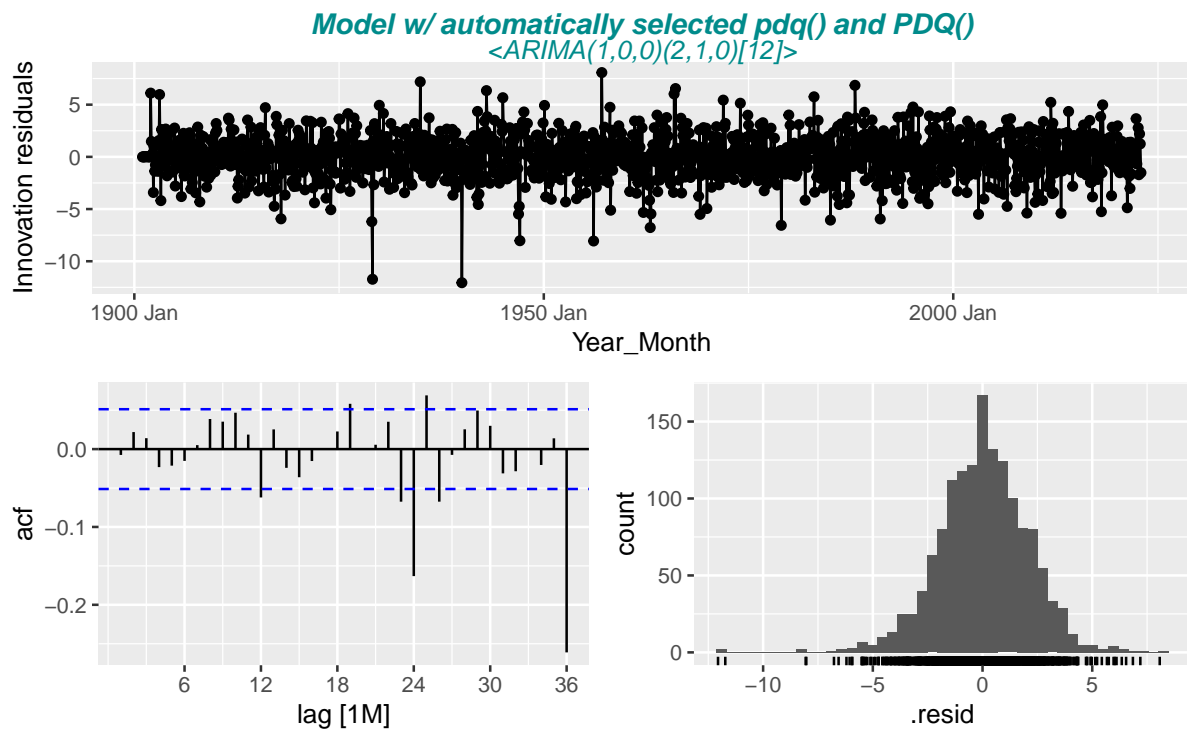
Non-seasonal ARIMA models are generally denoted $ARIMA(p,d,q)$ where parameters p , d , and q are non-negative integers, * p is the order (number of time lags) of the autoregressive model * d is the degree of differencing (number of times the data have had past values subtracted) * q is the order of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

```
#> # A tibble: 1 x 10
#>   City      Measure   .model sigma2 log_lik   AIC   AICc   BIC ar_roots  ma_ro~1
#>   <chr>   <fct>       <chr>   <dbl>   <dbl> <dbl> <dbl> <dbl> <list>   <list>
#> 1 Giessen Temperature arima     4.31 -3123. 6254. 6254. 6275. <cpl [25]> <cpl>
#> # ... with abbreviated variable name 1: ma_roots
#> Series: count
```

```
#> Model: ARIMA(1,0,0)(2,1,0)[12]
#>
#> Coefficients:
#>          ar1          sar1          sar2
#>         0.2252      -0.6535      -0.3004
#> s.e.    0.0256    0.0252    0.0252
#>
#> sigma^2 estimated as 4.313:  log likelihood=-3122.81
#> AIC=6253.62  AICc=6253.65  BIC=6274.74
```



```
#> Model Selection by Information Criterion - lowest AIC, AICc, BIC
#> choose p, q parameter accordingly - but only for same d, D values
#> # A tibble: 13 x 10
#>   City      Measure .model  sigma2 log_lik  AIC  AICc  BIC ar_ro~1 ma_ro~2
#>   <chr>    <fct>    <chr>    <dbl>  <dbl>  <dbl> <dbl> <dbl> <list>  <list>
#> 1 Giessen Temperature ARIMA_1~  3.05 -1184. 2379. 2379. 2401. <cpl> <cpl>
#> 2 Giessen Temperature ARIMA_0~  3.05 -1184. 2379. 2379. 2401. <cpl> <cpl>
#> 3 Giessen Temperature ARIMA_2~  3.06 -1184. 2379. 2379. 2401. <cpl> <cpl>
#> 4 Giessen Temperature ARIMA_1~  3.06 -1184. 2381. 2381. 2407. <cpl> <cpl>
#> 5 Giessen Temperature ARIMA_1~  4.11 -1251. 2510. 2510. 2528. <cpl> <cpl>
#> 6 Giessen Temperature ARIMA_1~  4.60 -1283. 2576. 2576. 2598. <cpl> <cpl>
#> 7 Giessen Temperature ARIMA_2~  4.60 -1283. 2576. 2576. 2598. <cpl> <cpl>
#> 8 Giessen Temperature ARIMA_3~  4.40 -1305. 2626. 2627. 2662. <cpl> <cpl>
#> 9 Giessen Temperature ARIMA_2~  5.74 -1346. 2700. 2700. 2718. <cpl> <cpl>
#> 10 Giessen Temperature ARIMA_1~  6.10 -1365. 2737. 2737. 2750. <cpl> <cpl>
#> 11 Giessen Temperature ARIMA_0~  6.10 -1366. 2737. 2737. 2750. <cpl> <cpl>
#> 12 Giessen Temperature ARIMA_0~  7.92 -1441. 2887. 2887. 2896. <cpl> <cpl>
#> 13 Giessen Temperature ARIMA_1~  8.66 -1466. 2936. 2936. 2945. <cpl> <cpl>
#> # ... with abbreviated variable names 1: ar_roots, 2: ma_roots
```

Good models are obtained by minimising the AIC, AICc or BIC (see `glance(fit_arma)` output). The preference is to use the AICc to select 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: 14 x 12
#>   City      Measure      .model .type      ME    RMSE    MAE    MPE    MAPE    MASE
#>   <chr>    <fct>      <chr>  <chr>    <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
#> 1 Giessen Temperature ARIMA_~ Trai~ 0.0290 1.72 1.35 48.2 79.8 0.702
#> 2 Giessen Temperature ARIMA_~ Trai~ 0.0272 1.72 1.35 48.6 80.4 0.703
#> 3 Giessen Temperature ARIMA_~ Trai~ 0.0301 1.72 1.35 50.5 81.8 0.703
#> 4 Giessen Temperature ARIMA_~ Trai~ 0.0296 1.72 1.35 48.9 80.5 0.703
#> 5 Giessen Temperature ARIMA_~ Trai~ 0.0377 2.00 1.57 62.8 97.3 0.814
#> 6 Giessen Temperature ARIMA_~ Trai~ 0.0192 2.09 1.64 36.7 84.1 0.850
#> 7 Giessen Temperature ARIMA_~ Trai~ 0.00203 2.12 1.65 37.2 77.5 0.856
#> 8 Giessen Temperature ARIMA_~ Trai~ 0.00203 2.12 1.65 37.2 77.5 0.856
#> 9 Giessen Temperature ARIMA_~ Trai~ -0.0131 2.36 1.84 27.8 69.0 0.956
#> 10 Giessen Temperature ARIMA_~ Trai~ -0.0134 2.44 1.87 78.8 126. 0.970
#> 11 Giessen Temperature ARIMA_~ Trai~ -0.0126 2.44 1.87 79.3 127. 0.970
#> 12 Giessen Temperature ARIMA_~ Trai~ -0.0104 2.78 2.14 64.9 116. 1.11
#> 13 Giessen Temperature ARIMA_~ Trai~ -0.00861 2.91 2.27 75.2 134. 1.18
#> 14 Giessen Temperature ARIMA_~ Trai~ NaN      NaN      NaN      NaN      NaN      NaN
#> # ... with 2 more variables: RMSSE <dbl>, ACF1 <dbl>
```

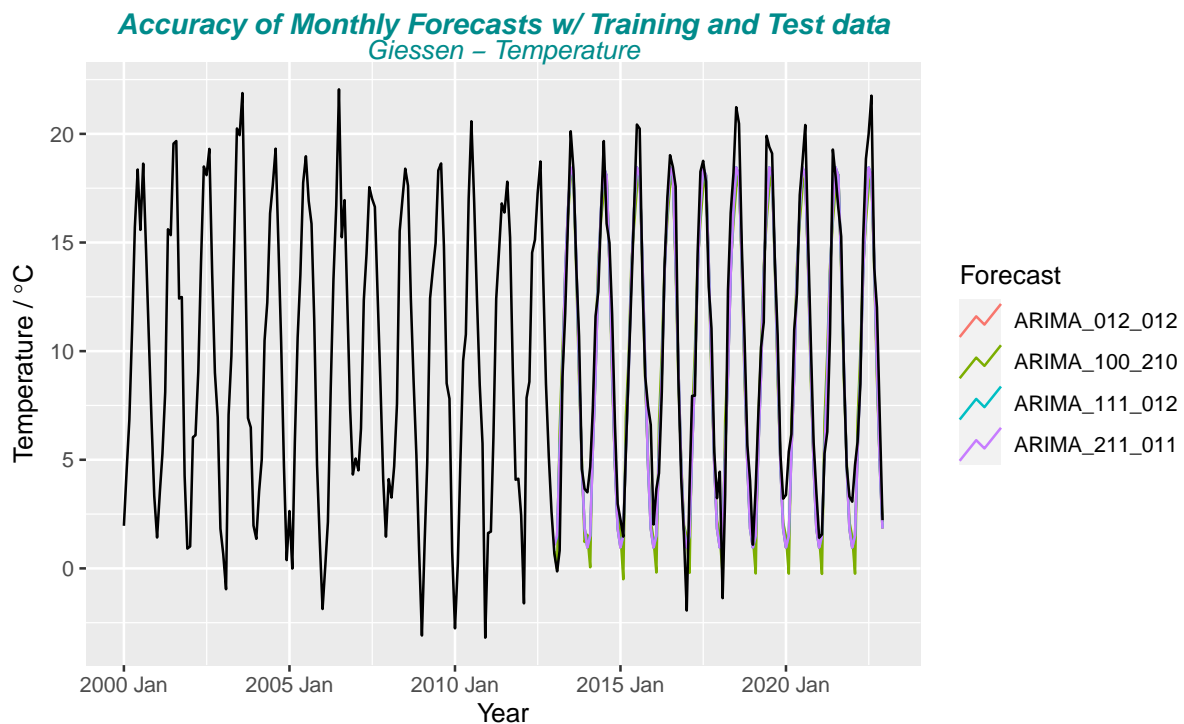
3.1.2 Ljung-Box Test - independence/white noise of the forecasts residuals

```
#> Null Hypothesis of independence/white noise for residuals - for p < 0.05: reject H_0
#> # A tibble: 14 x 5
#>   City      Measure      .model      lb_stat lb_pvalue
#>   <chr>    <fct>      <chr>      <dbl>    <dbl>
#> 1 Giessen Temperature ARIMA_012_012 26.7 6.38e- 1
#> 2 Giessen Temperature ARIMA_111_012 26.9 6.29e- 1
#> 3 Giessen Temperature ARIMA_211_011 27.8 5.81e- 1
#> 4 Giessen Temperature ARIMA_111_112 28.6 5.37e- 1
#> 5 Giessen Temperature ARIMA_301_200 88.3 1.18e- 7
#> 6 Giessen Temperature ARIMA_100_210 92.9 2.41e- 8
#> 7 Giessen Temperature ARIMA_100_110 110. 4.20e-11
#> 8 Giessen Temperature ARIMA_200_110 110. 4.20e-11
#> 9 Giessen Temperature ARIMA_010_110 281. 0
#> 10 Giessen Temperature ARIMA_012_010 192. 0
#> 11 Giessen Temperature ARIMA_110_010 333. 0
#> 12 Giessen Temperature ARIMA_111_010 193. 0
#> 13 Giessen Temperature ARIMA_210_110 179. 0
#> 14 Giessen Temperature ARIMA_002_200 NA NA
```

3.1.3 Forecast Accuracy with Training/Test Data

```
#> # A tibble: 4 x 12
#>   .model City Measure .type      ME    RMSE    MAE    MPE    MAPE    MASE    RMSSE    ACF1
#>   <chr>   <chr> <fct>  <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
```

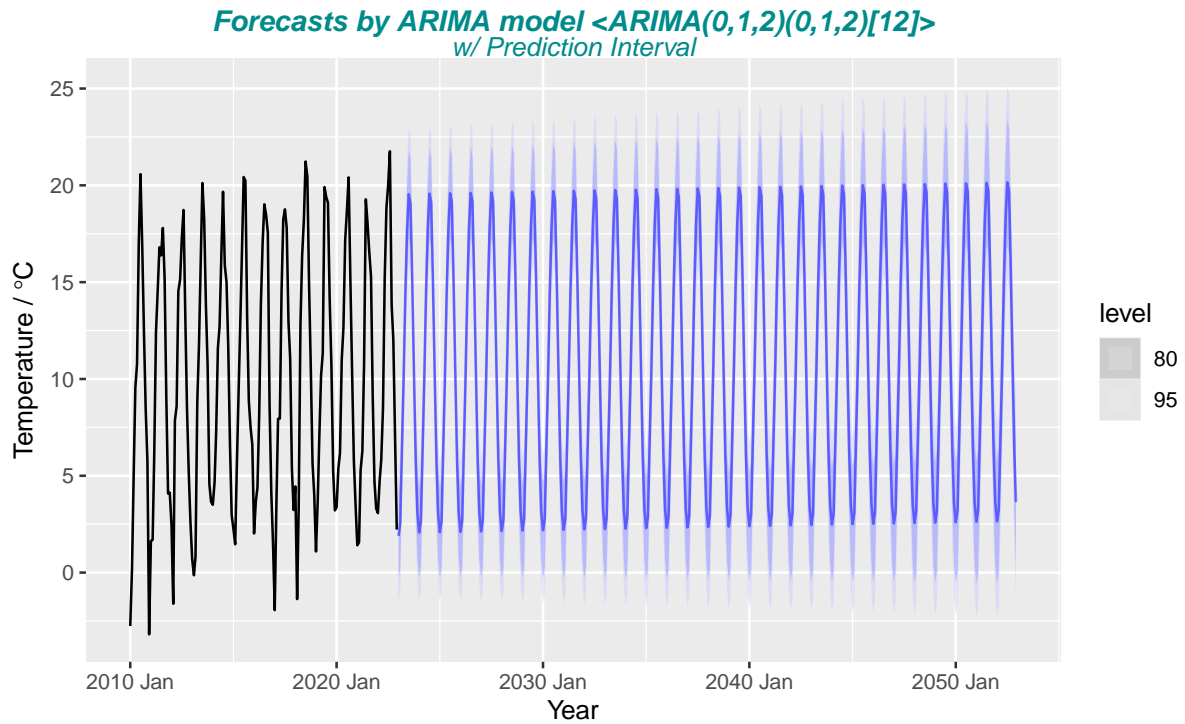
```
#> 1 ARIMA_11~ Gies~ Temper~ Test 0.792 1.82 1.46 20.5 36.7 0.762 0.735 0.169
#> 2 ARIMA_21~ Gies~ Temper~ Test 0.795 1.82 1.46 21.3 37.2 0.763 0.735 0.172
#> 3 ARIMA_01~ Gies~ Temper~ Test 0.806 1.83 1.47 20.7 36.7 0.765 0.738 0.170
#> 4 ARIMA_10~ Gies~ Temper~ Test 0.839 2.11 1.68 10.3 32.0 0.877 0.851 0.0864
```



3.2 Temperature - Forecasting with selected ARIMA model <ARIMA(0,1,2)(0,1,2)[12]>

3.2.1 Forecast Plot of selected ARIMA model

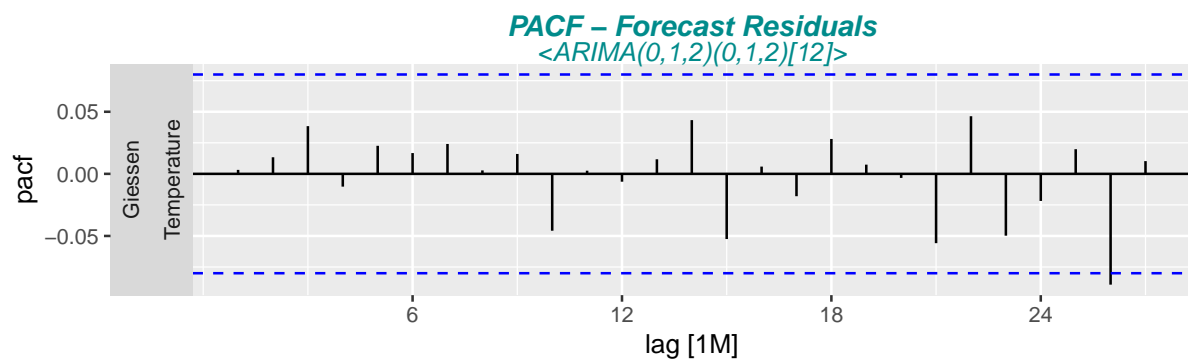
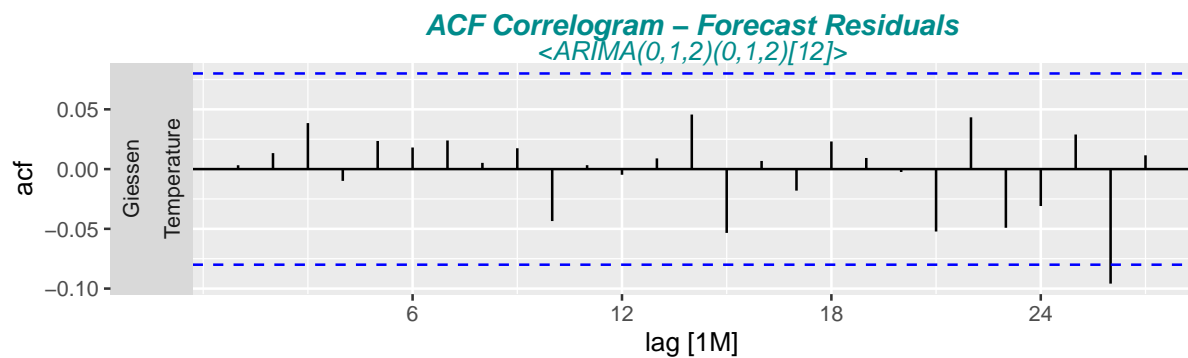
```
#> Provide model coefficients by report(fit_model)
#> Series: count
#> Model: ARIMA(0,1,2)(0,1,2)[12]
#>
#> Coefficients:
#>      ma1      ma2      sma1      sma2
#>    -0.8275 -0.1300 -1.0230  0.0230
#> s.e.    0.0411  0.0422  0.0532  0.0443
#>
#> sigma^2 estimated as 3.053: log likelihood=-1184.37
#> AIC=2378.75 AICc=2378.85 BIC=2400.62
```

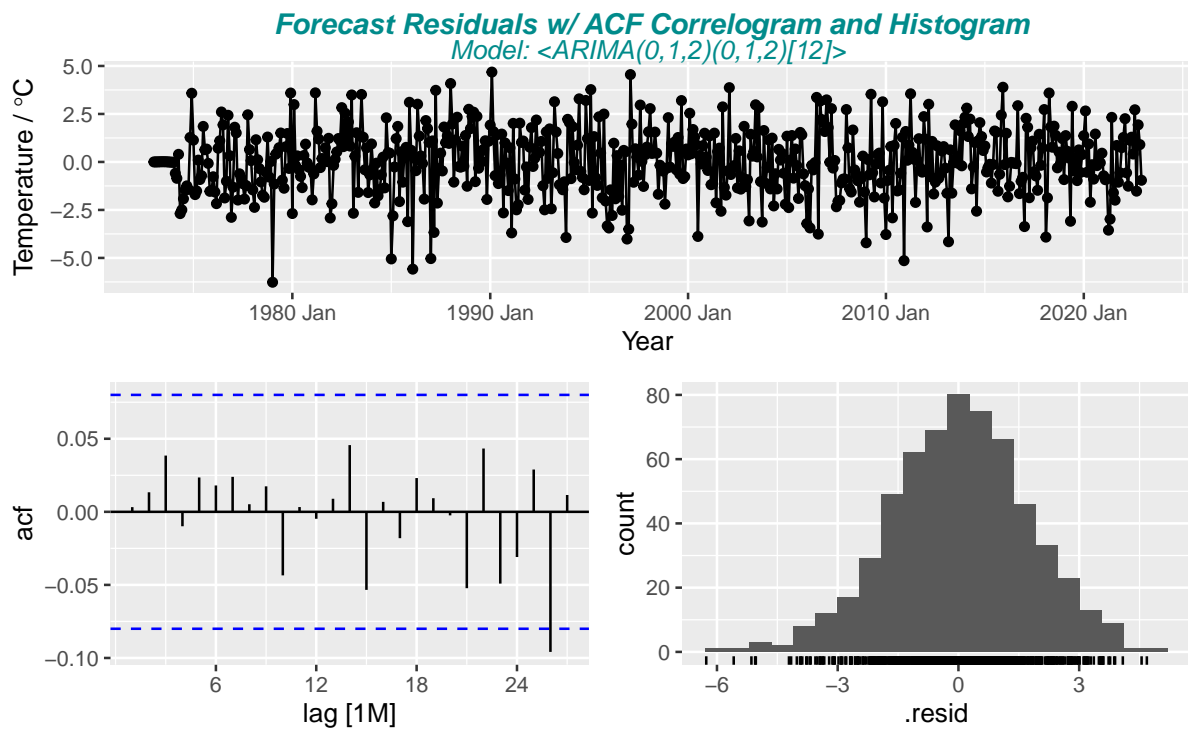



3.2.2 Residual Stationarity

Required checks to be ready for forecasting:

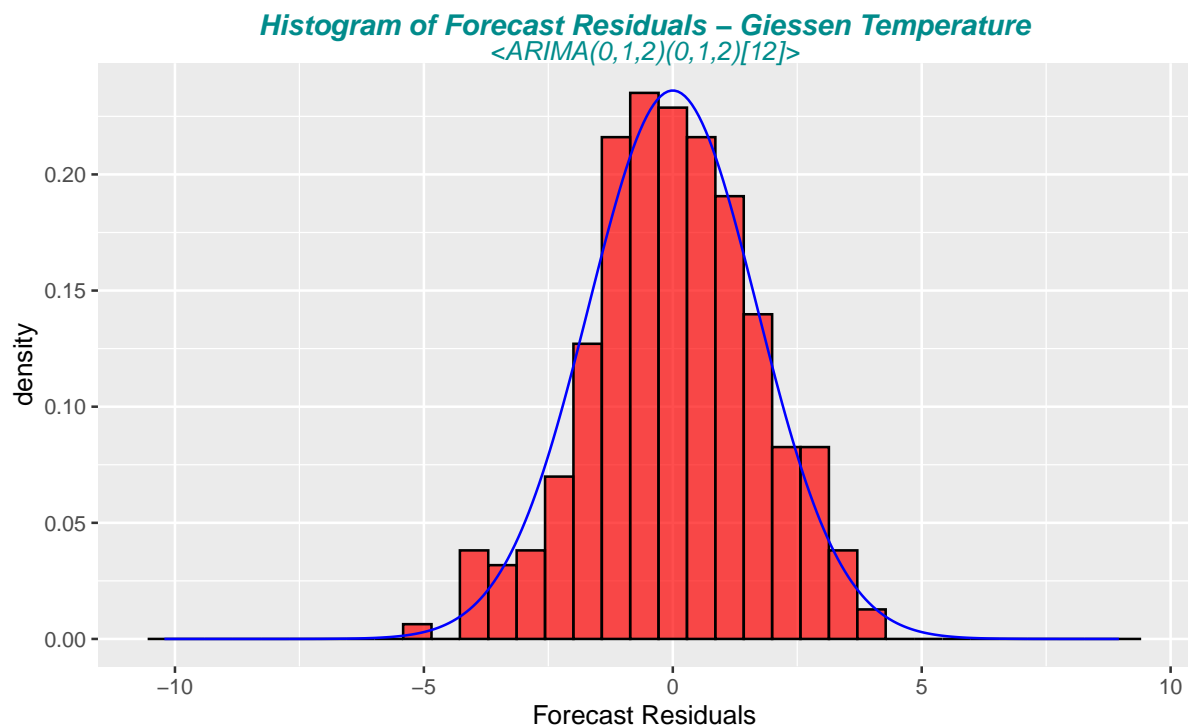
- ACF Forecast Residual: all spikes are within the significance limits, so the residuals appear to be white noise
- The Ljung-Box test also shows that the residuals have no remaining autocorrelations
- Forecast Residuals are more or less normally distributed with roughly centred on zero





3.2.3 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: 1 x 5
#>   City      Measure      .model lb_stat lb_pvalue
#>   <chr>    <fct>      <chr>   <dbl>   <dbl>
#> 1 Giessen Temperature arima      25.1     0.719
```



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 resp. ARIMA model the RMSE / MAE values. Lower values indicate a more accurate model based on the test set RMSE, ..., MASE.

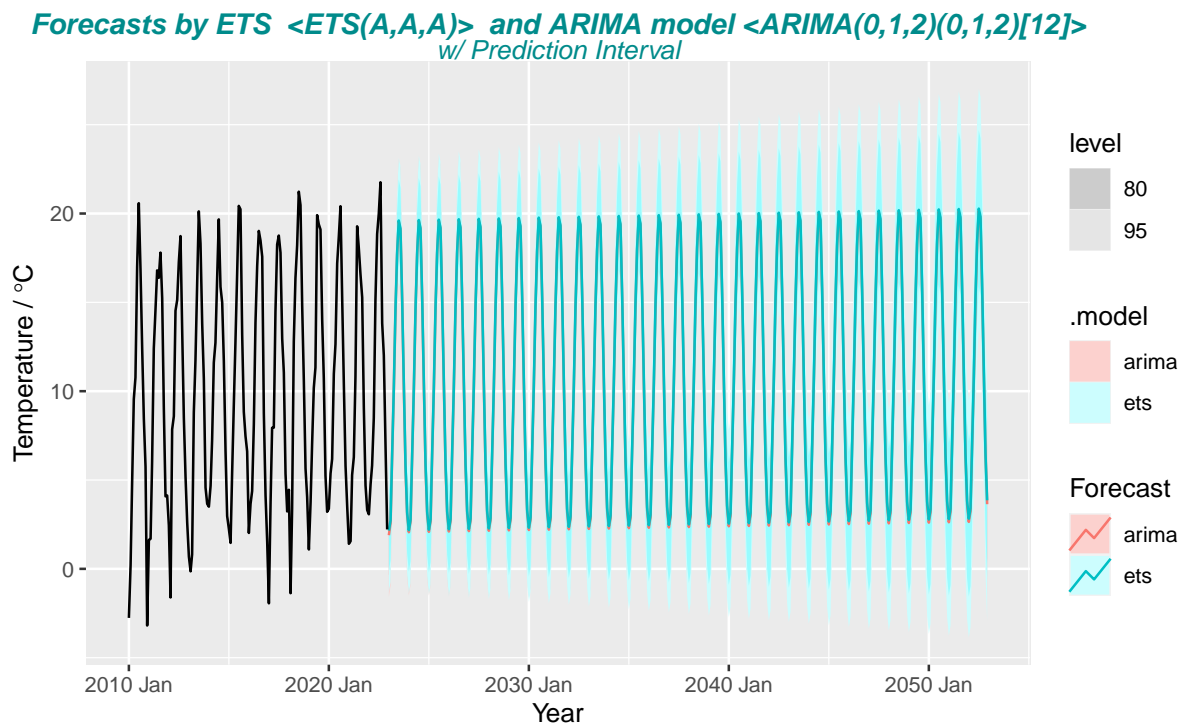
- Residual Accuracy with one-step-ahead fitted residuals
- Forecast Accuracy with Training/Test Data

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: 4 x 12
#>   City Measure .model .type      ME  RMSE  MAE  MPE  MAPE  MASE RMSSE  ACF1
#>   <chr> <fct>   <chr> <chr>   <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
#> 1 Gies~ Temper~ ets    Trai~ 0.00261 1.74 1.38 43.6 78.2 0.718 0.700 0.108
#> 2 Gies~ Temper~ arima  Trai~ 0.0272 1.72 1.35 48.6 80.4 0.703 0.692 0.00320
#> 3 Gies~ Temper~ ETS_A~ Test 0.886 1.87 1.51 23.4 38.2 0.787 0.756 0.185
#> 4 Gies~ Temper~ ARIMA~ Test 0.806 1.83 1.47 20.7 36.7 0.765 0.738 0.170
```

4.0.2 Forecast Plot of selected ETS and ARIMA model



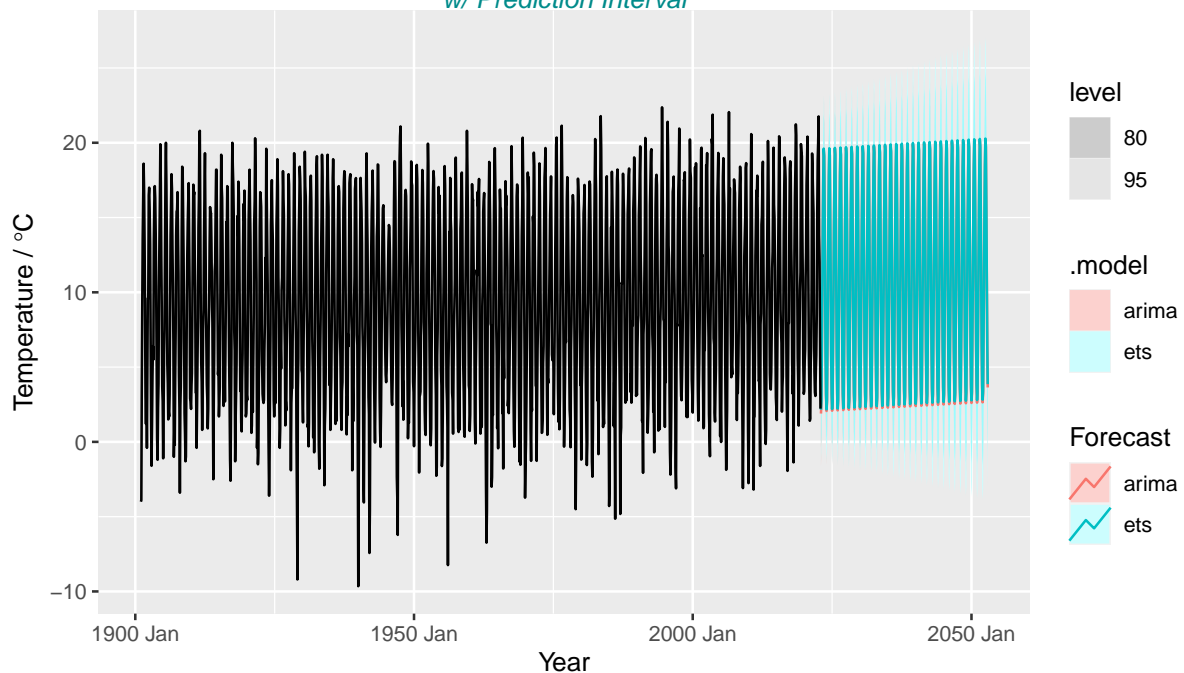
```
#> # A tsibble: 6 x 8 [1M]
#> # Key:      City, Measure, .model [2]
#> # Groups:   City, Measure, .model [2]
```

```

#>   City      Measure      .model Year_M~1      count .mean      '80%'
#>   <chr>    <fct>      <chr>    <mth>      <dist> <dbl>      <hilo>
#> 1 Giessen Temperature arima    2023 Jan N(1.9, 3.1)  1.89 [-0.3677544, 4.155698]80
#> 2 Giessen Temperature arima    2023 Feb N(2.6, 3.2)  2.58 [ 0.2865032, 4.876841]80
#> 3 Giessen Temperature arima    2023 Mrz N(6.3, 3.2)  6.27 [ 3.9752712, 8.569640]80
#> 4 Giessen Temperature ets      2023 Jan N(2.2, 3.1)  2.16 [-0.1079517, 4.418896]80
#> 5 Giessen Temperature ets      2023 Feb N(2.7, 3.1)  2.73 [ 0.4628109, 5.000388]80
#> 6 Giessen Temperature ets      2023 Mrz N(6.4, 3.1)  6.35 [ 4.0775217, 8.625837]80
#> # ... with 1 more variable: '95%' <hilo>, and abbreviated variable name
#> #   1: Year_Month
#> # A tibble: 6 x 8 [1M]
#> # Key:      City, Measure, .model [2]
#> # Groups:   City, Measure, .model [2]
#>   City      Measure      .model Year_M~1      count .mean      '80%'
#>   <chr>    <fct>      <chr>    <mth>      <dist> <dbl>      <hilo>
#> 1 Giessen Temperatu~ arima    2052 Okt  N(11, 6.5) 11.0 [ 7.7037624, 14.231954]80
#> 2 Giessen Temperatu~ arima    2052 Nov  N(6.4, 6.5)  6.39 [ 3.1262310,  9.660211]80
#> 3 Giessen Temperatu~ arima    2052 Dez  N(3.6, 6.5)  3.64 [ 0.3687402,  6.908729]80
#> 4 Giessen Temperatu~ ets      2052 Okt  N(11, 12) 11.2 [ 6.8096749, 15.658487]80
#> 5 Giessen Temperatu~ ets      2052 Nov  N(6.5, 12)  6.50 [ 2.0639852, 10.926187]80
#> 6 Giessen Temperatu~ ets      2052 Dez  N(3.9, 12)  3.86 [-0.5784961,  8.297103]80
#> # ... with 1 more variable: '95%' <hilo>, and abbreviated variable name
#> #   1: Year_Month

```

Forecasts by ETS <ETS(A,A,A)> and ARIMA model <ARIMA(0,1,2)(0,1,2)[12]> w/ Prediction Interval



```

#> # A tibble: 180 x 5
#> # Groups:   City, Measure, .model, Year [60]
#>   City      Measure      .model Year Year_avg
#>   <chr>    <fct>      <chr> <dbl> <dbl>
#> 1 Giessen Temperature arima    2023     1.89
#> 2 Giessen Temperature arima    2023     2.58
#> 3 Giessen Temperature arima    2023     6.27
#> 4 Giessen Temperature arima    2024     2.06
#> 5 Giessen Temperature arima    2024     2.65

```

```

#> 6 Giessen Temperature arima 2024 6.28
#> 7 Giessen Temperature arima 2025 2.08
#> 8 Giessen Temperature arima 2025 2.67
#> 9 Giessen Temperature arima 2025 6.30
#> 10 Giessen Temperature arima 2026 2.10
#> # ... with 170 more rows
#> # A tibble: 180 x 5
#> # Groups:   City, Measure, .model, Year [60]
#>   City Measure .model Year Year_avg
#>   <chr> <fct> <chr> <dbl> <dbl>
#> 1 Giessen Temperature arima 2023 10.3
#> 2 Giessen Temperature arima 2023 5.76
#> 3 Giessen Temperature arima 2023 3.05
#> 4 Giessen Temperature arima 2024 10.4
#> 5 Giessen Temperature arima 2024 5.81
#> 6 Giessen Temperature arima 2024 3.06
#> 7 Giessen Temperature arima 2025 10.4
#> 8 Giessen Temperature arima 2025 5.83
#> 9 Giessen Temperature arima 2025 3.08
#> 10 Giessen Temperature arima 2026 10.4
#> # ... with 170 more rows

```

4.0.3 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 Giessen Temperature arima 26.7 0.638
#> 2 Giessen Temperature ets 36.0 0.208

```

5 Yearly Data Forecasts with ARIMA and ETS

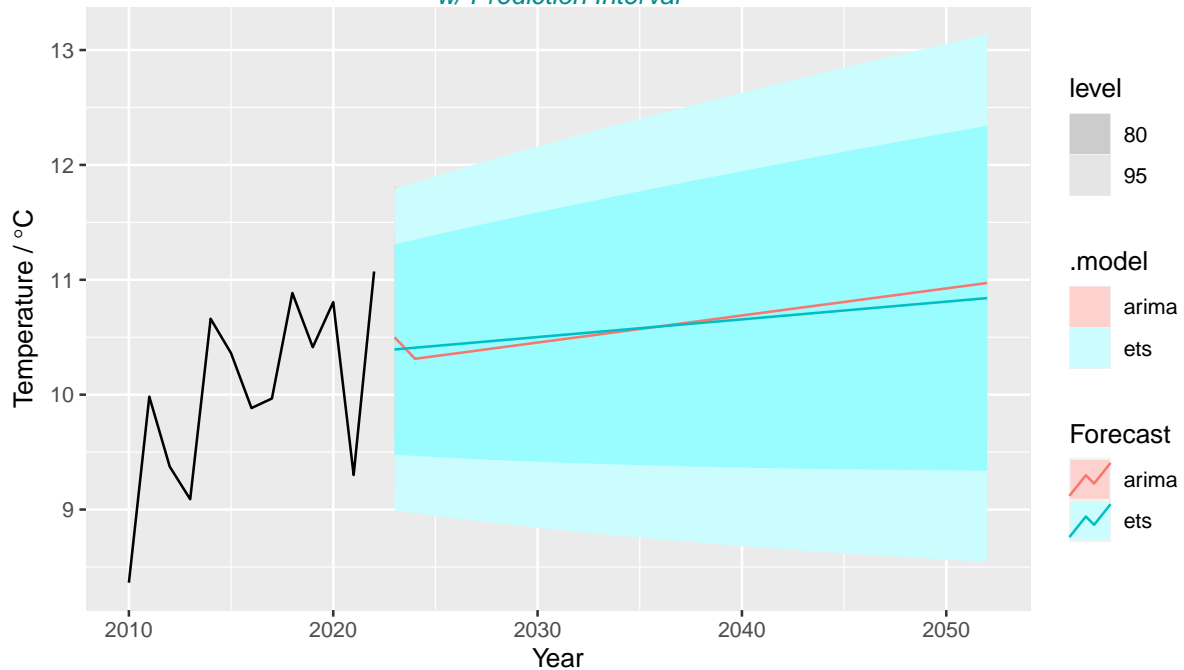
For yearly data the seasonal monthly data are replaced by the yearly average data. Therefore the seasonal component of the ETS and ARIMA model are to be taken out.

The ETS model $\langle ETS(A, A, N) \rangle$ 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 differencing had to be selected $\langle CO_2 \rangle \langle ARIMA(0,2,1) w/ poly \rangle$. For Temperature and Precipitation the same model as for monthly data can be taken by leaving out the seasonal term $\langle ARIMA(0,1,2)w/drift \rangle$.

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

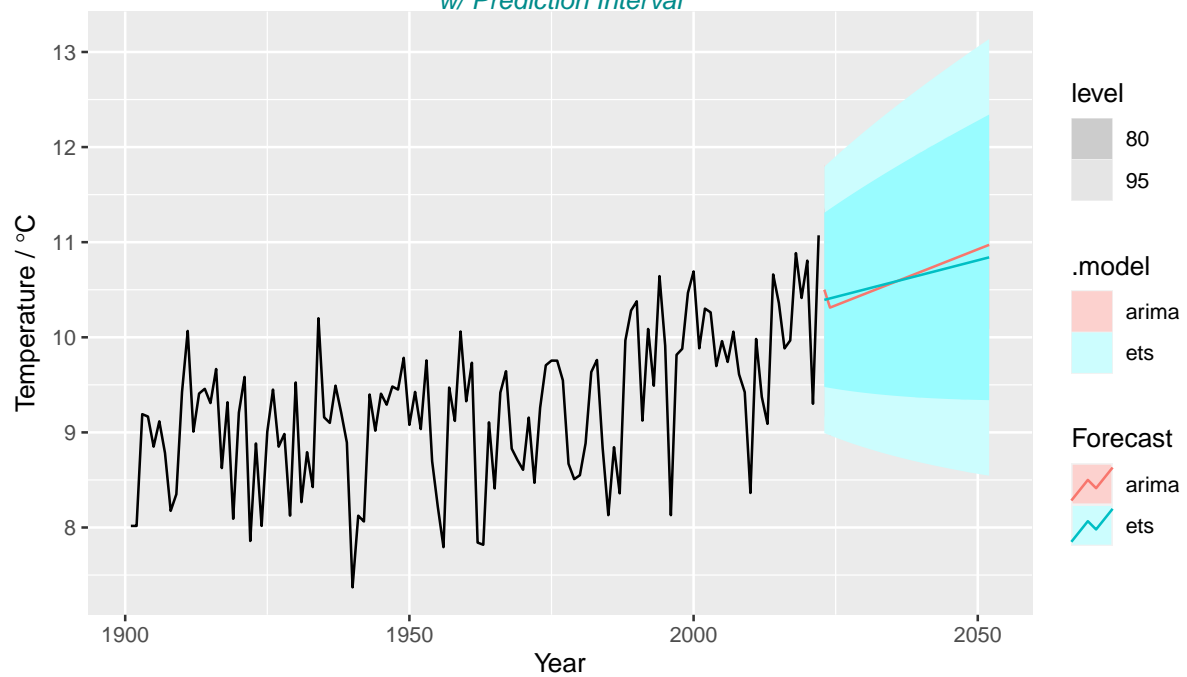
5.0.2 Forecast Plot of selected ETS and ARIMA model

arly Data Forecasts by ETS <ETS(A,A,N)> and ARIMA model <ARIMA(0,1,2) w/ drift>
w/ Prediction Interval



```
#> # A tsibble: 6 x 8 [1Y]
#> # Key:      City, Measure, .model [2]
#> # Groups:   City, Measure, .model [2]
#>   City      Measure      .model   Year   Year_avg .mean      '80%'
#>   <chr>    <fct>        <chr>  <dbl>   <dist> <dbl>    <hilo>
#> 1 Giessen Temperature arima    2023 N(11, 0.45) 10.5 [9.638375, 11.36206]80
#> 2 Giessen Temperature arima    2024 N(10, 0.48) 10.3 [9.428927, 11.19564]80
#> 3 Giessen Temperature arima    2025 N(10, 0.48) 10.3 [9.452491, 11.21920]80
#> 4 Giessen Temperature ets      2023 N(10, 0.51) 10.4 [9.477865, 11.30887]80
#> 5 Giessen Temperature ets      2024 N(10, 0.54) 10.4 [9.467302, 11.35027]80
#> 6 Giessen Temperature ets      2025 N(10, 0.57) 10.4 [9.457416, 11.39100]80
#> # ... with 1 more variable: '95%' <hilo>
#> # A tsibble: 6 x 8 [1Y]
#> # Key:      City, Measure, .model [2]
#> # Groups:   City, Measure, .model [2]
#>   City      Measure      .model   Year   Year_avg .mean      '80%'
#>   <chr>    <fct>        <chr>  <dbl>   <dist> <dbl>    <hilo>
#> 1 Giessen Temperature arima    2050 N(11, 0.48) 10.9 [10.041598, 11.80831]80
#> 2 Giessen Temperature arima    2051 N(11, 0.48) 10.9 [10.065162, 11.83188]80
#> 3 Giessen Temperature arima    2052 N(11, 0.48) 11.0 [10.088726, 11.85544]80
#> 4 Giessen Temperature ets      2050 N(11, 1.3) 10.8 [ 9.341714, 12.27780]80
#> 5 Giessen Temperature ets      2051 N(11, 1.3) 10.8 [ 9.340424, 12.30994]80
#> 6 Giessen Temperature ets      2052 N(11, 1.4) 10.8 [ 9.339307, 12.34190]80
#> # ... with 1 more variable: '95%' <hilo>
```

Early Data Forecasts by ETS <ETS(A,A,N)> and ARIMA model <ARIMA(0,1,2) w/ drift> w/ Prediction Interval



5.0.3 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 Giessen Temperature arima     33.1    0.318
#> 2 Giessen Temperature ets       31.2    0.406
```

6 Backup