

Modeling and Analysis of Time Series Data

Chapter 9: Case study: An association between unemployment and mortality?

STATS 531, Winter 2026

Edward Ionides

Historical investigations linking business cycles to mortality

- *Pro-cyclical* mortality occurs if death rates are statistically above trend when economic activity is above trend. An early report was Ogburn and Thomas [3].
- Procyclical mortality, if it exists, shows a key measure of population health is worse in economic booms than recessions.
- Both the economy and life expectancy have grown over the last century. However, these phenomena have not always occurred simultaneously. For example, 1950–1980 saw rapid growth in life expectancy in India and China, combined with relatively slow economic growth. Improvement in life expectancy has slowed during their recent economic surges.
- The link between economic growth and health improvement is controversial, since it has political implications. Economists and epidemiologists have argued both sides of this debate, using time series methods.

Implications of pro-cyclical mortality

- If our goal is population health and happiness, how much should our policies focus on gross domestic product (GDP) growth?
- Evidence supporting the view that economic growth is the critical engine for other improvements in living conditions would make a moral argument in favor of economic growth.
- Evidence that there are other major factors involved in improving living conditions suggest that economic growth should be only one political consideration, among others.

A time series of life expectancy in the USA

```
e_data = pd.read_csv("life_expectancy_usa.csv",
    sep='\s+', comment='#')
```

	Year	e0f	e0m	e0
0	1933	62.80	59.19	60.90
1	1934	62.30	58.29	60.19
2	1935	63.05	58.98	60.91

- Data are from the [Human Mortality Database](#).
- e0 is *life expectancy at birth (LEB)* for civilians.
- e0f and e0m are LEB for females and males. We focus on e0.

LEB is an actuarial calculation based on a fictitious individual having mortality rates at each age matching census age-specific mortality rates for the current year. This is a standard way to combine all the age-specific mortality rates into a single number.

```
u_data = pd.read_csv("unadjusted_unemployment.csv", comment='#')
```

	Year	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep
0	1948	4.0	4.7	4.5	4.0	3.4	3.9	3.9	3.6	3.4
1	1949	5.0	5.8	5.6	5.4	5.7	6.4	7.0	6.3	5.9
2	1950	7.6	7.9	7.1	6.0	5.3	5.6	5.3	4.1	4.0

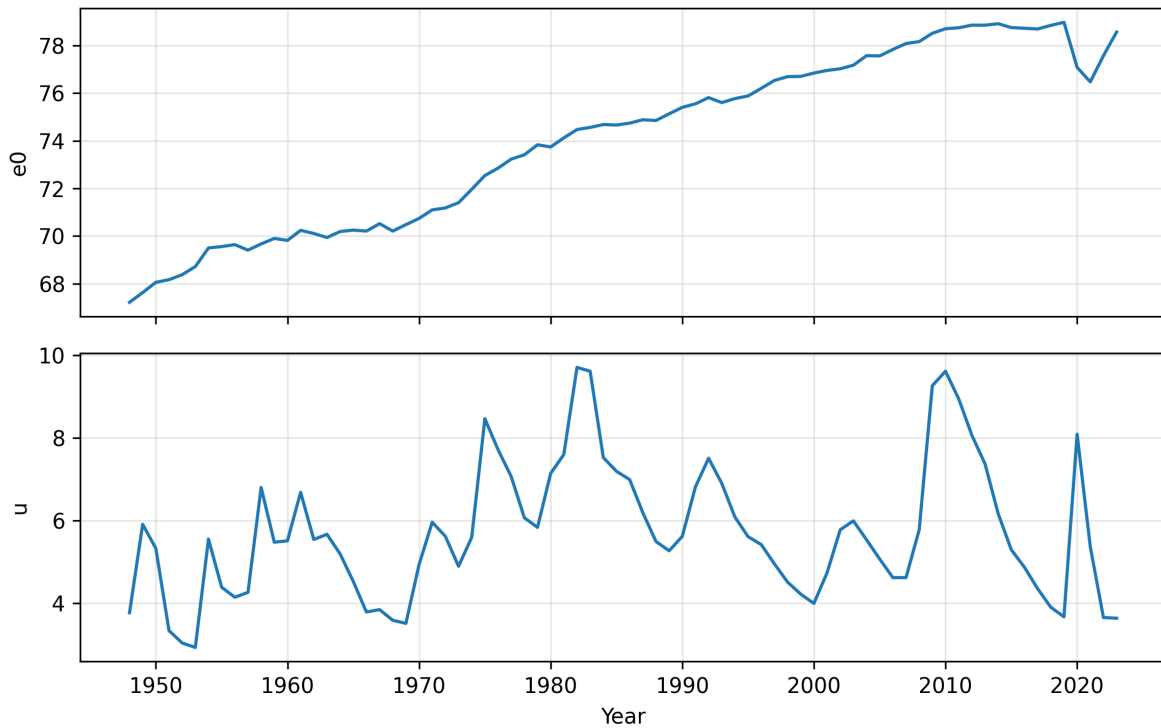
- We consider unadjusted unemployment from Bureau of Labor Statistics.
- Unemployment is just one component of the state of the economy. One could consider other measurements.
- Write e_n for life expectancy in year $t_n = 1947 + n$.
- Write u_n for mean unemployment in year t_n .

A time plot of the raw data

```

yr = np.intersect1d(e_data['Year'], u_data['Year'])
e = e_data.loc[e_data['Year'].isin(yr), 'e0'].values
u = u_data.loc[u_data['Year'].isin(yr),
               u_data.columns[1:13]].mean(axis=1).values

```



Allowing for COVID-19

- We are interested in changes over business cycle timescales, once trends have been removed.
- We wish to study the effect of economic fluctuations on population health.
- COVID-19 led to a dramatic effect of health on the economy.
- Therefore, we analyze data only up to 2019.
- The opioid crisis is another factor in recent mortality, worth study but not part of this investigation.

Detrending the data: The Hodrick-Prescott filter

- To extract the cyclical component, we use an econometric method: the *Hodrick-Prescott (HP) filter* [1].

- For a time series $y_{1:N}$, the HP filter is the time series $s_{1:N}$ defined as

$$s_{1:N} = \underset{s_{1:N}}{\operatorname{argmin}} \left\{ \sum_{n=1}^N (y_n - s_n)^2 + \lambda \sum_{n=2}^{N-1} (s_{n+1} - 2s_n + s_{n-1})^2 \right\}.$$

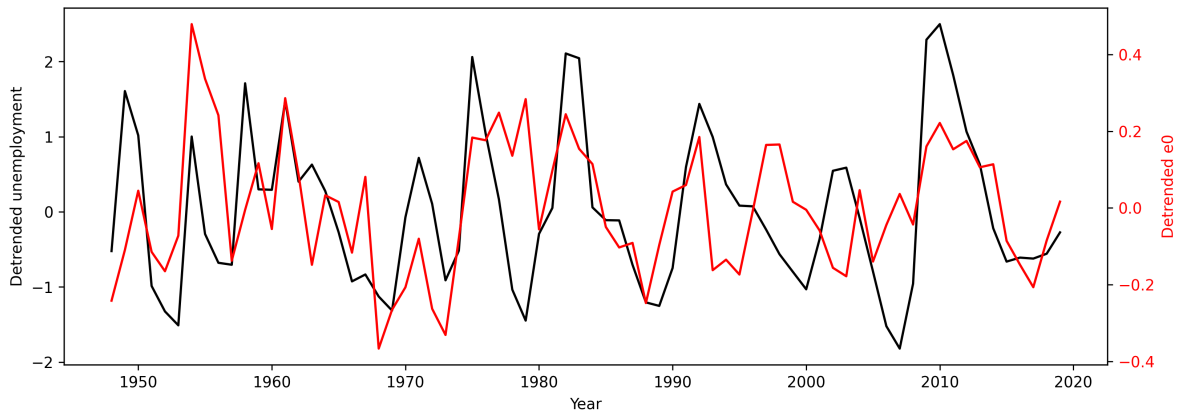
- The HP filter is a *smoothing spline*. Later, we see it can also be viewed as a state space model.
- A standard econometric choice of λ for removing nonlinear trend, for extracting the business cycle component, in annual data is $\lambda = 100$.
- A Python implementation of the Hodrick-Prescott filter is `hpfilter` from `statsmodels`.

- We use the Hodrick-Prescott filter to define the HP-detrended life expectancy, $e_{1:N}^{HP}$, and unemployment, $u_{1:N}^{HP}$.

```
e_hp_cycle, e_hp_trend = hpfilter(e, lamb=100)
u_hp_cycle, u_hp_trend = hpfilter(u, lamb=100)
```

- Plotting two time series on a single graph is not always advisable, but here it is helpful.

```
fig, ax1 = plt.subplots(figsize=(11, 4))
ax1.plot(yr, u_hp_cycle, 'k-')
ax1.set_xlabel('Year')
ax1.set_ylabel('Detrended unemployment', color='k')
ax1.tick_params(axis='y', labelcolor='k')
ax2 = ax1.twinx()
ax2.plot(yr, e_hp_cycle, 'r-')
ax2.set_ylabel('Detrended e0', color='r')
ax2.tick_params(axis='y', labelcolor='r')
plt.tight_layout()
plt.show()
```



Detrended unemployment (black; left axis) and detrended life expectancy at birth (red; right axis).

- Looking at this figure may suggest that detrended life expectancy and detrended unemployment cycle together.
- We make a formal statistical test to check our eyes are not deceiving us.

Hypothesis testing: regression with ARMA errors

-We investigate the dependence of $e_{1:N}^{HP}$ on $u_{1:N}^{HP}$ using a regression with ARMA errors model,

$$E_n^{HP} = \alpha + \beta u_n^{HP} + \epsilon_n, \quad (1)$$

where $\{\epsilon_n\}$ is a Gaussian ARMA process. We use an ARMA(1,0) model, as discussed in the supplementary analysis.

```
import pandas as pd
u_hp_df = pd.DataFrame(u_hp_cycle, columns=['u_hp'])
a0 = SARIMAX(e_hp_cycle, order=(1, 0, 0),
             exog=u_hp_df, trend='c').fit()
```

```
Coefficient of u_hp: 0.0707 (Std. error: 0.0198)
AR(1) coefficient: 0.4797
Residual std dev: 0.1340
Log-likelihood: 42.40
```

- The standard error (computed via observed Fisher information) gives a z -statistic of $0.0707 / 0.0198 = 3.57$ for the coefficient of detrended unemployment.
- We can also compute a p-value from a likelihood ratio test.

```
a0_null = SARIMAX(e_hp_cycle, order=(1, 0, 0),
                  trend='c').fit(dispatch=False)
log_lik_ratio = a0.llf - a0_null.llf
LRT_pval = 1 - stats.chi2.cdf(2 * log_lik_ratio, df=1)
print(f"LRT p-value: {LRT_pval:.5f}")
```

LRT p-value: 0.00018

- This gives a p-value of 0.00018.
- We have clear statistical evidence of a positive association for detrended unemployment and detrended life expectancy.
- For all observational studies, interpretation of association needs care.

Association and causation

We have been careful to talk about *association*, since observational data giving firm statistical evidence of an association between X and Y cannot readily distinguish between three possibilities:

1. X causes Y .
 2. Y causes X .
 3. Both X and Y are caused by a third variable Z that is unmeasured or has been omitted from the analysis. In this case, Z is called a *confounding variable*.
- Excluding the economic consequences of COVID-19, it is not considered plausible that mortality fluctuations drive economic fluctuations (the *reverse causation* possibility).
 - Unemployment is a *proxy variable* for economic fluctuations. Increased unemployment may not directly cause reduced mortality: all proxy variables for economic activity are confounded in this analysis.

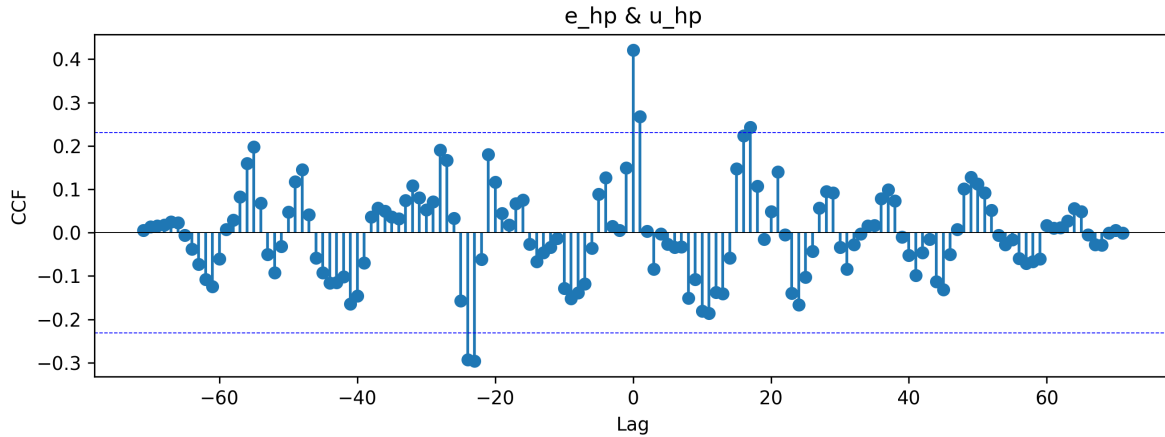
Potential lagged relationships

- A potential confounding variable is lagged economic activity. Theoretically, reduction in mortality for a current economic down-turn could result from delayed health progress caused by the previous economic boom.
- A lag relationship between two time series $x_{1:N}$ and $y_{1:N}$ can be identified from the *sample cross-correlation function (CCF)*

$$\hat{\rho}_{xy}(h) = \frac{\sum_{n=1}^{N-h} (x_{n+h} - \bar{x})(y_n - \bar{y})}{\sqrt{\sum_{n=1}^N (x_n - \bar{x})^2 \sum_{n=1}^N (y_n - \bar{y})^2}} \quad (2)$$

- $\hat{\rho}_{xy}(h)$ estimates $\rho_{XY}(h) = \text{Cor}(X_{n+h}, Y_n)$, the cross-correlation at lag h for a *bivariate stationary time series model*, consisting of a pair of random variables (X_n, Y_n) at each time n .

```
ccf_values = np.correlate(  
    e_hp_cycle - e_hp_cycle.mean(),  
    u_hp_cycle - u_hp_cycle.mean(),  
    mode='full') / (len(e_hp_cycle) *  
    e_hp_cycle.std() * u_hp_cycle.std())
```



- The strong positive cross-correlation at lag zero supports pro-cyclical mortality.
- The oscillatory pattern is not significant pointwise but might be more evident in the frequency domain.

Cross-covariance and the cross-spectrum

- The *cross-covariance function* of a stationary bivariate time series model, (X_n, Y_n) , is

$$\gamma_{XY}(h) = \text{Cov}(X_{n+h}, Y_n). \quad (3)$$

- The *cross-spectrum* is the Fourier transform of the cross-covariance,

$$\lambda_{XY}(\omega) = \sum_{h=-\infty}^{\infty} e^{-2\pi i \omega h} \gamma_{XY}(h). \quad (4)$$

- The cross-spectrum can be estimated by computing the cross-periodogram and smoothing it.

Coherency, coherence and phase

Definition: coherency is the normalized cross-spectrum,

$$\rho_{XY}(\omega) = \frac{\lambda_{XY}(\omega)}{\sqrt{\lambda_{XX}(\omega) \lambda_{YY}(\omega)}}. \quad (5)$$

Coherency measures correlation between frequency components of two time series at each frequency ω . It is complex-valued.

The magnitude of the coherency is called the **coherence**. It measures whether a large amplitude at frequency ω for $x_{1:N}$ is associated with a large amplitude at ω for $y_{1:N}$.

The angle of the coherency (in the complex plane) is called the **phase**. A phase of 0 means that peaks at frequency ω tend to occur simultaneously for $x_{1:N}$ and $y_{1:N}$. A phase of $\pm\pi$ at frequency ω means that peaks for the frequency component of $x_{1:N}$ coincide with troughs for $y_{1:N}$.

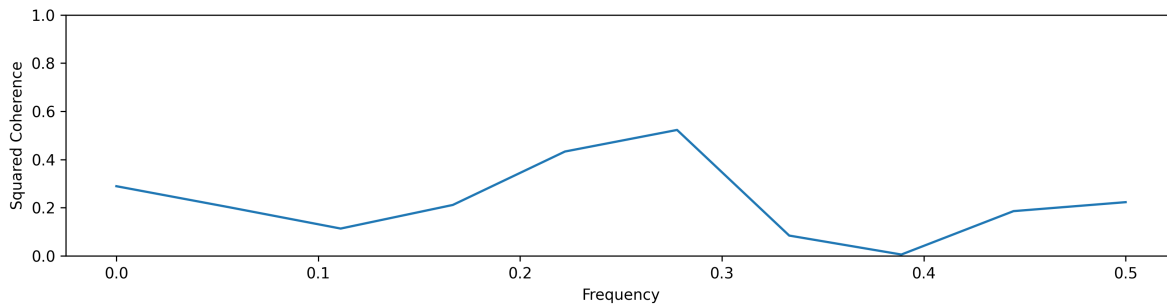
- The coherence and phase are estimated from the smoothed cross-periodogram and marginal periodograms.

```
# Compute cross-spectrum using Welch's method
freqs, Pxx = signal.welch(e_hp_cycle, nperseg=32)
freqs, Pyy = signal.welch(u_hp_cycle, nperseg=32)
freqs, Pxy = signal.csd(e_hp_cycle, u_hp_cycle,
                        nperseg=32)
```



```
# Compute squared coherence
coherence_sq = np.abs(Pxy)**2 / (Pxx * Pyy)

plt.figure(figsize=(11, 3))
plt.plot(freqs, coherence_sq, '-')
plt.xlabel('Frequency')
plt.ylabel('Squared Coherence')
plt.ylim(0, 1)
plt.tight_layout()
plt.show()
```

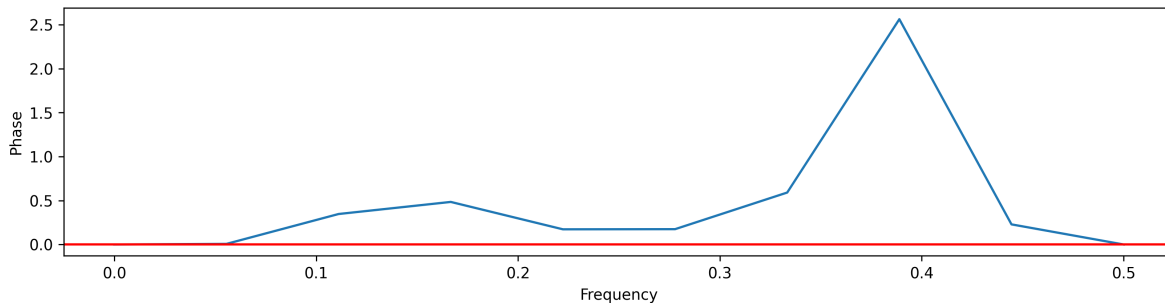


- Python calculates squared coherence, the squared absolute coherency. This is like reporting r^2 for regression, rather than $|r|$.

Question. Interpret the squared coherence plot.

```
# Compute phase
phase = np.angle(Pxy)

plt.figure(figsize=(11, 3))
plt.plot(freqs, phase, '-')
plt.axhline(y=0, color='red', linestyle='--')
plt.xlabel('Frequency')
plt.ylabel('Phase')
plt.tight_layout()
plt.show()
```



Question. Interpret the phase plot.

Conclusions

- There is strong evidence of pro-cyclical mortality at a national level in the USA from 1948 to 2019. For example, the Great Recession of 2009-2010 led to high unemployment, but these two years had above-trend values of life expectancy at birth.
- We have argued that this evidence supports a claim that above-trend economic growth CAUSES above-trend mortality.
- We CANNOT infer that unemployment reduces mortality for those who lose their jobs. Adverse individual-level effects of unemployment can be reconciled with our result [4].
- More data might give statistical precision to investigate sub-populations more accurately than can be done with a national-level dataset. For example, *panel data analysis* combining time series for each state [2].

Supplementary analysis

Explaining the choice of an ARMA(1,0) error model. Model selection by AIC for regression with ARMA errors follows the same approach as for ARMA models.

```
def aic_table(data, P, Q, xreg=None):
    table = np.zeros((P+1, Q+1))
    for p in range(P+1):
        for q in range(Q+1):
            if xreg is not None:
                model = SARIMAX(data, trend='c', exog=xreg,
                                order=(p, 0, q)).fit(dispatch=False)
            else:
                model = SARIMAX(data, trend='c',
                                order=(p, 0, q)).fit(dispatch=False)
```

```

        table[p, q] = model.aic
    return pd.DataFrame(table,
                        index=[f'AR{i}' for i in range(P+1)],
                        columns=[f'MA{i}' for i in range(Q+1)])

e_aic_table = aic_table(e_hp_cycle, 4, 5, xreg=u_hp_df)

```

	MA0	MA1	MA2	MA3	MA4	MA5
AR0	-60.21	-73.24	-75.36	-76.58	-76.27	-74.72
AR1	-76.80	-74.81	-74.47	-75.56	-81.87	-79.04
AR2	-74.81	-72.83	-74.67	-74.20	-69.86	-80.12
AR3	-74.53	-80.50	-72.47	-72.36	-71.14	-78.56
AR4	-76.74	-75.96	-74.84	-77.18	-78.40	-79.63

- ARMA(1,0) gives the best AIC among small models.
- Some larger models have better AIC, but notice inconsistencies in the AIC table. For example, consider ARMA(3,1) errors:

```

a3 = SARIMAX(e_hp_cycle, exog=u_hp_df, trend='c',
             order=(3, 0, 1)).fit()

```

```

Coefficient of u_hp: 0.081863
AR(1) coefficient:    1.3546
AR(2) coefficient:   -0.3841
AR(3) coefficient:   -0.1818
MA(1) coefficient:   -0.9950
Residual std dev:    0.122112
Log-likelihood:      47.25

```

- The estimated ARMA(3,1) is at the boundary of invertibility, with an MA1 coefficient on the unit circle.
- This is reminiscent of our earlier analysis of the Lake Huron depth data.

- Likely the ARMA(3,1) analysis is not very stable. A simulation study might find that the Fisher confidence intervals are not reliable.

Consistency through time

- A useful relationship should be consistent through time. We check this by repeating the analysis on temporal subsets.

```
t1 = slice(0, 36)
a1 = SARIMAX(e_hp_cycle[t1], exog=u_hp_df.iloc[t1],
             order=(1, 0, 0)).fit()
```

First half (1948-1983):

```
Coefficient: 0.0837
Std error:   0.0306
```

```
t2 = slice(36, 72)
a2 = SARIMAX(e_hp_cycle[t2], exog=u_hp_df.iloc[t2],
             order=(1, 0, 0)).fit()
```

Second half (1984-2019):

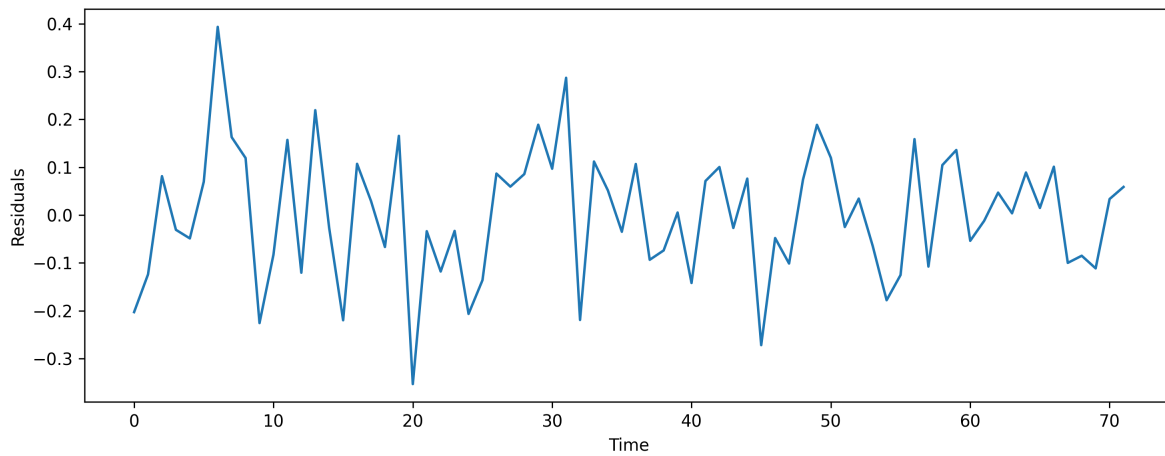
```
Coefficient: 0.0541
Std error:   0.0301
```

- The difference is small compared to the standard error. Overall there is consistency through this 72 year interval, though the pattern is weak in the 1990s.

Residual analysis

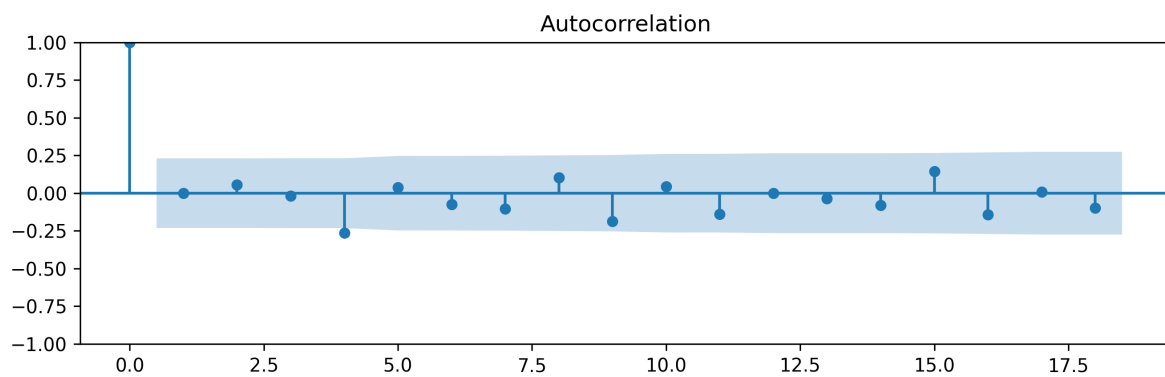
- We inspect the residuals for the fitted model, and look at their sample autocorrelation.

```
r = a0.resid
plt.figure(figsize=(10, 4))
plt.plot(r, '-')
plt.ylabel('Residuals')
plt.xlabel('Time')
plt.tight_layout()
plt.show()
```



- There is some evidence for fluctuations decreasing in amplitude over time. This is an example of *heteroskedasticity*. It is not extreme here, but could be studied in a future analysis.

```
fig, ax = plt.subplots(figsize=(9, 3))
plot_acf(r, ax=ax, lags=18)
plt.tight_layout()
plt.show()
```



- It is not a major model violation to have one out of 18 lags narrowly outside the dashed lines showing pointwise acceptance regions at the 5% level under a null hypothesis of Gaussian white noise.
- The presence of some small amount of sample autocorrelation is consistent with the AIC table, which finds the possibility of gains by fitting larger models to the regression errors.

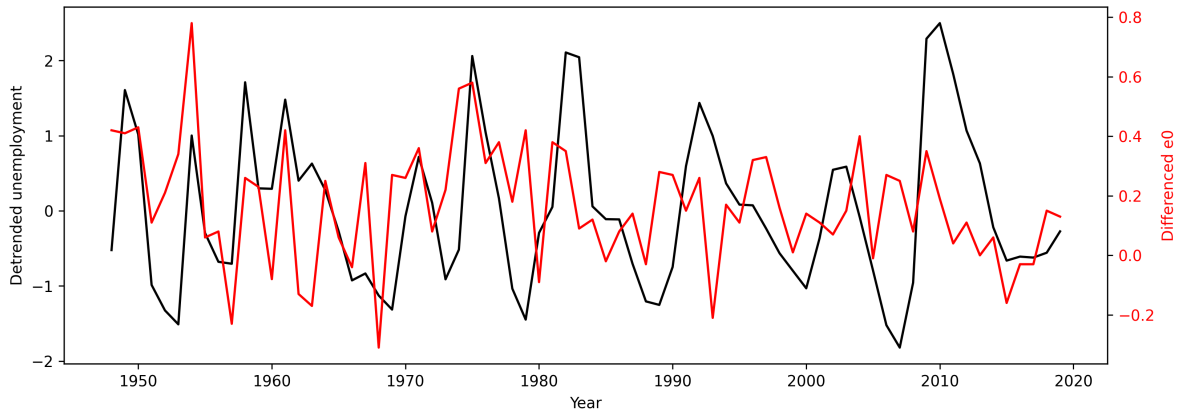
Analysis of temporal differences

- One might model annual changes in life expectancy, rather than difference from a trend. In this case, we consider the variable

$$\Delta e_n = e_n - e_{n-1}, \quad (6)$$

computed as

```
# Get full e_data for lagged values
yr_full = e_data['Year'].values
e_full = e_data['e0'].values
delta_e = e - e_full[np.isin(yr_full, yr-1)]
```



- The relationship between unemployment and differenced life expectancy is harder to see than when HP-detrended.
- The relationship is also harder to find by statistical methods:

```
a4 = SARIMAX(delta_e, exog=u_hp_df, order=(1, 0, 1)).fit()
```

Coefficient of u_hp: 0.0462
Standard error: 0.0253
z-statistic: 1.83
Residual std dev: 0.195513
Log-likelihood: 14.76

- Temporal differencing z -statistic of $0.0462 / 0.0253 = 1.83$ which is weaker evidence than the z -statistic of 3.57 for HP-detrended LEB.

Multiple testing considerations

- A scientific principle for interpreting experimental results is as follows: *An experiment which finds evidence of an effect is usually a better foundation for future investigations than one which fails to find evidence.*
- The experiment which found no evidence of an effect might have been a bad choice of experiment, or might have been carried out poorly.
- The principle of preference for methods giving positive results must be balanced with consideration of *multiple testing*. If we make 20 hypothesis tests, we expect one to be significant at the 5% level just by chance. There is a danger in trying many approaches and settling on one that claims statistical significance.
- The generalizability of any result is tentative until confirmed in other studies.

Acknowledgments

- Compiled on January 27, 2026 using Python.
- Licensed under the [Creative Commons Attribution-NonCommercial license](#). Please share and remix non-commercially, mentioning its origin.
- We acknowledge [previous versions of this course](#).

References

- [1] Robert J. Hodrick and Edward C. Prescott. “Postwar U.S. Business Cycles: An Empirical Investigation”. In: *Journal of Money, Credit and Banking* 29.1 (Feb. 1997), pp. 1–16.
- [2] E. L. Ionides, Z. Wang, and J. A. Tapia Granados. “Macroeconomic effects on mortality revealed by panel analysis with nonlinear trends”. In: *Annals of Applied Statistics* 7 (2013), pp. 1362–1385.

- [3] William F. Ogburn and Dorothy S. Thomas. “The Influence of the Business Cycle on Certain Social Conditions”. In: *Journal of the American Statistical Association* 18.139 (1922), pp. 324–340.
- [4] José A. Tapia Granados et al. “Individual Joblessness, Contextual Unemployment, and Mortality Risk”. In: *American Journal of Epidemiology* 180.3 (2014), pp. 280–287.