Using Simple Time Series Models

Martin Burger
R PROGRAMMING TUTOR
www.r-tutorials.com

Simple Models

Methods mean, naïve and drift

Analyzing random time series data

Model comparison

Checking the model quality

Demo

Primitive forecast methods

Random data

Advanced models for data with patterns

Simple models exploit one fact

- Last observation
- Mean
- General trend

Methods with Library Forecast

Naïve method

Returns the last observation as forecast value – naive()

Seasonal naïve method

Returns the last observation of the seasonal stage – snaive()

Mean method

Returns the mean as forecast value – meanf()

Drift method

Carries the change over first to last observation into the future – rwf()

```r
set.seed(50)


testts =
    ts(
        rnorm(300),
        start = c(1919,1),
        frequency = 4)
```

- Generating the data
- Set seed to make results reproducible
- Time series object
- Time series function
- 300 randomly distributed numbers
- Starts at Q1 1919
- Quarterly data

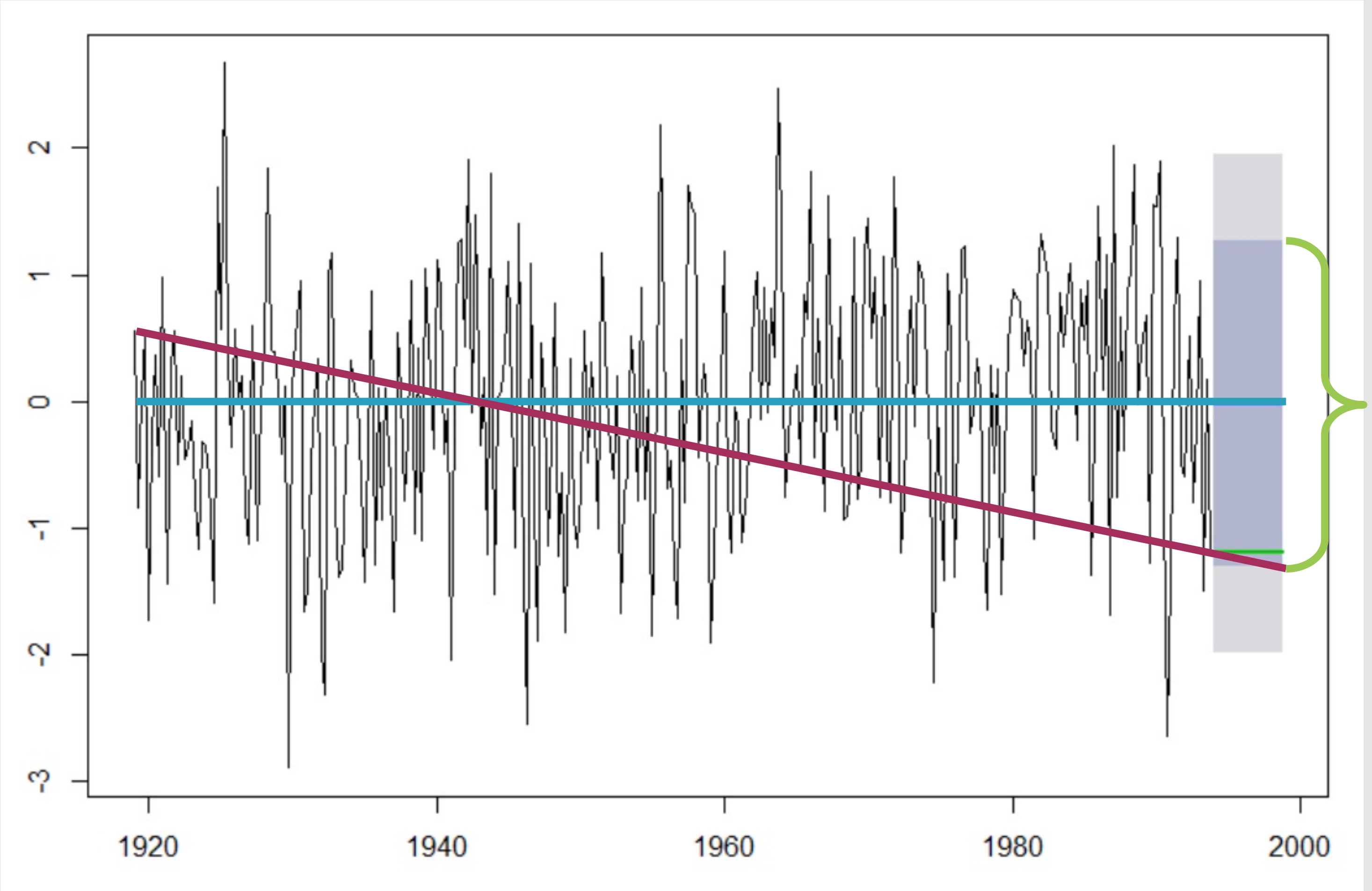
```r
library(forecast)

meanmodel =
    meanf(testts, h = 20)

naivemodel =
    naive(testts, h = 20)

driftmodel =
    rwf(testts, h = 20,
        drift = T)
```

- Setting up three models using tools from the library ‘forecast’
- Models: mean, naïve and drift
- Models use ‘testts’ as data
- Models forecast 20 observations
- Each object contains the original data and 20 forecast values

```r
plot(meanmodel, main = "")

lines(naivemodel$mean,
      col = 123, lwd = 2)

lines(driftmodel$mean,
      col = 22, lwd = 2)

legend("topleft", lty = 1,
       cex = 0.5, col = c(4,      123, 22),
       legend = c("Mean method",
       "Naïve method", "Drift   method"))
```

- Plots the mean model (original data and forecasted values) without a header
- Adds forecasted values of the naïve model to the plot
- Adds forecasted values of the drift model to the plot
- Adds a legend to the plot

Basic Methods on Random Data



Demo

Comparing forecasting models

Forecast accuracy

Error indicators:

- Mean Absolute Error - MAE
- Root Mean Squared Error – RMSE
- Mean Absolute Scale Error – MASE
- Mean Absolute Percentage Error – MAPE

$$MAE = \frac{\sum_{i=1}^{n} |e_i|}{n} = \frac{\sum_{i=1}^{n} |y_i - \hat{y}_i|}{n}$$

y_i	$\hat{y}_i$	$\lvert y_i - \hat{y}_i \rvert = \lvert e_i \rvert$
2.45	2.63	$\lvert -0.18 \rvert$
3.12	2.99	$\lvert 0.13 \rvert$
3.06	3.15	$\lvert -0.09 \rvert$
2.63	2.99	$\lvert -0.36 \rvert$
1.89	2.21	$\lvert -0.32 \rvert$
1.12	1.76	$\lvert -0.64 \rvert$
2.56	1.43	$\lvert 1.13 \rvert$

- Scale Dependent Errors: MAE
- The mean of all differences between actual and forecasted absolute values
- $|e_i|$ = absolute value of a forecast error
- $|e_i| = |y_i - \hat{y}_i|$
  - y_i = actual value
  - $\hat{y}_i$ = forecast of y_i
- $\frac{\sum_{i=1}^{n} |e_i|}{n} = \frac{2.85}{7} = 0.407 = MAE$

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{n} e_i^2}{n}} = \sqrt{\frac{\sum_{i=1}^{n} (y_i - \hat{y}_i)^2}{n}}$$

y_i	$\hat{y}_i$	$(y_i - \hat{y}_i)^2 = e_i^2$
2.45	2.63	-0.18^2
3.12	2.99	0.13^2
3.06	3.15	0.09^2
2.63	2.99	-0.36^2
1.89	2.21	-0.32^2
1.12	1.76	-0.64^2
2.56	1.43	1.13^2

- Scale Dependent Errors: RMSE
- Standard deviation of differences between actual and forecasted values
- e_i^2 = squared value of a forecast error
- $e_i^2 = (y_i - \hat{y}_i)^2$
  - y_i = actual value
  - $\hat{y}_i$ = forecast of y_i
- $\sqrt{\frac{\sum_{i=1}^{n} e_i^2}{n}} = \sqrt{\frac{1.976}{7}}$ = 0.531 = RMSE

Scale Dependent Errors

Observed values against their forecast

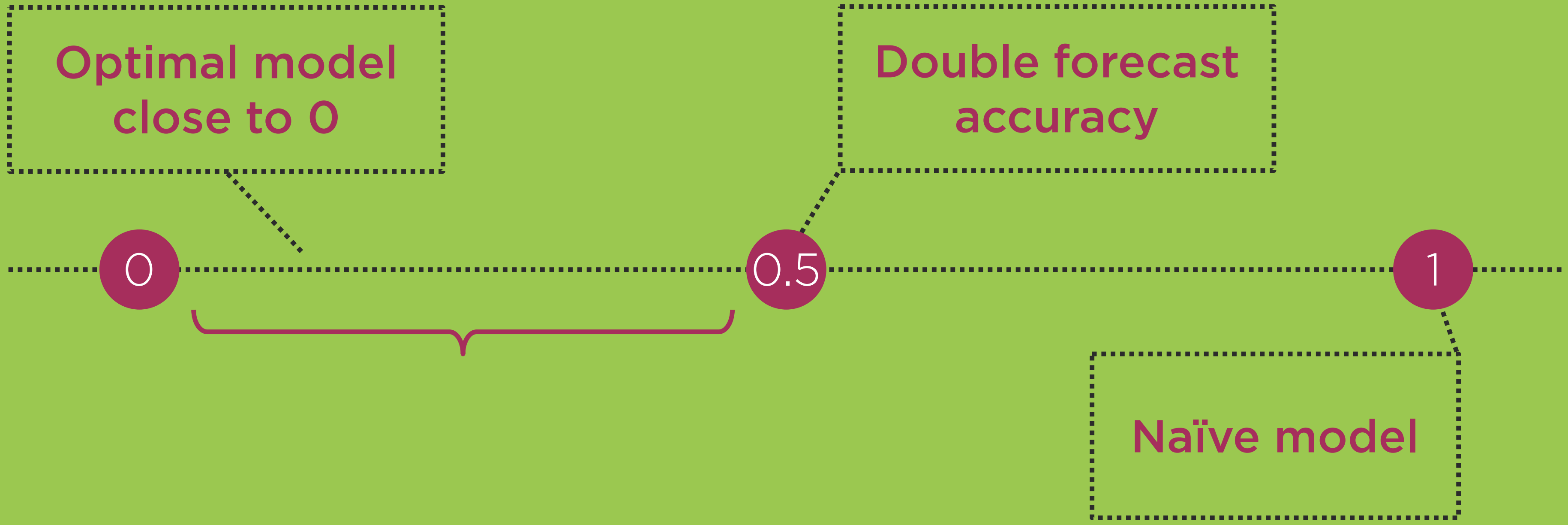
Forecast error against the error of a naïve model

MAE

RMSE

MASE

Mean Absolute Scale Error – MASE

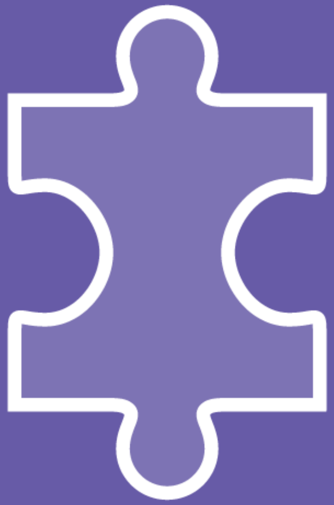


$$\text{MAPE} = \frac{\sum_{i=1}^{n} |p_i|}{n} = \frac{\sum_{i=1}^{n} \left|\frac{100e_i}{y_i}\right|}{n}$$

- Scale Independent Error: MAPE
- Measures the difference of forecast errors and divides it by the actual observation value
- $|p_i|$ = absolute value of forecast error differences

  y_i = actual value
- Does not allow for 0 values
- Puts more weight on extreme values and positive errors
- Use it to compare models on different datasets

Akaike Information Criterion – AIC



A measure of complexity

Complex models get penalized

The lower the AIC value the better

Getting the Forecast Errors in R

Function accuracy()

MAE

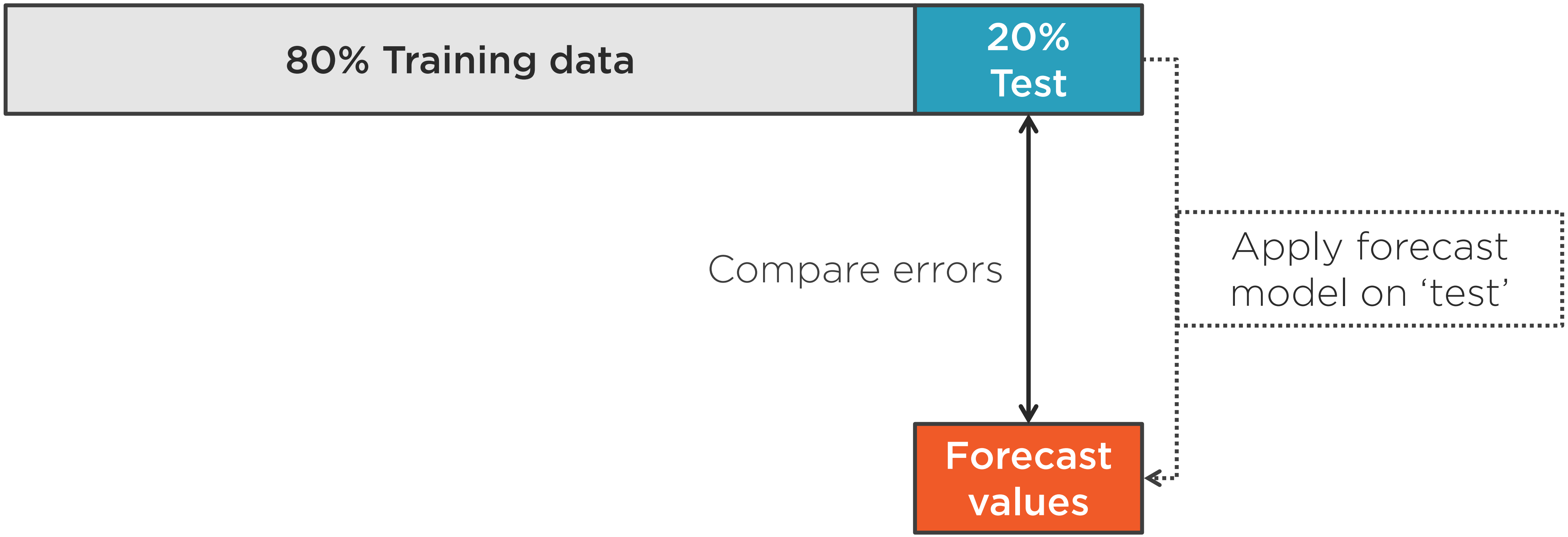
RMSE

MASE

MAPE

AIC

How Model Comparison Is Done



Machine learning and statistical modelling

Test on genuinely new data

- Split at ~80-20%

As simple as possible (AIC)

Well fitting ≠ great forecast power

Demo

Residuals

Indicator of model quality

All the patterns should be in the model, only randomness remains in the residuals

Container of randomness

Ideal Model

Zero mean
(Fix: addition or subtraction)

Constant variance
(Fix: transformation – *not always possible*)

Correlated residuals
(Fix: differencing)

Normal distribution
(Fix: transformation – *not always possible*)

The Mean Model on Random Data

Zero mean
mean()

Normal distribution
hist()

Equal variance
var() and plot()

No autocorrelation
acf()

Simple Models

Models naïve, mean and drift

Random data modeling

Model comparison with accuracy()

Model residuals