Lecture 6

Using simulation to evaluate and extend our methods

ADF test critical values

ADF test critical values: type="drift"

$$\Delta Y_t = \beta_0 + \varphi Y_{t-1} + U_t$$

$$H_0 : \varphi = 0 \quad \text{vs} \quad H_1 : \varphi < 0$$

```r
library(urca)
SignificanceLevels <- c(0.01, 0.05, 0.1)
cv_drift <- qunitroot(p=SignificanceLevels,
                      trend="c", statistic=
```

```
              1%     5%    10%
cv_drift -3.430 -2.861 -2.567
```

ADF test critical values: type="trend"

$$\Delta Y_t = \beta_0 + \beta_1 t + \varphi Y_{t-1} + U_t$$

$$H_0 : \varphi = 0 \quad \text{vs} \quad H_1 : \varphi < 0$$

```
cv_trend <- qunitroot(p=SignificanceLevels,
                              trend="ct", statistic
```

```
              1%      5%     10%
cv_trend -3.958 -3.410 -3.127
```

ADF test critical values

```
               1%      5%      10%
cv_drift -3.430 -2.861 -2.567
cv_trend -3.958 -3.410 -3.127
```

- These are *asymptotic* (approximate) critical values. Like “1.96” for standard t tests.
- Approximate critical values for any n can also be found. Like t critical values.

ADF test critical values

Example. $n = 100$

```r
cv_drift <- qunitroot(p=SignificanceLevels,
                                trend="c", N=100, sta
```

```
                1%      5%     10%
cv_drift -3.497 -2.891 -2.582
```

ADF test critical values

```
cv_drift
```

	1%	5%	10%
n=50	-3.568	-2.921	-2.599
n=100	-3.497	-2.891	-2.582
n=200	-3.463	-2.876	-2.574
n=400	-3.447	-2.869	-2.571
n=800	-3.438	-2.865	-2.569
n=Infinity	-3.430	-2.861	-2.567

Practically, allowing critical values to vary with n makes very little difference.

ADF test critical values

```
cv_trend

                 1%     5%     10%
n=50       -4.153 -3.502 -3.181
n=100      -4.052 -3.455 -3.153
n=200      -4.005 -3.432 -3.140
n=400      -3.981 -3.421 -3.133
n=800      -3.969 -3.415 -3.130
n=Infinity -3.958 -3.410 -3.127
```

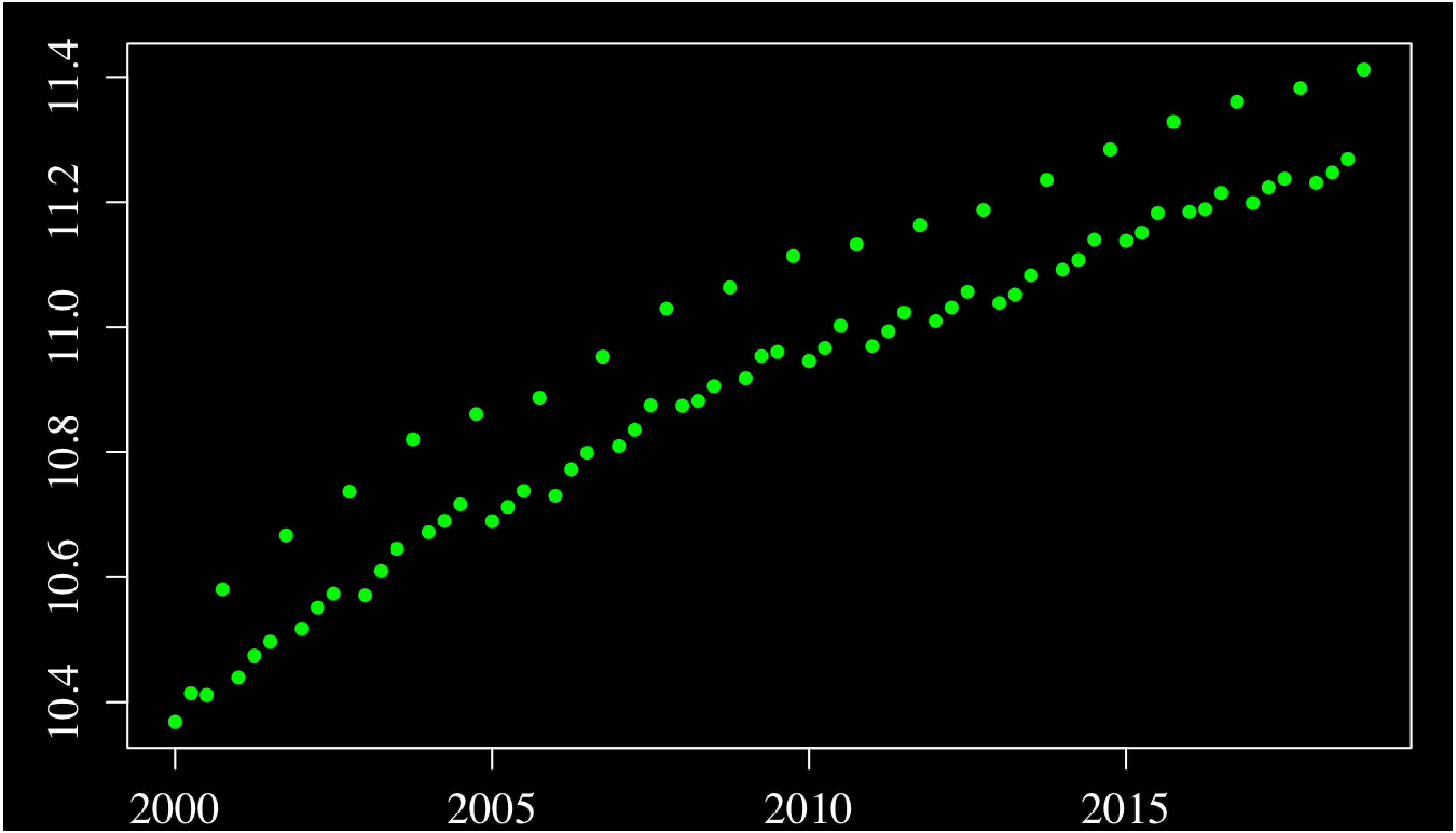
Practically, allowing critical values to vary with n makes very little difference.

ADF test critical values

```
               1%     5%     10%
cv_drift -3.430 -2.861 -2.567
cv_trend -3.958 -3.410 -3.127
```

- *Different* critical values for different trend functions.
  This is not the case for standard tests.
- These differences are large enough to matter in practice. We need the right critical value for the trend function used!

log of Retail Sales, 2000q1 - 2018q4



Trend function for $Y_t = \log \text{Retail Sales}$:

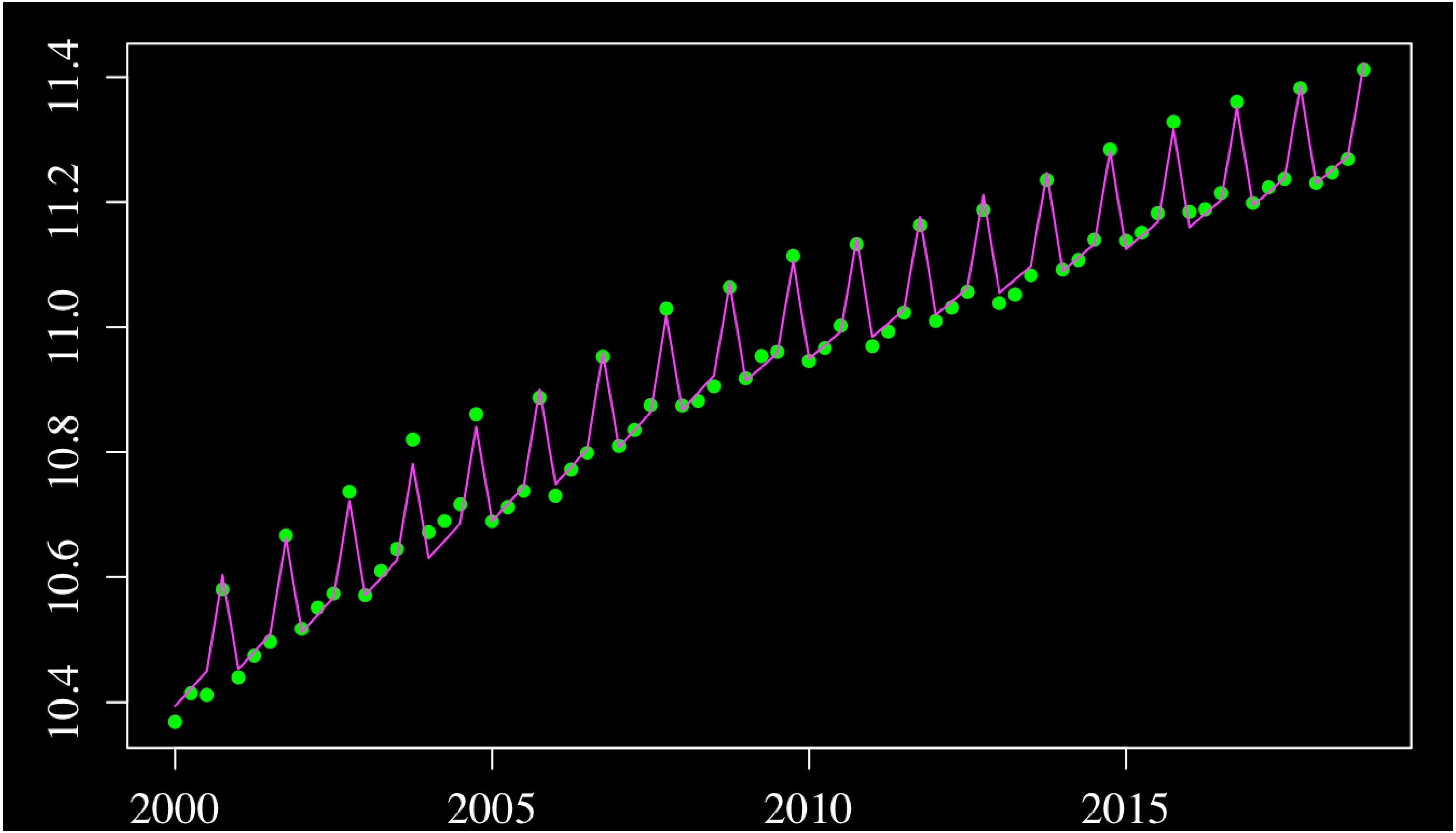
$$Y_t = \beta_0 + \beta_1 \text{Time}_t + \beta_2 \text{TimePostGFC}_t$$
$$+\delta_1 Q_{1,t} + \delta_2 Q_{2,t} + \delta_3 Q_{3,t} + Z_t$$

where

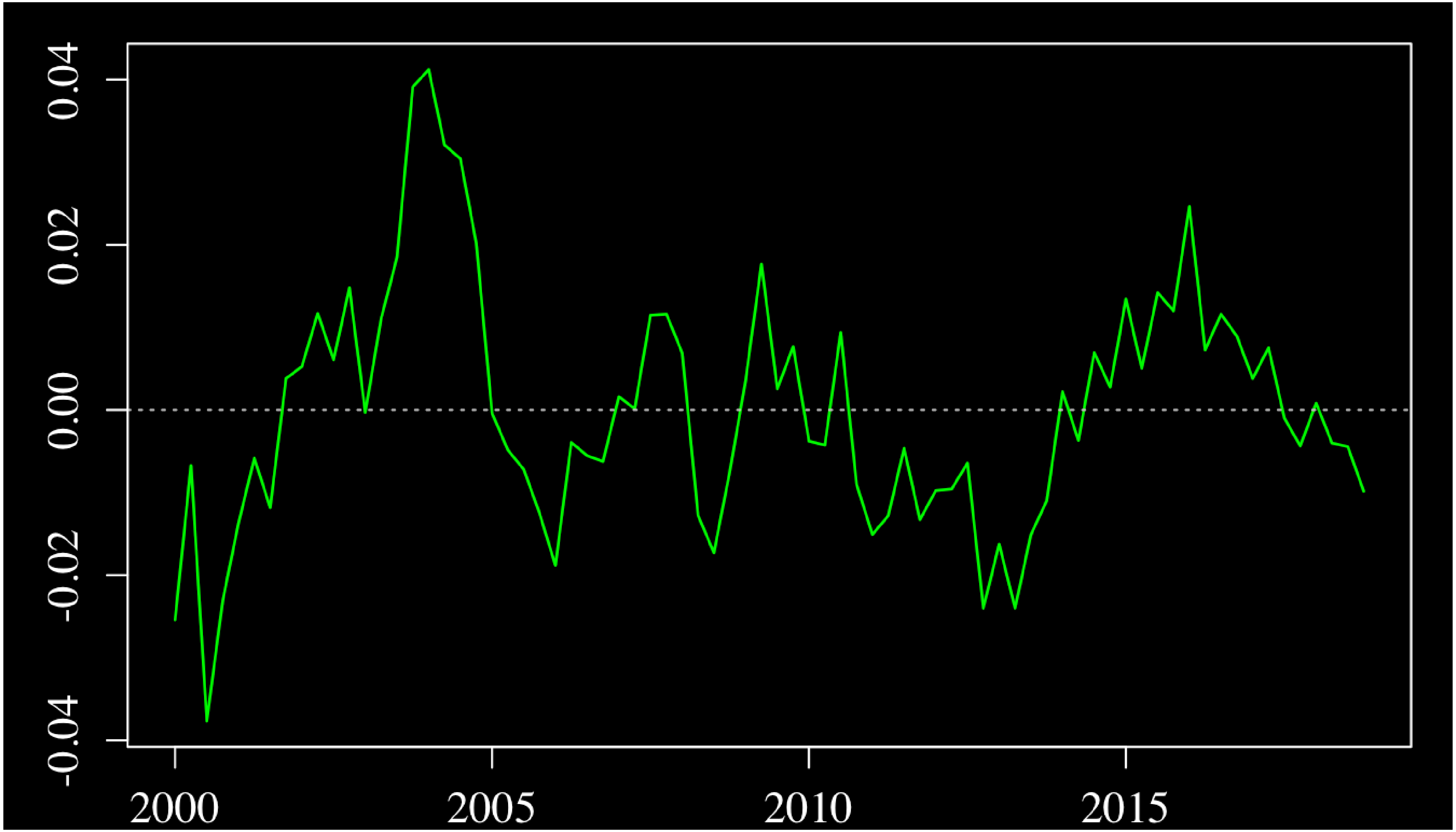
$$\text{TimePostGFC}_t$$
$$= (\text{Time}_t - 2008.5) \times 1(\text{Time}_t > 2008.5)$$

How to test for a unit root in Z_t?

log of Retail Sales, 2000q1 - 2018q4



Deviations from trend function: $\widehat{Z}_t$



Trend function for $Y_t = \log \text{Retail Sales}$:

$$Y_t = \beta_0 + \beta_1 \text{Time}_t + \beta_2 \text{TimePostGFC}_t \\ + \delta_1 Q_{1,t} + \delta_2 Q_{2,t} + \delta_3 Q_{3,t} + Z_t$$

- ur.df and qunitroot do not allow for custom trend functions

Trend function for $Y_t = \log \text{Retail Sales}$:

$$
\begin{aligned}
Y_t = \beta_0 + \beta_1 \text{Time}_t + \beta_2 \text{TimePostGFC}_t \\
+ \delta_1 Q_{1,t} + \delta_2 Q_{2,t} + \delta_3 Q_{3,t} + Z_t
\end{aligned}
$$

- ur.df and qunitroot do not allow for custom trend functions
- We need to **calculate the ADF test** allowing for this trend function.
- We need to find the **appropriate critical value** for this trend function.

What is a critical value anyway?

Components of a generic hypothesis test

- Hypotheses: H_0 vs H_1
- Significance level α
- A test statistic T_n
- A decision rule such as "reject H_0 if $T_n < c_\alpha$"

Components of a generic hypothesis test

- Hypotheses: H_0 vs H_1
- Significance level α
- A test statistic T_n
- A decision rule such as "reject H_0 if $T_n < c_\alpha$"
  (or "reject H_0 if $T_n > c_\alpha$"
  or "reject H_0 if $|T_n| > c_\alpha$")

Components of a generic hypothesis test

- A decision rule such as "reject H_0 if $T_n < c_\alpha$"

Critical value with significance level α :

$$P(T_n < c_\alpha \mid H_0 \text{ true}) = \alpha$$

Components of a generic hypothesis test

- A decision rule such as "reject H_0 if $T_n < c_\alpha$"

Critical value with significance level α :

$$P(T_n < c_\alpha \mid H_0 \text{ true}) = \alpha$$

- Reject H_0 when H_0 is true = Type 1 Error

Components of a generic hypothesis test

- A decision rule such as "reject H_0 if $T_n < c_\alpha$"

Critical value with significance level α :

$$P(T_n < c_\alpha \mid H_0 \text{ true}) = \alpha$$

- Frequentist probability: proportion of samples in which H_0 is rejected in repeated draws from the population distribution when H_0 is true.
- We can simulate this probability!

Example: simulation of ADF critical value

Simulated draw from a random walk

```r
Y <- arima.sim(n=100,
         model=list(order=c(0,1,0)))
```

```
Time Series:
Start = 0
End = 100
Frequency = 1
  [1]  0.000  1.371  0.806  1.169  1.802
  [6]  2.207  2.100  3.612  3.517  5.536
 [11]  5.473  6.778  9.064  7.676  7.397
 [16]  7.264  7.899  7.615  4.959  2.518
```

Simulated draw from a random walk

```
Y <- arima.sim(n=100,
        model=list(order=c(0,1,0)))
```

$\text{ARIMA}(0,1,0)$

```
Time Series:
Start = 0
End = 100
Frequency = 1
  [1]  0.000  1.371  0.806  1.169  1.802
  [6]  2.207  2.100  3.612  3.517  5.536
 [11]  5.473  6.778  9.064  7.676  7.397
 [16]  7.264  7.899  7.615  4.959  2.518
```

Simulated draw from a random walk

```r
Y <- arima.sim(n=100,
        model=list(order=c(0,1,0)))
```

```
Time Series:
Start = 0
End = 100
Frequency = 1
  [1]  0.000  1.371  0.806  1.169  1.802
  [6]  2.207  2.100  3.612  3.517  5.536
 [11]  5.473  6.778  9.064  7.676  7.397
 [16]  7.264  7.899  7.615  4.959  2.518
```

Sample size

Simulated draw from a random walk

```r
Y <- arima.sim(n=100,
        model=list(order=c(0,1,0)))
```

```
Time Series:
Start = 0
End = 100
Frequency = 1
  [1]  0.000  1.371  0.806  1.169  1.802
  [6]  2.207  2.100  3.612  3.517  5.536
 [11]  5.473  6.778  9.064  7.676  7.397
 [16]  7.264  7.899  7.615  4.959  2.518
```

includes $Y_0 = 0$

Simulated draw from a random walk

```
Y <- arima.sim(n=100,
         model=list(order=c(0,1,0)))[-1]
```

Remove Y_0

Simulated draw from a random walk

```
Y <- arima.sim(n=100,
          model=list(order=c(0,1,0)))[-1]
```

```
 [1]  1.371  0.806  1.169  1.802  2.207
 [6]  2.100  3.612  3.517  5.536  5.473
[11]  6.778  9.064  7.676  7.397  7.264
[16]  7.899  7.615  4.959  2.518  3.838
[21]  3.532  1.750  1.579  2.793  4.688
[26]  4.258  4.001  2.238  2.698  2.058
[31]  2.513  3.218  4.253  3.644  4.149
[36]  2.432  1.648  0.797 -1.618 -1.581
```

Simulated draw from a random walk

```
Y <- arima.sim(n=100,
        model=list(order=c(0,1,0)))[-1]
```

$$\text{ARIMA}(0,1,0): \quad Y_t = Y_{t-1} + U_t$$

Satisfies $H_0 : \phi_1 = 1$ (unit root) in

$$Y_t = \phi_1 Y_{t-1} + U_t$$

ADF test on simulated random walk

```r
Y <- arima.sim(n=100,
        model=list(order=c(0,1,0)))[-1]
ADF <- ur.df(Y, type="drift", lags=0)
```

Test equation:

```
          Estimate Std. Error t value Pr(>|t|)
drift       0.1755     0.1326  1.3236   0.1887
z.lag.1   -0.0701     0.0374 -1.8745   0.0639
```

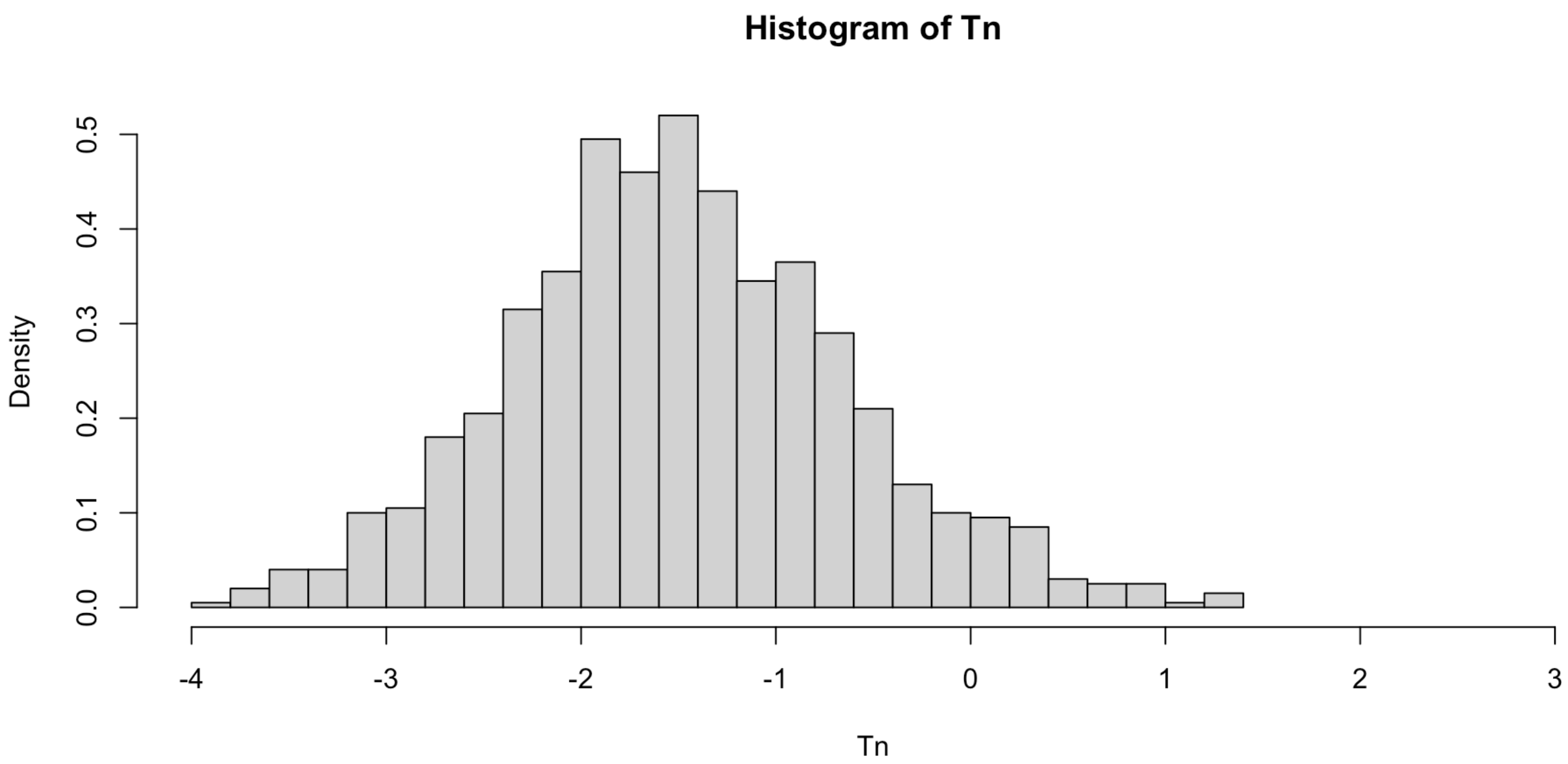
Test statistic : $T_n = -1.8745 \quad \uparrow$

Repeat ADF tests on simulated random walks

```
reps <- 1000
Tn <- matrix(nrow=reps, ncol=1)

# Repeat reps times
for (r in 1:reps){

  # Simulate random walk
  Y <- arima.sim(n=100,
      model=list(order=c(0,1,0)))[-1]

```

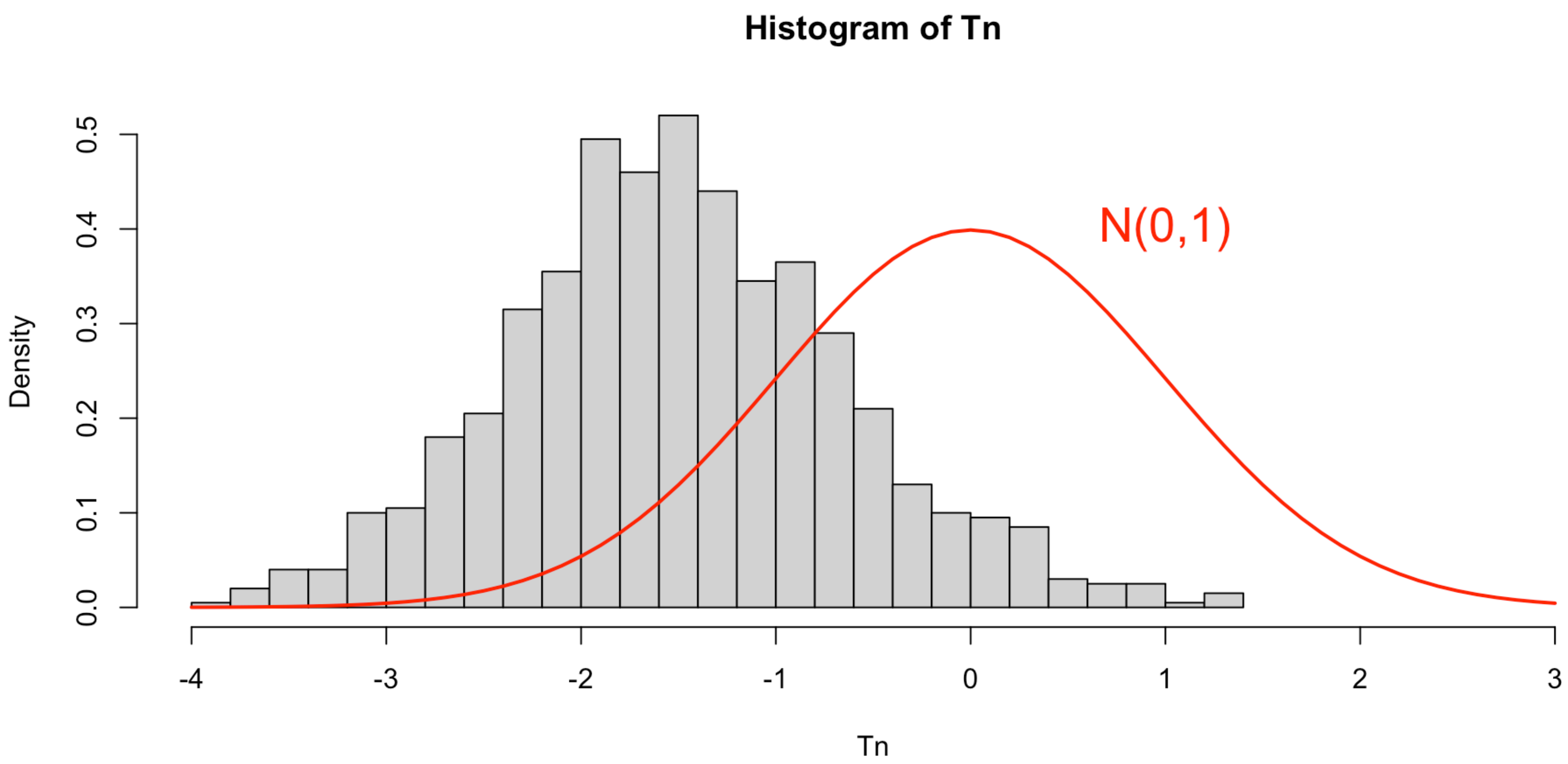
Sampling distribution of ADF test

```
1 hist(Tn)
```



Sampling distribution of ADF test

```
hist(Tn)
```



Simulated critical values of ADF test

```r
SignificanceLevels <- c(0.01, 0.05, 0.1)
cv_sim <- quantile(Tn,
                   probs=SignificanceLevels)
```

```
            1%     5%    10%
cv_sim   -3.462 -2.894 -2.587
```

Comparison:

```r
cv_drift <- qunitroot(p=SignificanceLevels,
            N=100, trend="c", statistic="t")
```

```
            1%     5%    10%
cv_drift -3.497 -2.891 -2.582
```

What is a p-value anyway?

Components of a generic hypothesis test

- Hypotheses: H_0 vs H_1
- Significance level α
- A test statistic T_n
- A decision rule such as "reject H_0 if $T_n < c_\alpha$"

Suppose we calculate the test statistic to be t.

p-value : $\quad p = P(\, T_n < t \mid H_0 \text{ true} \,)$

Decision rule: reject H_0 if $p < \alpha$.

Simulated *p*-value of ADF test

Eg. suppose we calculate $t = -1.8$.

Then $p = P(\, T_n < -1.8 \mid H_0 \text{ true} \,)$:

```
p_sim = mean(Tn < -1.8)
```

The proportion of the simulated statistics under H_0 that satisfy $T_n < -1.8$.

```
p_sim = 0.372
```

Simulated p-value of ADF test

Eg. suppose we calculate $t = -1.8$.

Then $p = P(\, T_n < -1.8 \mid H_0 \text{ true} \,)$:

```
p_sim = mean(Tn < -1.8)
```

```
p_sim = 0.372
```

Comparison:

```
p = punitroot(-1.8, N=100, trend="c",
                    statistic="t")
```

```
p     = 0.379
```

Notes on the ADF simulation process

$$\Delta Y_t = X_t'\beta + \varphi Y_{t-1} + \sum_{j=1}^{p-1} \psi_j \Delta Y_{t-j} + U_t$$

- The specification of X_t matters most.
- n has minor effect.
- p and $\psi_1, \ldots, \psi_{p-1}$: "asymptotically negligible".
- Distribution of U_t : "asymptotically negligible".

Notes on the ADF simulation process

Therefore under H_0 we simulate from

$$\Delta Y_t = X_t'\beta + \varphi Y_{t-1} + \sum_{j=1}^{p-1} \psi_j \Delta Y_{t-j} + U_t$$

Notes on the ADF simulation process

Therefore under $H_0 : \varphi = 0$ we simulate from

$$\Delta Y_t = X_t'\beta + \varphi Y_{t-1} + \sum_{j=1}^{p-1} \psi_j \Delta Y_{t-j} + U_t$$

Notes on the ADF simulation process

Therefore under $H_0 : \varphi = 0$ we simulate from

$$\Delta Y_t = X_t'\beta \qquad + \sum_{j=1}^{p-1} \psi_j \Delta Y_{t-j} + U_t$$

- p and $\psi_1, \ldots, \psi_{p-1}$: "asymptotically negligible".

Notes on the ADF simulation process

Therefore under $H_0 : \varphi = 0$ we simulate from

$$\Delta Y_t = X_t'\beta \qquad + \sum_{j=1}^{p-1} \psi_j \Delta Y_{t-j} + U_t$$

- p and $\psi_1, \ldots, \psi_{p-1}$: “asymptotically negligible”. Therefore set $p = 1$ ($\psi_j = 0$)

Notes on the ADF simulation process

Therefore under $H_0 : \varphi = 0$ we simulate from

$$\Delta Y_t = X_t' \beta \qquad\qquad\qquad\qquad + U_t$$

- Distribution of U_t : "asymptotically negligible".

Notes on the ADF simulation process

Therefore under $H_0 : \varphi = 0$ we simulate from

$$\Delta Y_t = X_t' \beta + U_t$$

- Distribution of U_t : "asymptotically negligible". We set $U_t \sim N(0, 1)$.

Notes on the ADF simulation process

Therefore under $H_0 : \varphi = 0$ we simulate from

$$\Delta Y_t = X_t' \beta + U_t, \quad U_t \sim N(0,1)$$

- The ADF test is *invariant* to β.
  (As long as we regress on X_t).

Notes on the ADF simulation process

Therefore under $H_0 : \varphi = 0$ we simulate from

$$\Delta Y_t = X_t'\beta + U_t, \quad U_t \sim N(0,1)$$

- The ADF test is *invariant* to β.
  (As long as we regress on X_t).
  Therefore we set $\beta = 0$.

Notes on the ADF simulation process

Therefore under $H_0 : \varphi = 0$ we simulate from

$$\Delta Y_t = +U_t, \quad U_t \sim N(0,1)$$

Summary

- Concepts of critical value and p-value.
- Simulation of critical/p-values by repeated generation from a model under H_0.
- Dependence of ADF critical/p-values on trend specification.