时间序列分析与预测

第五讲

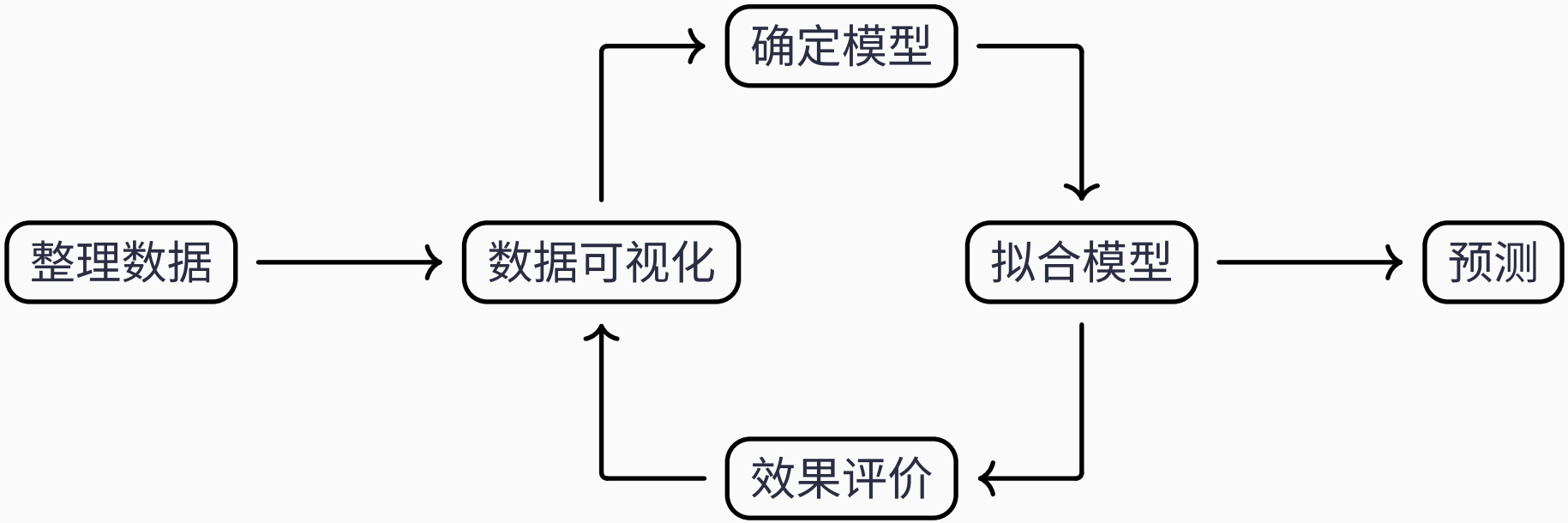
黄嘉平

深圳大学｜中国经济特区研究中心
粤海校区汇文楼办公楼 1510
课程网站 https://huangjp.com/TSAF/

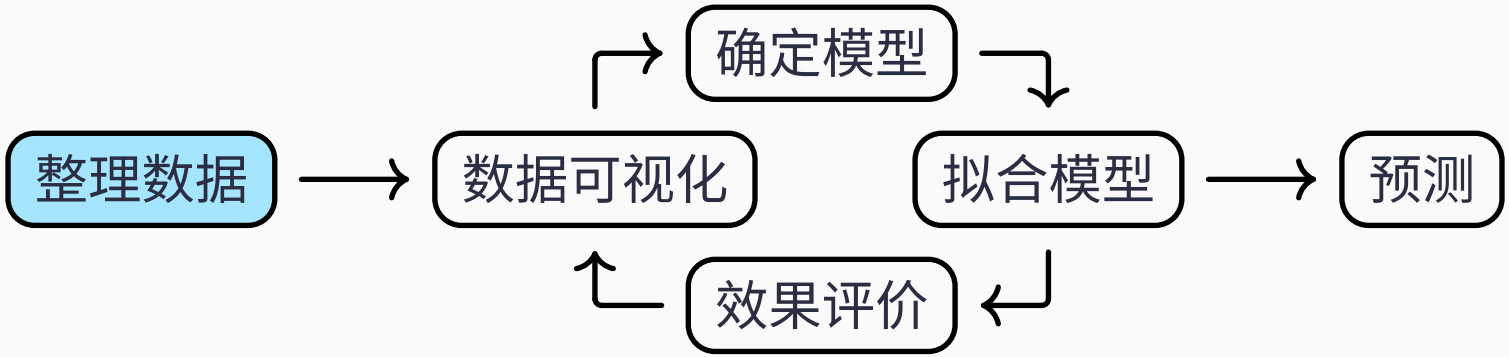
1. 预测的完整流程

1.1. 从获取数据到获得预测结果

从获取时间序列数据，到获得最终预测结果的流程可以分为以下几个步骤



1.2. 数据整理

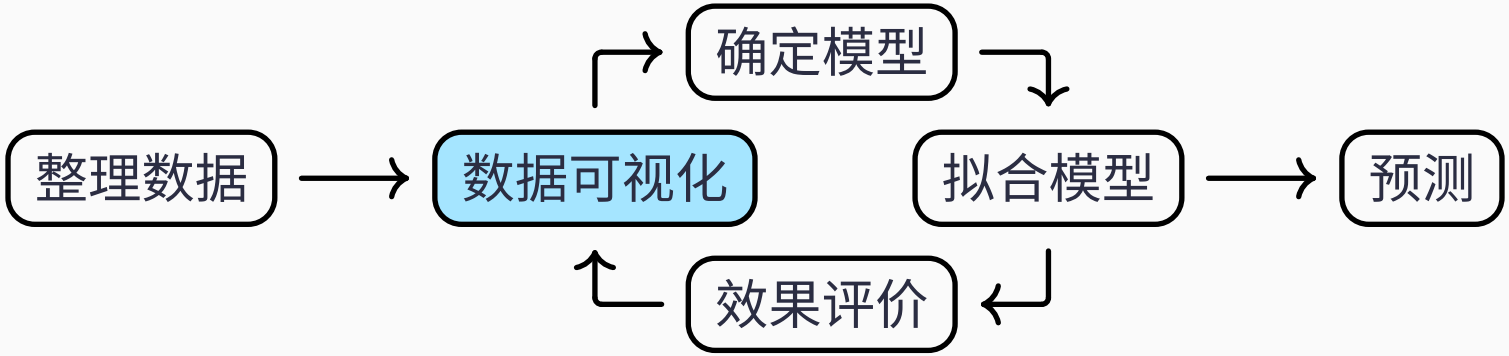


数据整理就是在分析前对数据做必要的处理，包括读取数据、处理缺失值、定义新变量、数据转换等。

下面我们以 `global_economy` 数据集为例展示每个过程

```
gdppc <- global_economy |>
  mutate(GDP_per_capita = GDP / Population)   # 定义新变量"人均GDP"
```

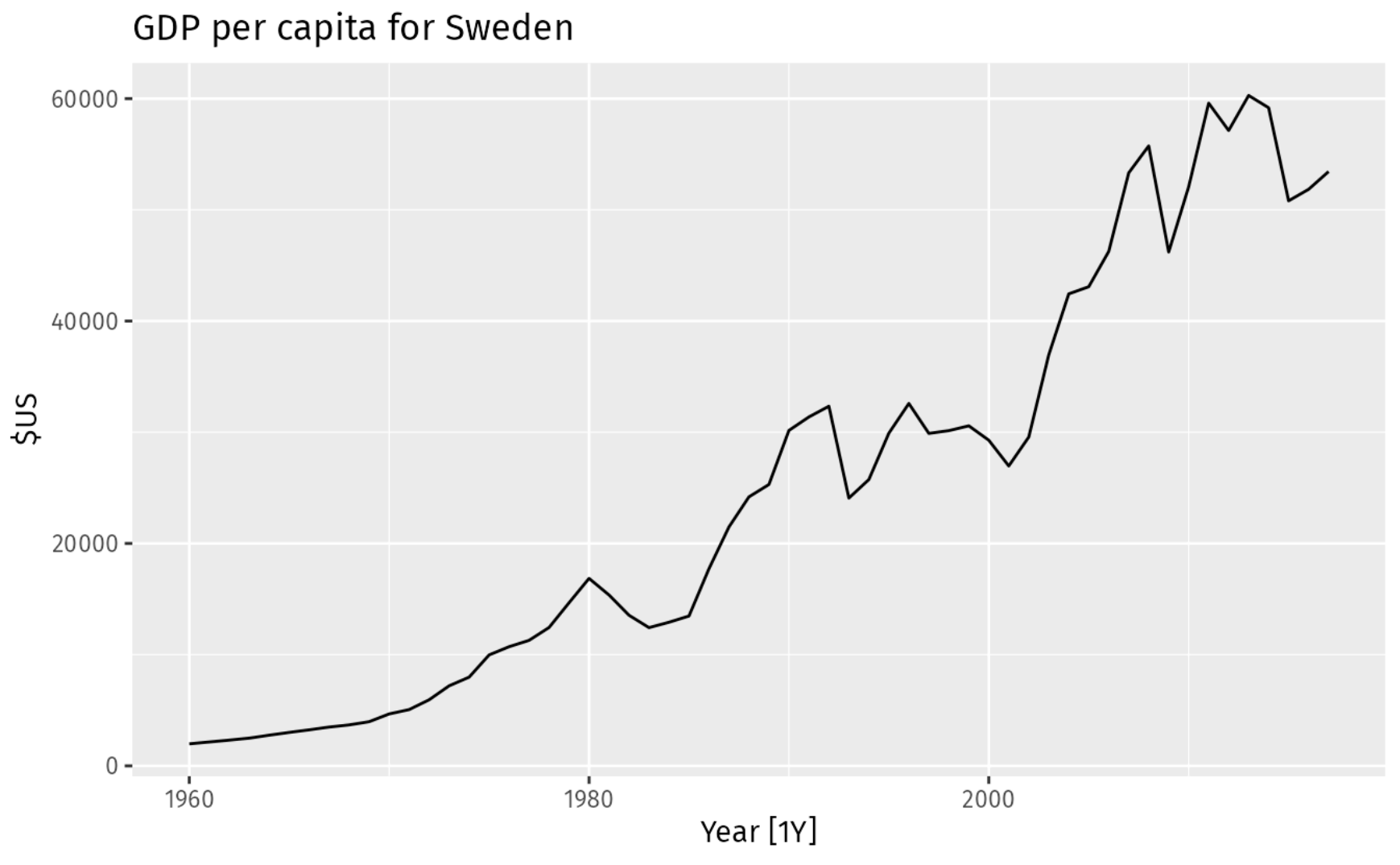
1.3. 数据可视化



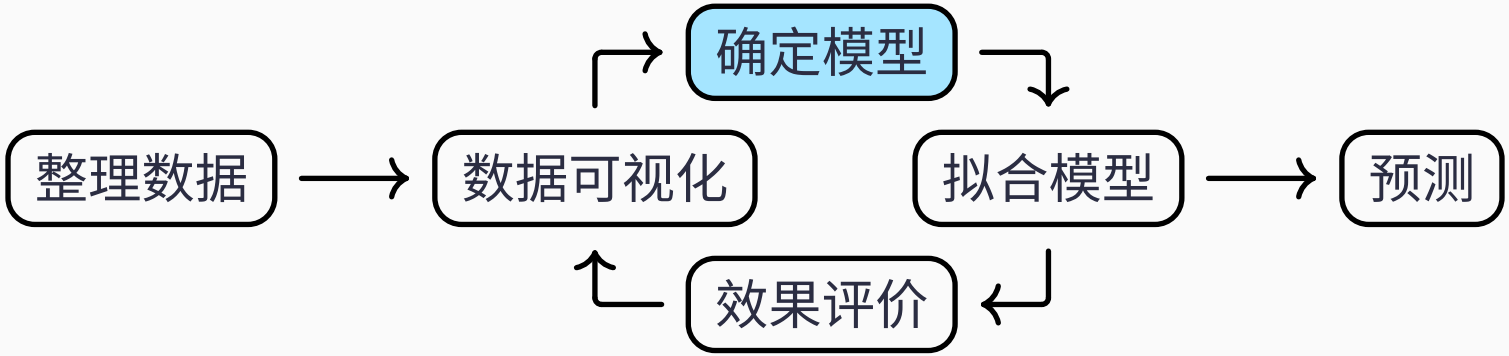
数据可视化的目的是让分析者直观地了解数据特征。

```
gdppc |>
  filter(Country == "Sweden") |>   # 选择瑞典的相关数据
  autoplot(GDP_per_capita) +       # 针对人均GDP进行绘图
  labs(y = "$US", title = "GDP per capita for Sweden")
```

1.3. 数据可视化


GDP per capita for Sweden
60000
40000
20000
0
$US
1960
1980
2000
Year [1Y]

1.4. 确定模型



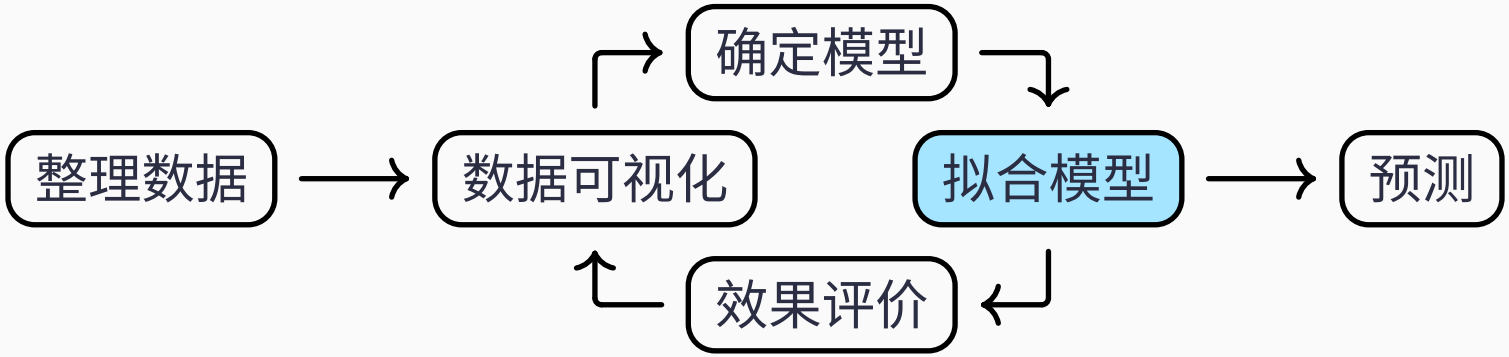
确定模型即建模，通常是从众多已知模型中选择最合适的。fable 包中包含很多模型函数，每个函数都需要通过 `formula(y ~ x)` 指定具体的函数形式。

例如 `TSLM(GDP_per_capita ~ trend())` 将线性趋势模型

$$\text{GDP}_{/\text{cap}} = \beta_0 + \beta_1 t + \text{error}$$

代入 `TSLM()` 函数（时间序列的线性回归模型）。

1.5. 拟合模型



拟合模型（model fitting/estimation）就是利用数据估计模型中的参数值，在数据科学领域也称之为**训练（training）**。

完成模型拟合的函数是 `model()`。下面的命令对每个国家拟合了一个线性趋势模型。

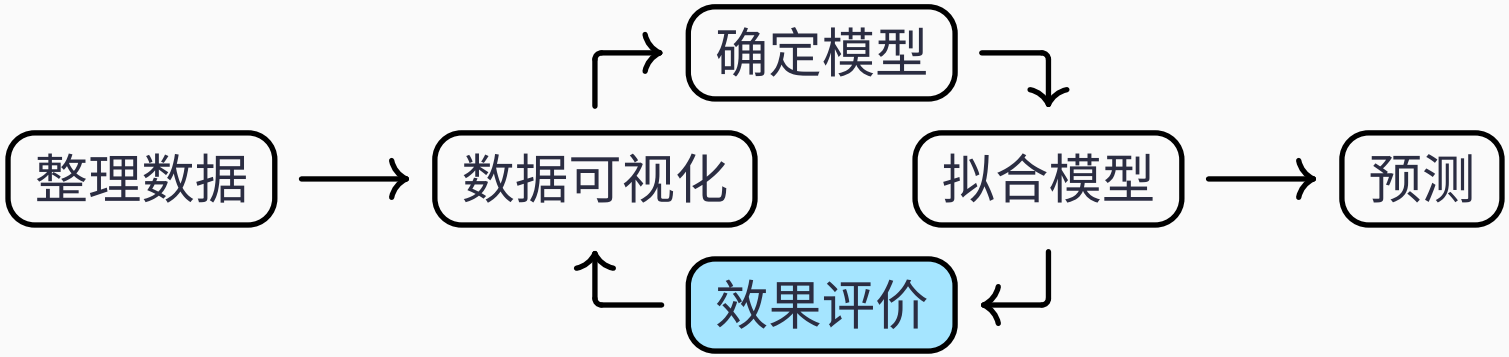
```
fit <- gdppc |>
  model(trend_model = TSLM(GDP_per_capita ~ trend()))
```

1.5. 拟合模型

```
fit
```

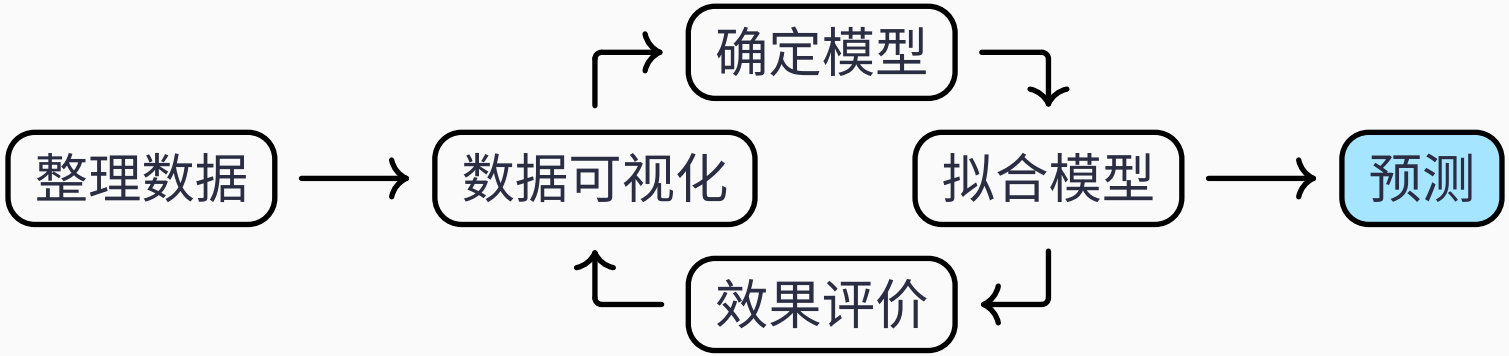
```
# A mable: 263 x 2
# Key:     Country [263]
   Country              trend_model
   <fct>                    <model>
 1 Afghanistan              <TSLM>
 2 Albania                  <TSLM>
 3 Algeria                  <TSLM>
 4 American Samoa           <TSLM>
 5 Andorra                  <TSLM>
 6 Angola                   <TSLM>
 7 Antigua and Barbuda      <TSLM>
 8 Arab World               <TSLM>
 9 Argentina                <TSLM>
10 Armenia                  <TSLM>
# ℹ 253 more rows
# ℹ Use print(n = …) to see more rows
```

1.6. 效果评价



效果评价是确认模型结合结果的好坏，或者从多个备选模型中挑出拟合/预测效果最好的。如果对模型的效果不满意，可以对模型进行调整，然后重复拟合-评价的过程。

1.7. 预测



完成了以上所有步骤后，我们就可以根据拟合好的模型进行**预测**了。这可以通过 `forecast()` 函数完成。通过参数 `h = 10` 或 `h = "2 years"` 指定需要预测的期数或时长。

例如：

```
fit |> forecast(h = "3 years")
```

1.7. 预测

```
fit |> forecast(h = "3 years")
```

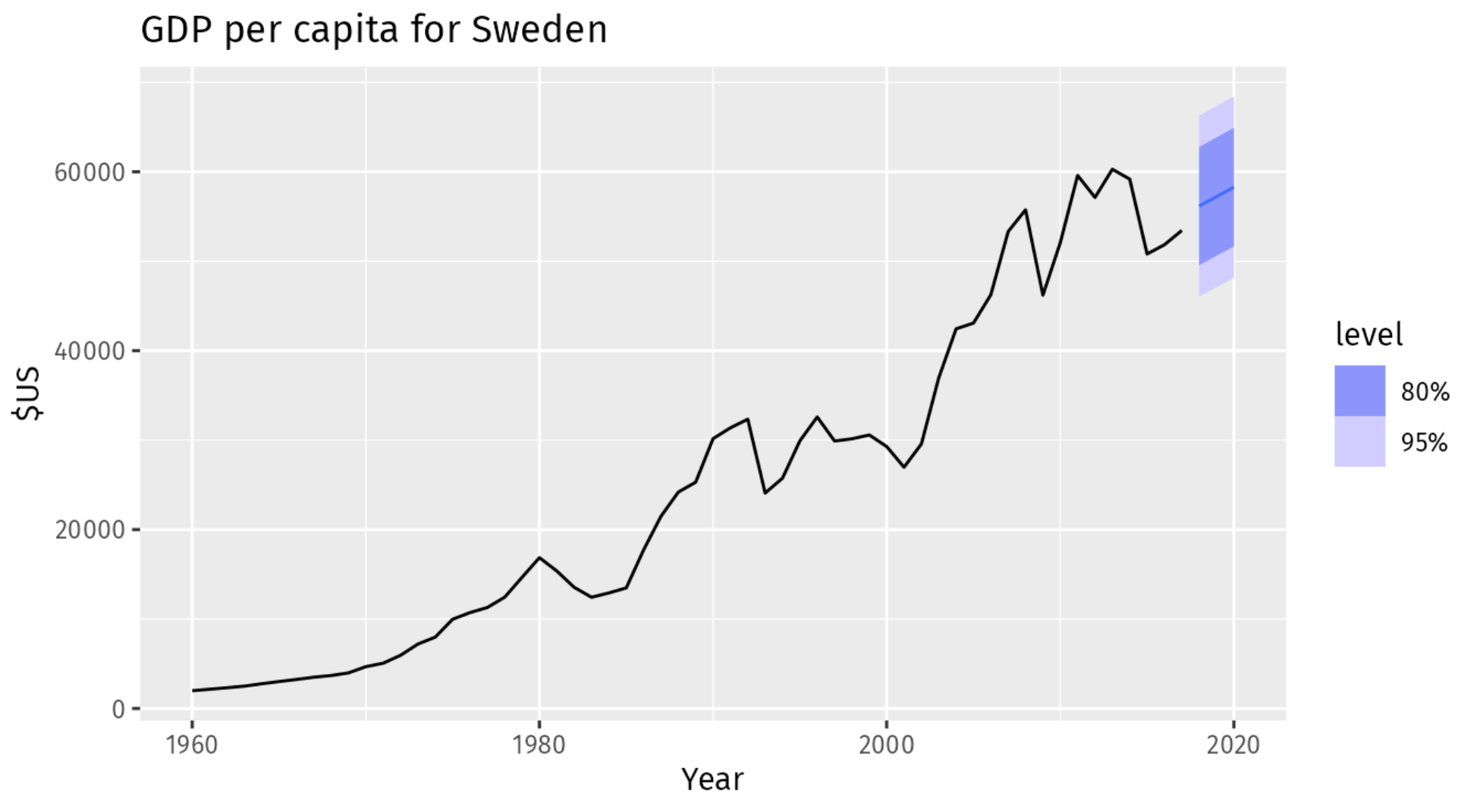
```
# A fable: 789 x 5 [1Y]
# Key:     Country, .model [263]
   Country  .model  Year   GDP_per_capita  .mean
   <fct>    <chr>  <dbl>           <dist>  <dbl>
 1 Afghani… trend…  2018     N(526, 9653)   526.
 2 Afghani… trend…  2019     N(534, 9689)   534.
 3 Afghani… trend…  2020     N(542, 9727)   542.
 4 Albania  trend…  2018  N(4716, 476419)  4716.
 5 Albania  trend…  2019  N(4867, 481086)  4867.
 6 Albania  trend…  2020  N(5018, 486012)  5018.
 7 Algeria  trend…  2018  N(4410, 643094)  4410.
 8 Algeria  trend…  2019  N(4489, 645311)  4489.
 9 Algeria  trend…  2020  N(4568, 647602)  4568.
10 America… trend…  2018 N(12491, 652926) 12491.
# ℹ 779 more rows
# ℹ Use print(n = …) to see more rows
```

1.7. 预测

下面的程序选取了瑞典的预测结果进行绘图。

```
fit |>
  forecast(h = "3 years") |>
  filter(Country == "Sweden") |>
  autoplot(gdppc) +
  labs(y = "$US", title = "GDP per capita for Sweden")
```

1.7. 预测


GDP per capita for Sweden
60000
40000
20000
0
$US
1960
1980
2000
2020
Year
level
80%
95%

2. 常用模型介绍

2.1. 四种简单模型

在学习常用的（原理比较复杂的）模型之前，我们首先了解几种简单的模型。在实践中，这些模型可以作为基准进行对比。

用作示例的数据是澳大利亚红砖产量数据：

```
bricks <- aus_production |>
  filter_index("1970 Q1" ~ "2004 Q4") |>
  select(Bricks)
```

这里用到了 tsibble 包提供的 `filter_index()` 函数，它可以更方便地提取一段时间内的观测值。

bricks

```
# A tsibble: 140 x 2 [1Q]
   Bricks Quarter
    <dbl>   <qtr>
 1    386 1970 Q1
 2    428 1970 Q2
 3    434 1970 Q3
 4    417 1970 Q4
 5    385 1971 Q1
 6    433 1971 Q2
 7    453 1971 Q3
 8    436 1971 Q4
 9    399 1972 Q1
10    461 1972 Q2
# i 130 more rows
# i Use print(n = …) to
see more rows
```

2.1. 四种简单模型

均值法：用历史数据的均值作为预测值的方法。其定义如下：

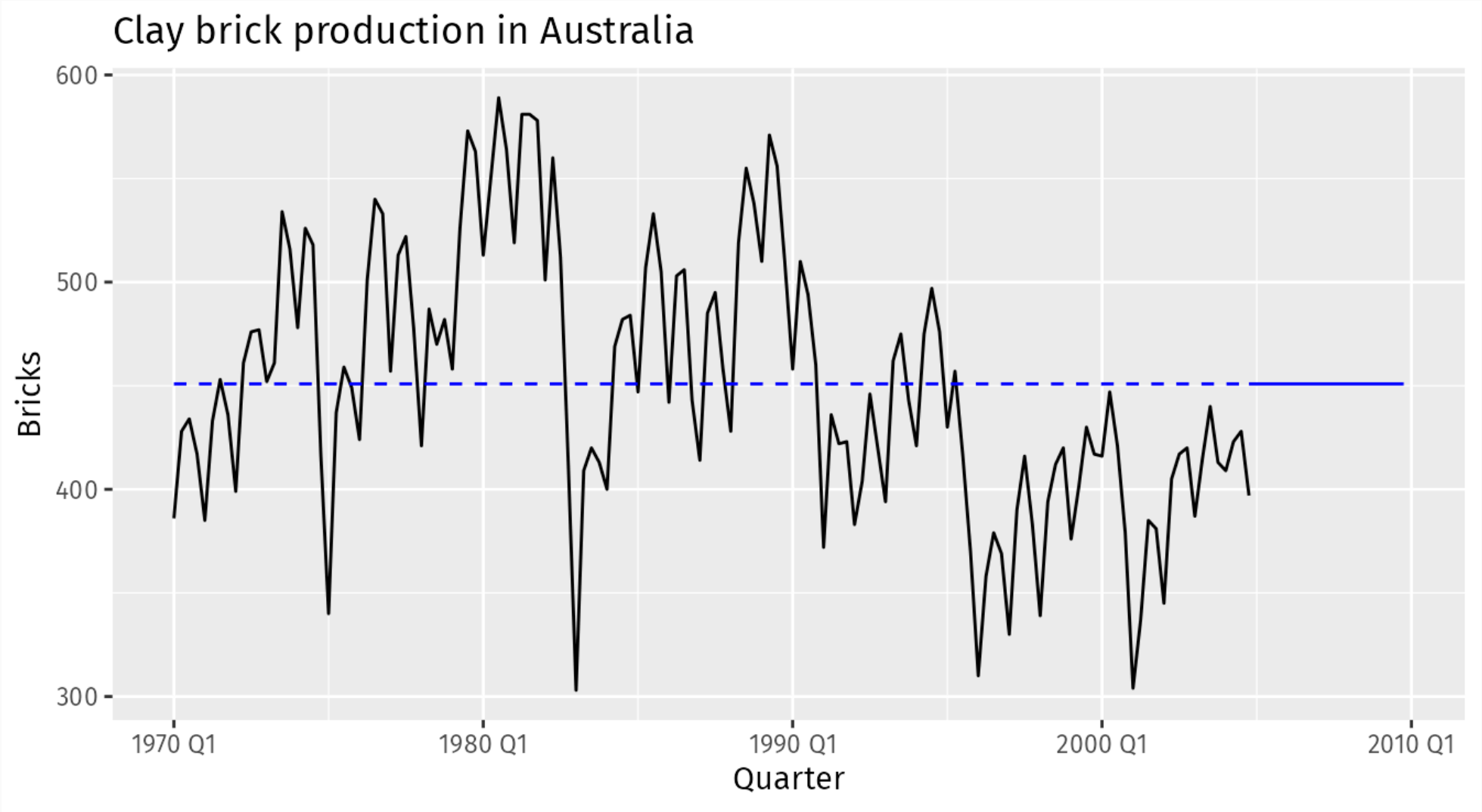
$$\hat{y}_{T+h|T} = \bar{y} = \frac{y_1 + \ldots + y_T}{T}$$

用均值法预测今后两年（8 个季度）的产量：

```
bricks |>
  model(MEAN(Bricks)) |>
  forecast(h = "2 year")
```

```
# A fable: 8 x 4 [1Q]
# Key:     .model [1]
  .model       Quarter        Bricks .mean
  <chr>          <qtr>        <dist> <dbl>
1 MEAN(Bricks) 2005 Q1 N(451, 4022)   451.
2 MEAN(Bricks) 2005 Q2 N(451, 4022)   451.
3 MEAN(Bricks) 2005 Q3 N(451, 4022)   451.
4 MEAN(Bricks) 2005 Q4 N(451, 4022)   451.
5 MEAN(Bricks) 2006 Q1 N(451, 4022)   451.
6 MEAN(Bricks) 2006 Q2 N(451, 4022)   451.
7 MEAN(Bricks) 2006 Q3 N(451, 4022)   451.
8 MEAN(Bricks) 2006 Q4 N(451, 4022)   451.
```

2.1. 四种简单模型


Clay brick production in Australia
600
500
400
300
Bricks
1970 Q1
1980 Q1
1990 Q1
2000 Q1
2010 Q1
Quarter

2.1. 四种简单模型

朴素法（naïve method）：用最终观测值作为预测值的方法。其定义如下：

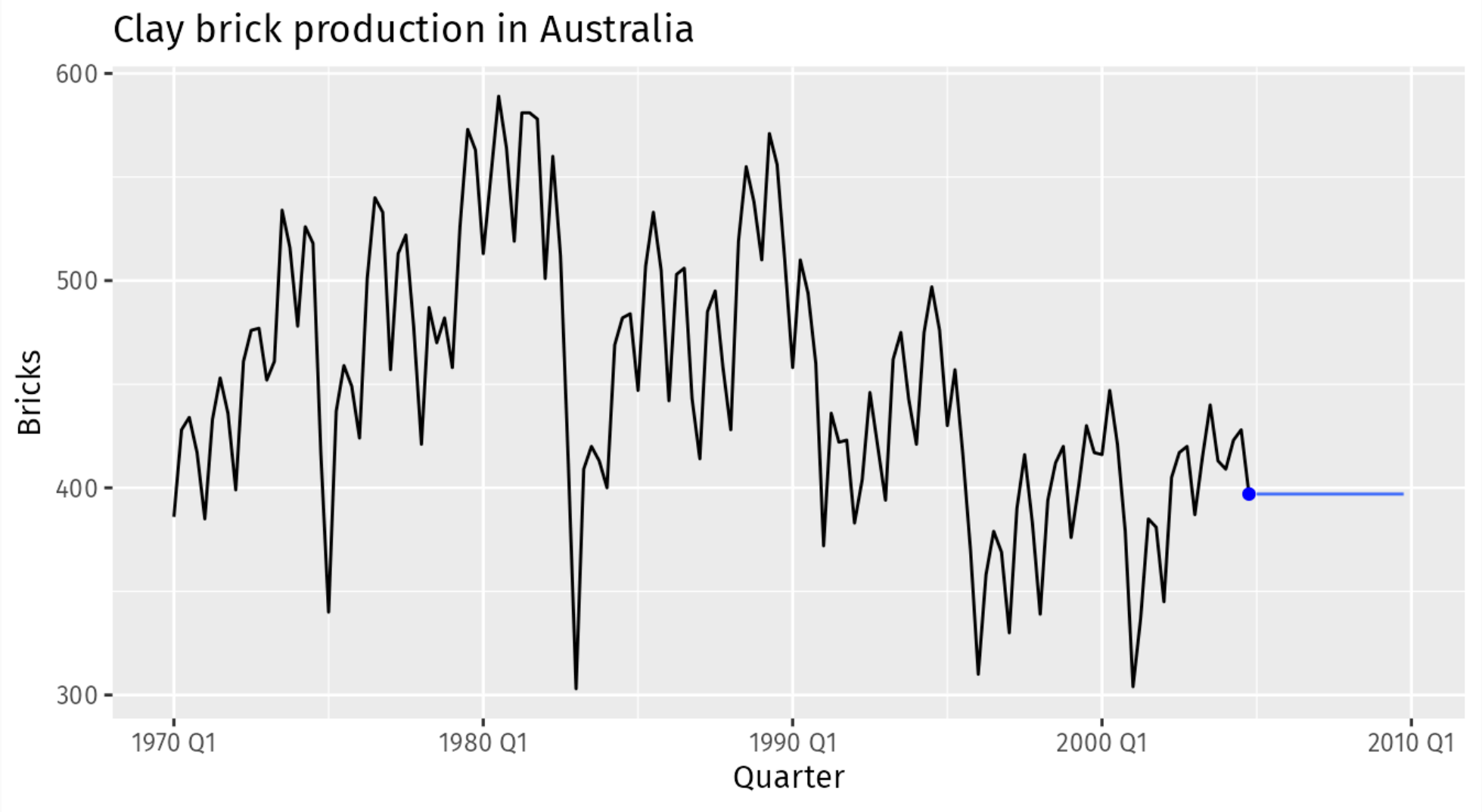
$$\hat{y}_{T+h|T} = y_T$$

用均值法预测今后两年（8 个季度）的产量：

```
bricks |>
  model(NAIVE(Bricks)) |>
  forecast(h = "2 year")
```

```
# A fable: 8 x 4 [1Q]
# Key:     .model [1]
  .model          Quarter        Bricks .mean
  <chr>             <qtr>        <dist> <dbl>
1 NAIVE(Bricks) 2005 Q1  N(397, 1960)    397
2 NAIVE(Bricks) 2005 Q2  N(397, 3920)    397
3 NAIVE(Bricks) 2005 Q3  N(397, 5880)    397
4 NAIVE(Bricks) 2005 Q4  N(397, 7840)    397
5 NAIVE(Bricks) 2006 Q1  N(397, 9801)    397
6 NAIVE(Bricks) 2006 Q2 N(397, 11761)    397
7 NAIVE(Bricks) 2006 Q3 N(397, 13721)    397
8 NAIVE(Bricks) 2006 Q4 N(397, 15681)    397
```

2.1. 四种简单模型


Clay brick production in Australia
600
500
400
300
Bricks
1970 Q1
1980 Q1
1990 Q1
2000 Q1
2010 Q1
Quarter

2.1. 四种简单模型

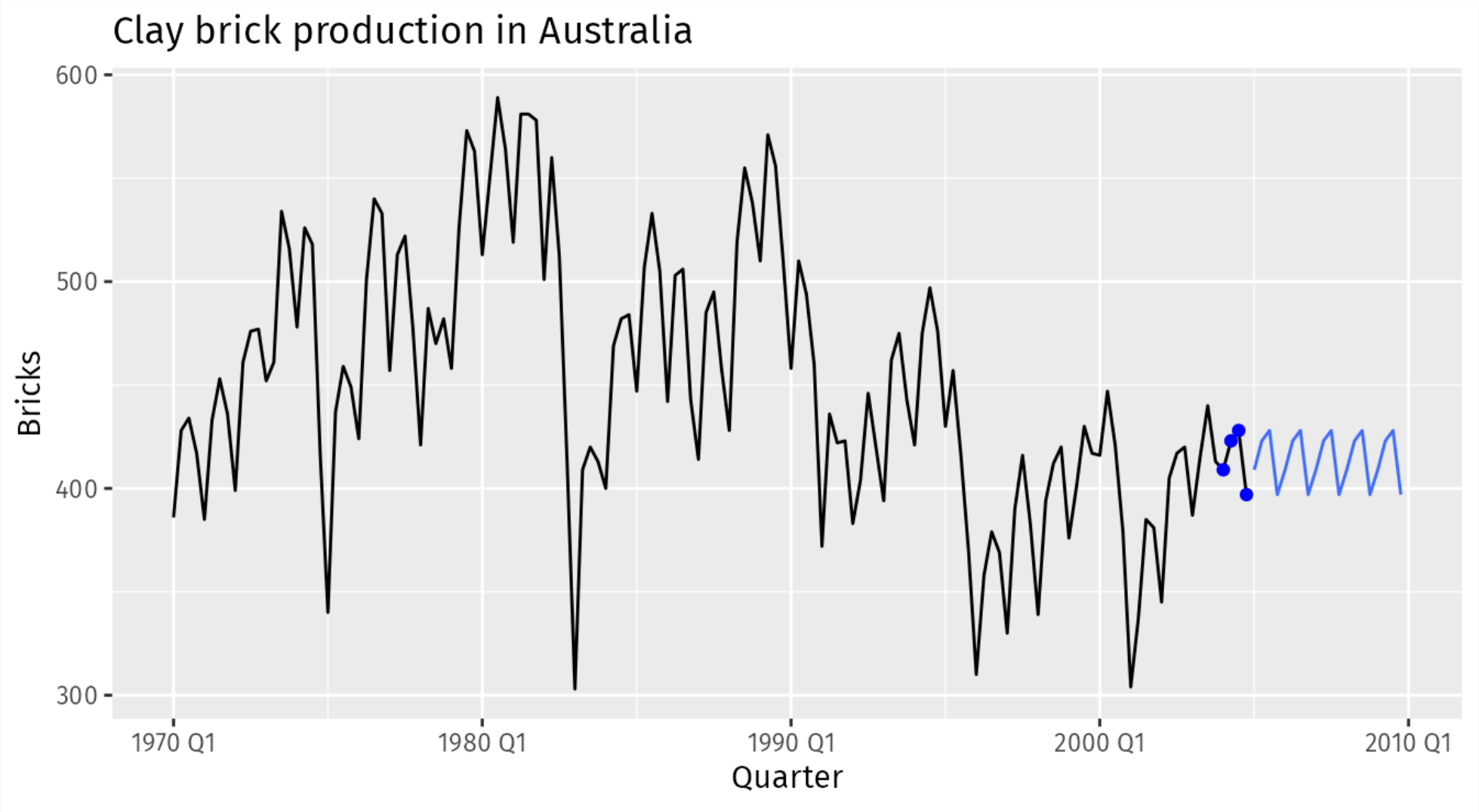
季节性朴素法（seasonal naïve method）：用每个季节的最终观测值作为该季节的预测值。令 m 为季节周期，k 为 $(h-1)/m$ 的整数部份，则预测值为

$$\hat{y}_{T+h|T} = y_{T+h-m(k+1)}$$

```
bricks |>
  model(SNAIVE(Bricks)) |>
  forecast(h = "2 year")
```

```
# A fable: 8 x 4 [1Q]
# Key:     .model [1]
  .model          Quarter        Bricks .mean
  <chr>             <qtr>        <dist> <dbl>
1 SNAIVE(Bricks) 2005 Q1 N(409, 3026)    409
2 SNAIVE(Bricks) 2005 Q2 N(423, 3026)    423
3 SNAIVE(Bricks) 2005 Q3 N(428, 3026)    428
4 SNAIVE(Bricks) 2005 Q4 N(397, 3026)    397
5 SNAIVE(Bricks) 2006 Q1 N(409, 6053)    409
6 SNAIVE(Bricks) 2006 Q2 N(423, 6053)    423
7 SNAIVE(Bricks) 2006 Q3 N(428, 6053)    428
8 SNAIVE(Bricks) 2006 Q4 N(397, 6053)    397
```

2.1. 四种简单模型


Clay brick production in Australia
600
500
400
300
Bricks
1970 Q1
1980 Q1
1990 Q1
2000 Q1
2010 Q1
Quarter

2.1. 四种简单模型

漂移法（drift method）：漂移指随时间推移而累积的位移，即序列的增加或减少。漂移法在朴素法的基础上利用历史数据的平均位移预测漂移项

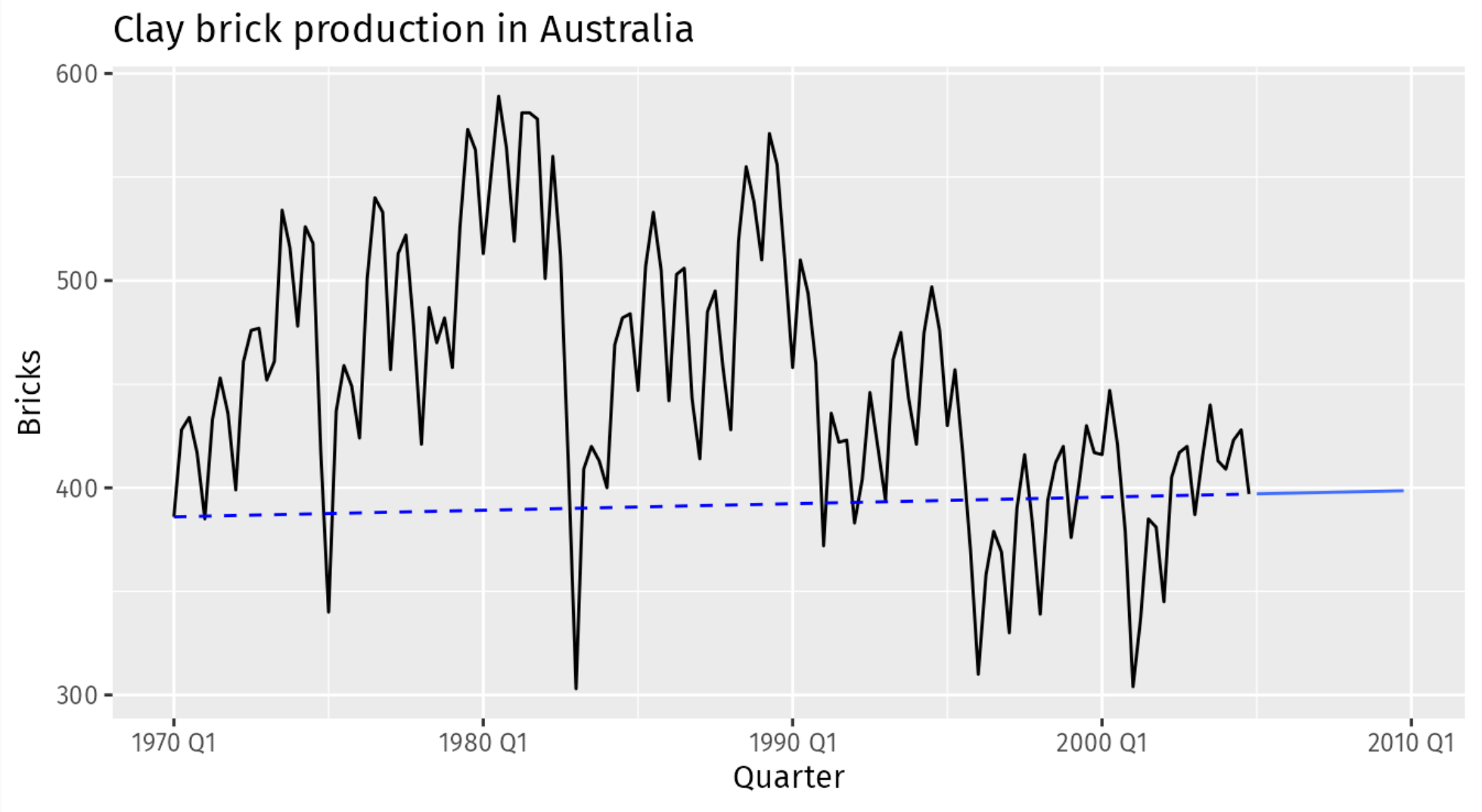
$$\hat{y}_{T+h|T} = y_T + \frac{h}{T-1}\sum_{t=2}^{T}(y_t - y_{t-1})$$

$$= y_T + h\,\frac{y_T - y_1}{T-1}$$

```
bricks |>
  model(DRIFT = RW(Bricks ~
  drift())) |>
  forecast(h = "2 year")
```

```
# A fable: 8 x 4 [1Q]
# Key:     .model [1]
  .model Quarter         Bricks .mean
  <chr>    <qtr>         <dist> <dbl>
1 DRIFT  2005 Q1  N(397, 1989)   397.
2 DRIFT  2005 Q2  N(397, 4005)   397.
3 DRIFT  2005 Q3  N(397, 6051)   397.
4 DRIFT  2005 Q4  N(397, 8124)   397.
5 DRIFT  2006 Q1 N(397, 10227)   397.
6 DRIFT  2006 Q2 N(397, 12357)   397.
7 DRIFT  2006 Q3 N(398, 14516)   398.
8 DRIFT  2006 Q4 N(398, 16703)   398.
```

2.1. 四种简单模型


Clay brick production in Australia
600
500
400
300
Bricks
1970 Q1
1980 Q1
1990 Q1
2000 Q1
2010 Q1
Quarter

2.2. 拟合值与残差

历史数据中的任意观测值 y_t 都可以利用之前的数据 $y_1, \ldots, y_{t-1}$ 进行预测，此时的预测值 $\hat{y}_{t|t-1}$ 称为**拟合值**（fitted value）。拟合值通常并不是真正的“预测值”，因为在估计模型参数时用到了所有历史数据（也包括 y_t 以及发生在它未来的观测值）。但是，在不包含参数的模型（如朴素法）中，拟合值就是预测值。

残差（residuals）是观测值和拟合值之差，即

$$e_t = y_t - \hat{y}_{t|t-1}$$

如果对数据进行了变换处理，则用处理后的观测值 $w_t = f(y_t)$ 计算的残差 $w_t - \hat{w}_t$ 称为**创新残差**（innovation residuals）。

拟合值和残差可以通过 `augment()` 函数获取。

2.3. 残差诊断

残差可以帮助我们判断预测模型的优劣。好的预测模型产生的创新残差满足下面两个性质：

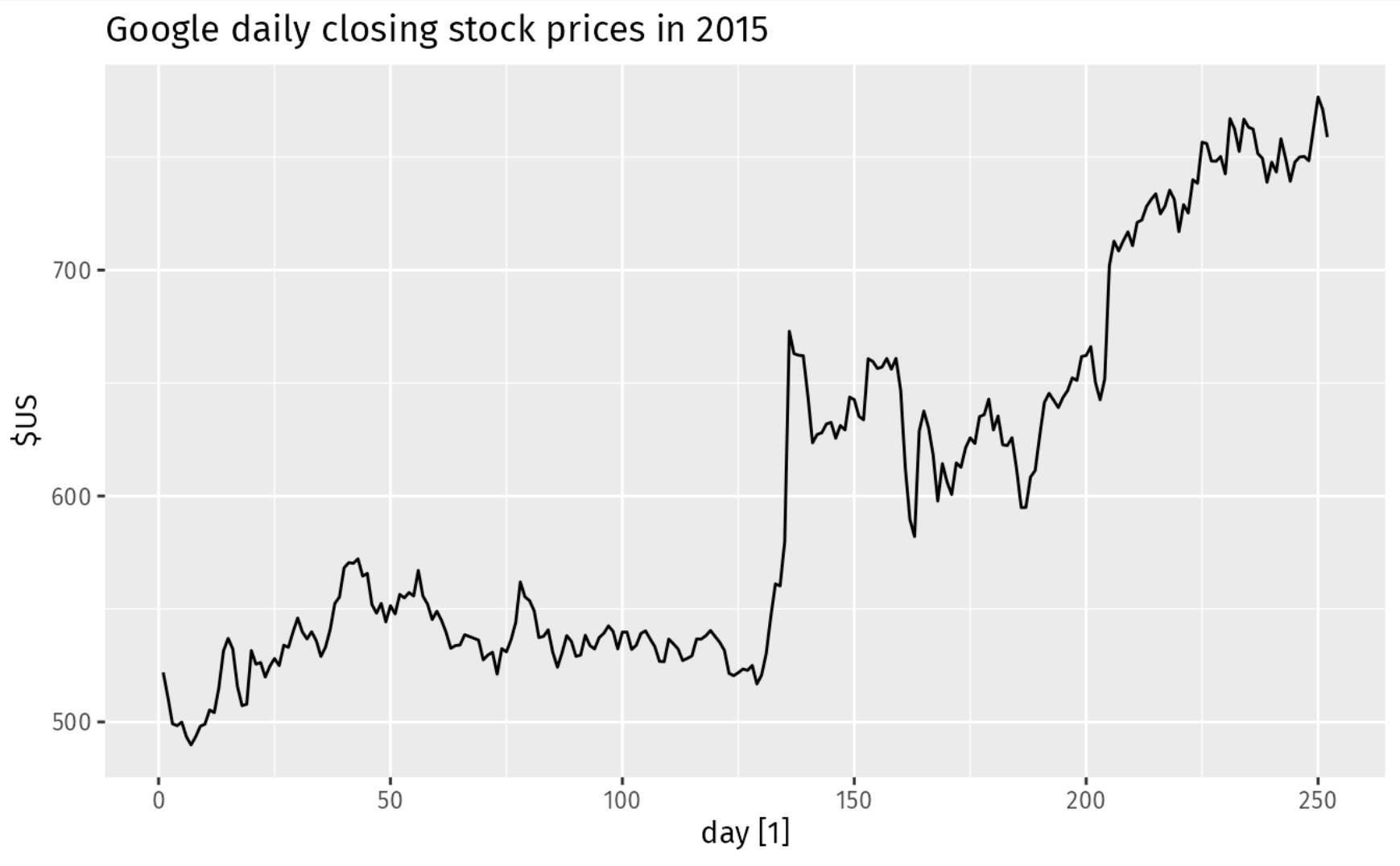
1. **创新产差的自相关为零**。若不为零，则说明残差中残留了一些本可以用来进行预测的信息。
2. **创新残差的均值为零**。若不为零，则说明预测值有偏差。

虽然如此，但不能仅凭残差的特征判断模型的好坏。满足以上特征的模型依然可以继续优化。

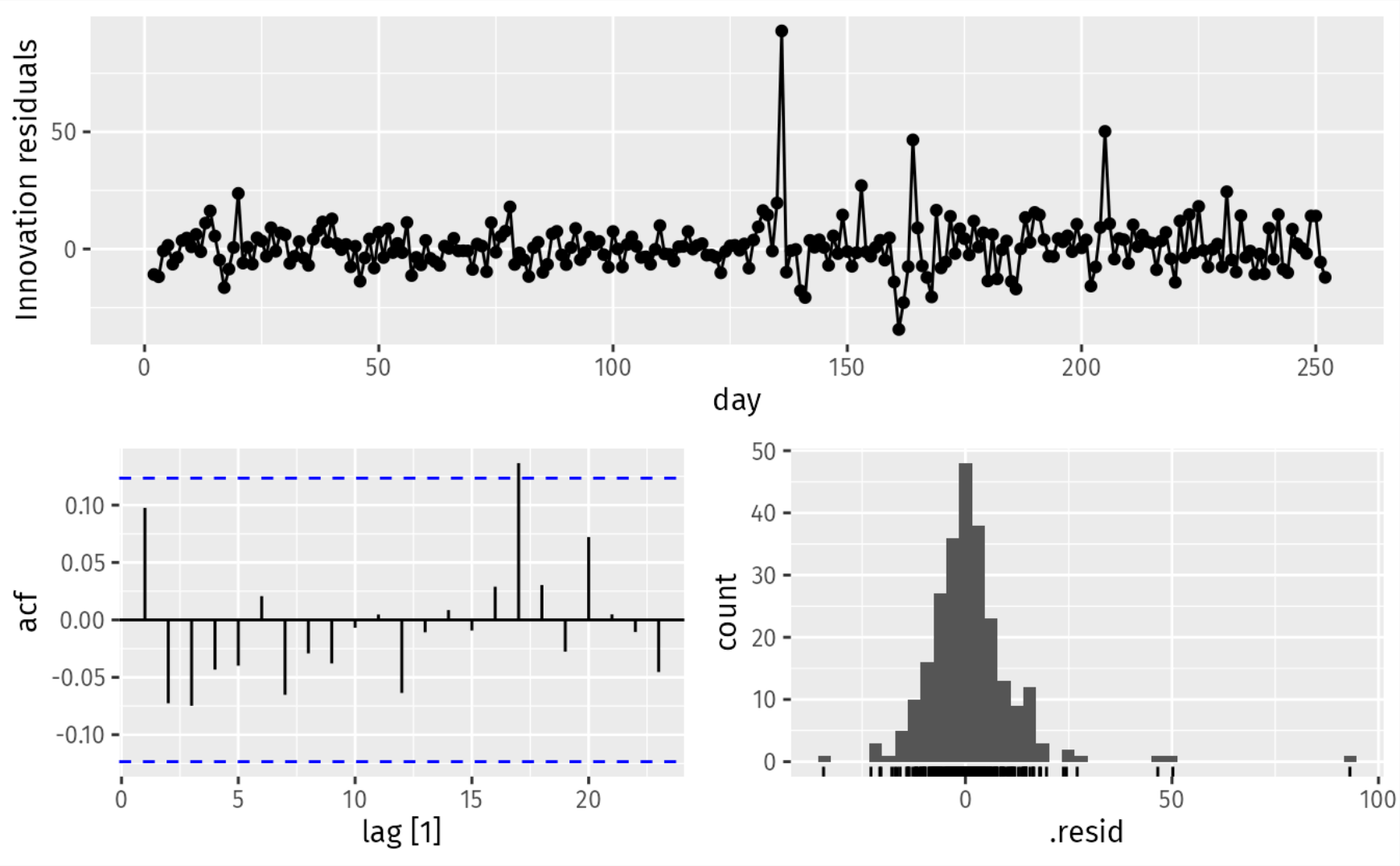
下面两个性质虽然不是必须的，但若满足则更好：

3. **创新残差的方差不变**（homoscedasticity/homoskedasticity）。
4. **创新残差服从正态分布**。

2.3. 残差诊断


Google daily closing stock prices in 2015
$US
700
600
500
0
50
100
150
200
250
day [1]

2.3. 残差诊断


Innovation residuals
50
0
0
50
100
150
200
250
day
acf
0.10
0.05
0.00
-0.05
-0.10
0
5
10
15
20
lag [1]
count
50
40
30
20
10
0
0
50
100
.resid

2.4. 区间预测

基于模型的预测大多是利用正态分布给出未来时间点上取值的分布信息，其中**点预测**（point forecast）通常是预测分布的均值，**区间预测**则是利用预测分布的标准差计算的。

如果时间点 $T+h$ 的预测分布是正态分布，其标准差为 $\hat{\sigma}_h$，则其预测区间为

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

教科书中经常展示的预测区间为 80% 和 95% 区间。不同区间对应的乘数 c 可以从正态分布表中查询，常用的乘数如右表所示。

%	乘数
50	0.67
60	0.84
70	1.04
80	1.28
90	1.64
95	1.96
96	2.05
97	2.17
98	2.33
99	2.58

2.4. 区间预测


Google daily closing stock price
800
700
600
500
$US
0
100
200
day
level
80%
95%

2.5. 合理利用数学变换和分解

在前面的学习中，我们了解到数学变换（如 Box-Cox 变换）可以让建模和预测变得更加容易。但需要注意的是，利用变换后的数据进行预测后，还需将预测值进行**逆变换**（fable 包会自动完成逆变换）。如果预测分布是正态分布，则逆变换后的分布就不是正态分布。同理，利用变换数据计算的预测区间端点也需要进行逆变换，而逆变换后的预测区间将不再以点预测值为中心。

当对某个时间序列进行分解时，我们可以获得

$$y_t = \hat{S}_t + \left(\hat{T}_t + \hat{R}_t\right) = y_t = \hat{S}_t + \hat{A}_t \quad \text{或} \quad y_t = \hat{S}_t\left(\hat{T}_t\hat{R}_t\right) = y_t = \hat{S}_t\hat{A}_t$$

$\hat{A}_t$ 是经过季节性调整的观测值。由于季节项 $\hat{S}_t$ 通常不随时间变化或变化非常缓慢，因此可以用季节性朴素法进行预测。而 $\hat{A}_t$ 则可以用任意一种非季节性模型（例如漂移法等）。

3. 预测效果的评价方法

3.1. 对点预测值的评价

不应用残差的大小评价预测效果的好坏，因为残差是根据已知数据计算的，而预测效果是针对未知的（发生在未来的）数据而定义的概念。

那么是否真的存在可以评价预测效果的方法呢？在实践中，常用的方法是人为地留出一部份数据用来评价模型，仅用剩下的数据进行拟合。用作拟合的数据称为**训练数据（training data）**或**训练集**，用作评价的数据称为**测试数据（test data）**或**测试集**。

测试数据通常是样本的 20%，但并不是绝对的。如果你想预测 n 期，那测试数据中应至少有 n 个值。

可以用来分割数据集的命令包括 `filter()`, `filter_index()`, `slice()` 等。

3.1. 对点预测值的评价

当训练集是 $\{y_1, \ldots, y_T\}$，测试集是 $\{y_{T+1}, \ldots, y_{T+H}\}$ 时，我们可以定义**预测误差（forecast error）**为观测值和预测值之差，即

$$e_{T+h} = y_{T+h} - \hat{y}_{T+h|T}$$

预测误差和残差的区别在于：

1. 残差是用训练集计算的，而预测误差是用测试集计算的。
2. 残差是用一步预测值 $\hat{y}_{t|t-1}$ 计算的，而预测误差可以用多步预测。

3.1. 对点预测值的评价

有多种方法可以将预测误差整合为一个测度指标并用来衡量预测的准确性。

- **标度依赖误差（scale-dependent error）**

  预测误差和观测值的单位相同，因此用其计算的准确度指标也是标度依赖的，无法在不同单位的变量间进行比较。具有标度依赖性的指标包括**平均绝对误差（mean absolute error/MAE）**和**均方根误差（root mean squared error/RMSE）**：

$$\text{MAE} = \text{mean}(|e_t|)$$

$$\text{RMSE} = \sqrt{\text{mean}(e_t^2)}$$

  令 MAE 最小化的模型会给出接近中位数的预测值，而令 RMSE 最小化的模型则会给出接近均值的预测值。另外，**均方误差（mean squared error/MSE）** 也很常见。

3.1. 对点预测值的评价

- **百分比误差（percentage error）**

  百分比误差是指 $p_t = 100e_t/y_t$。它不依赖任何数据单位，可以用作不同数据集间的比较。最常见的指标是**平均绝对百分比误差（mean absolute percentage error/MAPE）**：

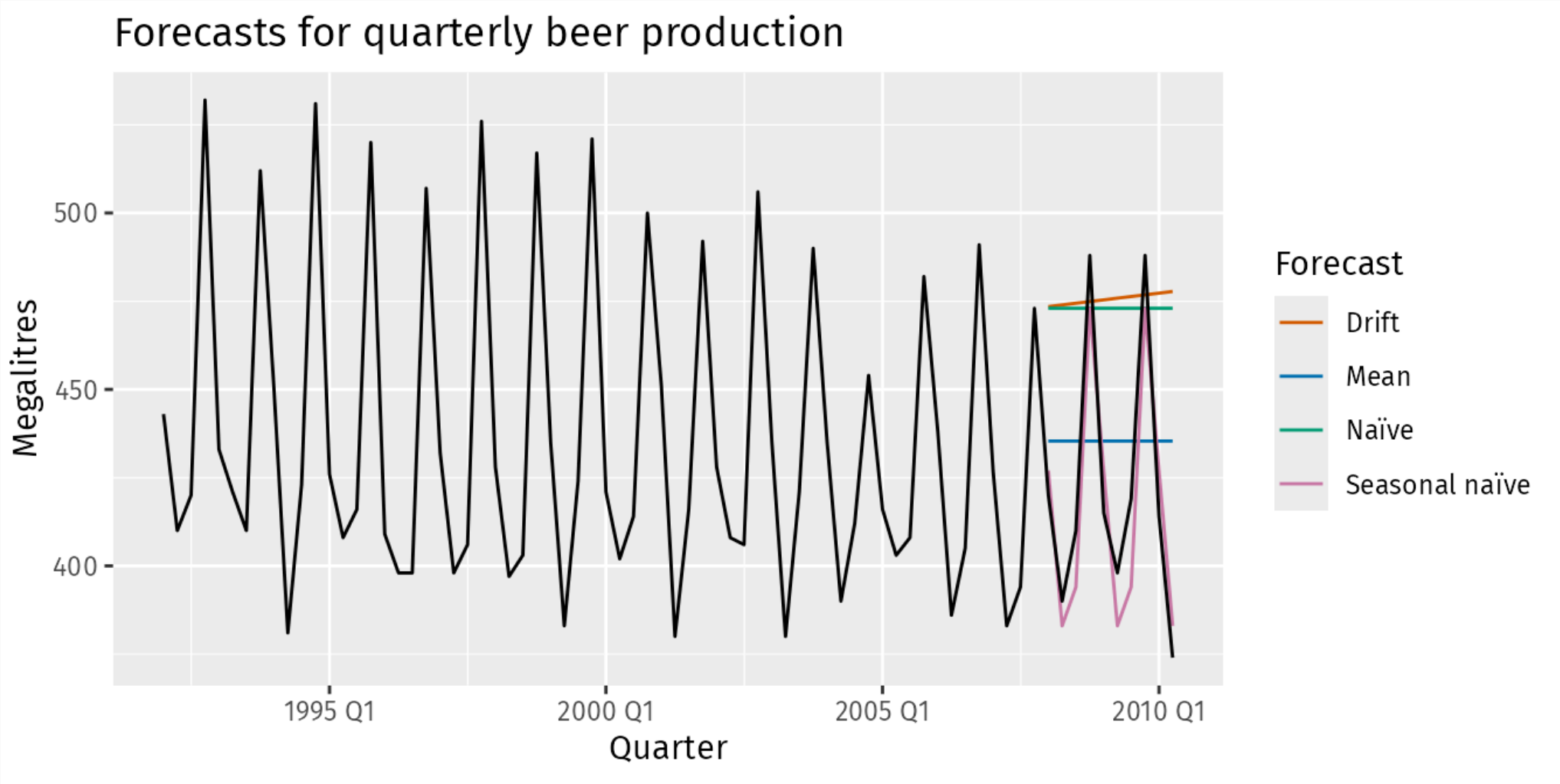
  $$\text{MAPE} = \text{mean}(|p_t|)$$

  百分比误差指标的最大缺点是，当 $y_t = 0$ 时无法定义 p_t。另一个经常被忽略的缺点是，它假设变量中的零值是有意义的。例如温度的单位尺度中的零值是任意定义的（包括华氏和摄氏），因此不适合用百分比误差来衡量。

- **去标度误差（scaled error）**

  针对百分比误差的缺点，教科书第 5.8 节还介绍了几种去标度误差，例如**平均绝对去标度误差（mean absolute scaled error/MASE）**。由于定义比较复杂，在这里就不详细介绍了。

3.1. 对点预测值的评价


Forecasts for quarterly beer production
Megalitres
500
450
400
1995 Q1
2000 Q1
2005 Q1
2010 Q1
Quarter
Forecast
Drift
Mean
Naïve
Seasonal naïve

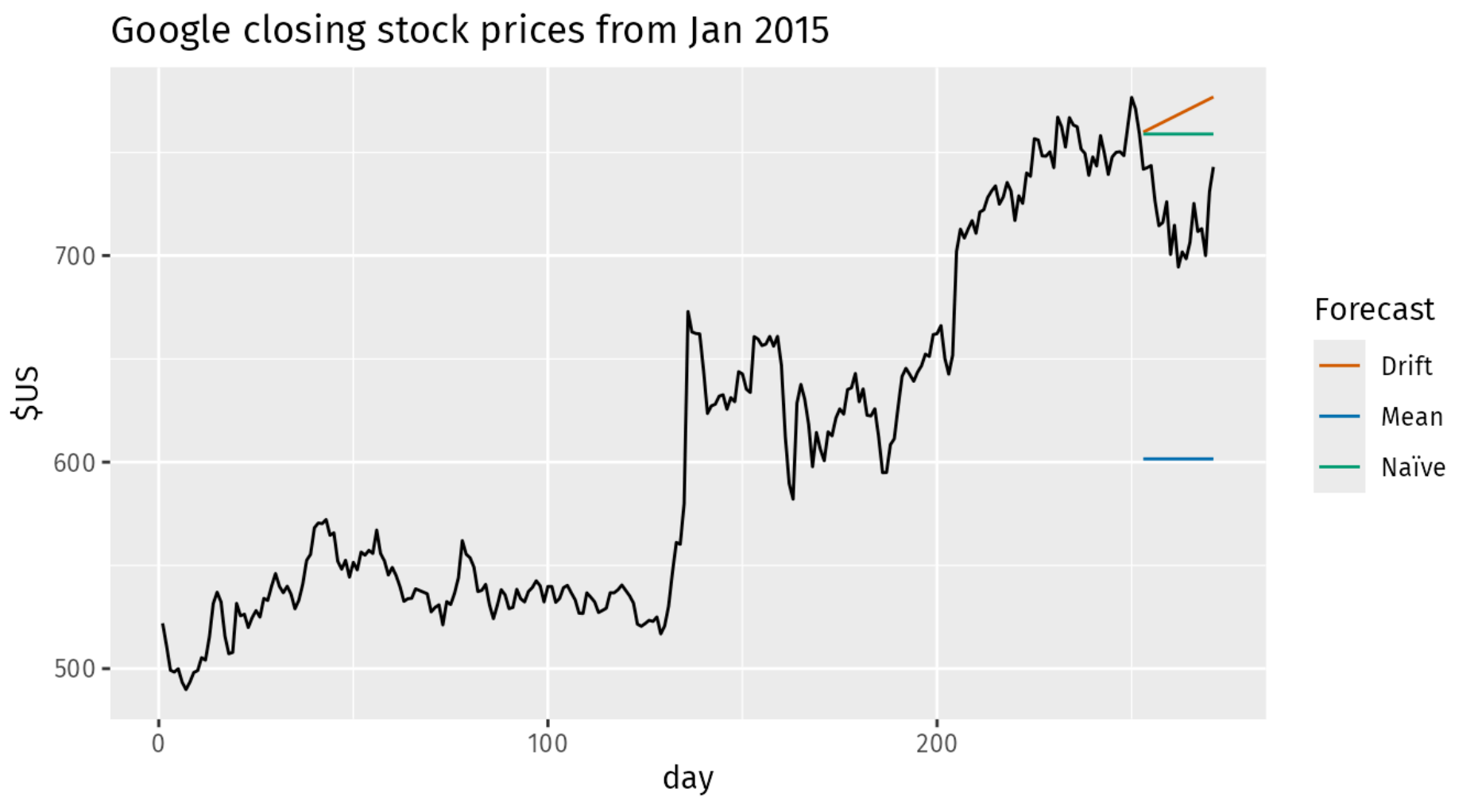
3.1. 对点预测值的评价

下表给出了对啤酒产量四种预测结果的预测误差

预测方法	RMSE	MAE	MAPE	MASE
漂移法	64.90	58.88	14.58	4.12
均值法	38.45	34.83	8.28	2.44
朴素法	62.69	57.40	14.18	4.01
季节性朴素法	14.31	13.40	3.17	0.94

由于数据包含明显的季节性特征，每种测度都显示季节性朴素法的预测效果最好。在季节特征并不明显的数据中，不同的测度可能提示不同的结果。

3.1. 对点预测值的评价


Google closing stock prices from Jan 2015
$US
700
600
500
0
100
200
day
Forecast
Drift
Mean
Naïve

3.1. 对点预测值的评价

下表给出了对谷歌股价三种预测结果的预测误差

预测方法	RMSE	MAE	MAPE	MASE
漂移法	53.07	49.82	6.99	6.99
均值法	118.03	116.95	16.24	16.41
朴素法	43.43	40.38	5.67	5.67

虽然四种测度都选择了朴素法，但不难看出这和测试集的选择是有很大关系的。

4. 课后练习

4. 课后练习

- 学习教科书第 5 章（The Forecaster’s Toolbox）中的内容，并尝试在自己的电脑上复现书中的结果。Slides 中省略了第 5.9 和 5.10 节的内容，感兴趣的同学可以自学。

- 思考下列问题，并尝试利用教科书中的数据检验你的想法：

  1. 在研究流程中的效果评价部份，为了计算误差测度，我们用到了**预测值**。这个预测值的计算和流程中的预测部份有什么异同?
  2. 四种基本预测模型显然都过于简单，那么如果让你基于这些基本模型开发一个更加实用的模型，你会怎样做？你的模型中是否包含参数？应该如何选择参数值?