ITSx: Policy Analysis Using Interrupted Time Series

Week 4 Slides

Michael Law, Ph.D.
The University of British Columbia

COURSE OVERVIEW

Layout of the weeks

1. Introduction, setup, data sources
2. Single series interrupted time series analysis
3. ITS with a control group
4. ITS Extensions
5. Regression discontinuities

INTRODUCTION TO THE EXAMPLES

Extensions to ITS

1. Model extensions
   - Wild points
   - Seasonal effects
   - Phase-in periods

2. Multiple interventions

3. Non-linear trends

EXAMPLE 1: WATER FLOW ON NILE

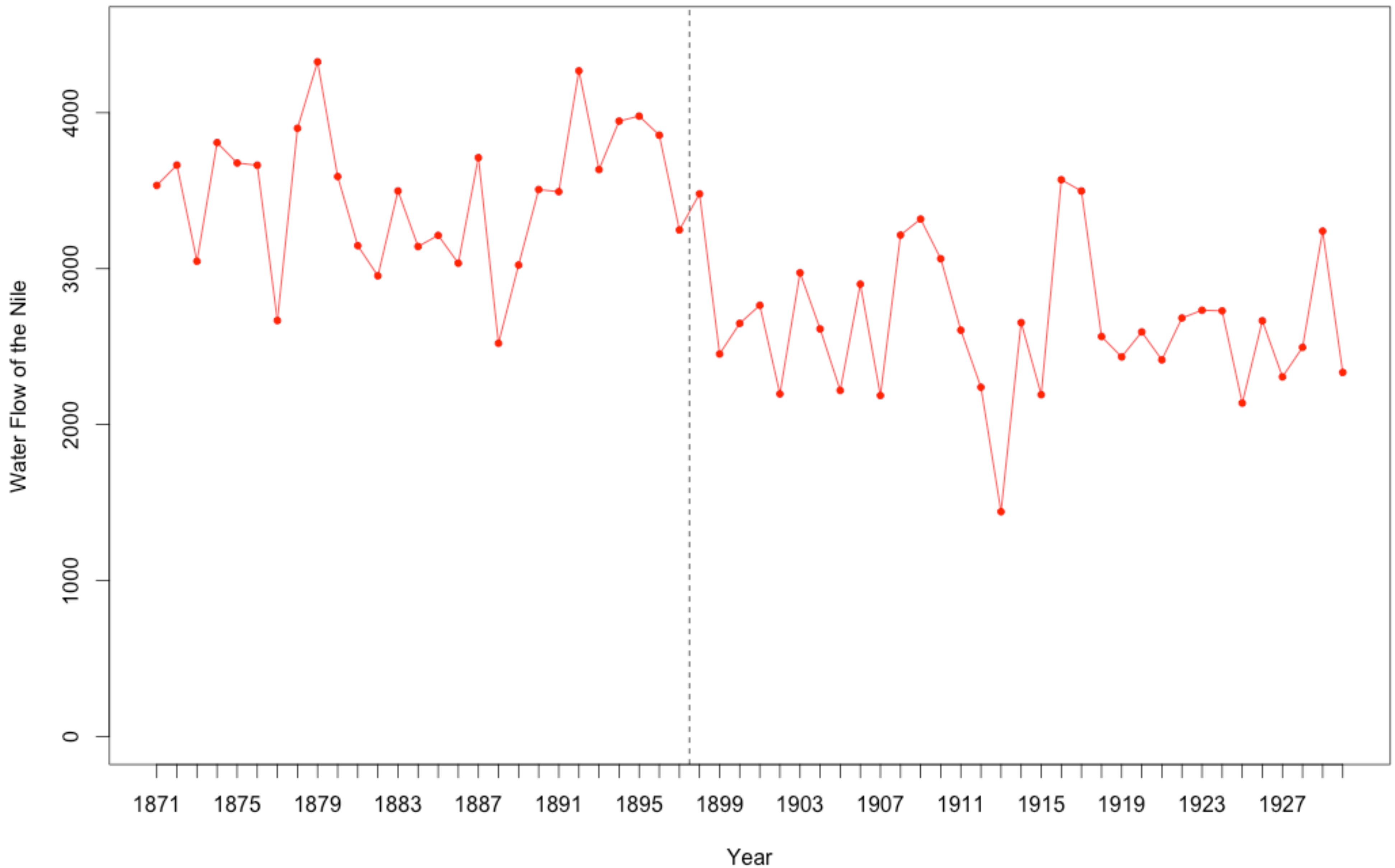
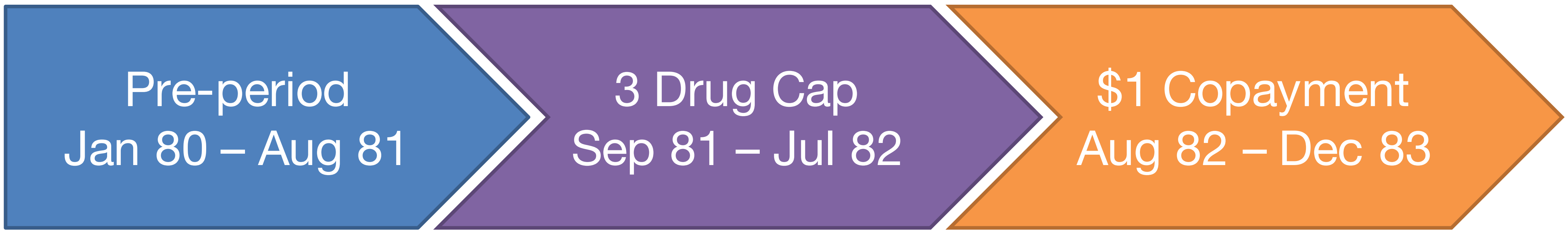
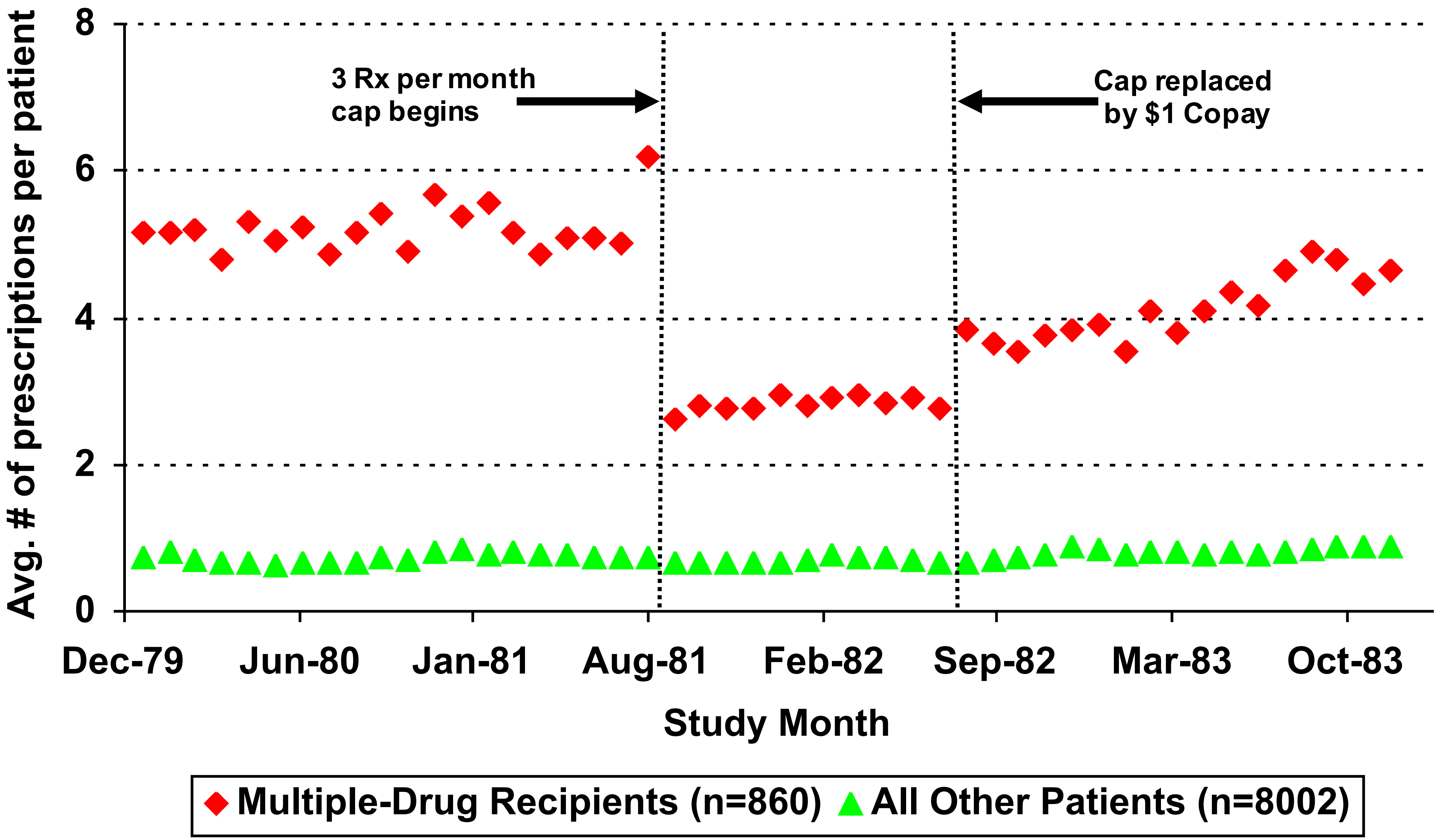

Water Flow of the Nile
0
1000
2000
3000
4000
1871
1875
1879
1883
1887
1891
1895
1899
1903
1907
1911
1915
1919
1923
1927
Year

Photo credit: https://commons.wikimedia.org/wiki/File%3AAswanHighDam_Egypt.jpg

EXAMPLE 2: NEW HAMPSHIRE MEDICAID DRUG CAP


Pre-period
Jan 80 – Aug 81
3 Drug Cap
Sep 81 – Jul 82
$1 Copayment
Aug 82 – Dec 83

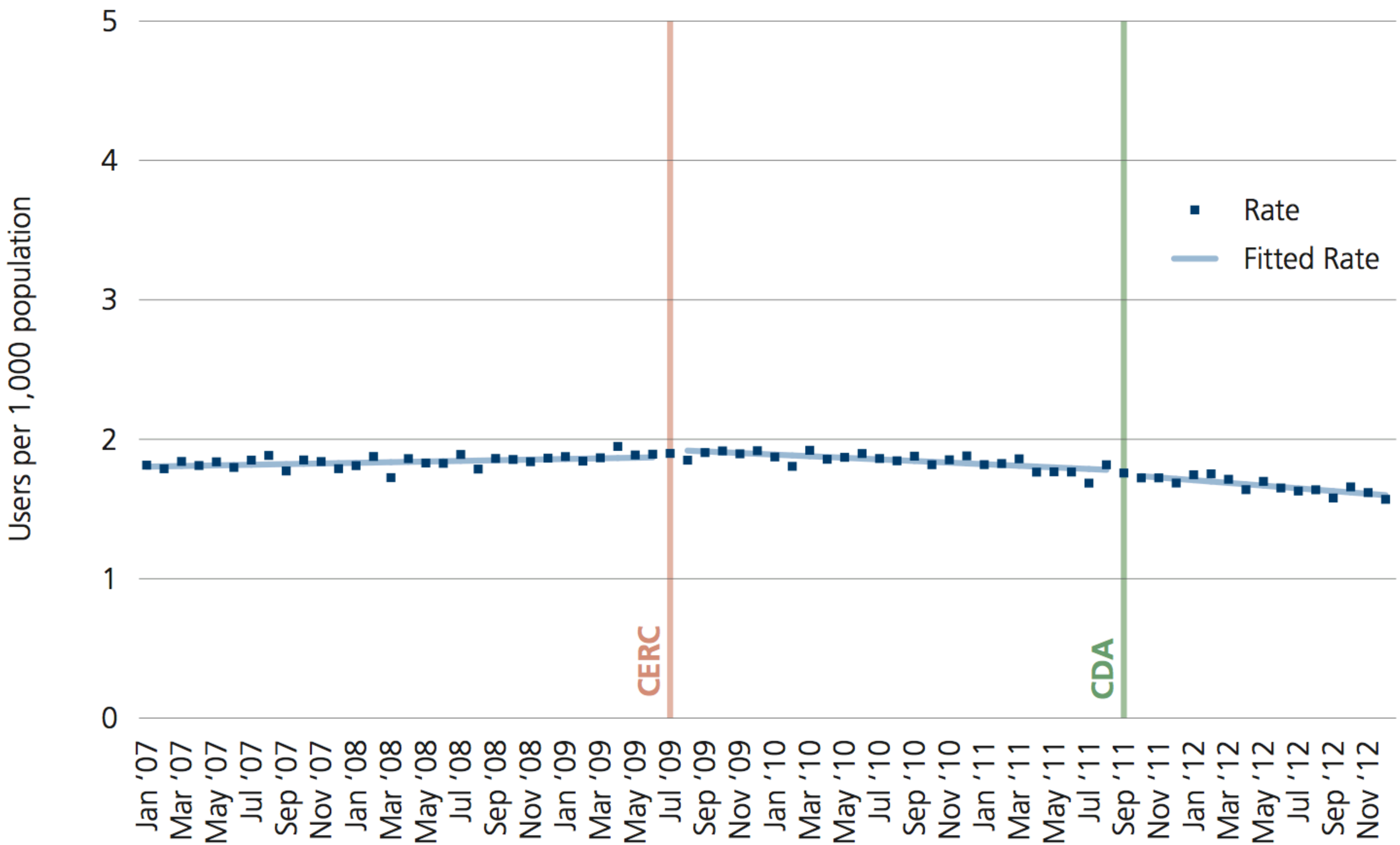


Source: Soumerai et al. NEJM 1987; 317(9):550-6

EXAMPLE 3: DIABETES TEST STRIP GUIDELINES

Photo Credit: https://www.flickr.com/photos/bodytel/5476255676


Pre-period
Jan 07 – Jun 09
CERC
Jul 09 – Aug 11
CDA
Sep 11 – Dec 12



Users per 1,000 population
5
4
3
2
1
0
CERC
CDA
Rate
Fitted Rate
Jan '07
Mar '07
May '07
Jul '07
Sep '07
Nov '07
Jan '08
Mar '08
May '08
Jul '08
Sep '08
Nov '08
Jan '09
Mar '09
May '09
Jul '09
Sep '09
Nov '09
Jan '10
Mar '10
May '10
Jul '10
Sep '10
Nov '10
Jan '11
Mar '11
May '11
Jul '11
Sep '11
Nov '11
Jan '12
Mar '12
May '12
Jul '12
Sep '12
Nov '12

EXAMPLE 4: UNIVERSAL HEALTH COVERAGE IN THAILAND

Garabedian et al. 2012

- In 2001, Thailand implemented universal health coverage
- Studied the impact on drug use for chronic conditions
  - We will look at data on insulin sales per 1000 population
- Original paper used a phase-in period & quadratic trend


Pre-period
1998Q2 – 2001Q1
Universal Coverage Scheme
2002Q1 – 2006Q3

EXAMPLE 5: MEDIA COVERAGE OF DRUG WARNINGS

Lu et al. 2014

- In 2003, the US FDA warned of an increased suicide risk from antidepressant use in young people

- Studied the impact on drug use, suicide attempts, and completed suicides
  - We will look at percent of young people receiving an antidepressant

- Original paper used a phase-in period & quadratic trend


Pre-period
2000Q1 – 2003Q3
Post Media Coverage
2005Q1 – 2010Q4

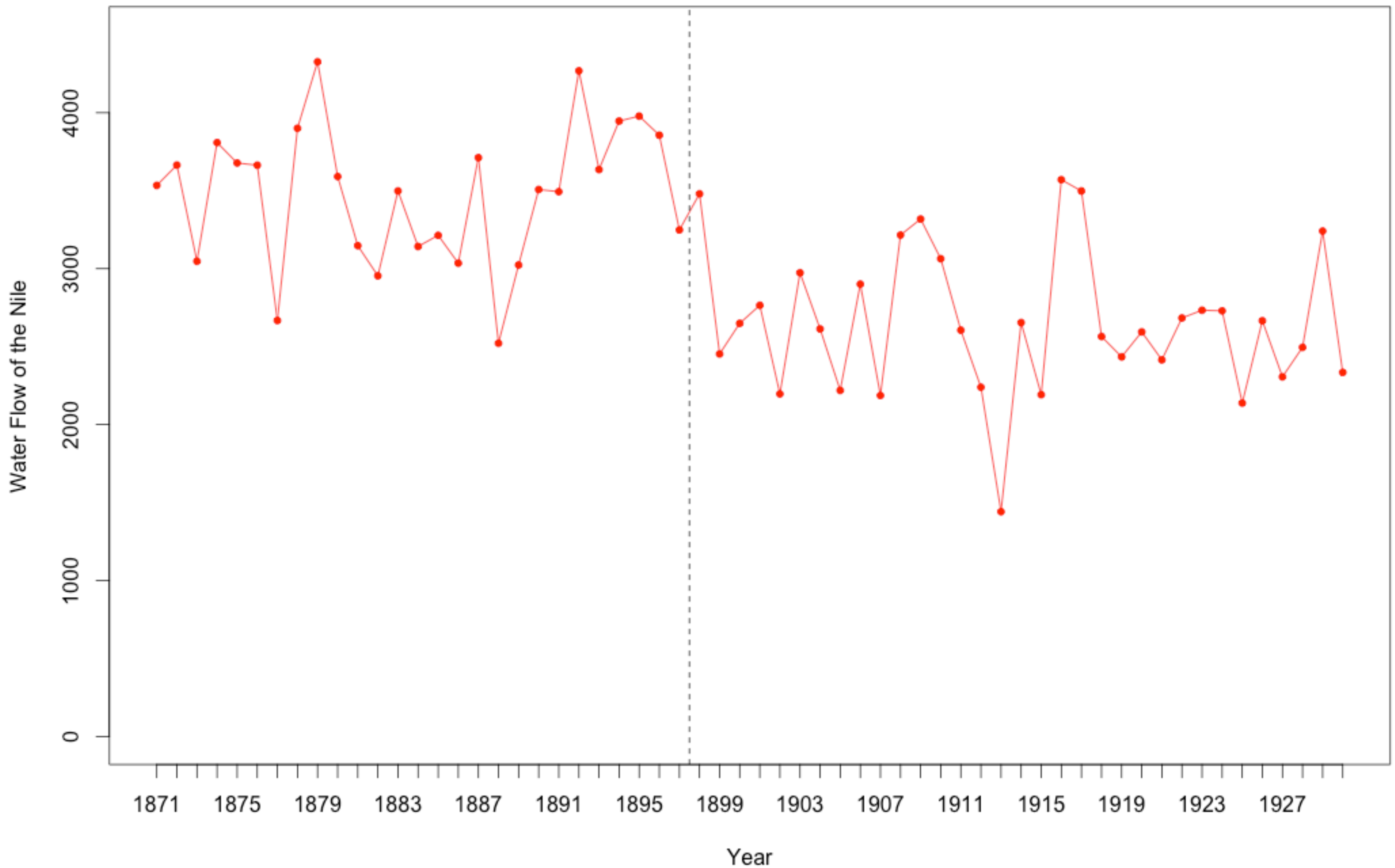
Extension 1, Part 1

WILD POINTS

“Wild” Points

- Can reflect:
  - Anticipatory effects
  - Data quality issues
  - Short-term history events

- Options:
  - Explicitly model them
  - Omit them out of your analysis



Water Flow of the Nile
0
1000
2000
3000
4000
1871
1875
1879
1883
1887
1891
1895
1899
1903
1907
1911
1915
1919
1923
1927
Year

Overview of steps

1. Determine time periods
2. Select analytic cohorts
3. Determine outcomes of interest
4. **Setup data**
5. Visually inspect the data
6. **Perform preliminary analysis**
7. Check for and address autocorrelation
8. **Run the final model**
9. **Plot the results**
10. Predict relative and absolute effects

Step 4: Setup Data

year	time	flow	level	trend	drought
1871	1	3533.9	0	0	0
…	…	…	…	…	…
1897	27	3448.2	0	0	0
1898	28	3479.8	1	1	0
…	…	…	…	…	…
1913	43	1441.25	1	16	1
…	…	…	…	…	…
1930	60	2334.4	1	33	0

Step 6: Perform Preliminary Analysis

```
#########################
# Preliminary Analysis
#########################

# Fit the OLS regression model
model_ols <- lm(flow ~ time + level + trend + drought,
                 data=data)


summary(model_ols)
```

Preliminary Analysis Results

```
Coefficients:
              Estimate  Std. Error t value  Pr(>|t|)
(Intercept)   3407.937     172.118  19.800   < 2e-16 ***
time             5.417      10.743   0.504   0.61610
level         -761.059     225.281  -3.378   0.00135 **
trend          -11.381      13.365  -0.852   0.39816
drought      -1256.478     441.627  -2.845   0.00622 **
```

Step 8: Run Final Model

```
########################
# Modeling
########################

# Fit the GLS regression model with p=10 as in Week 2
model_p10 <- gls(flow ~ time + level + trend + drought,
  data=data,
  correlation=corARMA(p=10,form=~time),
  method="ML")

summary(model_p10)
```

Final Model Results

Coefficients:

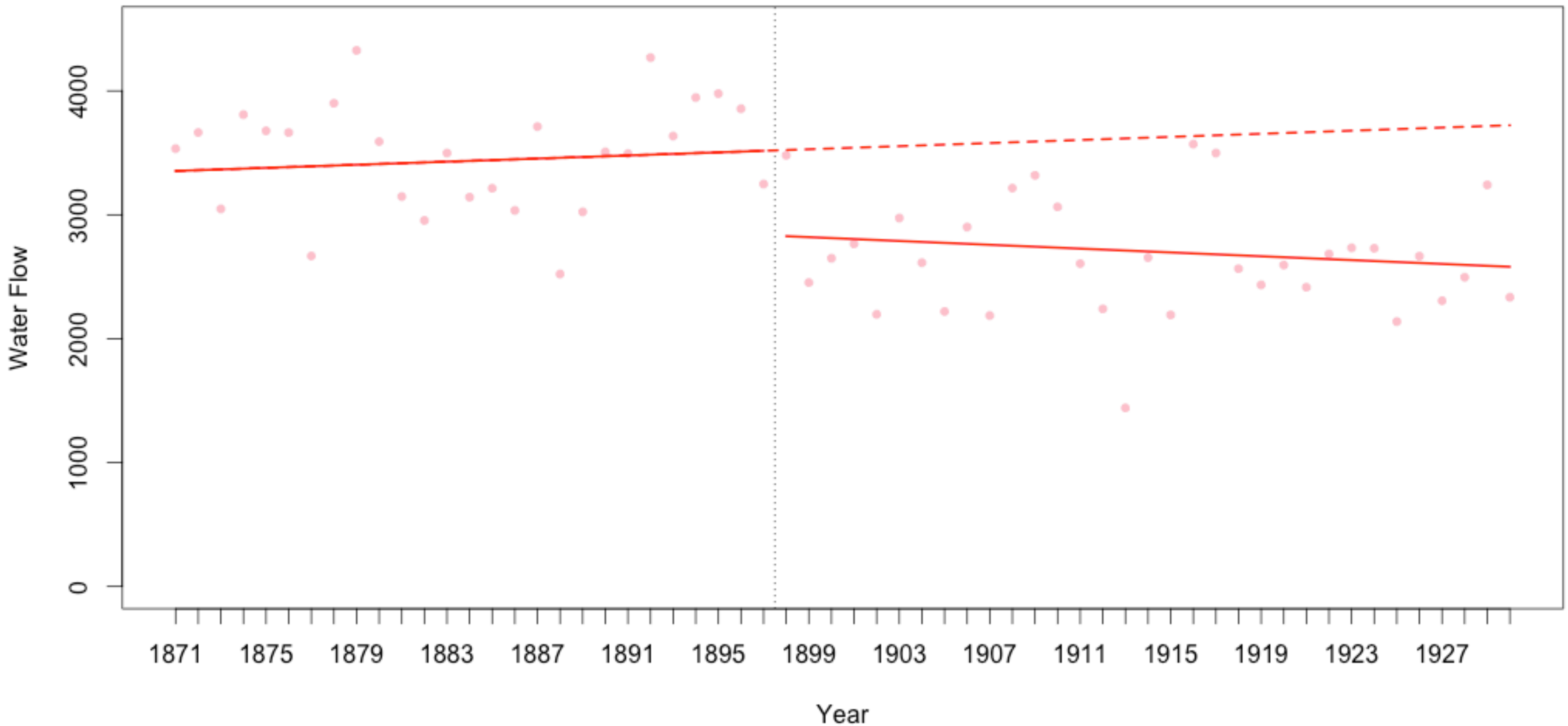
	Value	Std.Error	t-value	p-value
(Intercept)	3347.497	96.4774	34.69722	0.0000
time	6.252	6.2078	1.00714	0.3183
level	-680.943	133.0340	-5.11856	0.0000
trend	-13.983	6.6479	-2.10333	0.0400
drought	-1004.553	344.8281	-2.91320	0.0052

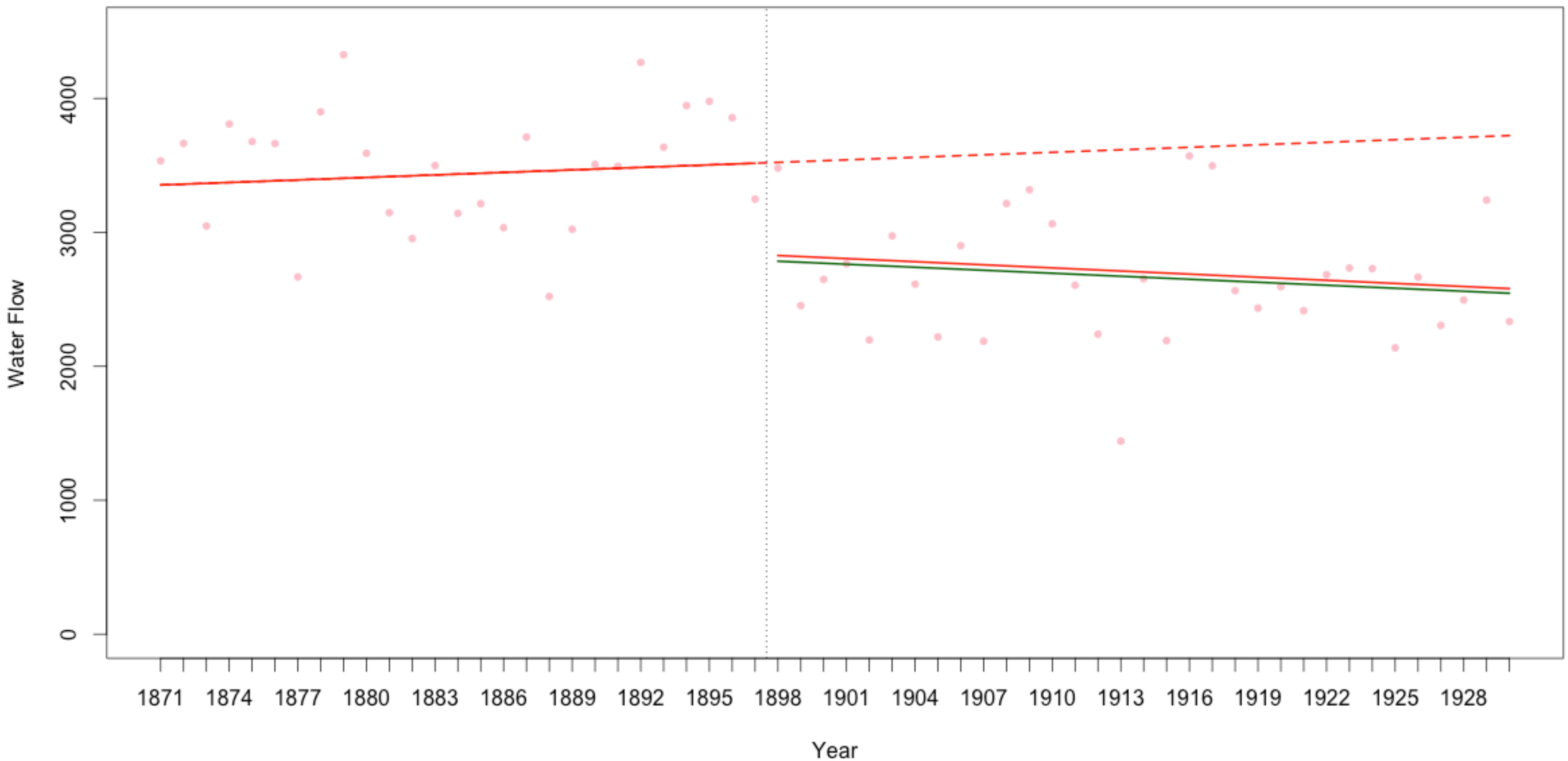
Step 9: Plot the Results

```
# Plot the first line segment
lines(data$time[1:27], fitted(model_p10)[1:27],
col="red",lwd=2)

# Plot the second line segment - Note what happens!
lines(data$time[28:60], fitted(model_p10)[28:60],
col="red",lwd=2)
```

Step 9: Plot the Results (2)

```
# An alternative  using model  coefficients
segments(28,
         model_p10$coef[1] + model_p10$coef[2]*28 +
         model_p10$coef[3] + model_p10$coef[4],
         60,
         model_p10$coef[1] + model_p10$coef[2]*60 +
         model_p10$coef[3] + model_p10$coef[4]*33,
         lty=1,
         lwd=2,
         col='red')
```


Water Flow
0
1000
2000
3000
4000
1871
1875
1879
1883
1887
1891
1895
1899
1903
1907
1911
1915
1919
1923
1927
Year



Water Flow
0
1000
2000
3000
4000
1871
1874
1877
1880
1883
1886
1889
1892
1895
1898
1901
1904
1907
1910
1913
1916
1919
1922
1925
1928
Year

Extension 1, Part 2

SEASONAL EFFECTS

Seasonal Effects

- Can reflect:
  - Natural patterns
  - Program design
- Options:
  - Model specific time periods
  - Add some type of function to model (sine, cosine, etc.)

Overview of steps

1. Determine time periods
2. Select analytic cohorts
3. Determine outcomes of interest
4. **Setup data**
5. Visually inspect the data
6. **Perform preliminary analysis**
7. Check for and address autocorrelation
8. **Run the final model**
9. **Plot the results**
10. Predict relative and absolute effects

year	time	flow	level	trend	elnino
1871	1	3533.9	0	0	0
1872	2	3664.3	0	0	0
…	…	…	…	…	…
1896	26	3856.0	0	0	1
1897	27	3248.2	0	0	1
1898	28	3479.8	1	1	0
1899	29	2453.0	1	2	1
…	…	…	…	…	…
1929	59	3241.1	1	32	0
1930	60	2334.4	1	33	1

Step 8: Run Final Model

```
########################
# Modeling
########################

# Fit the GLS regression model with p=10 as in Week 2
model_p10 <- gls(flow ~ time + level + trend + drought +
                 elnino,
                 data=data,
                 correlation=corARMA(p=10,form=~time),
                 method="ML")

summary(model_p10)
```

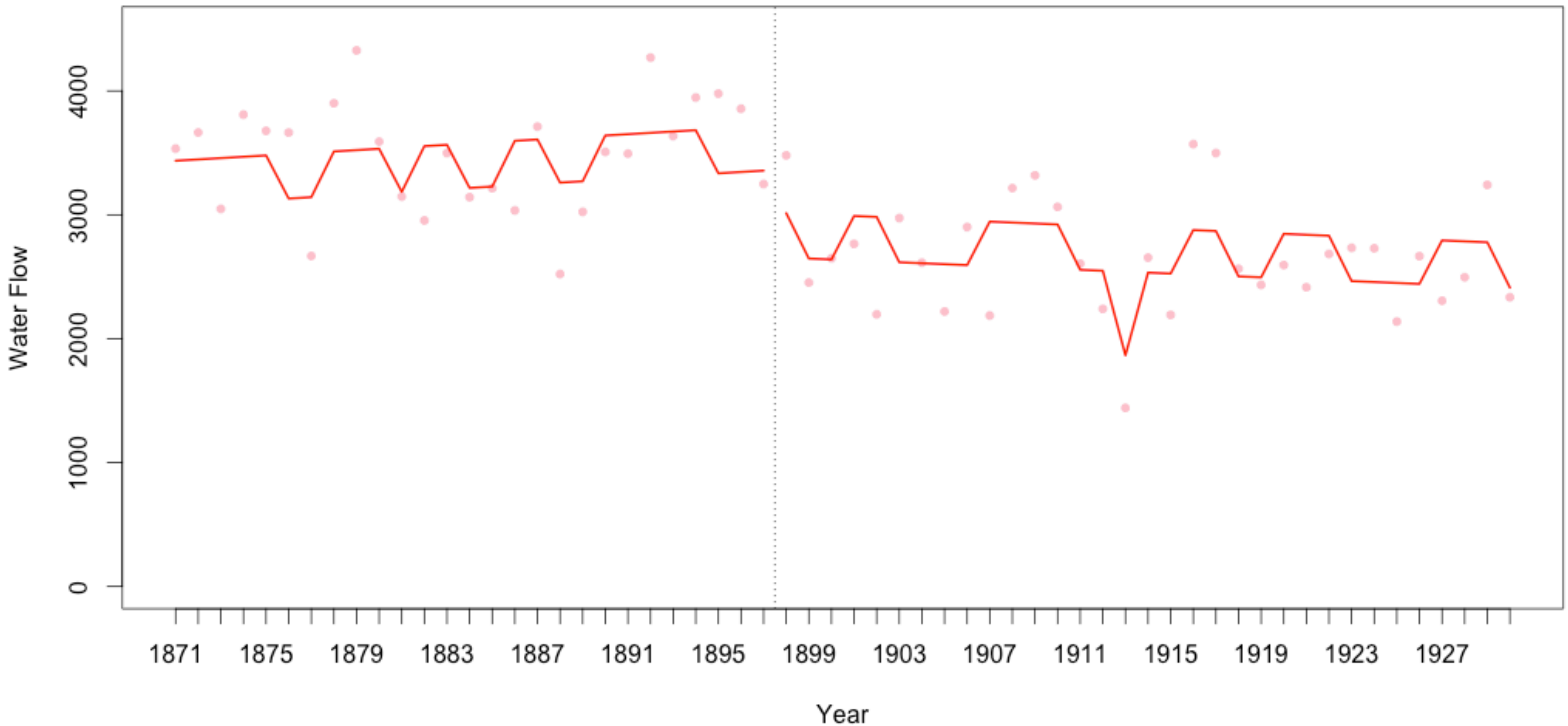
Final Model Results

Coefficients:

	Value	Std.Error	t-value	p-value
(Intercept)	3425.477	96.06013	35.65972	0.0000
time	10.723	6.24836	1.71617	0.0919
level	-694.370	130.95377	-5.30240	0.0000
trend	-18.302	6.70200	-2.73083	0.0085
drought	-1033.575	313.88560	-3.29284	0.0018
elnino	-358.669	90.37531	-3.96866	0.0002

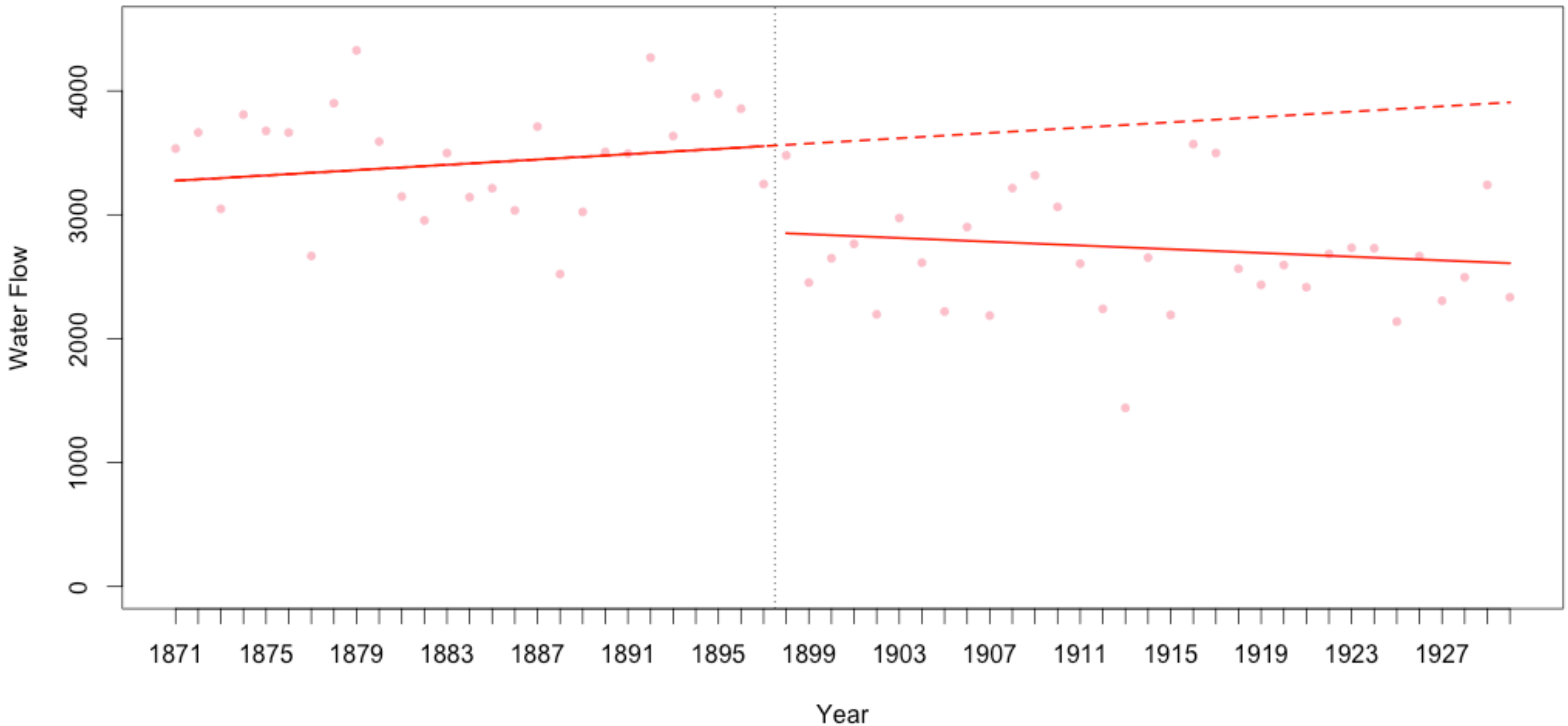
Step 9: Plotting (1)

```
# Plot the first line segment
lines(data$time[1:27], fitted(model_p10)[1:27],
col="red",lwd=2)

# Plot the second line segment
lines(data$time[28:60], fitted(model_p10)[28:60],
col="red",lwd=2)
```


Water Flow
0
1000
2000
3000
4000
1871
1875
1879
1883
1887
1891
1895
1899
1903
1907
1911
1915
1919
1923
1927
Year

Step 9: Plotting (2)

```
# Calculate the offset due to El Nino events
offset <- mean(data$elnino) * model_p10$coef[6]

# Plot the first line segment
segments(1, model_p10$coef[1] + model_p10$coef[2] + offset,
         27, model_p10$coef[1] + model_p10$coef[2]*27 + offset,
         lty=1, lwd=2, col='red')

# Plot the second line segment
segments(28, model_p10$coef[1] + model_p10$coef[2]*28 +
           model_p10$coef[3] + model_p10$coef[4] + offset,
         60, model_p10$coef[1] + model_p10$coef[2]*60 +
           model_p10$coef[3] + model_p10$coef[4]*33 + offset,
         lty=1, lwd=2, col='red')

# Plot the counterfactual
segments(1, model_p10$coef[1]+model_p10$coef[2] + offset,
         60, model_p10$coef[1]+model_p10$coef[2]*60 + offset,
         lty=2, lwd=2, col='red')
```

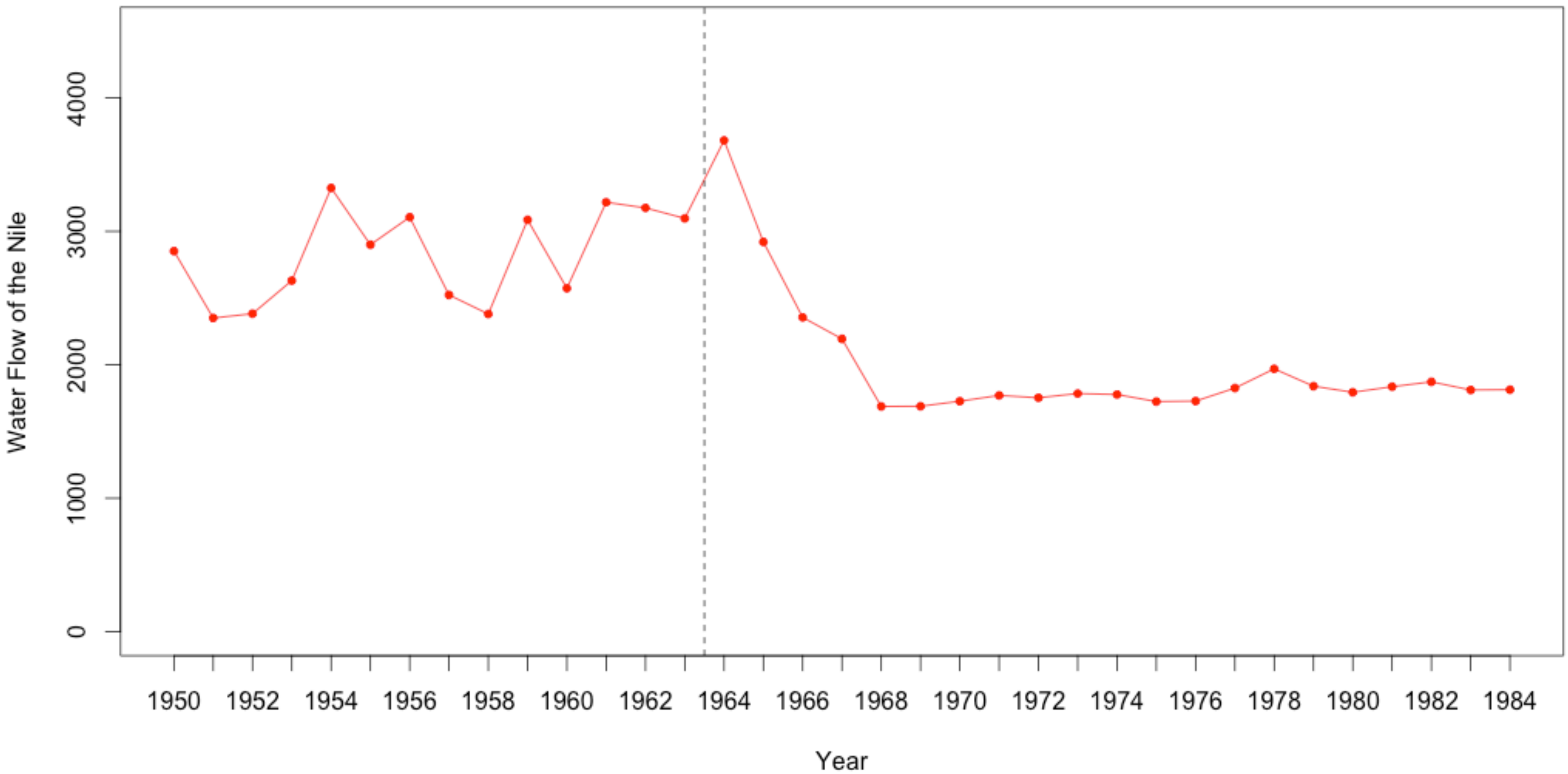

Water Flow
0
1000
2000
3000
4000
1871
1875
1879
1883
1887
1891
1895
1899
1903
1907
1911
1915
1919
1923
1927
Year

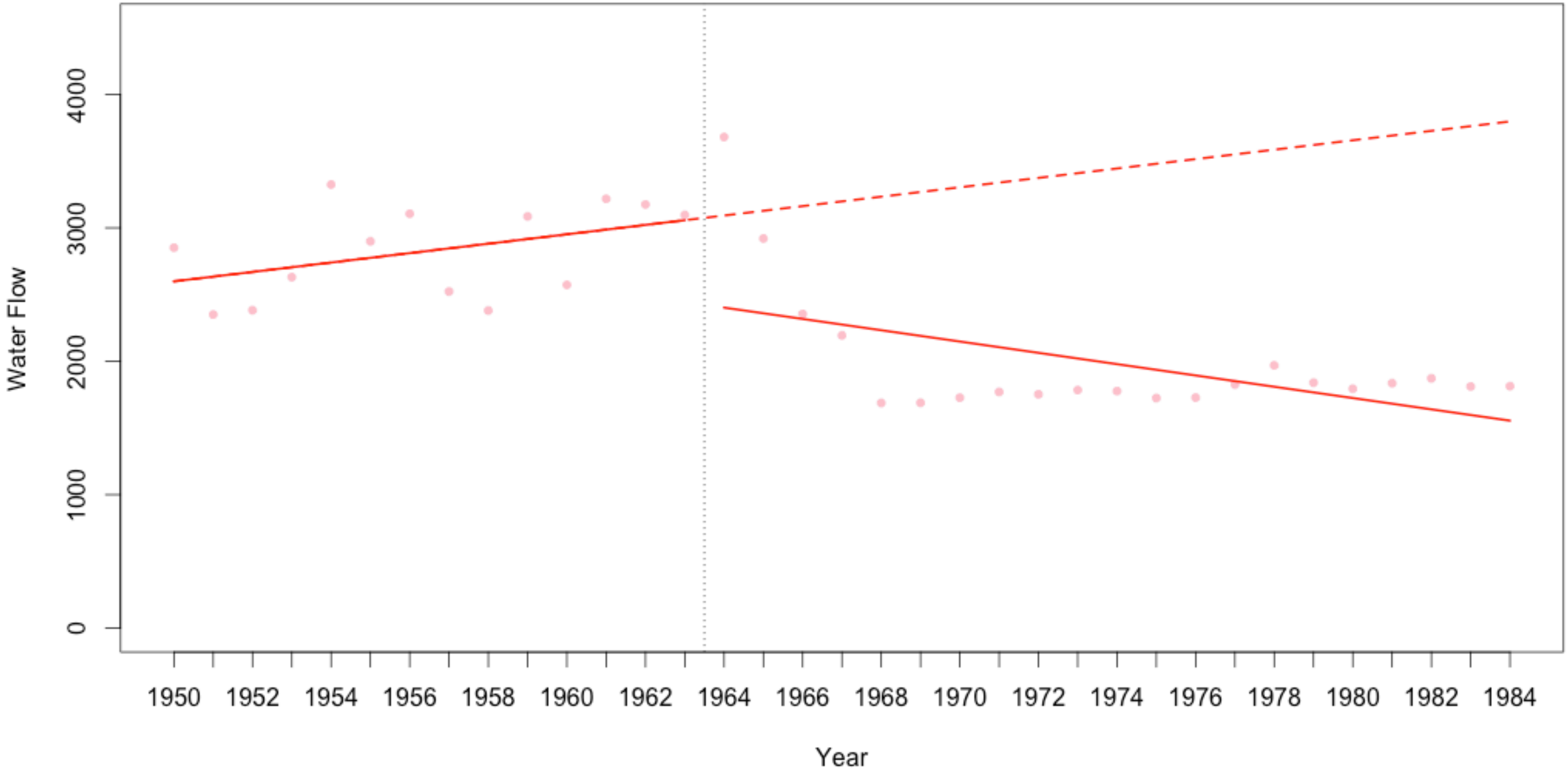
Extension 1, Part 3

PHASE-IN PERIODS

Phase-in Periods

- In many instances, policy implementation is not instantaneous
  - This often leads to a “delay” in seeing the impact
- Modeling options:
  - Exclude the data from your time series analysis
  - Model it as a separate segment


Water Flow of the Nile
0
1000
2000
3000
4000
1950
1952
1954
1956
1958
1960
1962
1964
1966
1968
1970
1972
1974
1976
1978
1980
1982
1984
Year



Water Flow
0
1000
2000
3000
4000
1950
1952
1954
1956
1958
1960
1962
1964
1966
1968
1970
1972
1974
1976
1978
1980
1982
1984
Year

Overview of steps

1. Determine time periods
2. Select analytic cohorts
3. Determine outcomes of interest
4. **Setup data**
5. Visually inspect the data
6. **Perform preliminary analysis**
7. Check for and address autocorrelation
8. **Run the final model**
9. **Plot the results**
10. Predict relative and absolute effects

Step 4: Setup Data

year	flow	flow	level	trend
...	...	...	...	...
1963	3097.17	14	0	0
1964	3681.42	15	1	1
1965	2920.67	16	1	2
1966	2355.08	17	1	...
1967	2194.25	18	1	4
1968	1687.83	19	1	5
1969	1689.33	20	1	6
...	...	...	...	...

Step 4: Setup Data

```
#########################
# Create New Dataset
#########################

# Make a vector of the rows we want to include
include <- c(1:14,19:35)

# Duplicate these rows into a new dataset
data_pi <- data[include,]
```

Step 6: Perform Preliminary Analysis

```
########################
# Modeling
########################

# A preliminary OLS regression
model_ols <- lm(flow ~ time + level + trend, data=data_pi)

summary(model_ols)
```

Step 8: Run the Final Model

```
########################
# Modeling
########################

# Fit the GLS regression  model
model_p4  <- gls(flow  ~ time + level + trend,
  data=data_pi,
  correlation=corARMA(p=4,form=~time),
  method="ML")

summary(model_p4)
```

Final Model Results

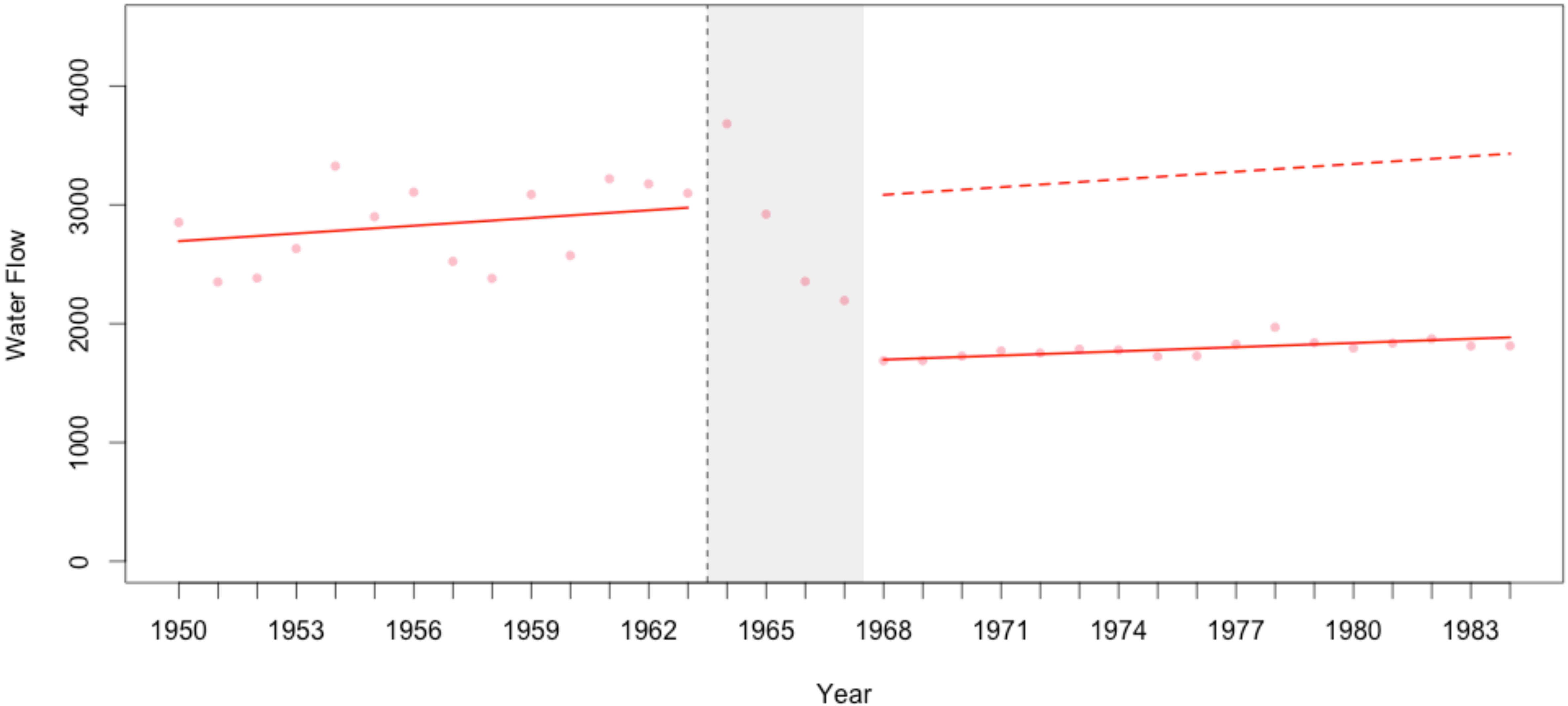
```
Coefficients:
                   Value  Std.Error    t-value  p-value
(Intercept)    2671.9331   52.57030   50.82591   0.0000
time             21.6551    6.40334    3.38184   0.0022
level         -1376.9521   79.06977  -17.41439   0.0000
trend            -9.9335    7.98571   -1.24391   0.2242
```

Step 9: Plot the Results

```
########################
# Plot results
#########################

# Produce the plot, first plotting the raw data points
plot(data$time,data$flow,
     ylim=c(0,4500),
     ylab="Water Flow",
     xlab="Year",
     pch=20,
     col="pink",
     xaxt="n")

# Add x axis with dates
axis(1, at=1:35, labels=data$year)
```

Step 9: Plot the Results

```
# Add line indicating upstream dam
abline(v=14.5,lty=2)

# Plot the first line segment
lines(data$time[1:14], fitted(model_p4)[1:14], col="red",lwd=2)

# Plot the second line segment
lines(data$time[19:39], fitted(model_p4)[15:35], col="red",lwd=2)

# And the counterfactual
segments(19, model_p4$coef[1]+model_p4$coef[2]*19,
         35, model_p4$coef[1]+model_p4$coef[2]*35,
         lty=2, lwd=2, col='red')

# Add a box to show phase-in period
rect(14.5,-500,18.5,5000 , border = NA, col= '#00000011')

# END
```



Water Flow
0
1000
2000
3000
4000
1950
1953
1956
1959
1962
1965
1968
1971
1974
1977
1980
1983
Year

Extension 2

MULTIPLE INTERVENTIONS

For intervention status *j* and *k*, at time *t:*

Pre-existing trend in the outcome of interest

Change in the level between pre and post first policy
* First variable of interest

Expected outcome at first time point

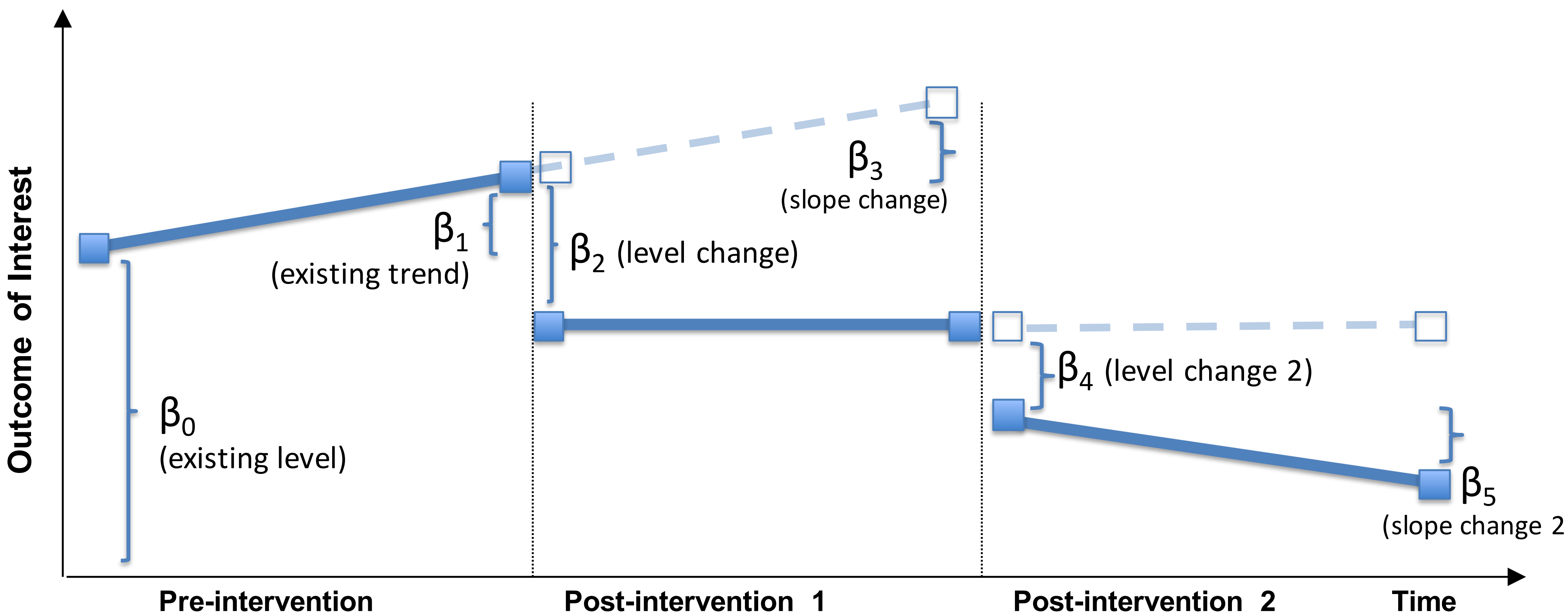
Change in the trend between pre and post first policy
* Second variable of interest

$$outcome_{jkt} = \beta_0 + \beta_1 \cdot time_t + \beta_2 \cdot level_j + \beta_3 \cdot level_j \cdot time_t$$
$$+\beta_4 \cdot level_k + \beta_5 \cdot level_k \cdot time_t + \varepsilon_{jkt}$$

Change in the level between first policy and second policy
* Third variable of interest

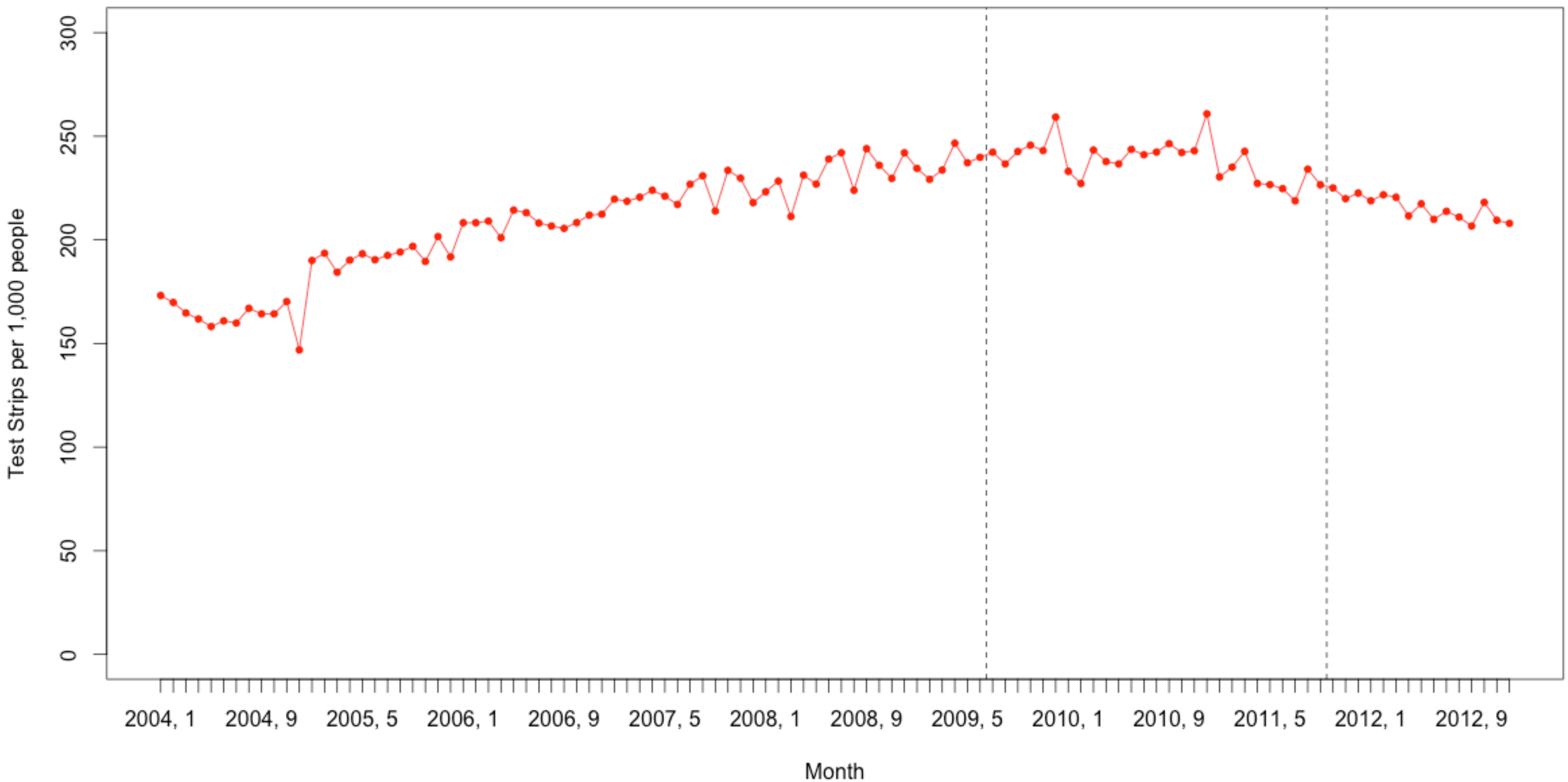
Change in the trend between first policy and second policy
* Fourth variable of interest

$$outcome_{jkt} = \beta_0 + \beta_1 \cdot time_t + \beta_2 \cdot level_j + \beta_3 \cdot level_j \cdot time_t$$
$$+\beta_4 \cdot level_k + \beta_5 \cdot level_k \cdot time_t + \varepsilon_{jkt}$$



Overview of steps

1. Determine time periods
2. Select analytic cohorts
3. Determine outcomes of interest
4. **Setup data**
5. Visually inspect the data
6. **Perform preliminary analysis**
7. Check for and address autocorrelation
8. **Run the final model**
9. **Plot the results**
10. Predict relative and absolute effects



Test Strips per 1,000 people
0
50
100
150
200
250
300
2004, 1
2004, 9
2005, 5
2006, 1
2006, 9
2007, 5
2008, 1
2008, 9
2009, 5
2010, 1
2010, 9
2011, 5
2012, 1
2012, 9
Month

Step 4: Setup Data

month	strips_pt	time	cerc	cerc_trend	cda	cda_trend
2007, 1	219.673	1	0	0	0	0
…	…	…	…	…	…	…
2009, 6	239.812	30	0	0	0	0
2009, 7	242.302	31	1	1	0	0
…	…	…	…	…	…	…
2011, 8	234.123	56	1	26	0	0
2011, 9	226.553	57	1	27	1	1
…	…	…	…	…	…	…
2012, 12	207.973	72	1	42	1	16

Step 6: Perform Preliminary Analysis

```
########################
# Modeling
########################

# A preliminary OLS regression
model_ols <- lm(strips_pt ~ time + cerc + cerc_trend + cda +
cda_trend, data=data)

summary(model_ols)
```

Preliminary Model Results

```
Coefficients:
             Estimate  Std. Error t value  Pr(>|t|)
(Intercept) 217.6591      2.6733   81.420   < 2e-16  ***
time          0.7374      0.1506    4.897  6.60e-06  ***
cerc          6.2255      3.8441    1.620     0.110
cerc_trend   -1.2769      0.2398   -5.324  1.31e-06  ***
cda          -6.0932      4.6281   -1.317     0.193
cda_trend    -0.5858      0.4298   -1.363     0.178
```

Step 8: Run the Final Model

```
########################
# Run the final model
########################

# Fit the GLS regression model
model_p4 <- gls(strips_pt ~ time + cerc + cerc_trend + cda +
cda_trend,
  data=data,
  correlation=corARMA(p=4,form=~time),
  method="ML")

summary(model_p4)
```

Final Model Results

Coefficients:

	Value	Std.Error	t-value	p-value
(Intercept)	217.38394	2.553315	85.13793	0.0000
time	0.75735	0.144097	5.25586	0.0000
cerc	5.31180	3.689251	1.43980	0.1546
cerc_trend	-1.24996	0.228423	-5.47213	0.0000
cda	-7.53190	4.476764	-1.68244	0.0972
cda_trend	-0.54965	0.416291	-1.32036	0.1913

Step 9: Plot the Results (1)

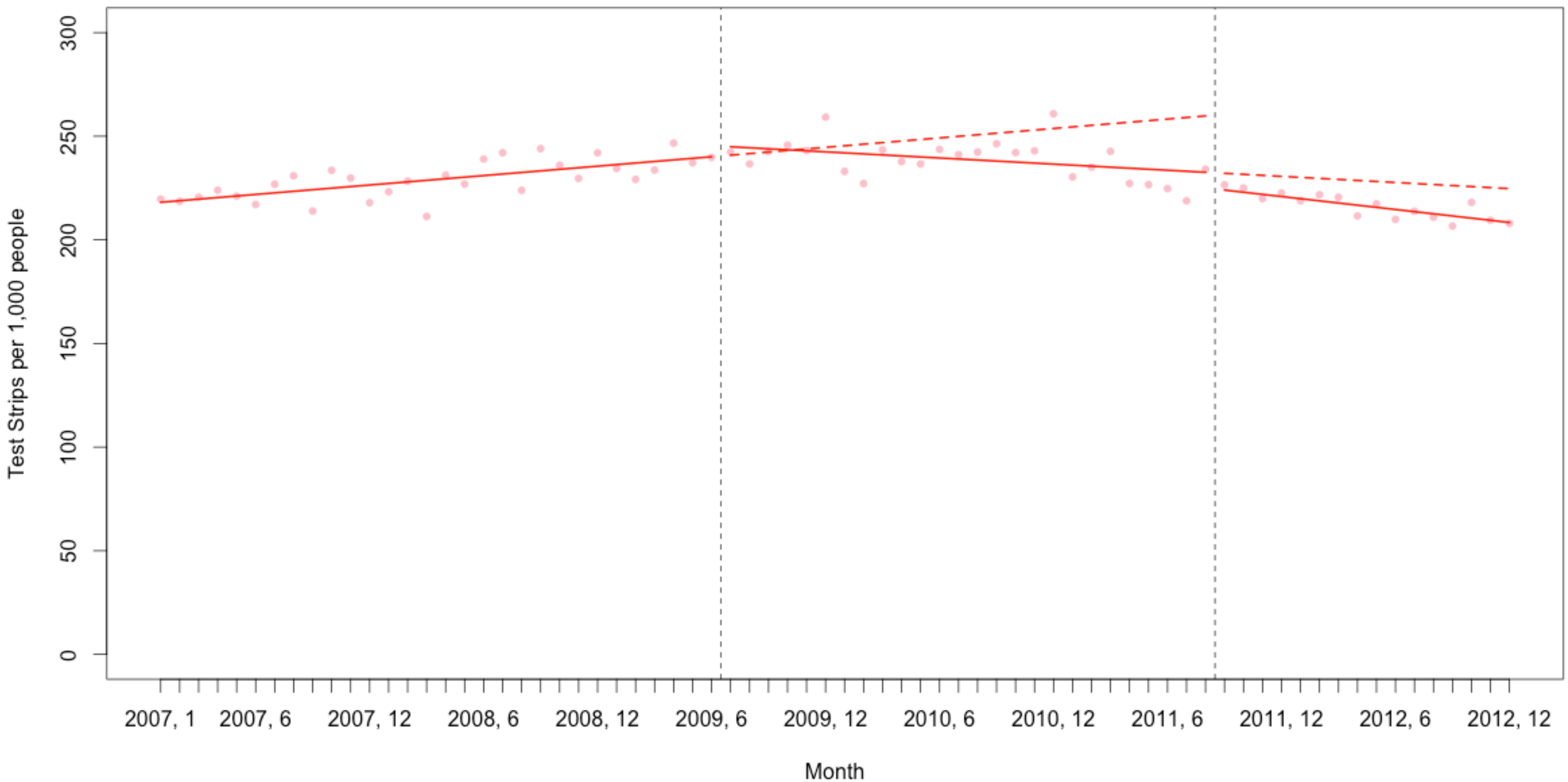
```
########################
# Plot results
#########################

# Produce the plot, first plotting the raw data points
plot(data$time,data$strips_pt,
     ylab="Test Strips per 1,000 people",
     ylim=c(0,300),
     xlab="Month",
     pch=20,
     col="pink",
     xaxt="n")

# Add x axis with dates
axis(1, at=1:72, labels=data$yearmonth)
```

Step 9: Plot the Results (2)

```
# Add line indicating the policy changes
abline(v=30.5,lty=2)
abline(v=56.5,lty=2)

# Plot the first line segment
lines(data$time[1:30], fitted(model_p4)[1:30], col="red",lwd=2)

# Plot the second line segment
lines(data$time[31:56], fitted(model_p4)[31:56],
col="red",lwd=2)

# Plot the third line segment
lines(data$time[57:72], fitted(model_p4)[57:72],
col="red",lwd=2)
```

Step 9: Plot the Results (3)

```
# And the first counterfactual
segments(31, model_p4$coef[1]+model_p4$coef[2]*31,
         56, model_p4$coef[1]+model_p4$coef[2]*56,
         lty=2, lwd=2, col='red')

# And the second counterfactual
segments(57, model_p4$coef[1] + model_p4$coef[2]*57 +
           model_p4$coef[3] + model_p4$coef[4]*27,
         72, model_p4$coef[1] + model_p4$coef[2]*72 +
           model_p4$coef[3] + model_p4$coef[4]*42,
         lty=2, lwd=2, col='red')

# END
```


Test Strips per 1,000 people
0
50
100
150
200
250
300
Month
2007, 1
2007, 6
2007, 12
2008, 6
2008, 12
2009, 6
2009, 12
2010, 6
2010, 12
2011, 6
2011, 12
2012, 6
2012, 12

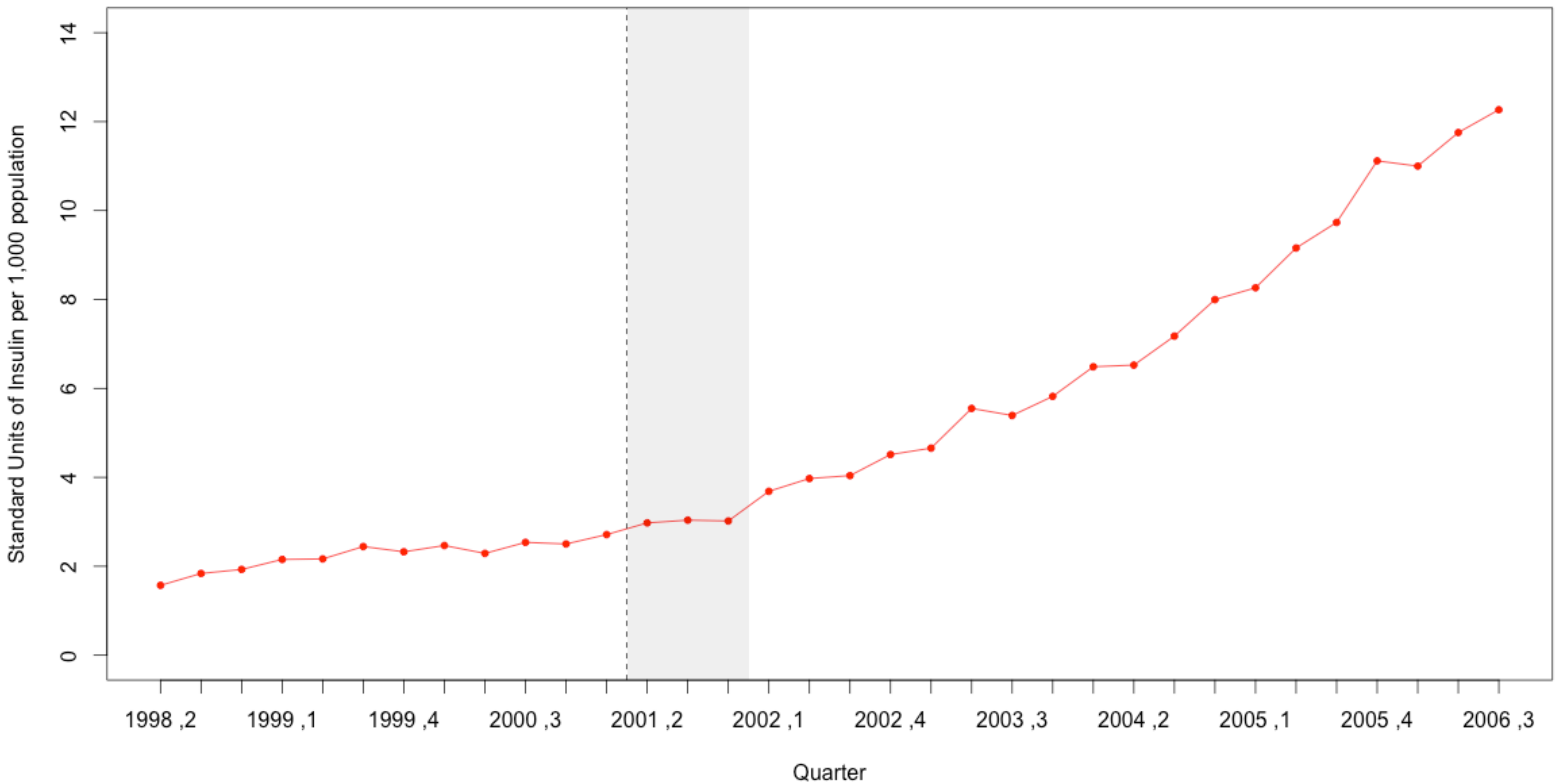
Extension 3:

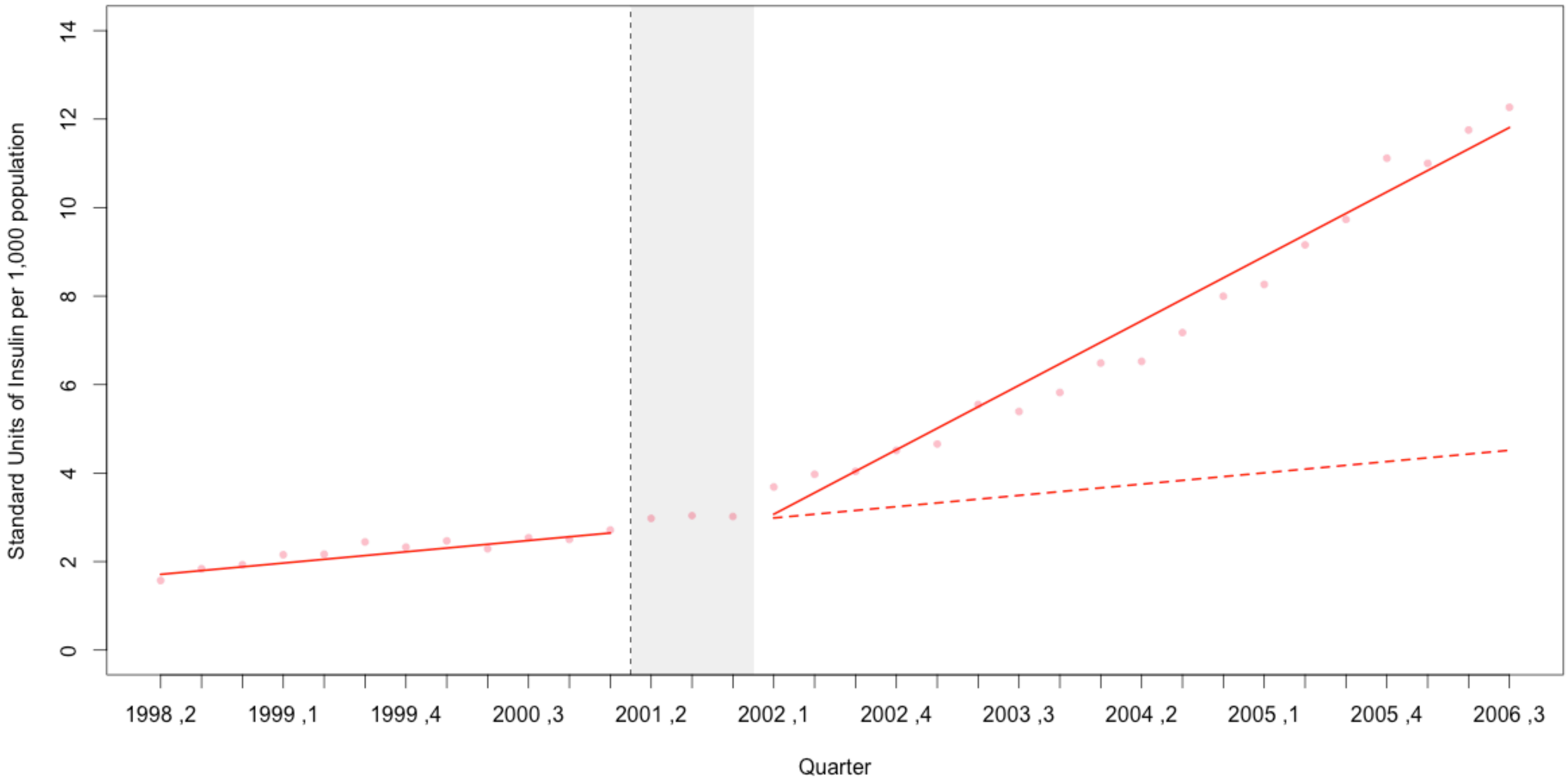
NON-LINEAR TRENDS

Addressing Non-linearity

- We will cover two different strategies:
  - Quadratic model terms
  - Differencing outcomes

QUADRATIC MODEL TERMS


Standard Units of Insulin per 1,000 population
0
2
4
6
8
10
12
14
1998 ,2
1999 ,1
1999 ,4
2000 ,3
2001 ,2
2002 ,1
2002 ,4
2003 ,3
2004 ,2
2005 ,1
2005 ,4
2006 ,3
Quarter



Standard Units of Insulin per 1,000 population
0
2
4
6
8
10
12
14
1998 ,2
1999 ,1
1999 ,4
2000 ,3
2001 ,2
2002 ,1
2002 ,4
2003 ,3
2004 ,2
2005 ,1
2005 ,4
2006 ,3
Quarter

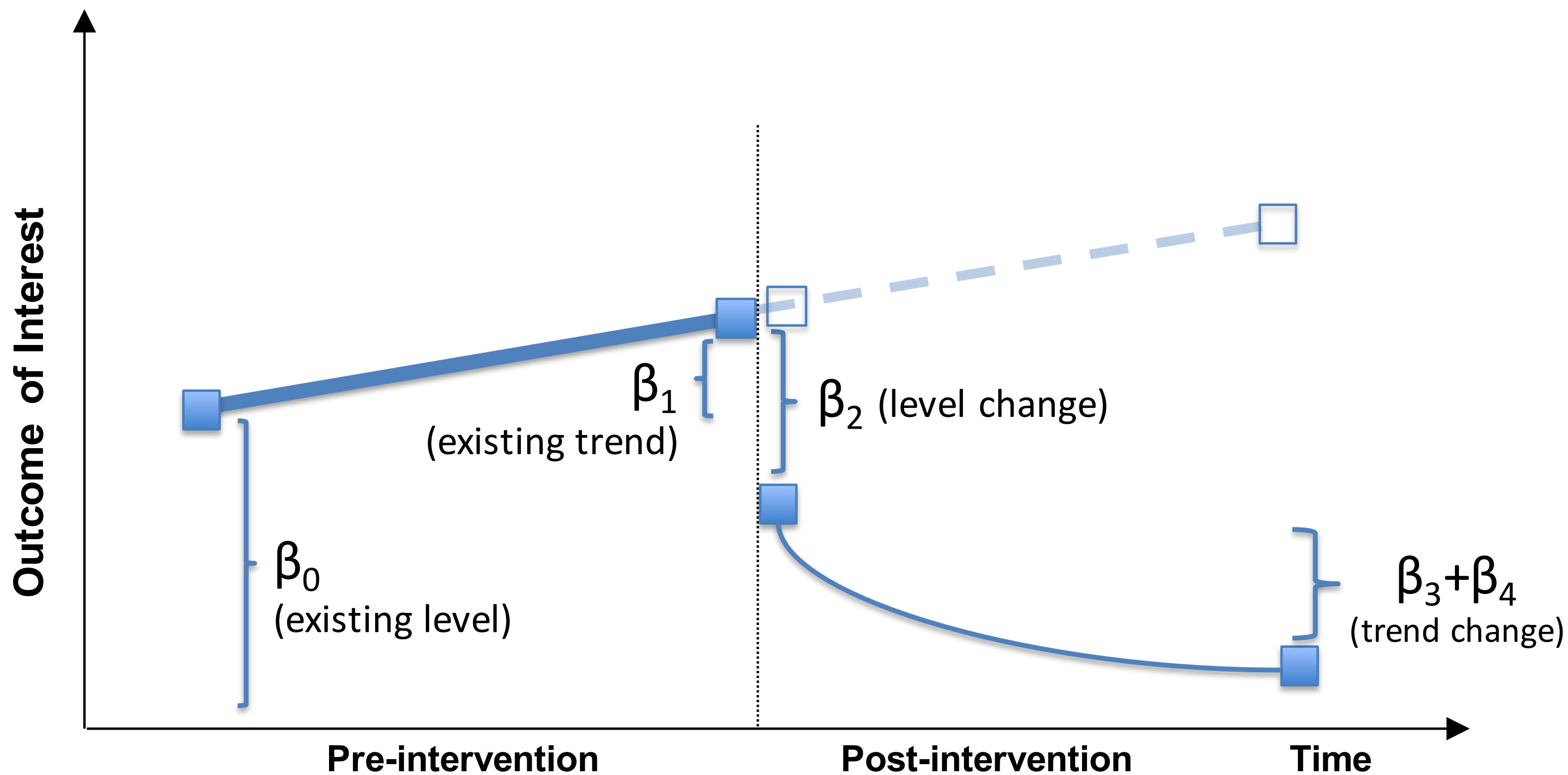
Quadratic Time Trend

- For intervention status *j*, at time *t:*

$$outcome_{jt} = \beta_0 + \beta_1 \cdot time_t + \beta_2 \cdot level_j + \beta_3 \cdot trend_{jt} +$$

$$+\beta_4 \cdot trend^2_{jt} + \varepsilon_{jt}$$

$$outcome_{jt} = \beta_0 + \beta_1 \cdot time_t + \beta_2 \cdot level_j + \beta_3 \cdot trend_{jt} +$$
$$+\beta_4 \cdot trend^2_{jt} + \varepsilon_{jt}$$



Overview of steps

1. Determine time periods
2. Select analytic cohorts
3. Determine outcomes of interest
4. **Setup data**
5. Visually inspect the data
6. **Perform preliminary analysis**
7. Check for and address autocorrelation
8. **Run the final model**
9. Plot the results
10. Predict relative and absolute effects

Step 4: Setup Data

yearqtr	time	stdunits	level	trend	trendsq
1998, 2	1	1.572	0	0	0
1998, 3	2	1.839	0	0	0
…	…		…	…	…
2000, 4	11	2.502	0	0	0
2001, 1	12	2.713	0	0	0
2002, 1	13	3.686	1	1	1
2002, 2	14	3.974	1	2	4
…	…		…	…	…
2006, 2	33	11.754	1	32	1024
2006, 3	34	12.267	1	33	1089

Step 6: Perform Preliminary Analysis

```
##############################
# Modeling - with square term
##############################

# A preliminary OLS regression
model_ols <- lm(stdunits ~ time + level + trend + trendsq,
data=data_pi)

summary(model_ols)
```

Preliminary Model Results

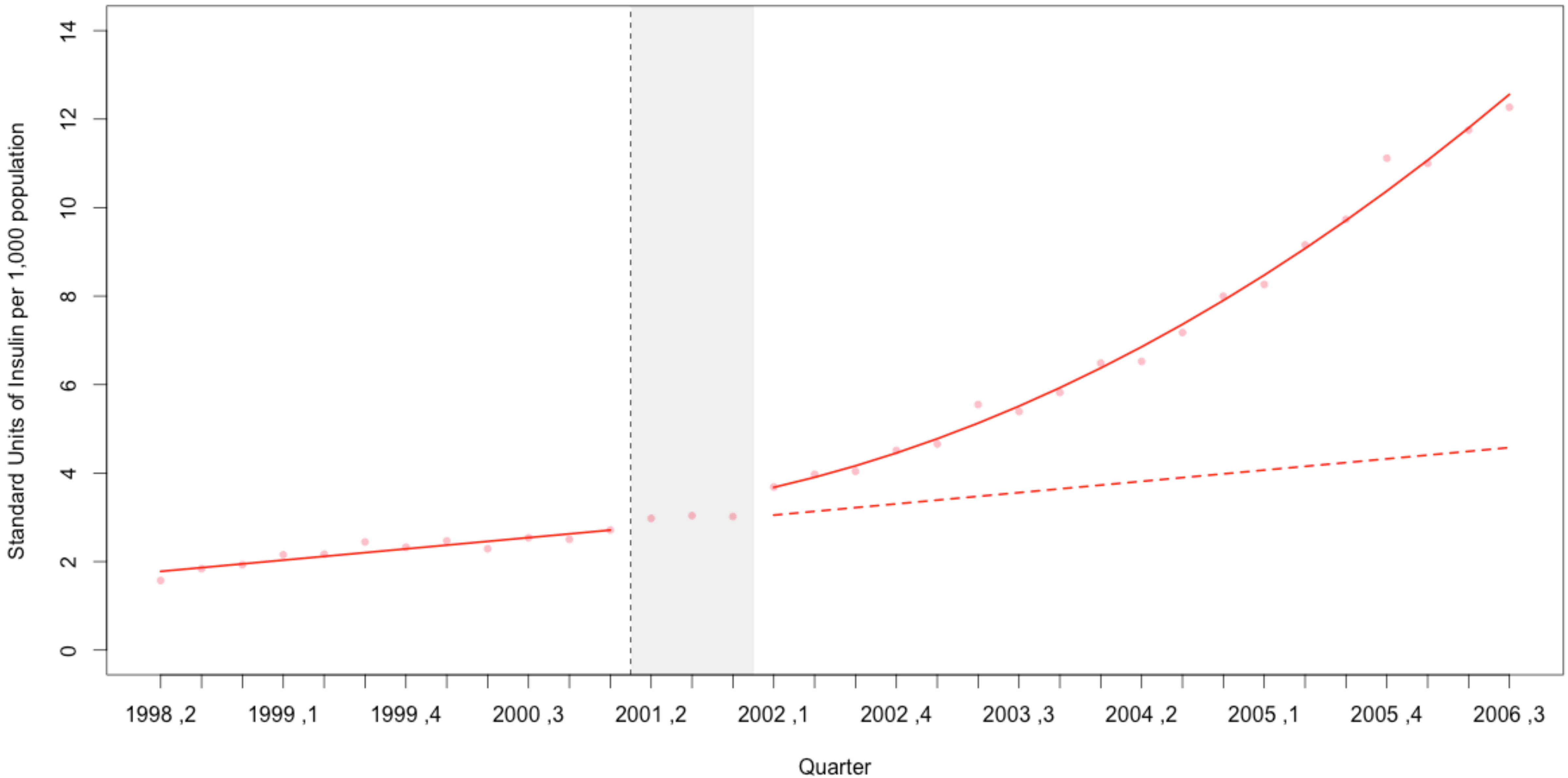
```
Coefficients:
             Estimate  Std. Error t value Pr(>|t|)
(Intercept)  1.693955    0.136137   12.443 1.87e-12  ***
time         0.084776    0.018497    4.583 0.000101  ***
level        0.374562    0.231034    1.621 0.117032
trend        0.002304    0.053536    0.043 0.966000
trendsq      0.015630    0.001899    8.230 1.03e-08  ***
```

Step 8: Run the Final Model

```
########################
# Modeling
########################

# Fit the GLS regression model with square term
model_sq <- gls(stdunits ~ time + level + trend + trendsq,
                data=data_pi,
                method="ML")

summary(model_sq)
```

Final Model Results

```
Coefficients:
             Estimate  Std. Error  t value  Pr(>|t|)
(Intercept)  1.693955    0.136137   12.443  1.87e-12  ***
time         0.084776    0.018497    4.583  0.000101  ***
level        0.374562    0.231034    1.621  0.117032
trend        0.002304    0.053536    0.043  0.966000
trendsq      0.015630    0.001899    8.230  1.03e-08  ***
```


Standard Units of Insulin per 1,000 population
0
2
4
6
8
10
12
14
1998 ,2
1999 ,1
1999 ,4
2000 ,3
2001 ,2
2002 ,1
2002 ,4
2003 ,3
2004 ,2
2005 ,1
2005 ,4
2006 ,3
Quarter

Quadratic trends

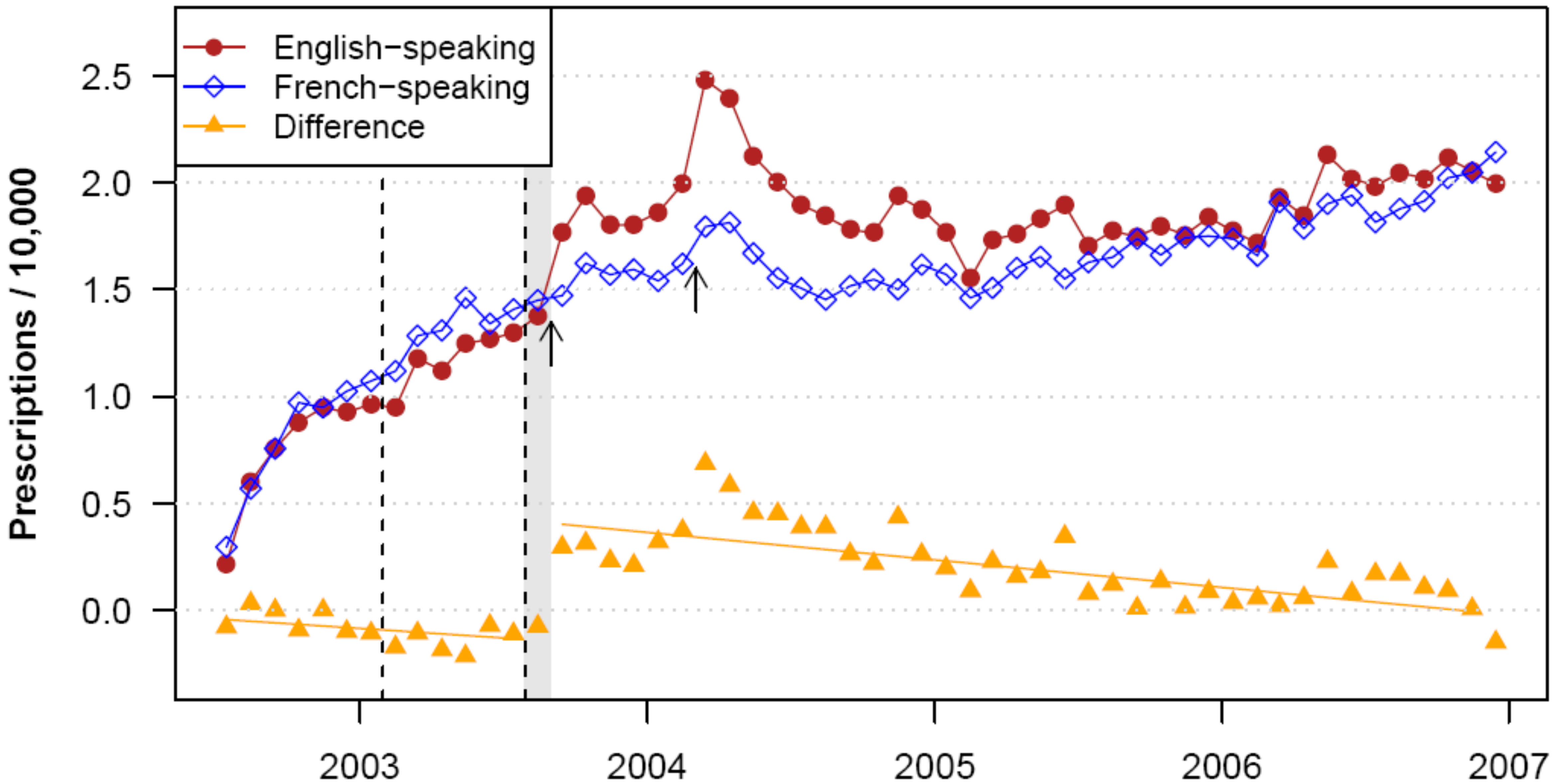
- Advantages
  - Relatively easy to model and predict
  - Avoids transforming an outcome variable

- Disadvantages
  - Makes interpreting model output more difficult
  - Can lead to strange projections

DIFFERENCING OUTCOMES

Differencing

- An alternative tactic if you have a control group
- Simple to employ:
  - Difference the outcomes between your intervention and control group
  - Model this difference as a single ITS


English−speaking
French−speaking
Difference
Prescriptions / 10,000
2.5
2.0
1.5
1.0
0.5
0.0
2003
2004
2005
2006
2007

Month	English	French	Difference
...	...	...	...
Sep-03	1.767	1.472	0.295
Oct-03	1.938	1.624	0.314
Nov-03	1.802	1.570	0.232
Dec-03	1.804	1.595	0.210
Jan-04	1.861	1.541	0.320
Feb-04	1.994	1.621	0.373
...	...	...	...