# Probabilistic forecasts for anomaly detection

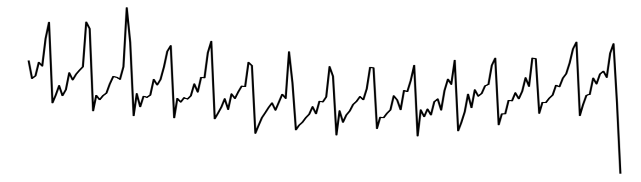
Rob J Hyndman

3 July 2024

MONASH University

# Time series anomaly detection paradigms

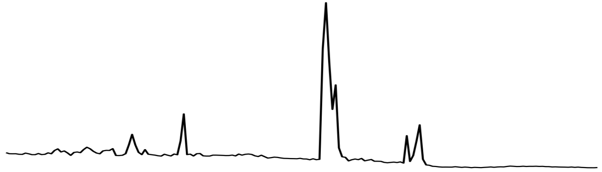
1. **Identify anomalies within a time series in real time:** use one-step forecast distributions



# Time series anomaly detection paradigms

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

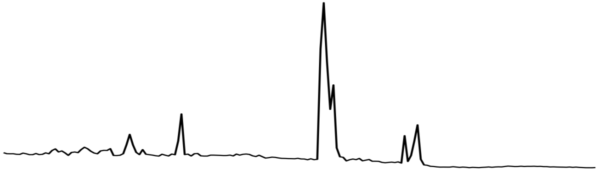
2. **Identify anomalies within a time series in historical data:** use residual distributions



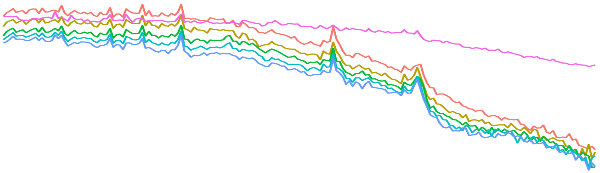
# Time series anomaly detection paradigms

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



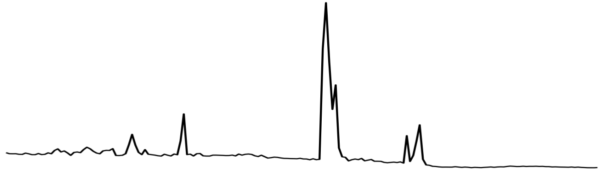
3. **Identify an anomalous time series in a collection of time series:**
   use feature-based approach



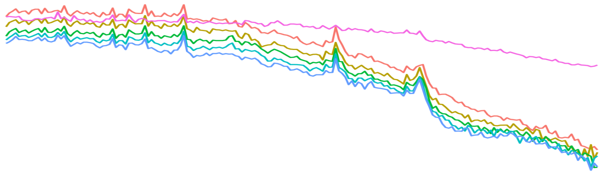
# Time series anomaly detection paradigms

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



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

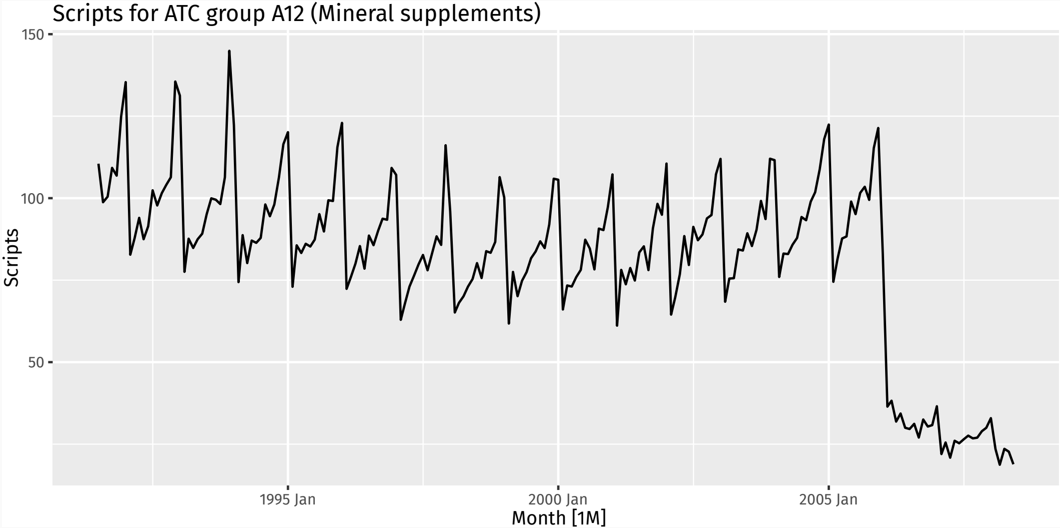


# Australian PBS data

```
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
```

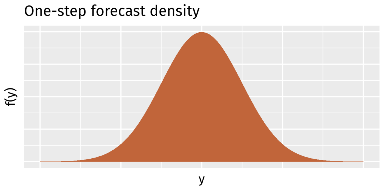
# Australian PBS data



# Anomaly score distribution

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

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



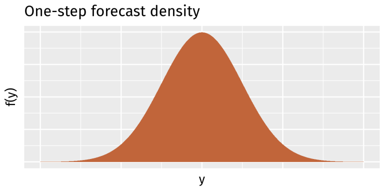
# Anomaly score distribution

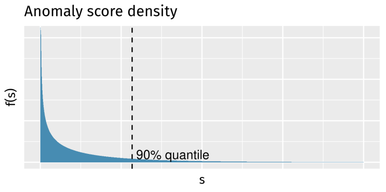
## One-step forecast distribution: $N(\mu_t, \sigma^2)$

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

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

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



# Anomaly score distribution

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

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

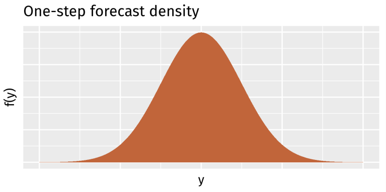
## Anomaly score distribution: $S \sim \frac{1}{2}\chi_1^2 + c$

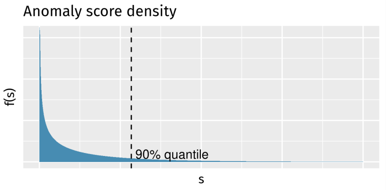
$$s_t = -\log f(y_t | y_1, \dots, y_{t-1}) = \frac{1}{2}\left(\frac{y_t - \mu_t}{2\sigma}\right)^2 + \frac{1}{2}\log(2\pi\sigma^2)$$

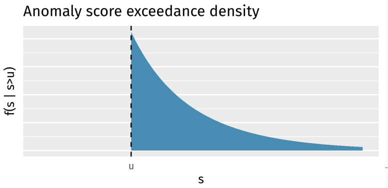
## Extreme anomaly score distribution

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

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



# Anomaly detection algorithm

For each $t$:

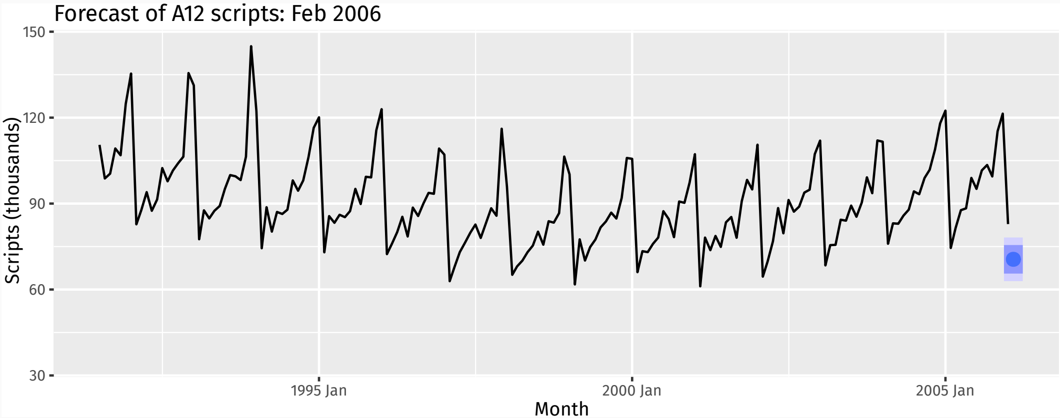
- Estimate one-step forecast density: $f(y_t|y_1, \dots, y_{t-1})$.
- Anomaly score: $s_t = -\log \hat{f}(y_t|y_1, \dots, y_{t-1})$.
- High anomaly score indicates potential anomaly.
- Fit a Generalized Pareto Distribution to the top 10% of anomaly scores seen so far.
- $y_t$ is anomaly if $P(S > s_t)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

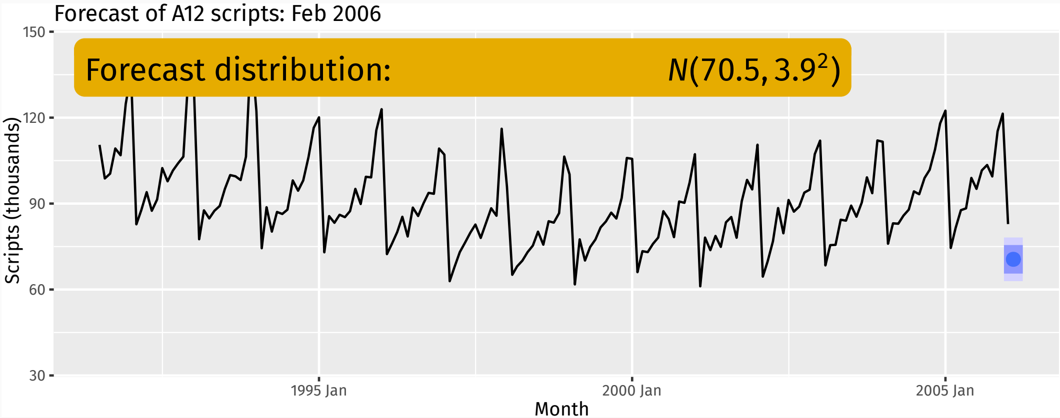
# Example

```
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)
```



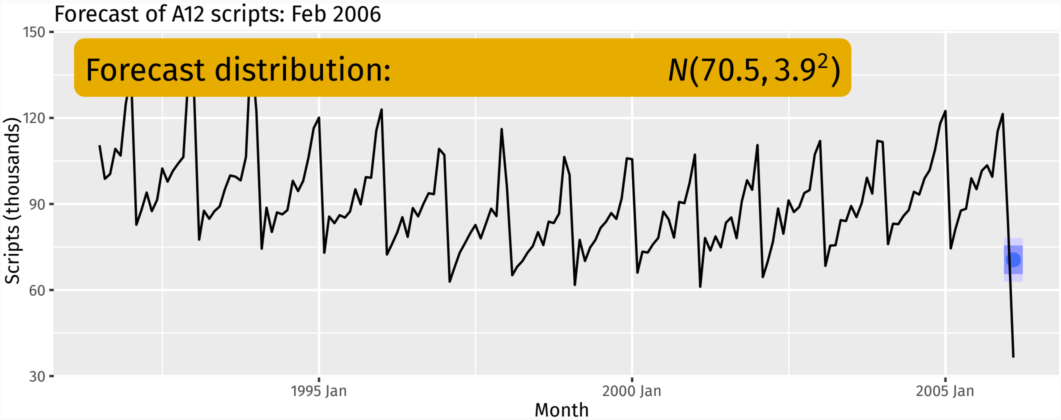
# Example

```
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)
```



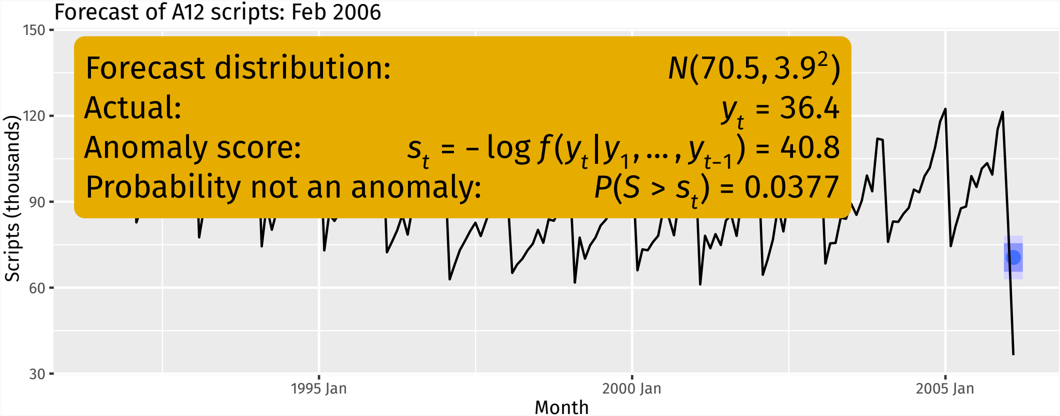
# Example

```
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)
```



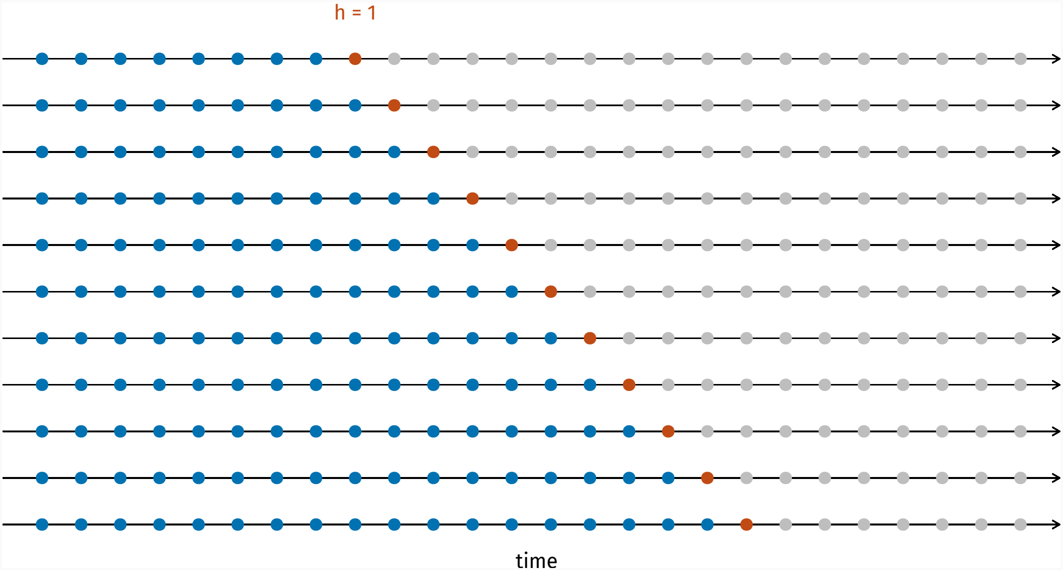
# Example

```
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: | $s_t = -\log f(y_t \mid y_1, \dots, y_{t-1}) = 40.8$ |
| Probability not an anomaly: | $P(S > s_t) = 0.0377$ |

# 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

```r
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")) |>
  group_by(.id) |>
  mutate(
    s = -log_likelihood(Scripts, actual),
    prob = lookout(density_scores = s, threshold = 0.9)
  ) |>
  ungroup()
```

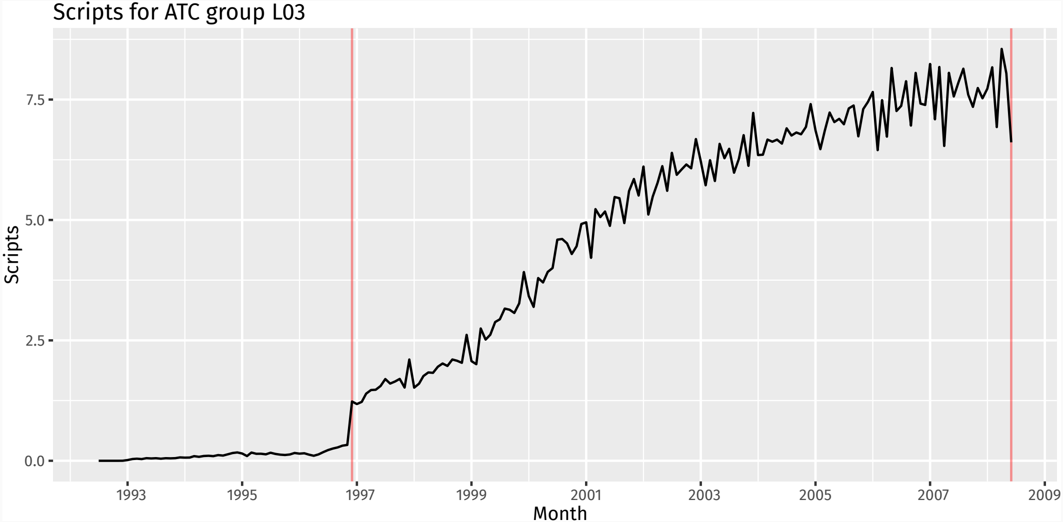
```
# A tsibble: 14,076 x 7 [1M]
# Key:       .id, ATC2 [14,076]
     .id ATC2     Month        Scripts actual     s  prob
   <int> <chr>    <mth>         <dist>  <dbl> <dbl> <dbl>
 1     1 A01   1994 Jul    N(23, 2.1)  20.9   2.46 1
 2     1 A02   1994 Jul  N(590, 1054) 516.    6.97 0.575
 3     1 A03   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
```

# PBS anomalies

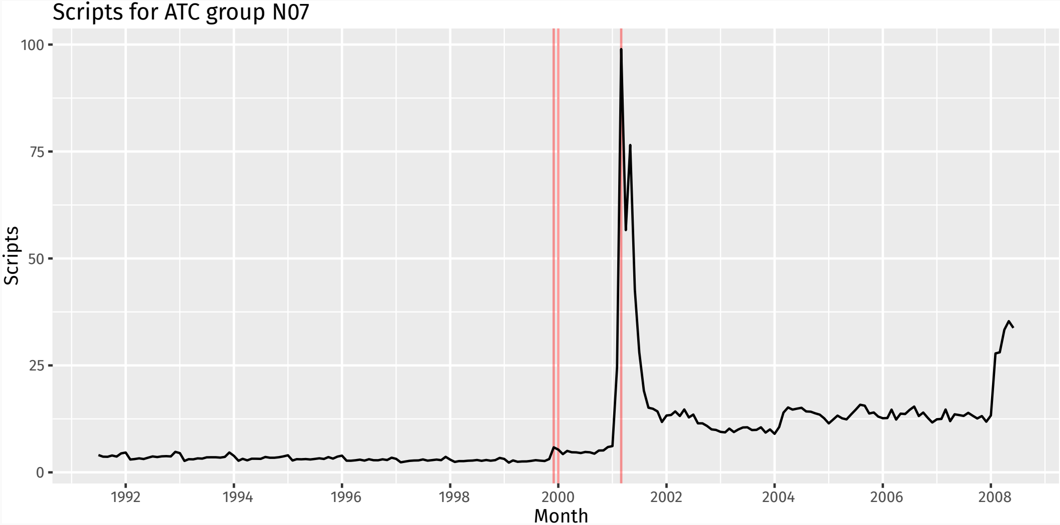
```
pbs_scores |> filter(prob < 0.05)
```

```
# A tsibble: 149 x 7 [1M]
# Key:       .id, ATC2 [149]
     .id ATC2     Month           Scripts  actual     s       prob
   <int> <chr>    <mth>            <dist>   <dbl> <dbl>      <dbl>
 1     1 S01   1994 Jul     N(403, 356)    347.    8.29 0.0249
 2     2 J01   1994 Aug   N(1382, 5743)   1616.   10.0  0.0232
 3     3 J01   1994 Sep   N(1558, 8941)   1552.    5.47 0.0330
 4     4 J01   1994 Oct   N(1496, 7931)   1327.    7.21 0.0405
 5     5 H01   1994 Nov N(0.77, 0.0039)      1.05  7.89 0.0405
 6     6 C02   1994 Dec      N(142, 39)    108.   17.1  0.00190
 7     6 C07   1994 Dec     N(413, 324)    320.   17.0  0.0380
 8     7 C07   1995 Jan     N(414, 541)    328.   10.9  0.0000572
 9     8 R03   1995 Feb     N(416, 966)    507.    8.66 0.000770
10     9 C10   1995 Mar     N(205, 214)    260.   10.6  0.0261
# i 139 more rows
```

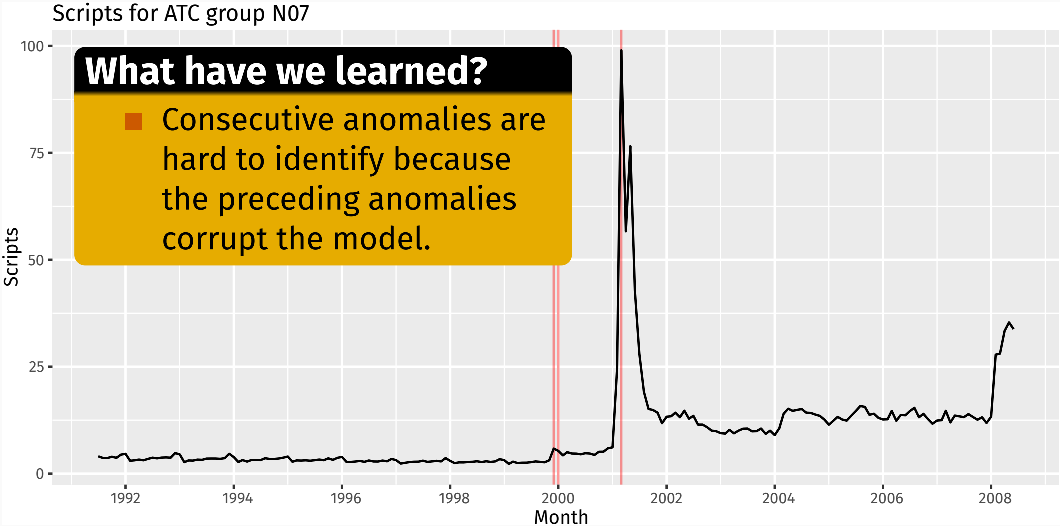
# PBS anomalies



# PBS anomalies



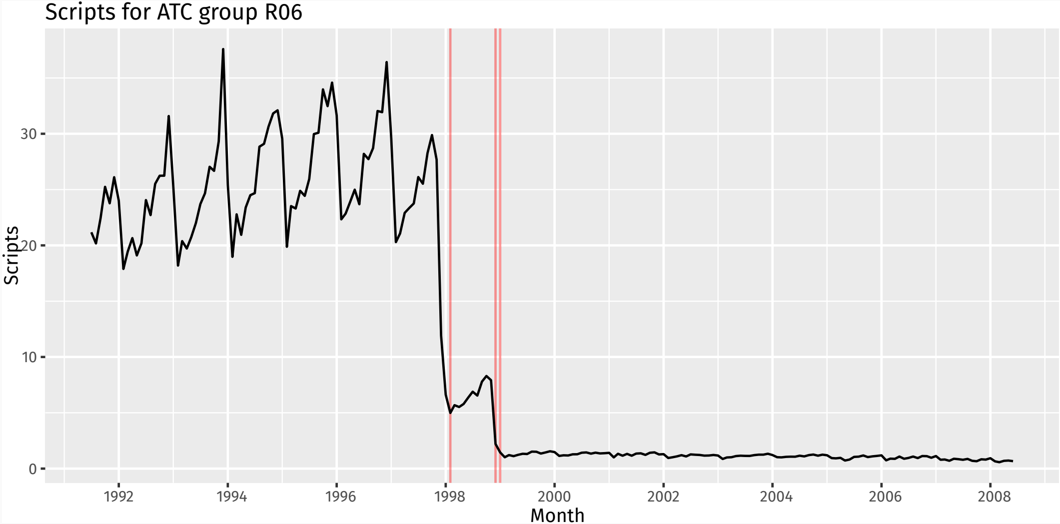
# PBS anomalies



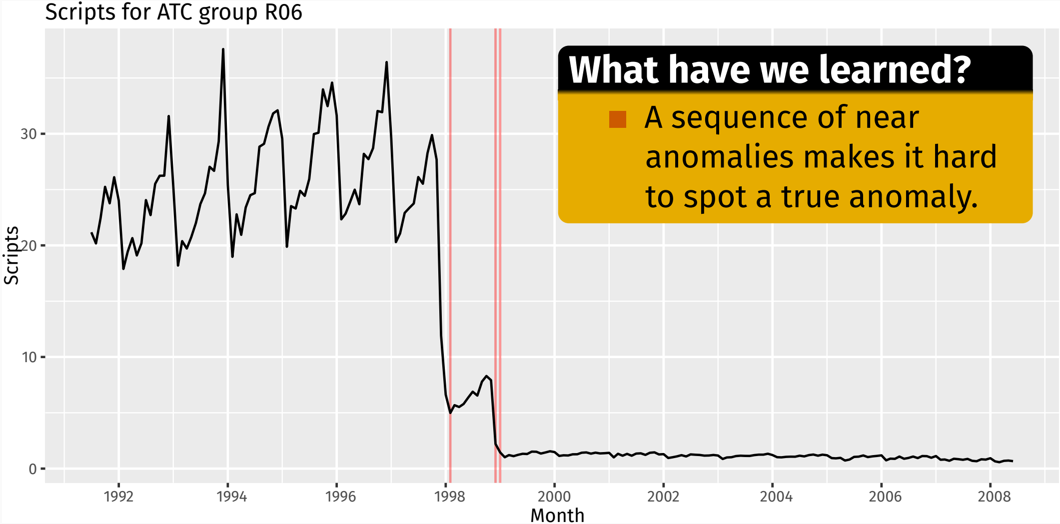

## What have we learned?

- Consecutive anomalies are hard to identify because the preceding anomalies corrupt the model.

# PBS anomalies



# PBS anomalies




**What have we learned?**

- A sequence of near anomalies makes it hard to spot a true anomaly.

# Modified 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.
- **If $y_t$ is anomaly, set $y_t$ to missing.**
- **Repeat**

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

# Time series anomaly detection paradigms

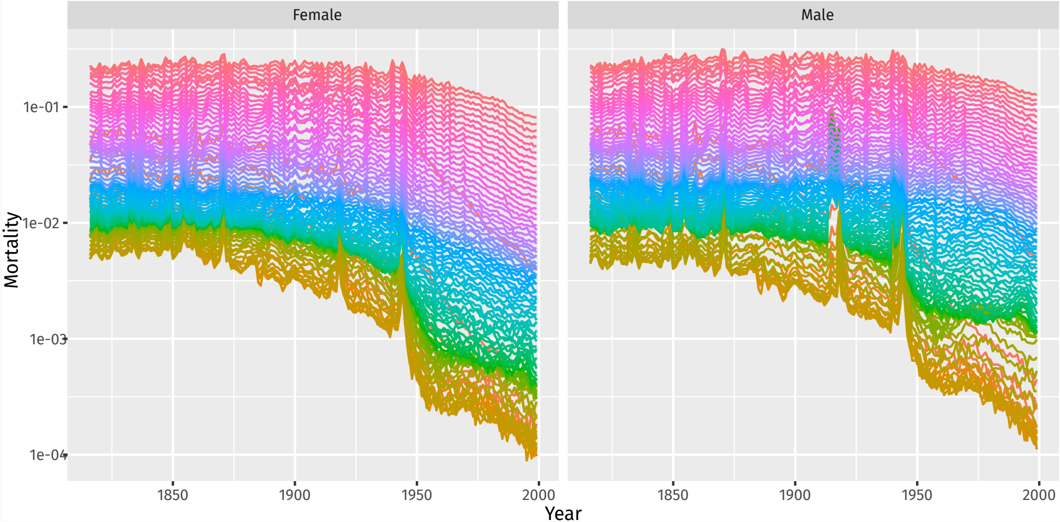
1. **Identify anomalies within a time series in real time:** use one-step forecast distributions
2. **Identify anomalies within a time series in historical data:** use residual distributions
3. **Identify an anomalous time series in a collection of time series:** use feature-based approach

# Example: French mortality

```r
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

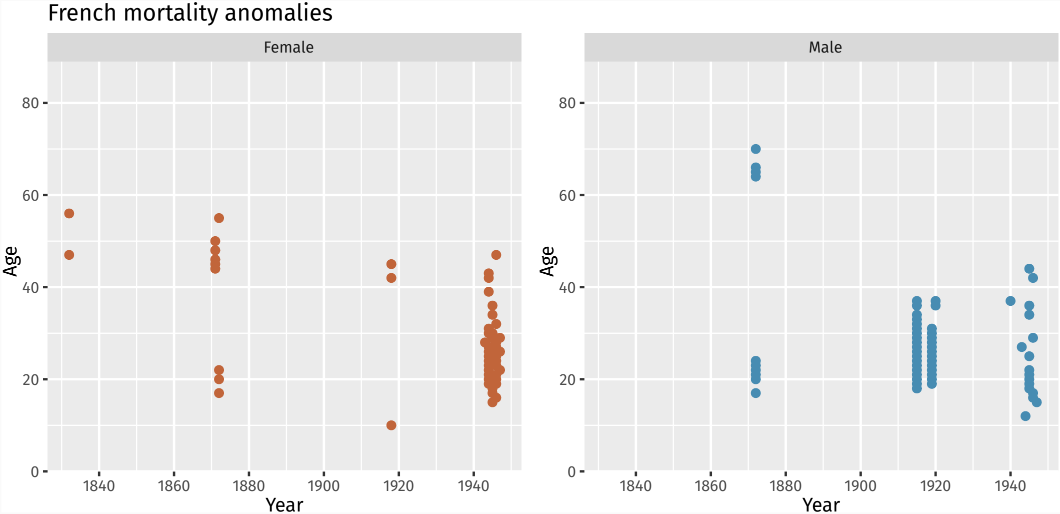
```r
fr_fit <- fr_mortality |> model(arima = ARIMA(log(Mortality)))
fr_sigma <- augment(fr_fit) |>
  group_by(Age, Sex) |>
  summarise(sigma = sd(.innov, na.rm = TRUE), .groups = "drop")
fr_scores <- augment(fr_fit) |>
  left_join(fr_sigma) |>
  mutate(
    s = -log(dnorm(.innov / sigma)),
    prob = lookout(density_scores = s, threshold_probability = 0.9)
  )
```

# Example: French mortality

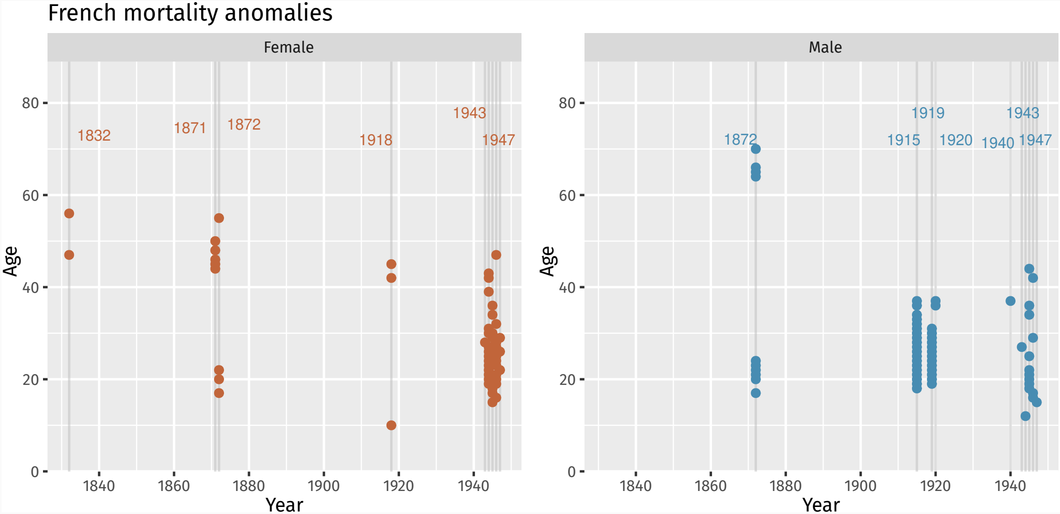
```
fr_scores |> arrange(prob)
```

```
# A tibble: 31,648 x 7
     Age Sex    Year Mortality .innov     s     prob
   <int> <chr> <int>     <dbl>  <dbl> <dbl>    <dbl>
 1    18 Male   1914    0.0798   2.77  55.6 0.000487
 2    19 Male   1914    0.0906   2.74  42.2 0.00110
 3    32 Male   1914    0.0550   2.00  29.3 0.00311
 4    30 Male   1914    0.0591   2.13  29.3 0.00312
 5    31 Male   1914    0.0578   2.03  28.9 0.00325
 6    20 Male   1914    0.0741   2.50  28.1 0.00352
 7    29 Male   1914    0.0597   2.13  26.9 0.00398
 8    33 Male   1914    0.0493   1.83  26.8 0.00401
 9    28 Male   1914    0.0611   2.15  24.4 0.00522
10    27 Male   1914    0.0613   2.19  24.3 0.00528
# i 31,638 more rows
```

# Example: French mortality



# Example: French mortality



# More information

- **Slides:** robjhyndman.com/isf2024
- **Incomplete book:** OTexts.com/weird