# Probabilistic forecasts for anomaly detection

Rob J Hyndman

3 July 2024

MONASH University

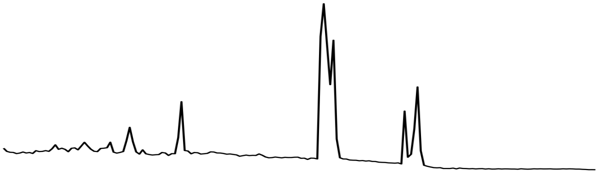
# Time series anomaly detection paradigms

1 **Identify anomalies within a time series in real time:** use one-step forecast distributions

# Time series anomaly detection paradigms

1 **Identify anomalies within a time series in real time:** use one-step forecast distributions

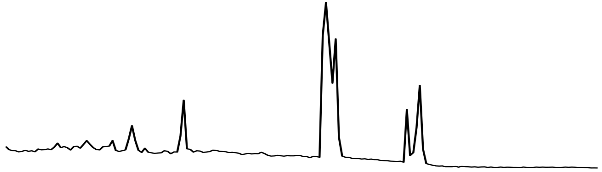
2 **Identify anomalies within a time series in historical data:** use residual distributions from smoothing method



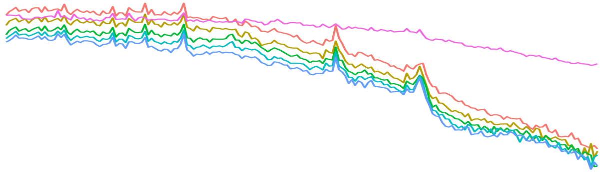
# Time series anomaly detection paradigms

1. **Identify anomalies within a time series in real time:** use one-step forecast distributions

2. **Identify anomalies within a time series in historical data:** use residual distributions from smoothing method



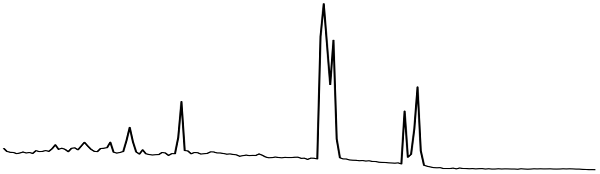
3. **Identify an anomalous time series in a collection of time series:** use feature-based approach



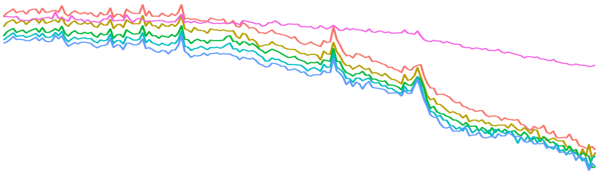
# Time series anomaly detection paradigms

1. **Identify anomalies within a time series in real time:** use one-step forecast distributions

2. **Identify anomalies within a time series in historical data:** use residual distributions from smoothing method



3. **Identify an anomalous time series in a collection of time series:** use feature-based approach

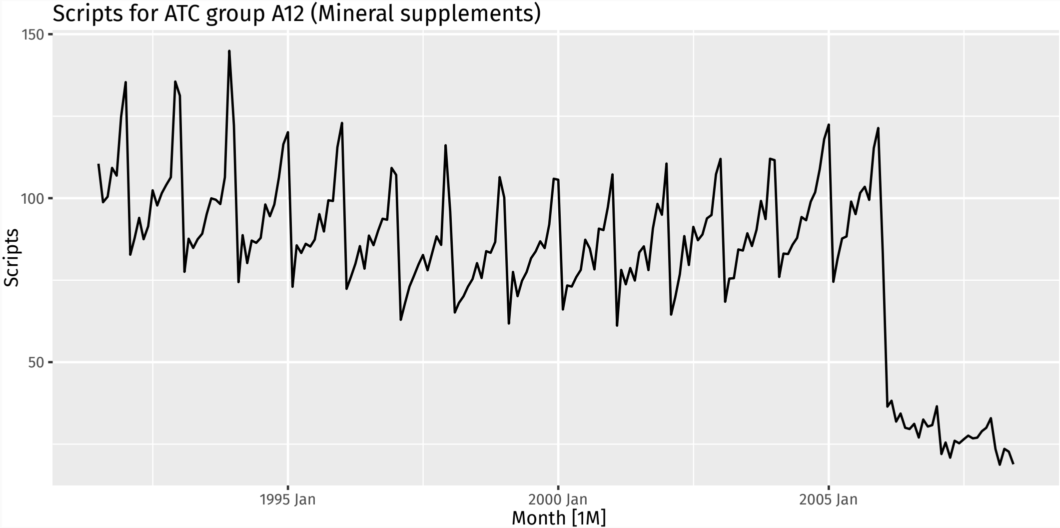


# Australian PBS data

```
pbs
```

```
# A tsibble: 17,016 x 3 [1M]
# Key:       ATC2 [84]
   ATC2     Month Scripts
   <chr>    <mth>   <dbl>
 1 A01   1991 Jul    22.6
 2 A01   1991 Aug    20.4
 3 A01   1991 Sep    21.4
 4 A01   1991 Oct    23.7
 5 A01   1991 Nov    23.5
 6 A01   1991 Dec    26.3
 7 A01   1992 Jan    22.0
 8 A01   1992 Feb    16.4
 9 A01   1992 Mar    17.2
10 A01   1992 Apr    18.8
# i 17,006 more rows
```

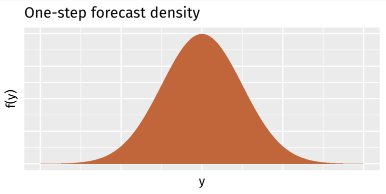
# Australian PBS data



# Anomaly score distribution

## One-step forecast distribution: $N(\mu_t, \sigma^2)$

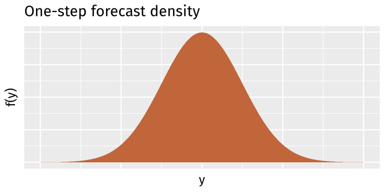
$$f(y_t|y_1,\dots,y_{t-1}) = \phi\left(\frac{y_t - \mu_t}{\sigma}\right) = \frac{1}{\sigma\sqrt{2\pi}}\exp\left\{\frac{(y_t-\mu_t)^2}{2\sigma^2}\right\}$$



# Anomaly score distribution

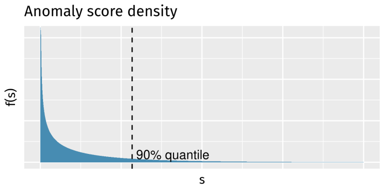
## One-step forecast distribution: $N(\mu_t, \sigma^2)$

$$f(y_t|y_1,\dots,y_{t-1}) = \phi\left(\frac{y_t - \mu_t}{\sigma}\right) = \frac{1}{\sigma\sqrt{2\pi}}\exp\left\{\frac{(y_t-\mu_t)^2}{2\sigma^2}\right\}$$




## Anomaly score distribution: $S \sim \frac{1}{2}\chi_1^2 + c$

$$s_t = -\log f(y_t|y_1,\dots,y_{t-1}) = \frac{1}{2}\left(\frac{y_t-\mu_t}{2\sigma}\right)^2 + \frac{1}{2}\log(2\pi\sigma^2)$$



# Anomaly score distribution

## One-step forecast distribution: $N(\mu_t, \sigma^2)$

$$f(y_t|y_1,\dots,y_{t-1}) = \phi\left(\frac{y_t-\mu_t}{\sigma}\right) = \frac{1}{\sigma\sqrt{2\pi}}\exp\left\{\frac{(y_t-\mu_t)^2}{2\sigma^2}\right\}$$

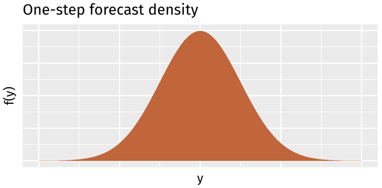
## Anomaly score distribution: $S \sim \frac{1}{2}\chi^2_1 + c$

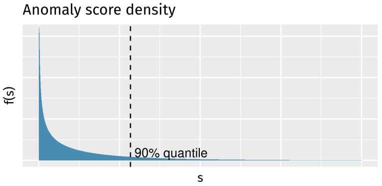
$$s_t = -\log f(y_t|y_1,\dots,y_{t-1}) = \frac{1}{2}\left(\frac{y_t-\mu_t}{2\sigma}\right)^2 + \frac{1}{2}\log(2\pi\sigma^2)$$

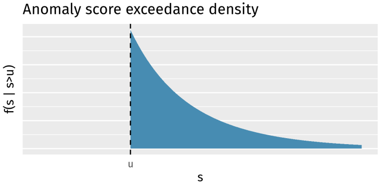
## Extreme anomaly score distribution

$$H(x) = P(S \le u + x \mid S > u)$$

→ Generalized Pareto Distribution for almost all forecast distributions $f$.



# Anomaly detection algorithm

For each $t$:

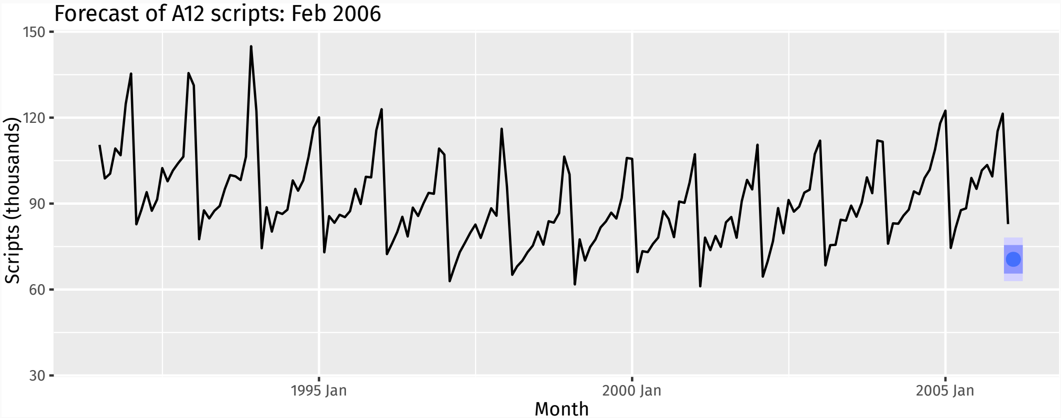
- Estimate one-step forecast density: $f(y_t|y_1,\dots,y_{t-1})$.
- Anomaly score: $s_t = -\log \hat{f}(y_t|y_1,\dots,y_{t-1})$.
- High anomaly score indicates potential anomaly.
- Fit a Generalized Pareto Distribution to the top 10% of anomaly scores seen so far.
- $y_t$ is anomaly if $P(S > s_t) < 0.05$ under GPD.

# Example

```
a12 ← pbs ▷ filter(ATC2 == "A12", Month <= yearmonth("2006 Jan"))
a12plus ← pbs ▷ filter(ATC2 == "A12", Month <= yearmonth("2006 Feb"))
fc ← a12 ▷ model(ets = ETS(Scripts)) ▷ forecast(h = 1)
```

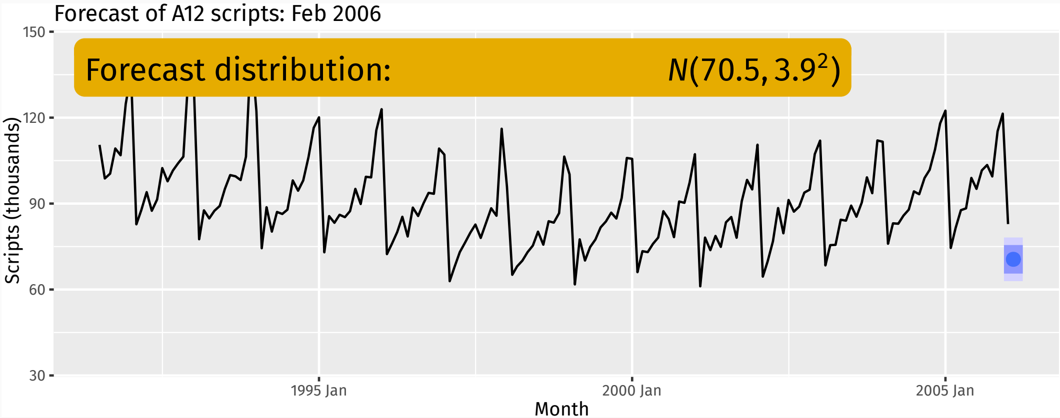
# Example

```
a12 ← pbs ▷ filter(ATC2 == "A12", Month <= yearmonth("2006 Jan"))
a12plus ← pbs ▷ filter(ATC2 == "A12", Month <= yearmonth("2006 Feb"))
fc ← a12 ▷ model(ets = ETS(Scripts)) ▷ forecast(h = 1)
fc ▷ autoplot(a12)
```



# Example

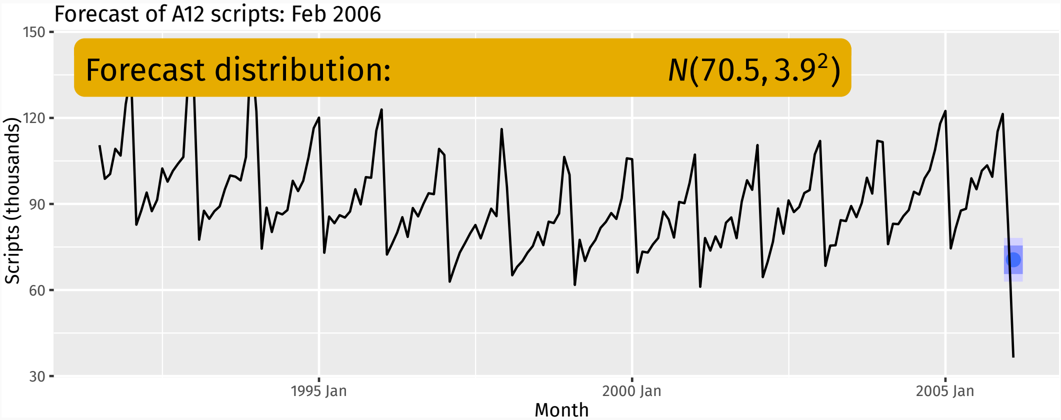
```
a12 ← pbs ▷ filter(ATC2 == "A12", Month <= yearmonth("2006 Jan"))
a12plus ← pbs ▷ filter(ATC2 == "A12", Month <= yearmonth("2006 Feb"))
fc ← a12 ▷ model(ets = ETS(Scripts)) ▷ forecast(h = 1)
fc ▷ autoplot(a12)
```




Forecast distribution: $N(70.5, 3.9^2)$

# Example

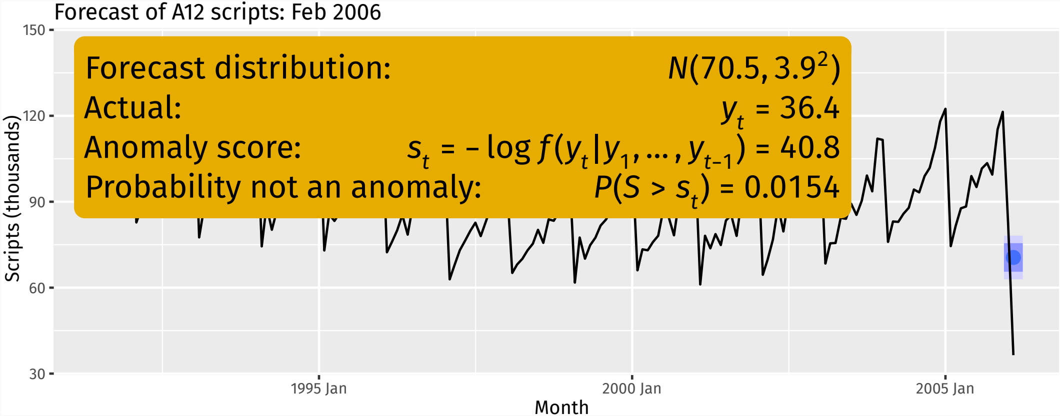
```
a12 ← pbs ▷ filter(ATC2 == ”A12”, Month <= yearmonth(”2006 Jan”))
a12plus ← pbs ▷ filter(ATC2 == ”A12”, Month <= yearmonth(”2006 Feb”))
fc ← a12 ▷ model(ets = ETS(Scripts)) ▷ forecast(h = 1)
fc ▷ autoplot(a12plus)
```




**Forecast distribution:** $N(70.5, 3.9^2)$

# Example

```
a12 <- pbs |> filter(ATC2 == "A12", Month <= yearmonth("2006 Jan"))
a12plus <- pbs |> filter(ATC2 == "A12", Month <= yearmonth("2006 Feb"))
fc <- a12 |> model(ets = ETS(Scripts)) |> forecast(h = 1)
fc |> autoplot(a12plus)
```



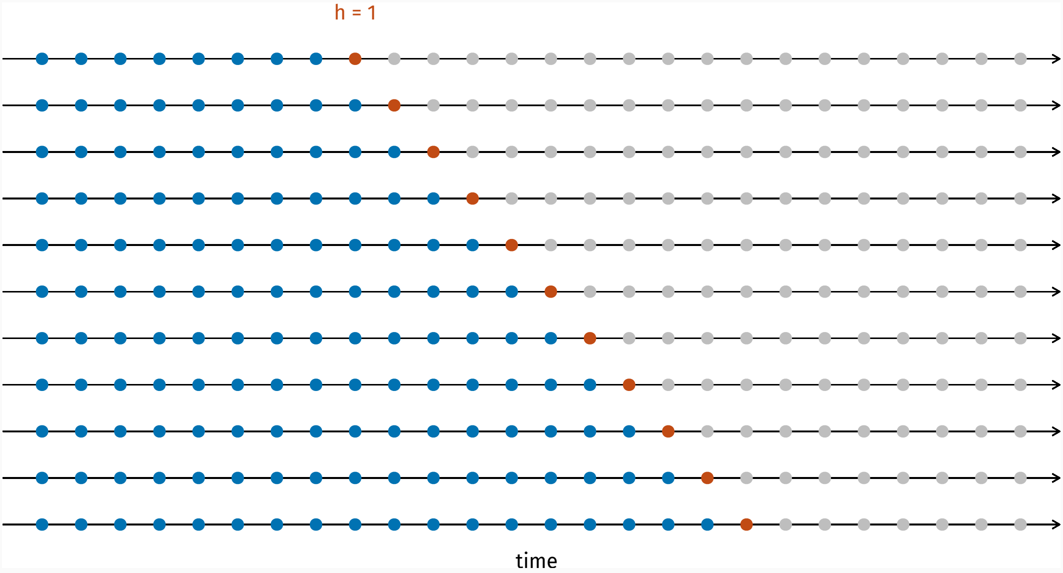

Forecast distribution: $N(70.5, 3.9^2)$

Actual: $y_t = 36.4$

Anomaly score: $s_t = -\log f(y_t|y_1,\dots,y_{t-1}) = 40.8$

Probability not an anomaly: $P(S > s_t) = 0.0154$

# Rolling origin forecasts



# Rolling origin forecasts

```r
pbs_stretch <- stretch_tsibble(pbs, .step = 1, .init = 36)
```

```
# A tsibble: 1,684,884 x 4 [1M]
# Key:       .id, ATC2 [14,076]
   ATC2      Month Scripts   .id
   <chr>     <mth>   <dbl> <int>
 1 A01    1991 Jul    22.6     1
 2 A01    1991 Aug    20.4     1
 3 A01    1991 Sep    21.4     1
 4 A01    1991 Oct    23.7     1
 5 A01    1991 Nov    23.5     1
 6 A01    1991 Dec    26.3     1
 7 A01    1992 Jan    22.0     1
 8 A01    1992 Feb    16.4     1
 9 A01    1992 Mar    17.2     1
10 A01    1992 Apr    18.8     1
# i 1,684,874 more rows
```

# Rolling origin forecasts

```r
pbs_fit <- pbs_stretch |> model(ets = ETS(Scripts))
```

```
# A mable: 14,076 x 3
# Key:     .id, ATC2 [14,076]
     .id ATC2             ets
   <int> <chr>        <model>
 1     1 A01     <ETS(M,N,A)>
 2     1 A02     <ETS(M,A,M)>
 3     1 A03     <ETS(M,A,M)>
 4     1 A04     <ETS(M,N,A)>
 5     1 A05    <ETS(A,Ad,N)>
 6     1 A06     <ETS(M,A,M)>
 7     1 A07     <ETS(M,N,M)>
 8     1 A09     <ETS(M,A,M)>
 9     1 A10     <ETS(M,A,M)>
10     1 A11     <ETS(M,A,M)>
# i 14,066 more rows
```

# Rolling origin forecasts

```
pbs_fc <- forecast(pbs_fit, h = 1)
```

```
# A fable: 14,076 x 4 [1M]
# Key:      .id, ATC2 [14,076]
      .id ATC2      Month          Scripts
    <int> <chr>     <mth>           <dist>
 1      1 A01    1994 Jul     N(23, 2.1)
 2      1 A02    1994 Jul   N(590, 1054)
 3      1 A03    1994 Jul      N(84, 19)
 4      1 A04    1994 Jul      N(69, 15)
 5      1 A05    2003 Jul N(1.4, 0.014)
 6      1 A06    1994 Jul     N(33, 4.2)
 7      1 A07    1994 Jul      N(74, 17)
 8      1 A09    1994 Jul N(3.7, 0.029)
 9      1 A10    1994 Jul     N(166, 54)
10      1 A11    1994 Jul       N(30, 3)
# i 14,066 more rows
```

# PBS anomalies

```r
pbs_scores <- pbs_fc |>
  left_join(pbs |> rename(actual = Scripts), by = c("ATC2", "Month")) |>
  mutate(
    s = -log_likelihood(Scripts, actual),
    prob = lookout(density_scores = s, threshold = 0.9)
  )
```

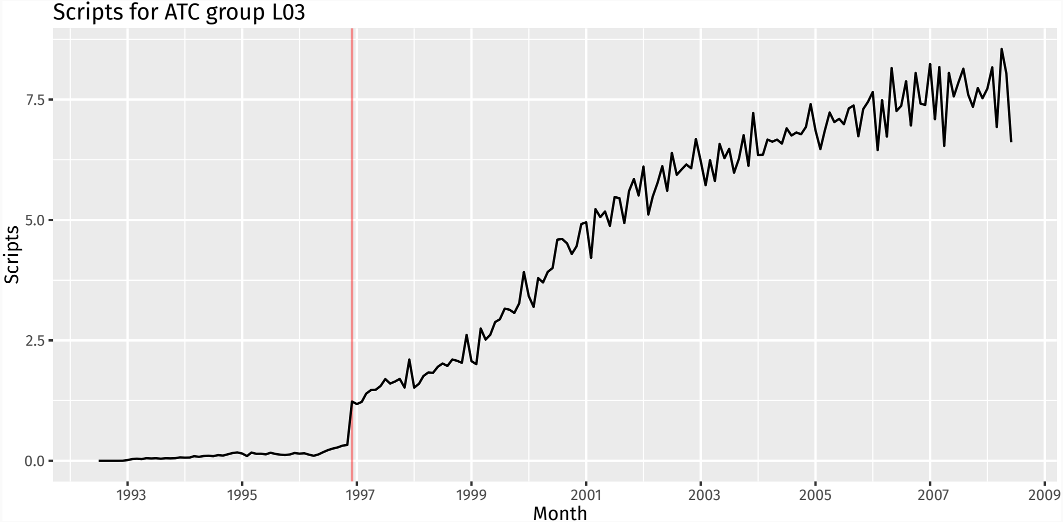
```
# A fable: 14,076 x 7 [1M]
# Key:     .id, ATC2 [14,076]
    .id ATC2      Month         Scripts actual     s  prob
  <int> <chr>     <mth>          <dist>  <dbl> <dbl> <dbl>
1     1 A01    1994 Jul    N(23, 2.1)   20.9    2.46 1
2     1 A02    1994 Jul  N(590, 1054) 516.      6.97 0.296
3     1 A03    1994 Jul     N(84, 19)   80.5    2.75 1
4     1 A04    1994 Jul     N(69, 15)   66.1    2.62 1
5     1 A05    2003 Jul N(1.4, 0.014)    1.47 -1.05 1
6     1 A06    1994 Jul    N(33, 4.2)   29.2    3.41 1
7     1 A07    1994 Jul     N(74, 17)   68.5    3.09 1
8     1 A09    1994 Jul N(3.7, 0.029)    3.32  1.46 1
```

# PBS anomalies

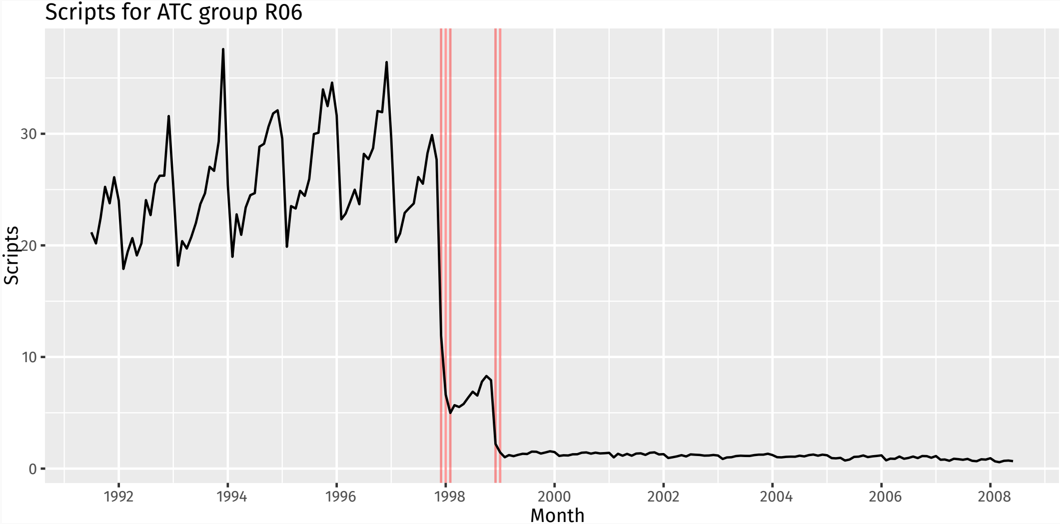
```
pbs_scores |> filter(prob < 0.05)
```

```
# A fable: 67 x 7 [1M]
# Key:      .id, ATC2 [67]
      .id ATC2     Month               Scripts actual     s     prob
    <int> <chr>    <mth>                <dist>  <dbl> <dbl>    <dbl>
 1     11 P03   1995 May     N(2.3, 0.045)  3.83   26.1 0.0278
 2     13 D05   1995 Jul  N(0.33, 0.0039)  0.781  24.3 0.0307
 3     18 A11   1995 Dec        N(46, 6.6) 25.1    34.4 0.0192
 4     18 C05   1995 Dec        N(33, 4.9)  2.46   98.9 0.00510
 5     18 D02   1995 Dec        N(43, 5.9) 10.0    97.2 0.00522
 6     18 D06   1995 Dec      N(6.7, 0.17)  4.24   18.5 0.0455
 7     18 D08   1995 Dec      N(5.4, 0.11)  1.40   71.4 0.00759
 8     18 G04   1995 Dec        N(54, 8.4)  9.67  121.  0.00399
 9     18 L03   1996 Dec N(0.33, 0.00054)  1.23  756.  0.000463
10     19 D02   1996 Jan         N(38, 26)  8.07   19.8 0.0412
# i 57 more rows
```

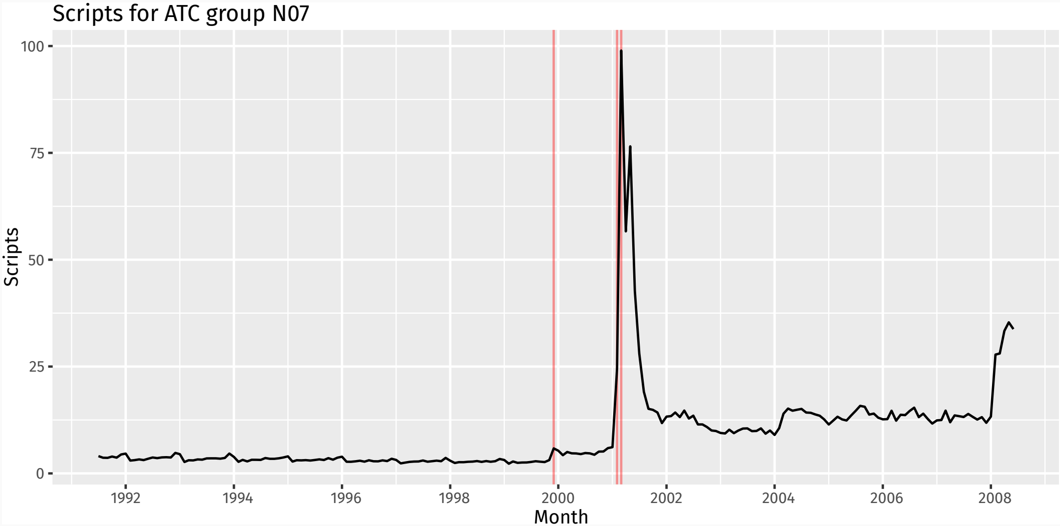
# PBS anomalies



# PBS anomalies



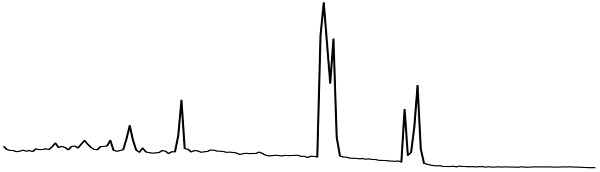
# PBS anomalies



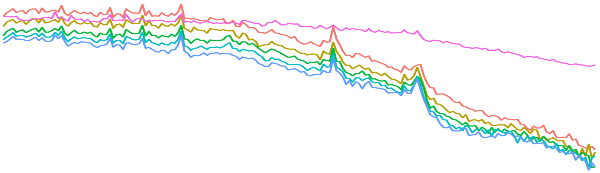
# Time series anomaly detection paradigms

1. **Identify anomalies within a time series in real time:** use one-step forecast distributions

2. **Identify anomalies within a time series in historical data:** use residual distributions from smoothing method

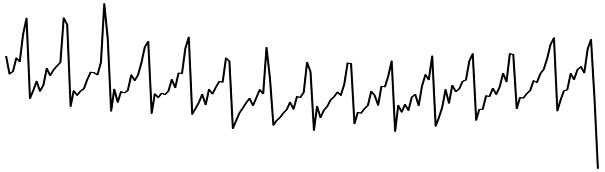


3. **Identify an anomalous time series in a collection of time series:** use feature-based approach

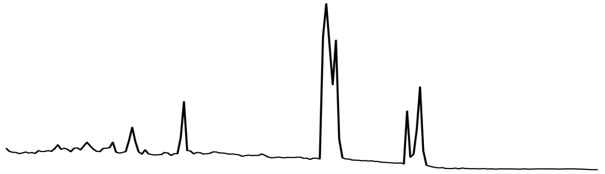


# Time series anomaly detection paradigms

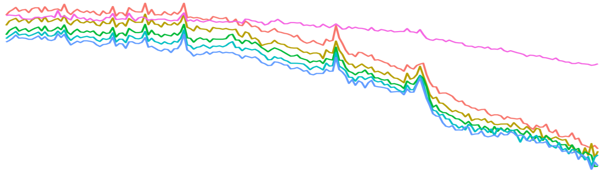
1. **Identify anomalies within a time series in real time:** use one-step forecast distributions



2. **Identify anomalies within a time series in historical data:** use residual distributions from smoothing method



3. **Identify an anomalous time series in a collection of time series:** use feature-based approach

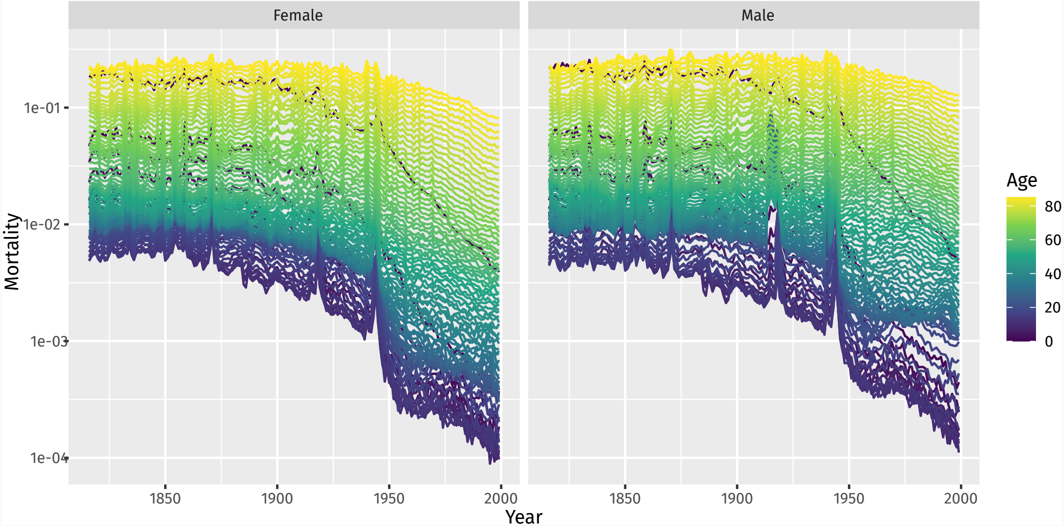


# Example: French mortality

```
fr_mortality
```

```
# A tsibble: 31,648 x 4 [1Y]
# Key:       Age, Sex [172]
    Year   Age Sex    Mortality
   <int> <int> <chr>      <dbl>
 1  1816     0 Female     0.187
 2  1817     0 Female     0.182
 3  1818     0 Female     0.186
 4  1819     0 Female     0.197
 5  1820     0 Female     0.181
 6  1821     0 Female     0.182
 7  1822     0 Female     0.207
 8  1823     0 Female     0.192
 9  1824     0 Female     0.199
10  1825     0 Female     0.194
# i 31,638 more rows
```

# Example: French mortality



# Example: French mortality

```
fr_fit <- fr_mortality |>
  model(stl = STL(log(Mortality)))
```

```
fr_fit
```

```
# A mable: 172 x 3
# Key:     Age, Sex [172]
     Age Sex       stl
   <int> <chr>  <model>
 1     0 Female   <STL>
 2     0 Male     <STL>
 3     1 Female   <STL>
 4     1 Male     <STL>
 5     2 Female   <STL>
 6     2 Male     <STL>
 7     3 Female   <STL>
 8     3 Male     <STL>
 9     4 Female   <STL>
10     4 Male     <STL>
```

# Example: French mortality

```r
augment(fr_fit)
```

```
# A tsibble: 31,648 x 8 [1Y]
# Key:       Age, Sex, .model [172]
     Age Sex    .model  Year Mortality .fitted   .resid  .innov
   <int> <chr>  <chr>  <int>     <dbl>   <dbl>    <dbl>   <dbl>
 1     0 Female stl     1816     0.187   0.193 -0.00650 -0.0342
 2     0 Female stl     1817     0.182   0.193 -0.0108  -0.0580
 3     0 Female stl     1818     0.186   0.192 -0.00595 -0.0316
 4     0 Female stl     1819     0.197   0.191  0.00603  0.0311
 5     0 Female stl     1820     0.181   0.190 -0.00895 -0.0483
 6     0 Female stl     1821     0.182   0.189 -0.00713 -0.0385
 7     0 Female stl     1822     0.207   0.188  0.0192   0.0973
 8     0 Female stl     1823     0.192   0.187  0.00500  0.0263
 9     0 Female stl     1824     0.199   0.186  0.0123   0.0639
10     0 Female stl     1825     0.194   0.185  0.00905  0.0477
# i 31,638 more rows
```

# Example: French mortality

```
fr_sigma <- augment(fr_fit) |>
  group_by(Age, Sex) |>
  summarise(sigma = IQR(.innov)/1.349, .groups = "drop")
```

```
# A tibble: 172 x 3
     Age Sex     sigma
   <int> <chr>   <dbl>
 1     0 Female 0.0643
 2     0 Male   0.0616
 3     1 Female 0.0894
 4     1 Male   0.0788
 5     2 Female 0.0900
 6     2 Male   0.0931
 7     3 Female 0.0925
 8     3 Male   0.0864
 9     4 Female 0.0963
10     4 Male   0.0931
# i 162 more rows
```

# Example: French mortality

```
fr_scores <- augment(fr_fit) |>
  left_join(fr_sigma) |>
  mutate(
    s = -log(dnorm(.innov / sigma)),
    prob = lookout(density_scores = s, threshold_probability = 0.9)
  )
```

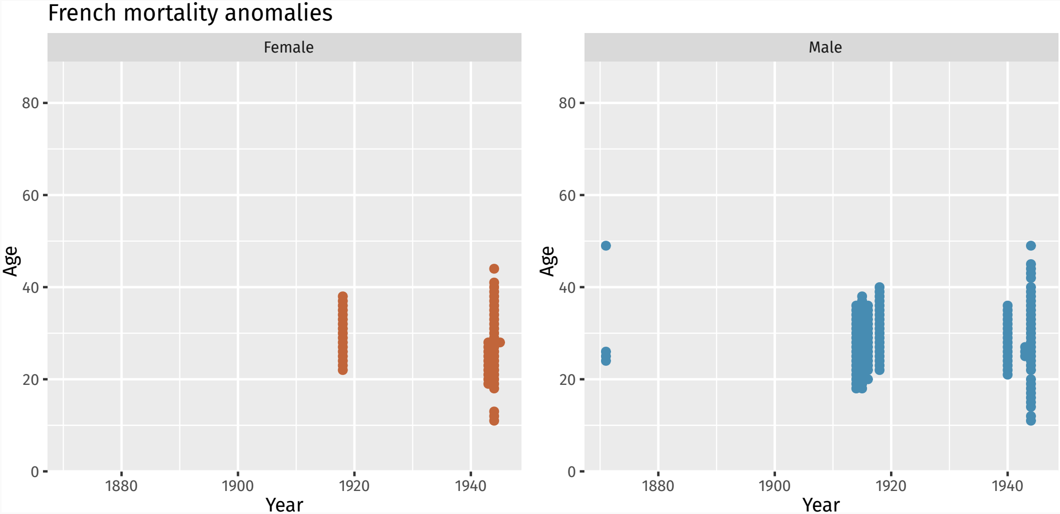
```
# A tibble: 31,648 x 7
     Age Sex     Year Mortality  .innov     s  prob
   <int> <chr>  <int>     <dbl>   <dbl> <dbl> <dbl>
 1     0 Female  1816     0.187 -0.0342  1.06     1
 2     0 Female  1817     0.182 -0.0580  1.32     1
 3     0 Female  1818     0.186 -0.0316  1.04     1
 4     0 Female  1819     0.197  0.0311  1.04     1
 5     0 Female  1820     0.181 -0.0483  1.20     1
 6     0 Female  1821     0.182 -0.0385  1.10     1
 7     0 Female  1822     0.207  0.0973  2.06     1
 8     0 Female  1823     0.192  0.0263  1.00     1
 9     0 Female  1824     0.199  0.0639  1.41     1
```

# Example: French mortality

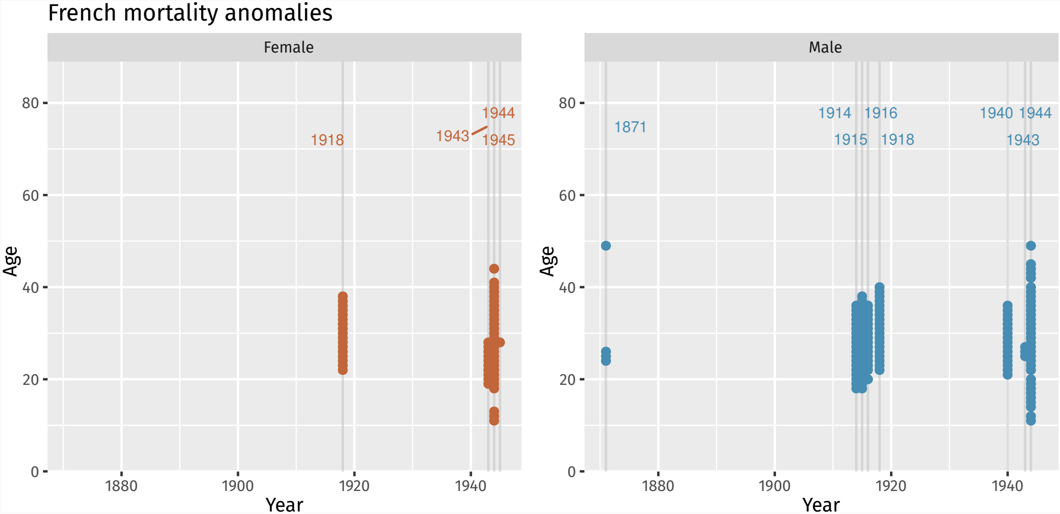
```r
fr_scores |> arrange(prob)
```

```
# A tibble: 31,648 x 7
     Age Sex     Year Mortality .innov     s    prob
   <int> <chr>  <int>     <dbl>  <dbl> <dbl>   <dbl>
 1    28 Female  1944    0.0170   1.45  373. 0.00737
 2    25 Female  1944    0.0191   1.59  331. 0.00831
 3    26 Female  1944    0.0176   1.50  266. 0.0104
 4    24 Female  1944    0.0150   1.40  259. 0.0106
 5    27 Female  1944    0.0178   1.50  228. 0.0121
 6    25 Male    1944    0.0432   1.89  170. 0.0163
 7    18 Male    1914    0.0798   2.06  170. 0.0163
 8    21 Female  1944    0.0120   1.29  168. 0.0165
 9    27 Male    1944    0.0388   1.78  168. 0.0165
10    23 Female  1944    0.0134   1.29  167. 0.0166
# i 31,638 more rows
```

# Example: French mortality

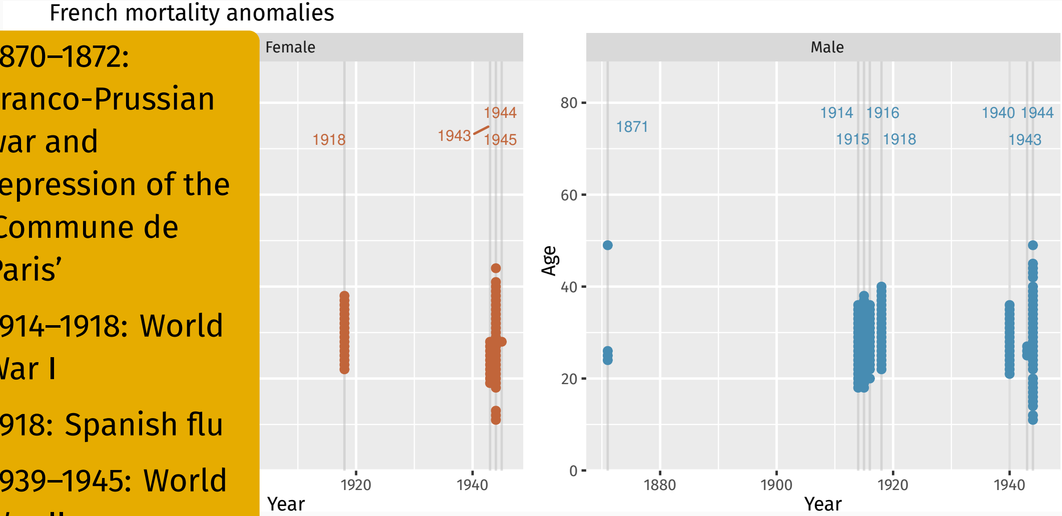


# Example: French mortality

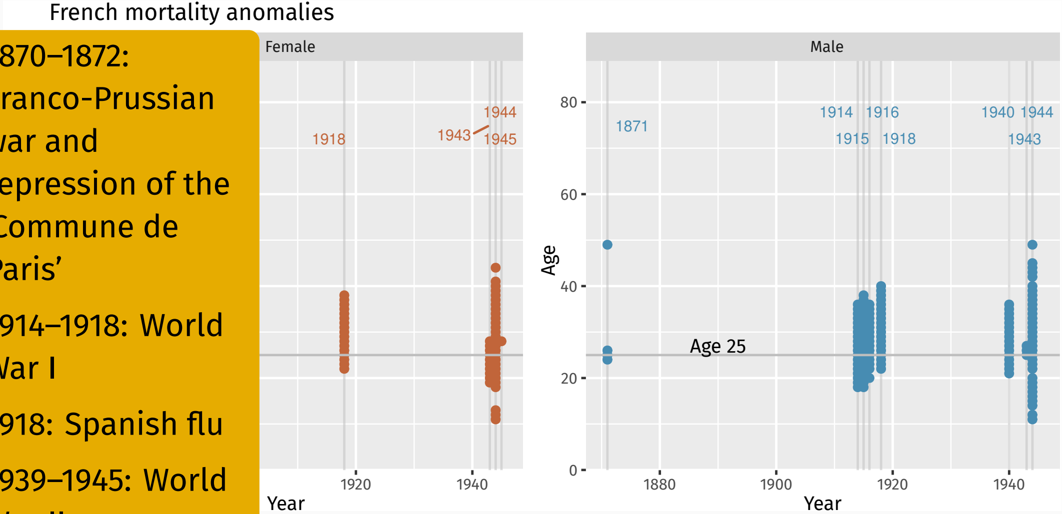


# Example: French mortality

- 1870–1872: Franco-Prussian war and repression of the ‘Commune de Paris’
- 1914–1918: World War I
- 1918: Spanish flu
- 1939–1945: World War II

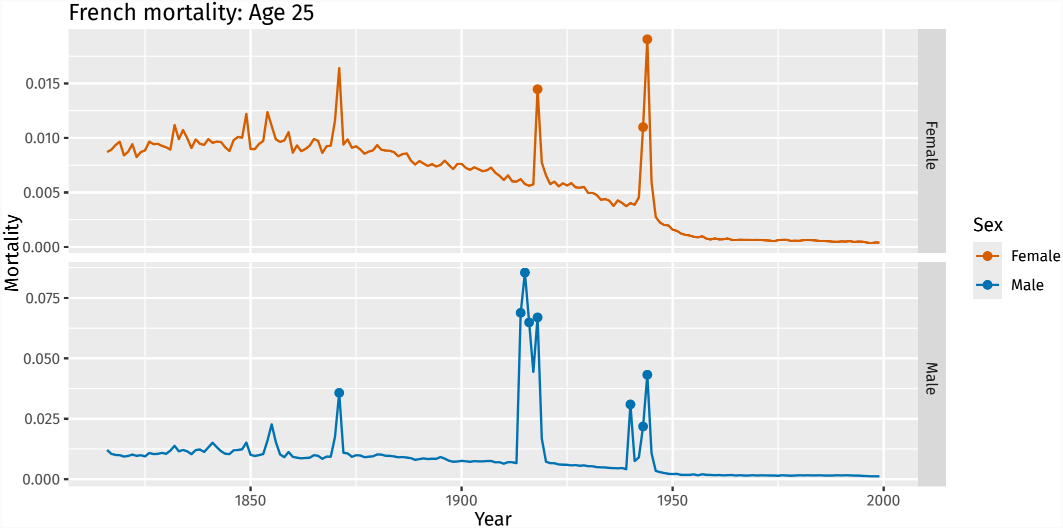


# Example: French mortality




- 1870–1872: Franco-Prussian war and repression of the ‘Commune de Paris’
- 1914–1918: World War I
- 1918: Spanish flu
- 1939–1945: World War II

# Example: French mortality



# More information

# More information

**Slides:** robjhyndman.com/isf2024

**Incomplete book:** OTexts.com/weird

**fable:** fable.tidyverts.org

**weird:** pkg.robjhyndman.com/weird

# More information

| | |
|---|---|
| **Slides:** | robjhyndman.com/isf2024 |
| **Incomplete book:** | OTexts.com/weird |
| **fable:** | fable.tidyverts.org |
| **weird:** | pkg.robjhyndman.com/weird |


## Find me at ...

robjhyndman.com
@robjhyndman
@robjhyndman
rob.hyndman@monash.edu