Modeling and Analysis of Time Series Data
Chapter 2: Estimating trend and autocovariance

STATS 531, Winter 2026

Edward Ionides

General notation for time series data and models

Definition: A **time series** is a sequence of numbers, called **data**. In general, we will suppose that there are N numbers, $y_1^*, y_2^*, \ldots, y_N^*$, collected at an increasing sequence of times, $t_1, t_2, \ldots, t_N$.

- We write $1{:}N$ for the sequence $\{1, 2, \ldots, N\}$ and we write the collection of numbers $\{y_n^*, n = 1, \ldots, N\}$ as $y_{1:N}^*$.
- A **time series model** is a collection of jointly defined random variables, $Y_1, Y_2, \ldots, Y_N$.
- We write this collection of random variables as $Y_{1:N}$.

Joint density function

- Like all jointly defined random variables, the distribution of $Y_{1:N}$ is defined by a joint density function, which we write as

$$f_{Y_{1:N}}(y_1, \ldots, y_N \,;\, \theta). \tag{1}$$

- Here, θ is a vector of parameters.
- The density is defined for arbitrary outcomes $y_{1:N}$. Its value at the data, $y^*_{1:N}$, is called the **likelihood**,

$$L(\theta) = f_{Y_{1:N}}(y^*_1, \ldots, y^*_N \,;\, \theta). \tag{2}$$

Notation for densities

- Our notation for densities generalizes. We write $f_Y(y)$ for the density of a random variable Y evaluated at y, and $f_{YZ}(y, z)$ for the joint density of the pair of random variables (Y, Z) evaluated at (y, z). We can also write $f_{Y|Z}(y \mid z)$ for the conditional density of Y given Z.
- For discrete data, such as count data, our model may also be discrete and we interpret the density function as a probability mass function. Expectations and probabilities are integrals for continuous models, and sums for discrete models. Otherwise, everything remains the same. We will write formulas only for the continuous case. You can swap integrals for sums when working with discrete models.

- Scientifically, we consider the hypothesis that $y^*_{1:N}$ is well described as a realization of $Y_{1:N}$ for an unknown value of θ.
- Our notation distinguishes between the model, $Y_{1:N}$; an arbitrary realization of the model, $y_{1:N}$; the specific sequence of numbers that we observed as data, $y^*_{1:N}$.
- Time series texts commonly ignore these distinctions. For example, Shumway and Stoffer [2] call all these quantities y_t.

Review: Random variables

Question. What is a random variable?

Question. What is a collection of jointly defined random variables?

Question. What is a probability density function? What is a joint density function? What is a conditional density function?

Question. What does it mean to say that "θ is a vector of parameters?"

There are different answers to these questions, but you should be able to write down an answer that you are satisfied with.

Review: Expectation

Random variables usually have an **expected value**, and in this course they always do. We write $\mathrm{E}[X]$ for the expected value of a random variable X.

Question. Review question: What is expected value? How is it defined? How can it fail to exist for a properly defined random variable?

Trend, also called the mean function

Definition: The **mean function**, for $n \in 1{:}N$, is

$$\mu_n = \mathrm{E}[Y_n] = \int_{-\infty}^{\infty} y_n \, f_{Y_n}(y_n) \, dy_n$$

- **Mean function** and **trend** are synonyms. We can say **model trend** to clarify that this is a property of the model, not data.
- We say "function" since μ_n is formally a function of n.
- Sometimes, it makes sense to think of time as continuous. Then, we write $\mu(t)$ for the expected value of an observation at time t. We only make observations at the discrete collection of times $t_{1:N}$ and so we require $\mu(t_n) = \mu_n$.
- A time series may have measurements evenly spaced in time, but our notation does not insist on this. Time series data may contain missing values or unequally spaced observations.

- μ_n may depend on θ, the parameter vector. We can write $\mu_n(\theta)$ or $\mu(t\,;\,\theta)$ to make this explicit.
- We write $\widehat{\mu}_n(y_{1:N})$ to be some estimator of μ_n, i.e., a map which is applied to data to give an estimate of μ_n. An appropriate choice of $\widehat{\mu}_n$ will depend on the data and the model.
- The **estimated mean function** or **estimated trend** is the value of the estimator when applied to our data, written as

$$\widehat{\mu}_n = \widehat{\mu}_n(y^*_{1:N}). \tag{3}$$

- $\widehat{\mu}_n$ denotes both the estimator function and its value evaluated at $y^*_{1:N}$. This is standard terminology for data analysis. The ambiguity can be clarified when it is helpful.

Mean stationary models

Definition: If $\mu_n = \mu$, so the mean is assumed constant, the model is called **mean stationary**.

In this case, we might estimate μ using the mean estimator,

$$\hat{\mu}(y_{1:N}) = \frac{1}{N}\sum_{n=1}^{N} y_n. \tag{4}$$

- $\hat{\mu} = \hat{\mu}(y^*_{1:N})$ is the **sample mean** or **data mean**.
- We can compute $\hat{\mu}$ for any dataset. It is only a reasonable estimate of the mean function when a mean stationary model is appropriate.
- Data cannot formally be mean stationary. Mean stationarity is a property of a model.

Properties of models vs properties of data

Question. Properties of models vs properties of data.

Consider these two statements. Does is matter which we use?

1. "The data look mean stationary."
2. "A mean stationary model looks appropriate for these data."

Autocovariance and autocorrelation

Definition: Assuming that variances and covariances exist for the random variables $Y_{1:N}$, we write

$$\gamma_{m,n} = \mathrm{Cov}(Y_m, Y_n) = \mathrm{E}\big[(Y_m - \mu_m)(Y_n - \mu_n)\big]. \tag{5}$$

This is the **autocovariance** function, a function of m and n.

- We write Γ for the $N \times N$ matrix whose (m, n) entry is $\gamma_{m,n}$.

Definition: If the covariance between two observations depends only on their time difference, the time series model is **covariance stationary**. For observations equally spaced in time, the autocovariance function is a function of a lag, h,

$$\gamma_h = \gamma_{n,n+h}. \tag{6}$$

- For a covariance stationary model, and a mean estimate $\hat{\mu}_n$, an estimate for γ_h is the **sample autocovariance function**,

$$\hat{\gamma}_h = \frac{1}{N} \sum_{n=1}^{N-h} (y_n - \hat{\mu}_n)\,(y_{n+h} - \hat{\mu}_{n+h}). \tag{7}$$

Definition: Dividing the autocovariance by the variance gives the **autocorrelation function** ρ_h,

$$\rho_h = \frac{\gamma_h}{\gamma_0}.$$

We can analogously construct the standard autocorrelation estimator,

$$\hat{\rho}_h(y_{1:N}) = \frac{\hat{\gamma}_h(y_{1:N})}{\hat{\gamma}_0(y_{1:N})},$$

which leads to an estimate known as the **sample autocorrelation**,

$$\hat{\rho}_h = \hat{\rho}_h(y^*_{1:N}) = \frac{\hat{\gamma}_h}{\hat{\gamma}_0}.$$

- It is common to use ACF as an acronym for any or all of the autocorrelation function, sample autocorrelation function, autocovariance function, and sample autocovariance function. **If you use the acronym ACF, it is good to remove ambiguity by defining it**.

Sample statistics exist without a model

- The sample autocorrelation and sample autocovariance functions are statistics computed from the data. They exist, and can be computed, even when the data are not well modeled as covariance stationary. However, in that case, it does not make sense to view them as estimators of the autocorrelation and autocovariance functions (which exist as functions of a lag h only for covariance stationary models).
- Formally, we should not talk about the correlation or covariance of data. These are properties of models. We can talk about the sample autocorrelation or sample autocovariance of data.

Least squares estimation of a trend

We analyze a time series of global mean annual temperature from https://climate.nasa.gov/vital-signs/global-temperature/.
These data are in degrees Celsius measured as an anomaly from a 1951-1980 base. This is climatology jargon for saying that the sample mean of the temperature over the interval 1951-1980 was subtracted from all time points.

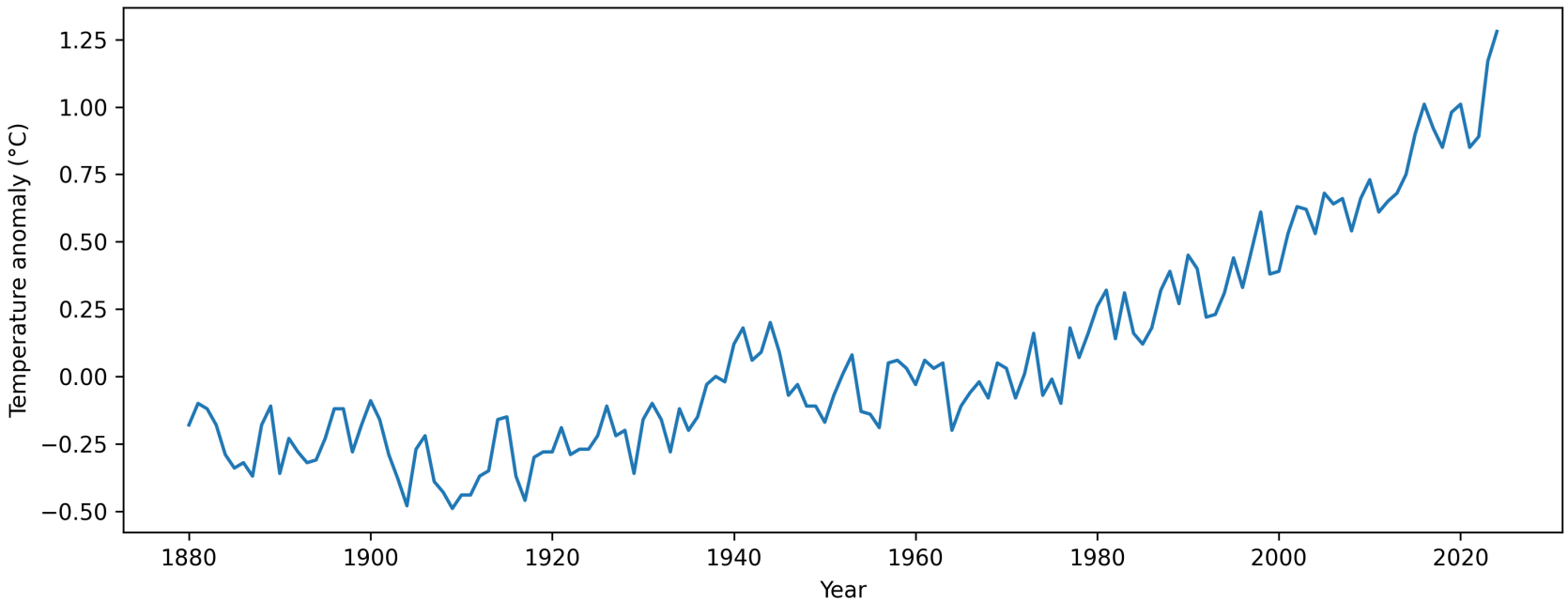
```
global_temp = pd.read_csv("Global_Temperature.txt",
  sep='\s+', comment='#', engine='python')
print(global_temp.head(3))
```

```
   Year  Annual  Smoothed
0  1880   -0.18     -0.11
1  1881   -0.10     -0.14
2  1882   -0.12     -0.18
```

Time series plot code

```
plt.figure(figsize=(10, 4))
plt.plot(global_temp['Year'],
    global_temp['Annual'], '-')
plt.xlabel('Year')
plt.ylabel('Temperature anomaly (°C)')
plt.tight_layout()
plt.show()
```

Mean global temperature anomaly, degrees C



Climate change and statistical analysis

- These data should make all of us pause for thought about the future of our planet.
- Understanding climate change involves understanding a highly complex system of physical, chemical and biological processes. It is hard to know if gigantic models that attempt to capture all important parts of the global climate dynamics are an accurate description of what is happening.
- There is value in statistical analysis, which can tell us what evidence there is for how things are, or are not, changing.

A quote from *Science* (18 December 2015, volume 350, page 1461) remains pertinent:

"Scientists are still debating whether—and, if so, how—warming in the Arctic and dwindling sea ice influences extreme weather events at midlatitudes. **Model limitations**, **scarce data** on the warming Arctic, and the **inherent variability** of the systems make answers elusive."

Fitting a least squares model with a quadratic trend

Perhaps the simplest trend model that makes sense looking at these data is a quadratic trend,

$$\mu(t) = \beta_0 + \beta_1 t + \beta_2 t^2. \tag{8}$$

To write the least squares estimate of β_0, β_1 and β_2, we set up matrix notation. Write

$$\mu = (\mu_1, \mu_2, \dots, \mu_N)^\top \tag{9}$$

for the column vector describing the mean function, and similarly,

$$\beta = (\beta_0, \beta_1, \beta_2)^\top. \tag{10}$$

Design matrix

Then, defining

$$Z = \begin{pmatrix} 1 & 1880 & 1880^2 \\ 1 & 1881 & 1881^2 \\ 1 & 1882 & 1882^2 \\ \vdots & \vdots & \vdots \end{pmatrix}, \tag{11}$$

we can write

$$\mu = Z\beta.$$

OLS estimator

We write $y_{1:N}$ as a column vector,

$$y \quad = \quad (y_1, y_2, \dots, y_N)^\top. \tag{12}$$

The ordinary least squares (OLS) estimator of β is

$$\hat{\beta}_{\mathrm{OLS}}(y_{1:N}) = (Z^\top Z)^{-1} Z^\top y, \tag{13}$$

with corresponding OLS estimate

$$\hat{\beta}_{\mathrm{OLS}} = \hat{\beta}_{\mathrm{OLS}}(y^*_{1:N}) = (Z^\top Z)^{-1} Z^\top y^*.$$

Fitting the model in Python

We can carry out this computation in Python by

```
global_temp['Year_sq'] = global_temp['Year'] ** 2
lm_fit = ols('Annual ~ Year + Year_sq',
    data=global_temp).fit()
```

```
print(lm_fit.summary())
```

OLS Regression Results

Dep. Variable:	Annual	R-squared:	0.906
Model:	OLS	Adj. R-squared:	0.905
Method:	Least Squares	F-statistic:	687.0
Date:	Fri, 16 Jan 2026	Prob (F-statistic):	9.64e-74
Time:	11:46:31	Log-Likelihood:	102.02
No. Observations:	145	AIC:	-198.0
Df Residuals:	142	BIC:	-189.1
Df Model:	2		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	338.8755	24.425	13.874	0.000	290.592	387.159
Year	-0.3555	0.025	-14.200	0.000	-0.405	-0.306
Year_sq	9.315e-05	6.41e-06	14.528	0.000	8.05e-05	0.000

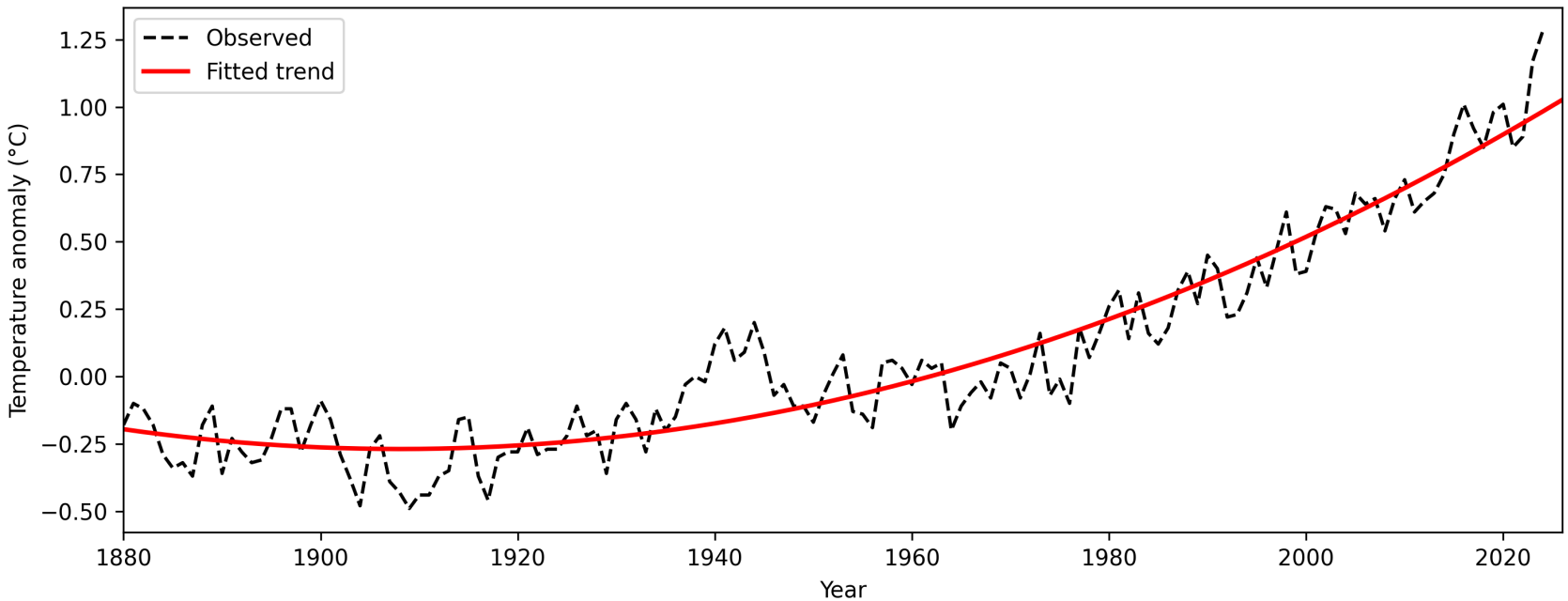
Omnibus:	4.057	Durbin-Watson:	0.862
Prob(Omnibus):	0.132	Jarque-Bera (JB):	3.777
Skew:	0.394	Prob(JB):	0.151
Kurtosis:	3.069	Cond. No.	9.28e+09

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctl
[2] The condition number is large, 9.28e+09. This might indicate that there are strong multicollinearity or other numerical problems.

We can check visually how well this model fits the data.

```
yr = np.arange(1880, 2027)
X_pred = np.column_stack([np.ones(len(yr)),yr,yr**2])
beta = lm_fit.params.values
prediction = X_pred @ beta
plt.figure(figsize=(10, 4))
plt.plot(global_temp['Year'], global_temp['Annual'],
    'k--', label='Observed')
plt.plot(yr, prediction, 'r-', linewidth=2,
    label='Fitted trend')
plt.xlabel('Year')
plt.ylabel('Temperature anomaly (°C)')
plt.xlim(yr.min(), yr.max())
plt.legend()
plt.tight_layout()
plt.show()
```

Fitted trend plot



Assessment

- The overall estimated trend seems a reasonable fit for the data.
- If we want to attach uncertainty to our parameter estimates, and consequently to our forecast, we need a time series model $Y_{1:N}$, which we write in column vector form as

$$Y = (Y_1, Y_2, \dots, Y_N)^\top.$$

White noise error model

The usual model behind OLS is the independent error model, known in time series analysis as the **white noise** error model:

[L1] $$Y = Z\beta + \epsilon,$$

where $\epsilon = \epsilon_{1:N}$ is a vector of independent identically distributed (iid) random variables with mean zero and constant variance, $\mathrm{E}[\epsilon_n] = 0$ and $\mathrm{Var}(\epsilon_n) = \sigma^2$.

- Standard linear model software, such as `ols` in Python's `statsmodels`, provides confidence intervals based on this model.

For model L1, the estimator $\widehat{\beta}_{\mathrm{OLS}}(y_{1:N})$ is unbiased, since

$$\begin{aligned}
\mathrm{E}[\widehat{\beta}_{\mathrm{OLS}}(Y_{1:N})] &= \mathrm{E}[(Z^\top Z)^{-1} Z^\top Y] \\
&= \mathrm{E}[(Z^\top Z)^{-1} Z^\top \{Z\beta + \epsilon\}] \\
&= (Z^\top Z)^{-1} Z^\top \{Z\beta + \mathrm{E}[\epsilon]\} \\
&= (Z^\top Z)^{-1} (Z^\top Z)\beta \\
&= \beta
\end{aligned}$$

- A result for linear models is that $\widehat{\beta}_{\mathrm{OLS}}(y_{1:N})$ is the minimum variance unbiased estimator for model L1.

Variance/covariance for the white noise error model

- The variance/covariance matrix of $\hat{\beta}_{\mathrm{OLS}}(Y_{1:N})$ under this model [1] is

$$\mathrm{Var}[\hat{\beta}_{\mathrm{OLS}}(Y_{1:N})] = \sigma^2 (Z^\top Z)^{-1},$$

which is estimated using an estimate for σ of

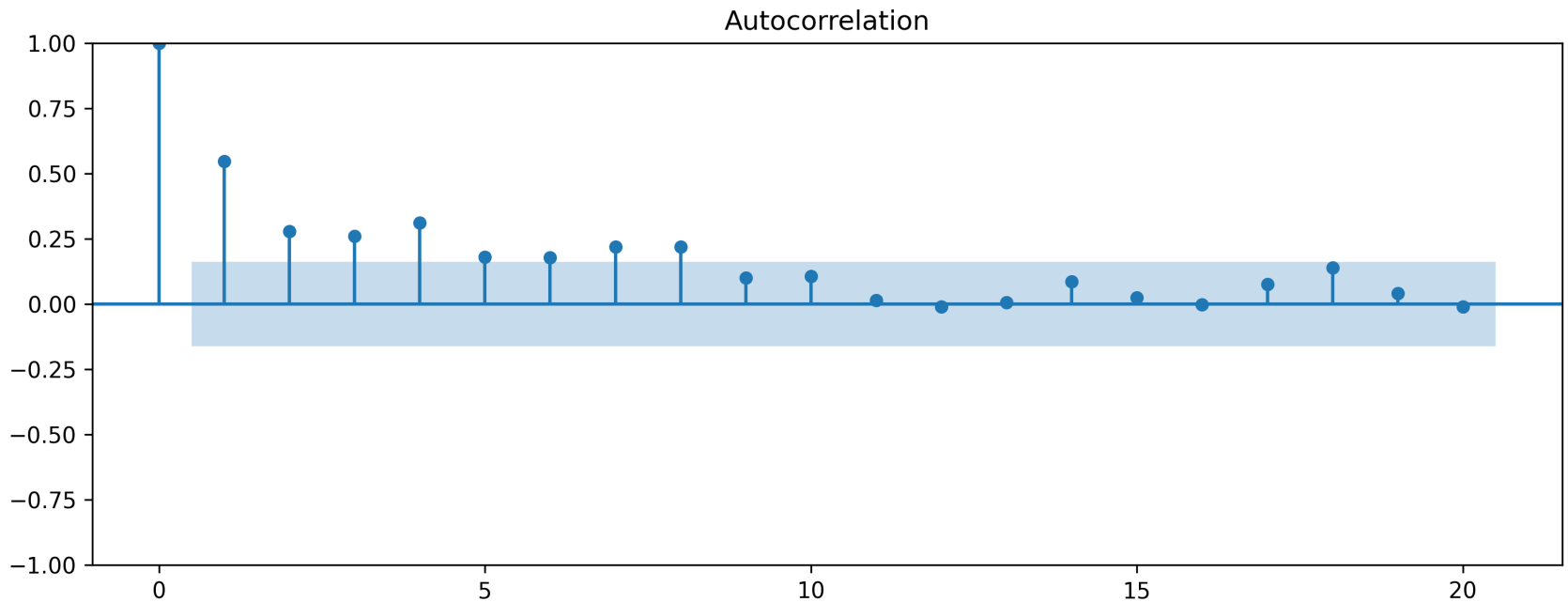
$$\hat{\sigma}_{\mathrm{OLS}} = \sqrt{\frac{1}{N-d}(y - Z\hat{\beta}_{\mathrm{OLS}})^\top (y - Z\hat{\beta}_{\mathrm{OLS}})},$$

where d is the number of covariates, i.e., the number of columns of Z.

Autocorrelated residuals and trend estimation

Let's look at the residuals to assess how appropriate this model is here.

```
fig, ax = plt.subplots(figsize=(10, 4))
plot_acf(lm_fit.resid, ax=ax, lags=20,
    bartlett_confint=False)
plt.tight_layout()
plt.show()
```



Investigating the ACF plot

- The horizontal shaded region on the sample autocorrelation function (ACF) gives a measure of chance variation under the null hypothesis that the residuals are iid.
- At each lag h, the chance that the estimated ACF falls within this band is approximately 95%, under the null hypothesis.
- Thus, under the null hypothesis, one expects a fraction of $1/20$ of the lags of the sample ACF to fall outside this band.
- Here, the sample ACF confirms what we can probably see from the plot of the fitted model: the variation around the fitted model is clustered in time, so the sample ACF of the residuals is not consistent with a model having independent error terms.

Question. How does Python/statsmodels construct these horizontal dashed lines?

Understanding confidence bands

- The confidence bands use a normal distribution approximation for the sample autocorrelation estimator, with mean zero and standard deviation $1/\sqrt{N}$.
- For 95% confidence, the bands are approximately at $\pm 1.96/\sqrt{N}$.
- This approximation is appropriate when the residuals are truly independent (under the null hypothesis).

Generalized least squares for trend estimation

Suppose that we knew the covariance matrix, Γ, for a model with dependent errors,

[L2] $$Y = Z\beta + \zeta, \qquad \zeta \sim N[0, \Gamma].$$

We read "$\zeta \sim N[0, \Gamma]$" as "ζ follows a multivariate normal distribution with mean zero and covariance matrix Γ."

- The minimum variance unbiased estimator of β for model L2 is the generalized least square (GLS) estimator [1, Chapter 5]

$$\hat{\beta}_{\mathrm{GLS}}(y_{1:N}) = \left(Z^\top \Gamma^{-1} Z\right)^{-1} Z^\top \Gamma^{-1} y.$$

- The OLS estimator remains unbiased for L2 (you can check this as an exercise). In this sense it remains a reasonable estimator. It is often a practical solution to use the OLS estimator, especially for preliminary data analysis. We don't know Γ so can't necessarily make a good estimator based on the GLS model. It might be easier to get an estimate of Γ once we have a reasonable estimate of the trend.

Variance under model L2

- For model L2, the variance of the OLS estimator is

$$\mathrm{Var}[\hat{\beta}_{\mathrm{OLS}}(Y_{1:N})] = (Z^\top Z)^{-1} Z^\top \Gamma Z (Z^\top Z)^{-1}. \qquad (14)$$

  This is different from the variance under model L1.

- **CONCLUSION. It is okay to do ordinary linear regression for data which are not well modeled with uncorrelated errors. However, if we do so, we should not trust the error estimates coming from L1.**

- This is an example of a situation where some parts of the output from statistical software are reasonable (here, the parameter estimates from `ols`) and other parts are unreasonable (the corresponding standard errors and any tests based on them). The theory helps us decide which bits of computer output to use and which to ignore.

References and Acknowledgements

- Compiled on January 16, 2026.
-
- We acknowledge students and instructors for previous versions of this course.

[1] Julian James Faraway. *Practical regression and ANOVA using R*. 2002. URL: https://cran.r-project.org/doc/contrib/Faraway-PRA.pdf.

[2] Robert H Shumway and David S Stoffer. *Time Series Analysis and its Applications: With R Examples*. 4th. Springer, 2017.