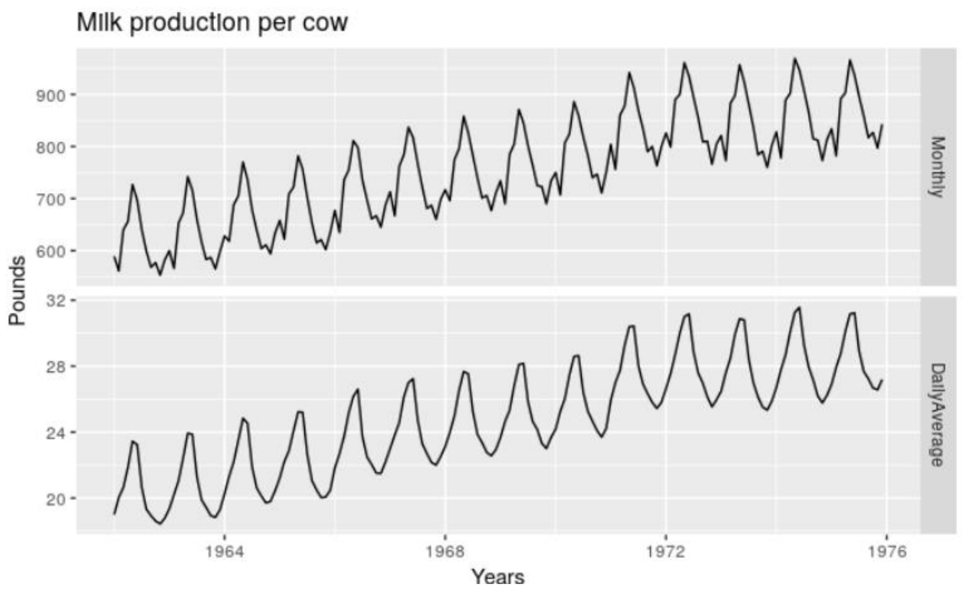
M7
Predictive Analytics

ENİS KAYIŞ

Transformations

- Calendar Adjustments
  - E.g.: Different number of days in each month
  - Could use average numbers instead.



- Population Adjustments (could use per capita)
- Inflation Adjustments (could use price index)

Transformations

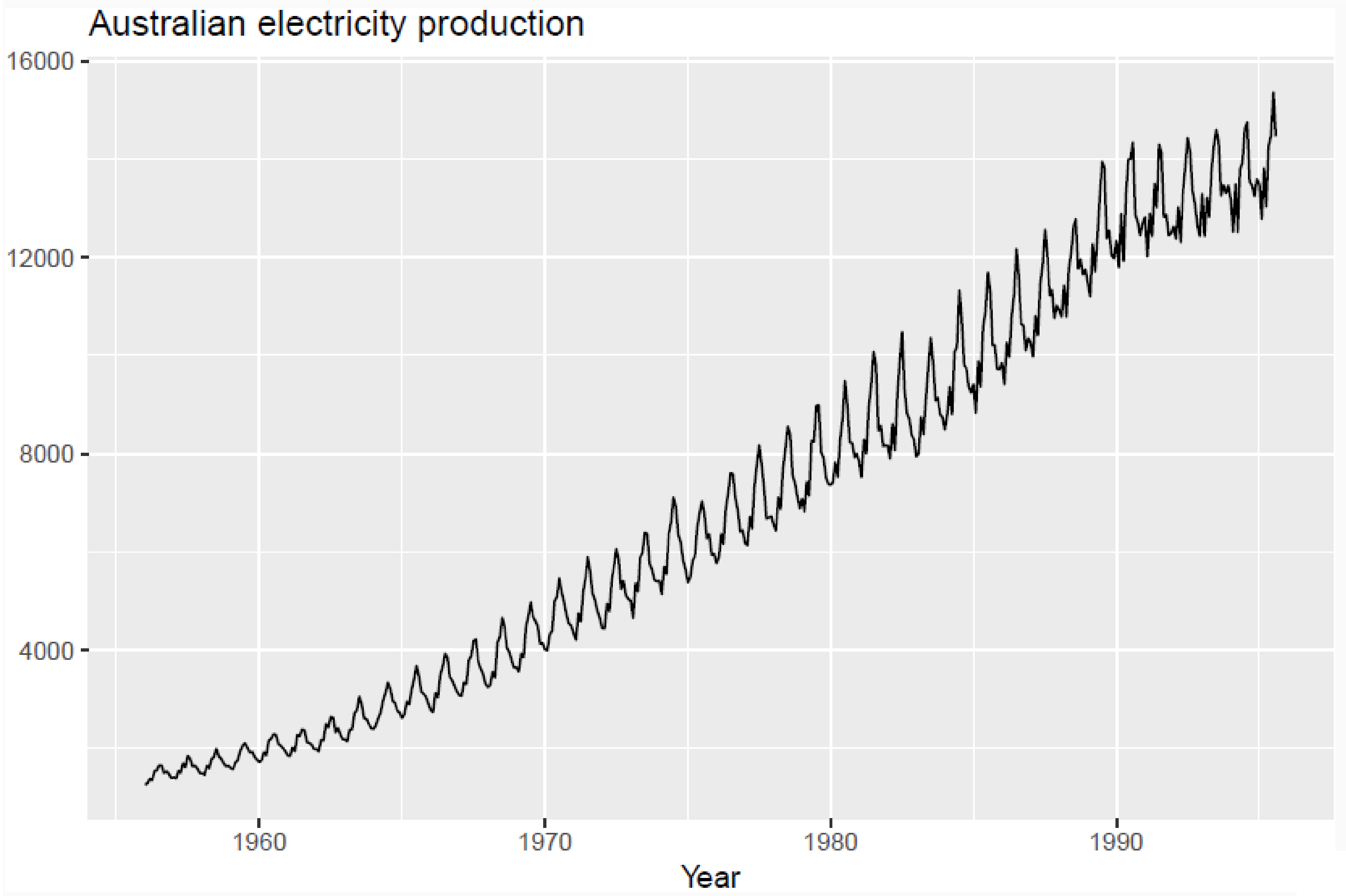
- Box-Cox Transformations
  - If the data show different variation at different levels of the series, then a transformation can be useful.
  - Denote original observations as $y_1, \ldots y_n$ and transformed observations as $w_1, \ldots w_n$.

Mathematical transformations for stabilizing variation

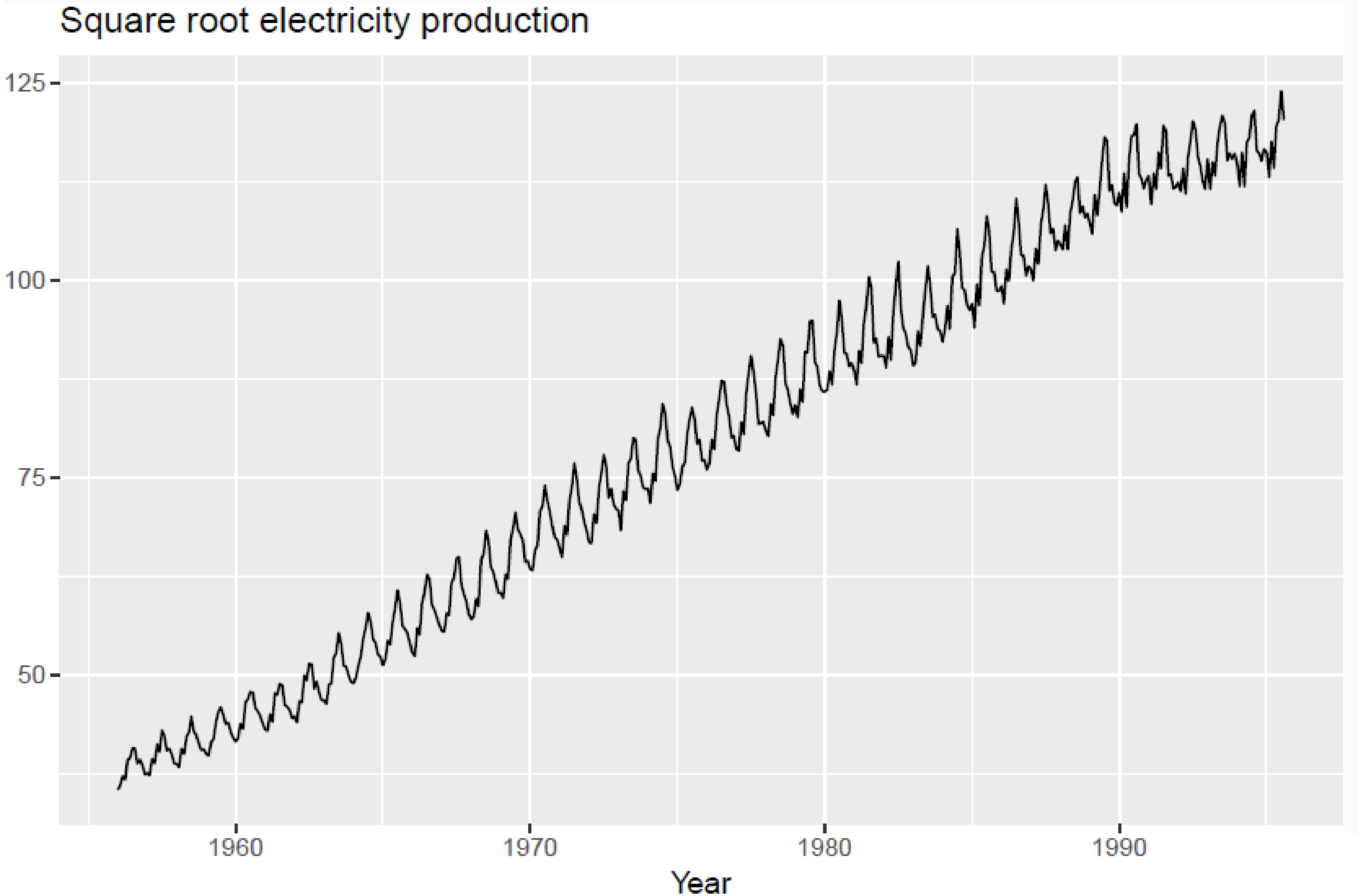
Square root	$w_t = \sqrt{y_t}$	$\downarrow$
Cube root	$w_t = \sqrt[3]{y_t}$	Increasing
Logarithm	$w_t = \log(y_t)$	strength

  - Logarithms, in particular, are useful because they are more interpretable: changes in a log value are **relative (percent) changes on the original scale.**

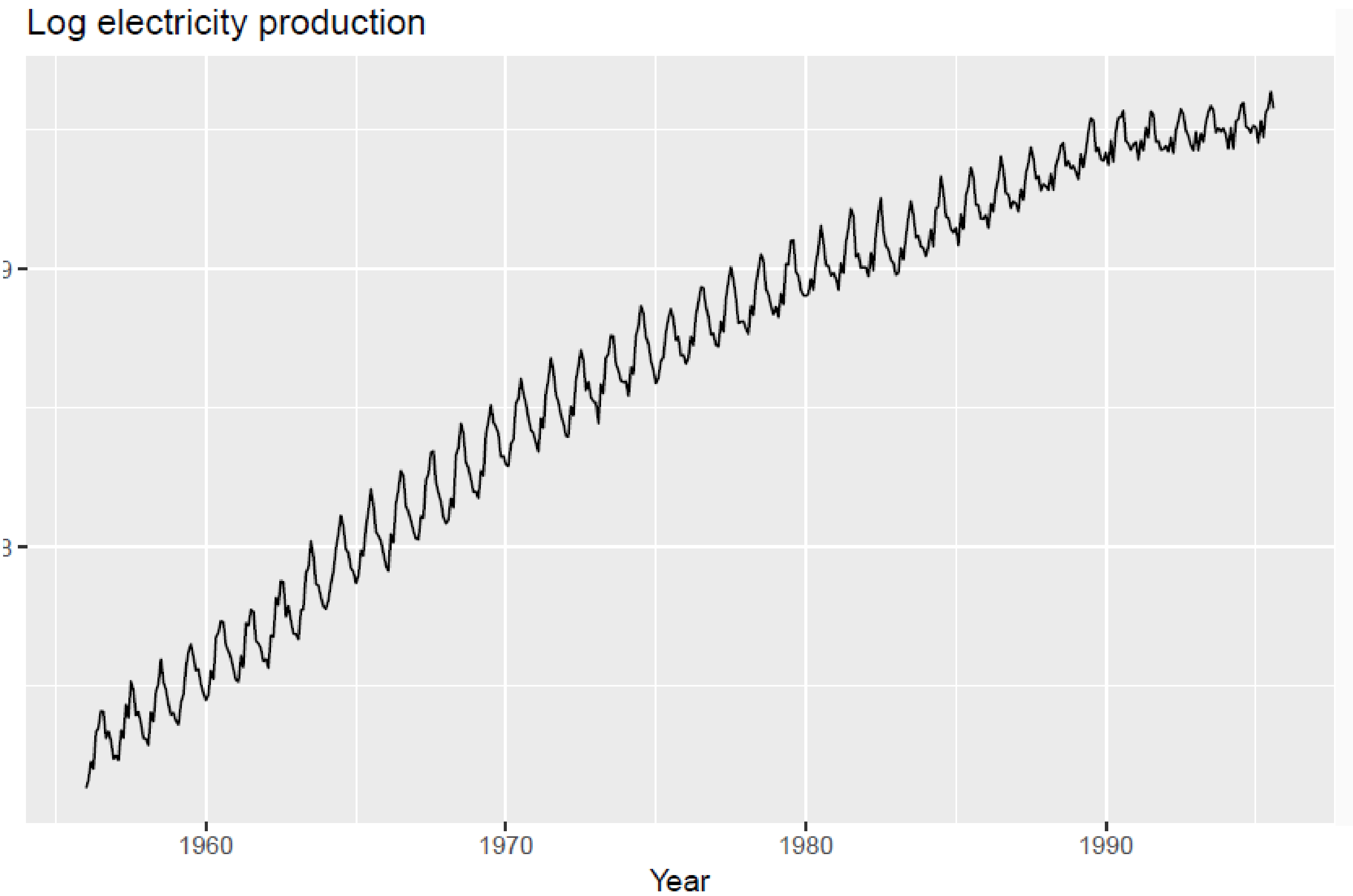
Transformations



Transformations



Transformations



Transformations

- Each of these transformations is close to a member of the family of **Box-Cox transformations**:

$$w_t = \begin{cases} \log(y_t), & \lambda = 0; \\ (y_t^\lambda - 1)/\lambda, & \lambda \neq 0. \end{cases}$$

- y_t^λ for λ close to zero behaves like logs.
- If some $y_t = 0$, then must have $\lambda > 0$
- If some $y_t < 0$, no power transformation is possible unless all y_t adjusted by **adding a constant to all values**.
- Simple values of λ are easier to explain.
- Results are relatively insensitive to λ.
- Often no transformation ($\lambda = 1$) is needed.
- Transformation make little difference to the forecasts but can have very large effect on prediction interval.

Transformations

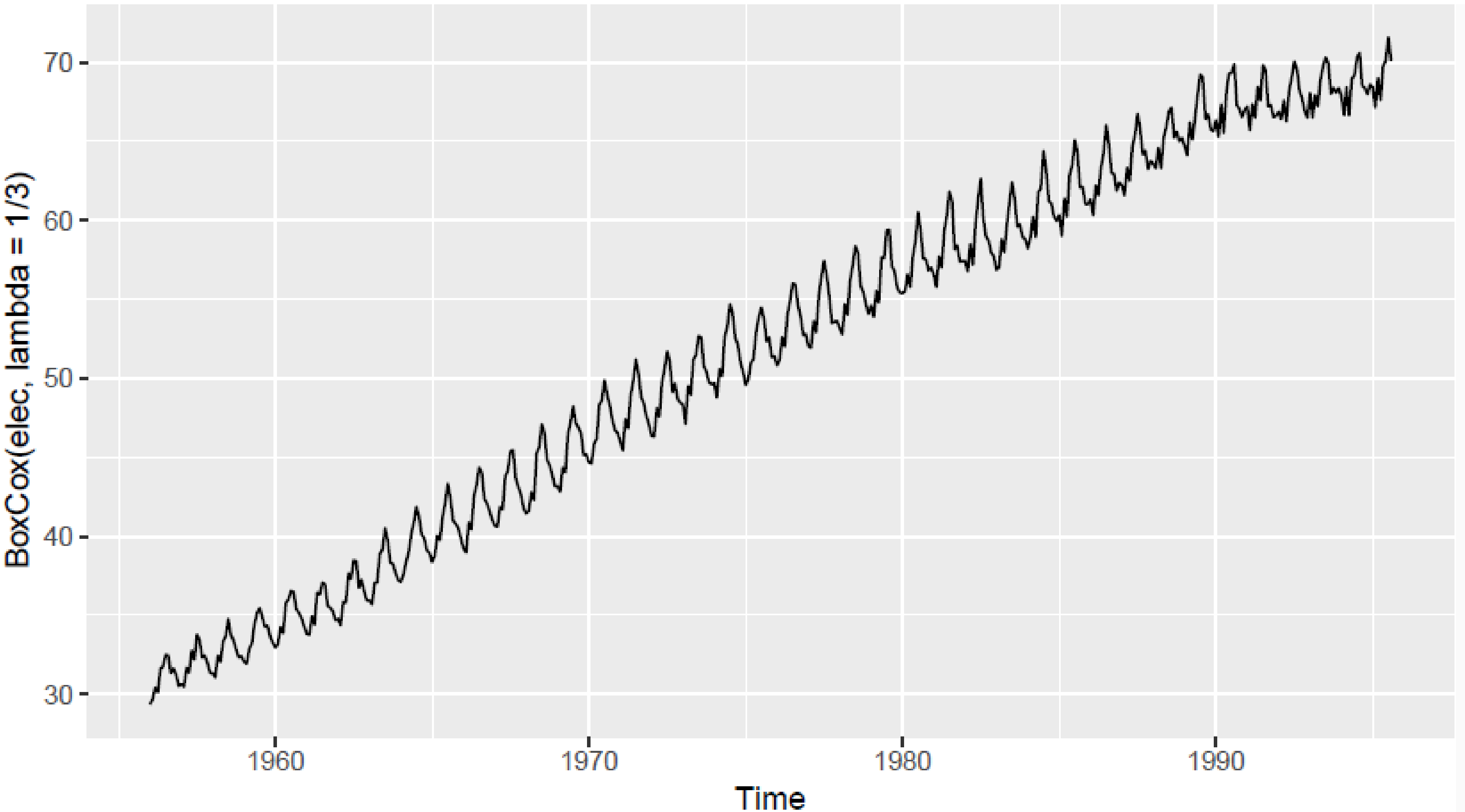
- Automated Box-Cox transformations

```r
(BoxCox.lambda(elec))
```

```
## [1] 0.2654076
```

- This attempts to balance the seasonal fluctuations and random variation across the series.
- Always check the results.
- A low value of λ can give extremely large prediction intervals.

Transformations



Transformations

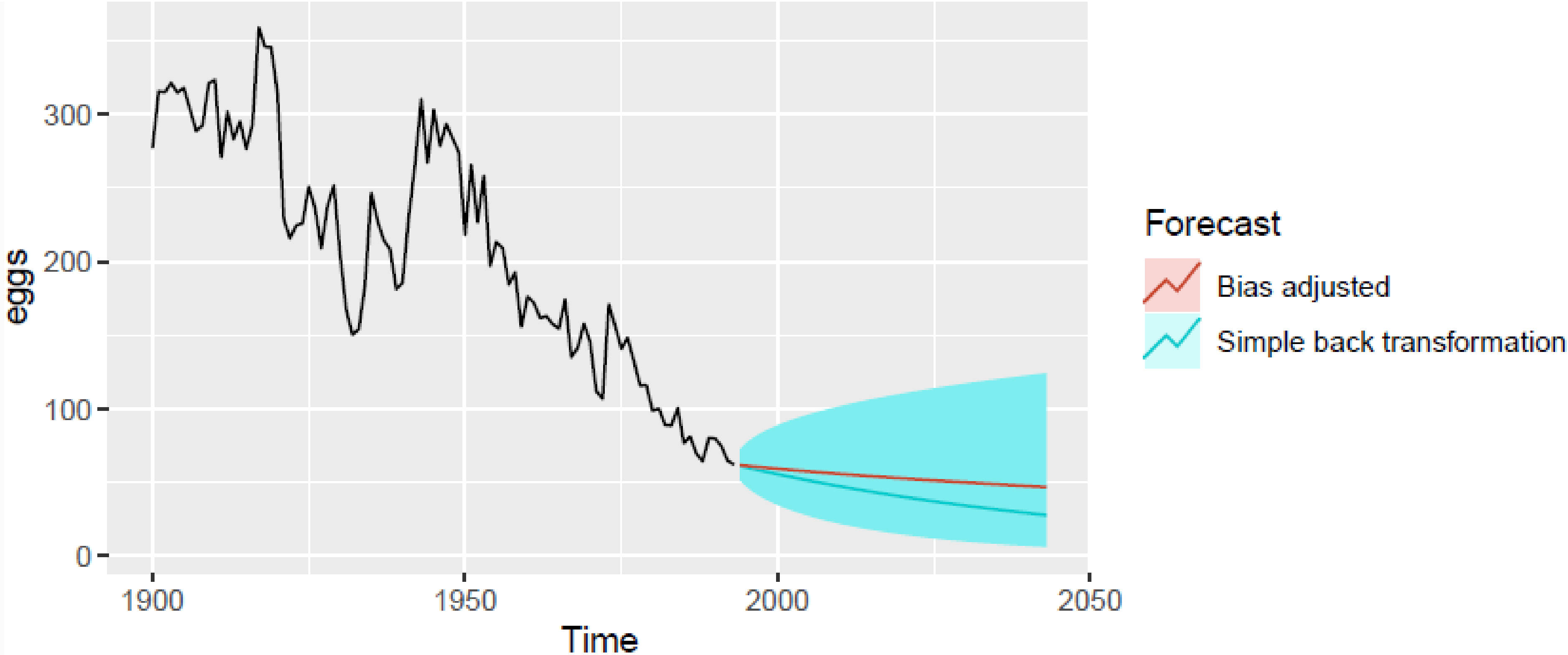
- Back-transformation
  - We must reverse the transformation (or *back-transform*) to obtain forecasts on the original scale. The reverse Box-Cox transformations are given by

$$y_t = \begin{cases} \exp(w_t), & \lambda = 0; \\ (\lambda W_t + 1)^{1/\lambda}, & \lambda \neq 0. \end{cases}$$

  - Bias Adjustment:
    - Back-transformed point forecasts are medians.
    - Need to use the following formula to get mean forecast

$$y_t = \begin{cases} \exp(w_t)\left[1 + \frac{\sigma_h^2}{2}\right] & \text{if } \lambda = 0; \\ (\lambda w_t + 1)^{1/\lambda}\left[1 + \frac{\sigma_h^2(1-\lambda)}{2(\lambda w_t+1)^2}\right] & \text{otherwise;} \end{cases}$$

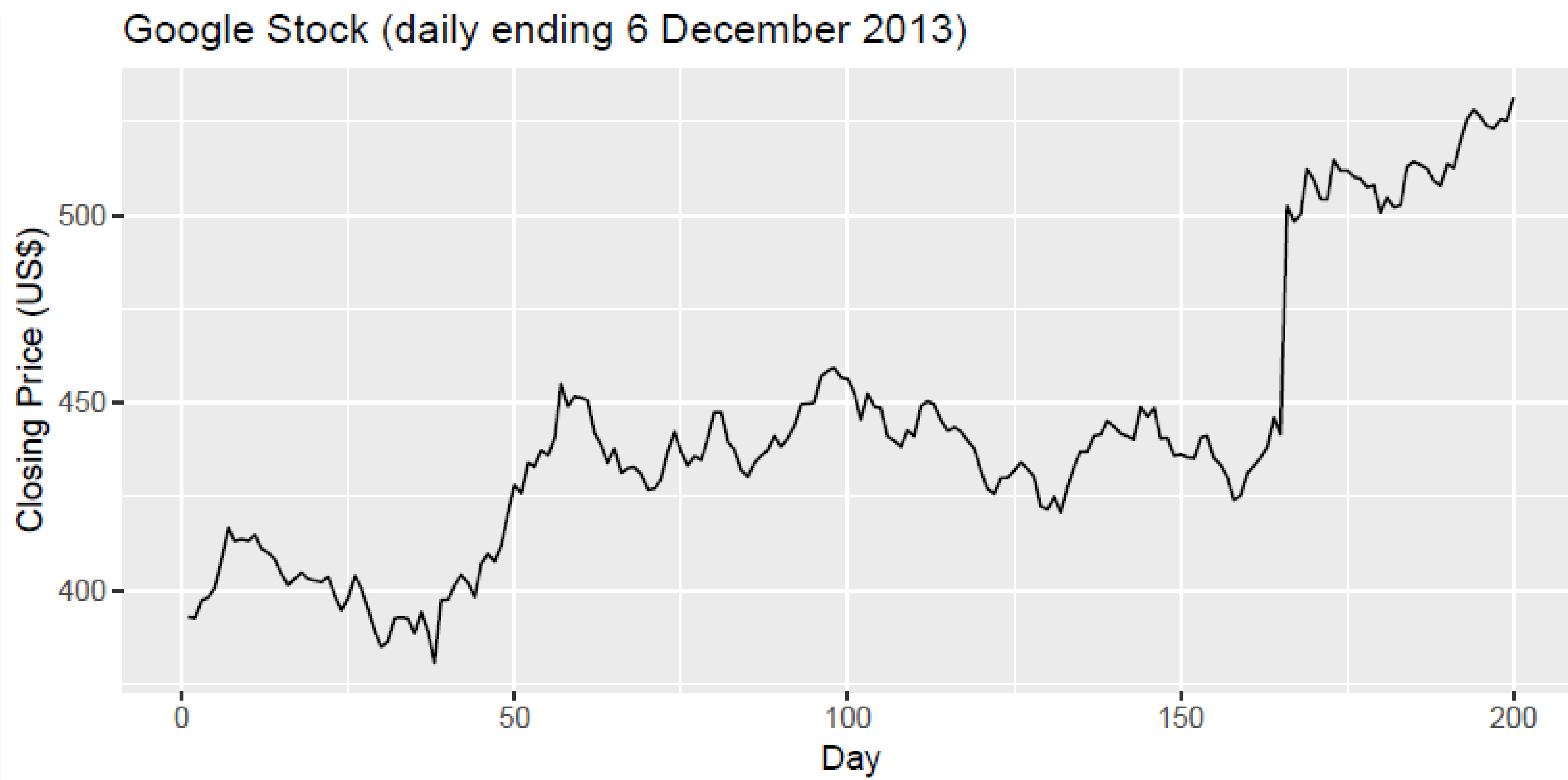
Transformations



Residual Diagnostics

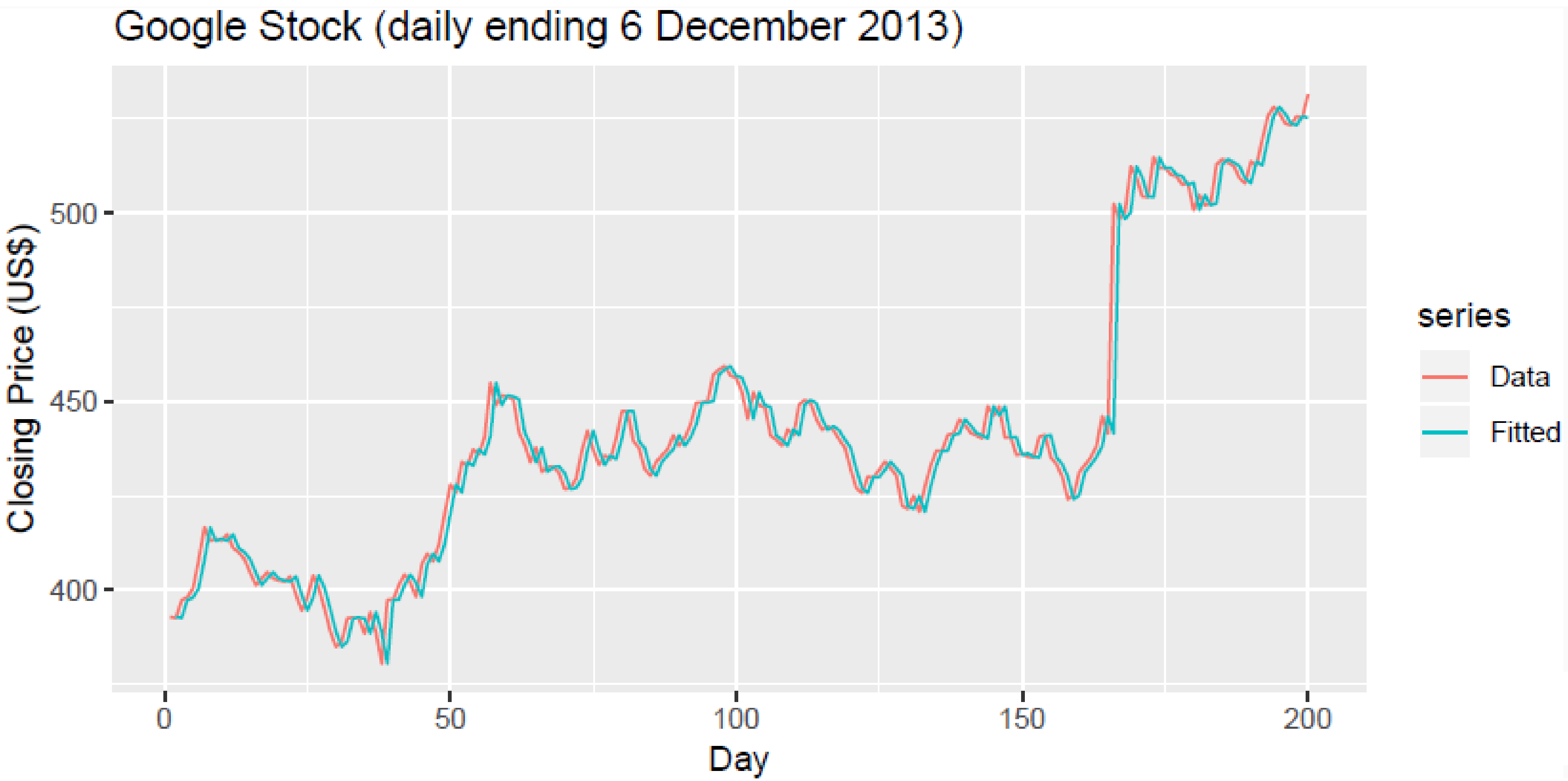
- $\hat{y}_{t|t-1}$ is the forecast of y_t based on observations $y_1, \ldots y_{t-1}$: (fitted values).
- Sometimes drop the subscript: $\hat{y}_t \equiv \hat{y}_{t|t-1}$.
- **Residuals in forecasting:** difference between observed value and its fitted value: $e_t = y_t - \hat{y}_{t|t-1}$
- Assumptions:
  1. $\{e_t\}$ are uncorrelated. If they aren't, then information left in residuals that should be used in computing forecasts.
  2. $\{e_t\}$ have mean zero. If they don't, then forecasts are biased.
  3. $\{e_t\}$ have constant variance.
  4. $\{e_t\}$ are normally distributed
  - 3-4 are only needed for prediction intervals

Residual Diagnostics

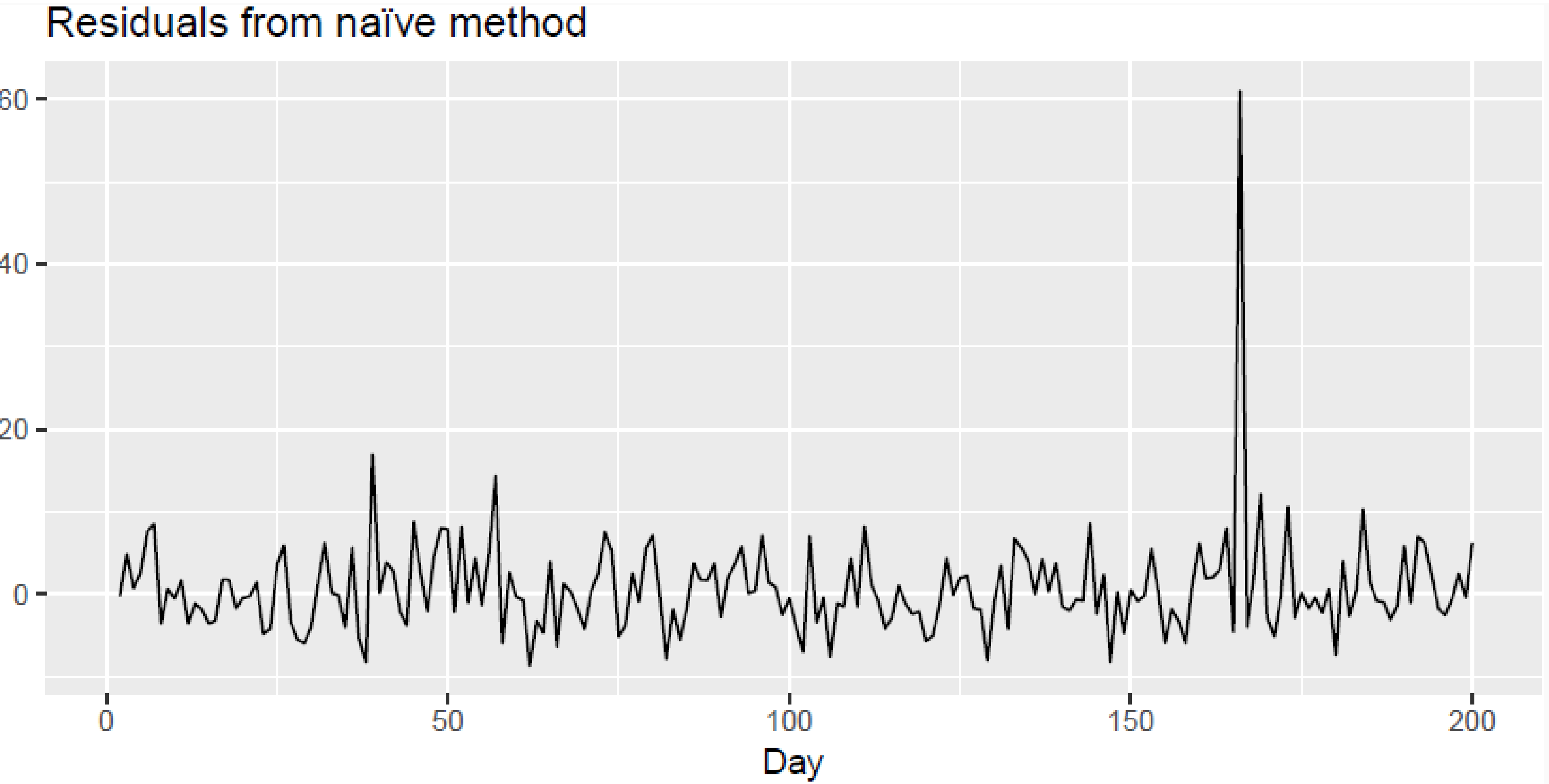


Residual Diagnostics

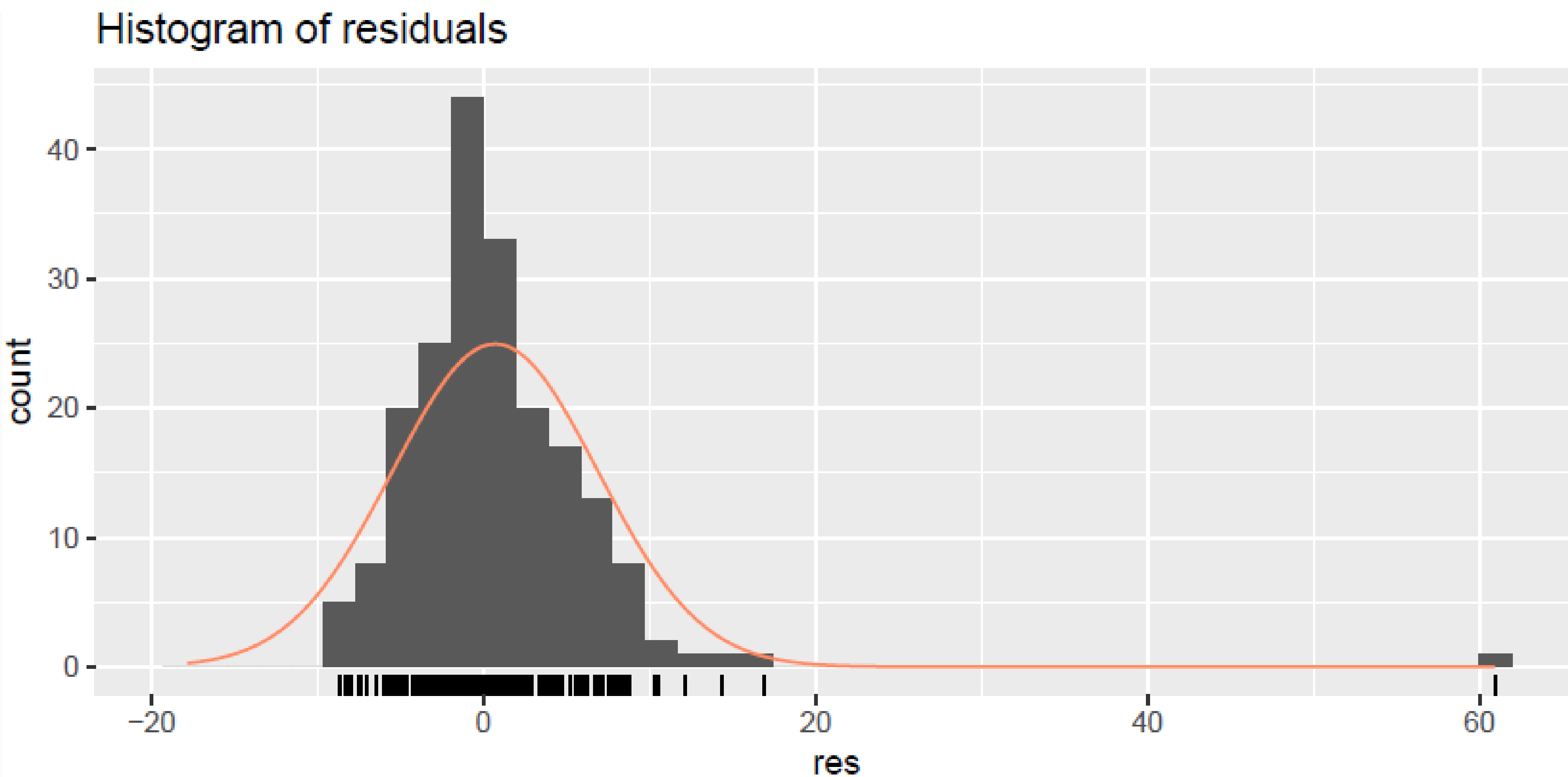
- Let's use naïve forecast



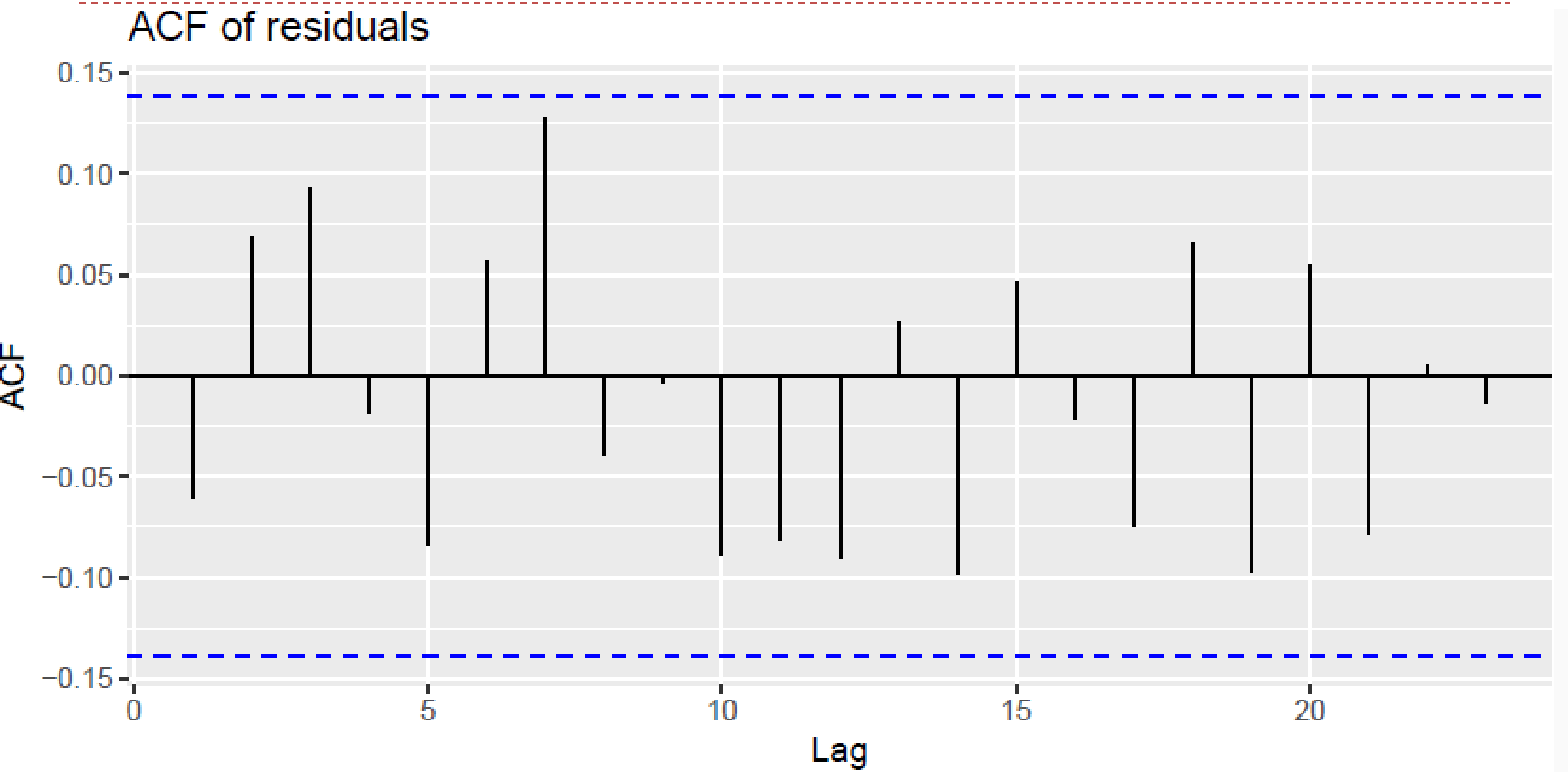
Residual Diagnostics



Residual Diagnostics



Residual Diagnostics



Residual Diagnostics

- We assume that the residuals are white noise (uncorrelated, mean zero, constant variance). If they aren't, then there is information left in the residuals that should be used in computing forecasts.
- So a standard residual diagnostic is to check the ACF of the residuals of a forecasting method.
- We *expect* these to look like white noise.
- Remember to conduct Portmanteau tests as well

```r
# lag=h and fitdf=K
Box.test(res, lag=10, fitdf=0, type="Lj")
```

```
##
##  Box-Ljung test
##
## data:  res
## X-squared = 11.031, df = 10, p-value =
## 0.3551
```

Prediction Intervals (PIs)

- A forecast $\hat{y}_{T+h|T}$ is (usually) the mean of the conditional distribution $y_{T+h} \mid y_1, \dots y_T$.
- A prediction interval gives a region within which we expect y_{T+h} to lie with a specified probability.
- Assuming forecast errors are normally distributed, then a 95% PI is

$$\hat{y}_{T+h|T} \pm 1.96\hat{\sigma}_h$$

where $\hat{\sigma}_h$ is the s.d. of the residuals for the h-step forecasts.

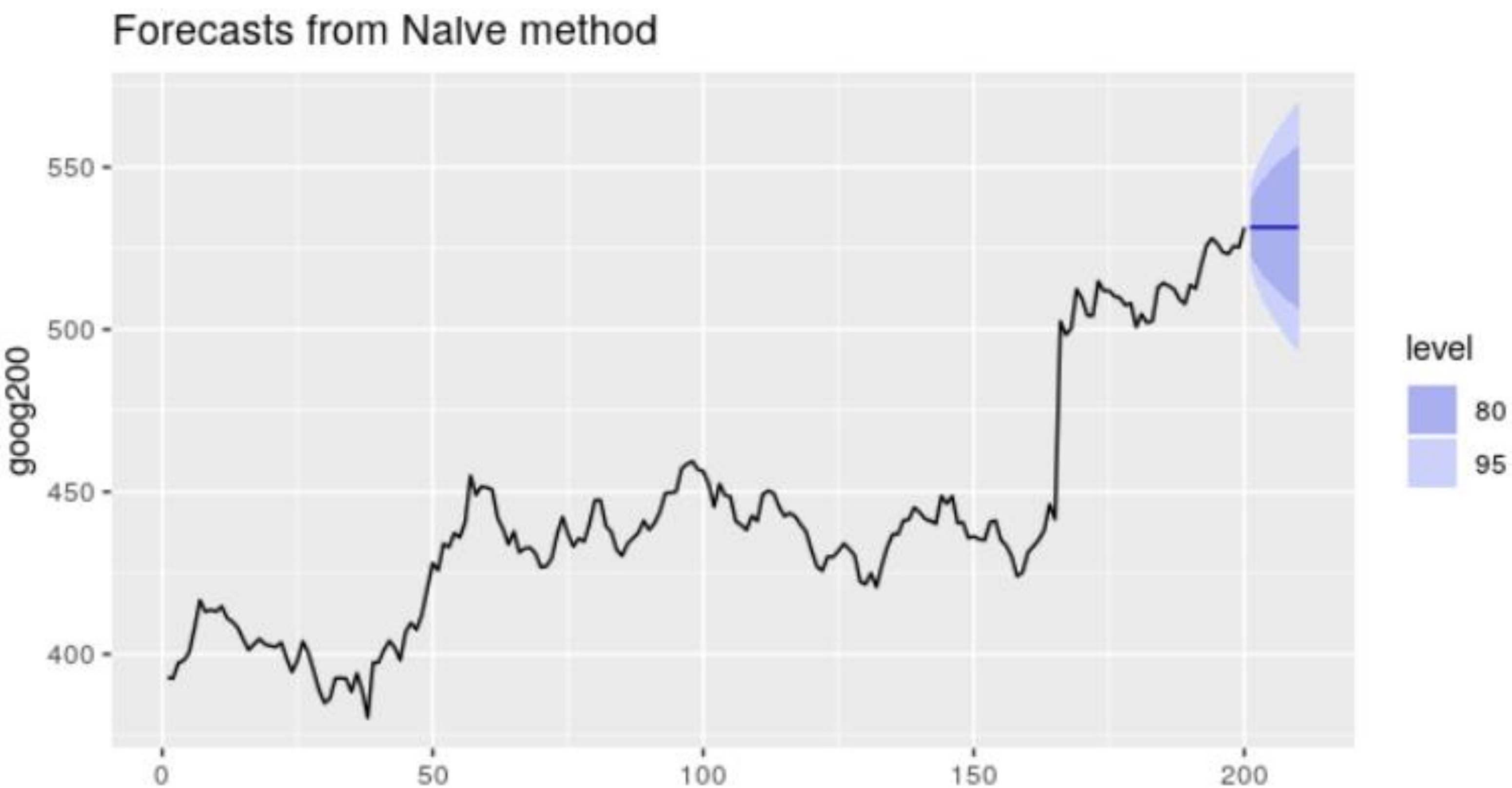
- Point forecasts are often useless without prediction intervals.
- Multi-step forecasts for time series require a more sophisticated approach (with PI getting wider as the forecast horizon increases).

Prediction Intervals

```r
naive(goog200, level=95)
```

```
##     Point Forecast    Lo 95    Hi 95
## 201       531.4783 519.3104 543.6461
## 202       531.4783 514.2703 548.6862
## 203       531.4783 510.4029 552.5536
## 204       531.4783 507.1425 555.8140
## 205       531.4783 504.2701 558.6865
## 206       531.4783 501.6732 561.2833
## 207       531.4783 499.2851 563.6714
```

Prediction Intervals



Prediction Intervals

- When a normal distribution for the forecast errors is an unreasonable assumption, one alternative is to use bootstrapping, which only assumes that the forecast errors are uncorrelated.
- We can use $y_{T+1} = \hat{y}_{T+1|T} + e_{T+1}$ where $\hat{y}_{T+1|T}$ is the forecast and e_{T+1} is unknown but could be replaced by a sampled value from past errors.
- Next, we could use $y_{T+2} = \hat{y}_{T+2|T+1} + e_{T+2}$ similarly.
- Continuing in this fashion, we obtain many possible futures and compute prediction intervals by percentiles for each forecast horizon.