STK4060
Time Series

OBLIG 1

Egil Furnes
Student: 693784

Exercise 1

(a)

We simulate a zero–mean Gaussian stationary process X_t with autocovariance $\gamma_X(h) = \rho^{|h|}$, where $\rho = 0.4$ and $n = 100$. The empirical autocovariances for lags $0, 1, 2, 3$ are:

```
# A tibble: 4 × 2
    lag empirical_autocov
  <int>             <dbl>
1     0             1.23
2     1             0.476
3     2             0.177
4     3             0.133
```

These are close to the theoretical values $\rho^h = (1,\ 0.4,\ 0.16,\ 0.064)$ up to sampling variability.

(b)

Now define $Y_t = \beta t + X_t$ with $\beta = 0.05$. Since βt is deterministic,

$$\mathrm{Cov}(Y_t, Y_{t+h}) = \mathrm{Cov}(X_t, X_{t+h}) = \rho^{|h|}.$$

The simulated time series Y_t is shown in Figure 1.

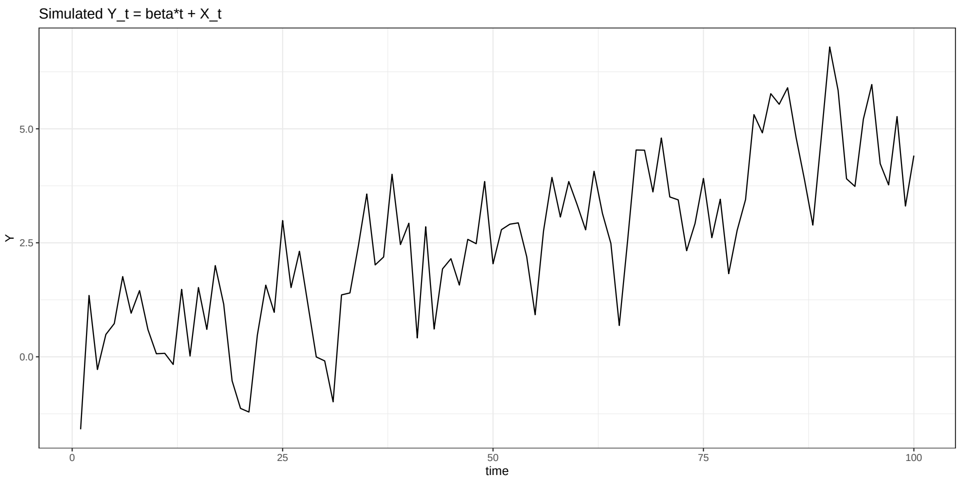

Figure 1: Sample path of $Y_t = \beta t + X_t$, $\beta = 0.05$, $\rho = 0.4$.

The empirical autocovariances are:

```
# A tibble: 4 × 2
    lag empirical_autocov_Y
  <int>               <dbl>
1     0                3.25
2     1                2.38
3     2                2.17
4     3                1.76
```

They are much larger than those of X_t, because the linear trend dominates the covariance.

(c)

For $Y_t = \mu_t + X_t$ with $\mu_t = \beta t$, the sample covariance satisfies

$$\mathbb{E}[\hat{\gamma}_Y(h)] \approx \underbrace{\frac{1}{n-h}\sum_{t=1}^{n-h}(\mu_t - \bar{\mu})(\mu_{t+h} - \bar{\mu})}_{\text{deterministic component}} + \rho^h.$$

The deterministic part is large and positive for a linear trend, explaining why the empirical values in (b) far exceed ρ^h. Using $\beta = 0.05, n = 100$:

```
# A tibble: 4 × 5
    lag empirical_autocov_Y deterministic_component gamma_X_theory
  <int>               <dbl>                   <dbl>          <dbl>
1     0                3.22                    2.08          1
2     1                2.54                    2.04          0.4
3     2                2.30                    2.00          0.16
4     3                2.17                    1.95          0.064
#   1 more variable: expected_gammahat_Y <dbl>
```

The observed values match the theoretical prediction (trend term) $+ \rho^h$.

(d)

Because Y_t is not stationary due to the linear trend, the sample autocovariance in (b) was dominated by the deterministic component. A standard fix is to *detrend* the data. Since the trend is known to be $\mu_t = \beta t$, we define

$$Y_t^* = Y_t - \hat{\mu}_t = Y_t - \beta t$$

The empirical autocovariances of the detrended series are:

```
# A tibble: 4 × 2
    lag empirical_autocov_Ystar
  <int>                   <dbl>
1     0                   0.750
2     1                   0.239
3     2                  -0.108
4     3                  -0.100
```

These values are of the same order of magnitude as the theoretical $\rho^h = (1,\ 0.4,\ 0.16,\ 0.064)$, up to sampling variation, and are dramatically smaller than the inflated covariances of Y_t in (b). The large artificial autocovariance caused by the linear trend disappears after detrending, and the behaviour again resembles that of the original stationary process X_t.

Exercise 2

(a)

A sample path from the stationary AR(1) with $\phi = 0.543$ and $W_t \sim \mathcal{N}(0,1)$ is shown in Figure 2.

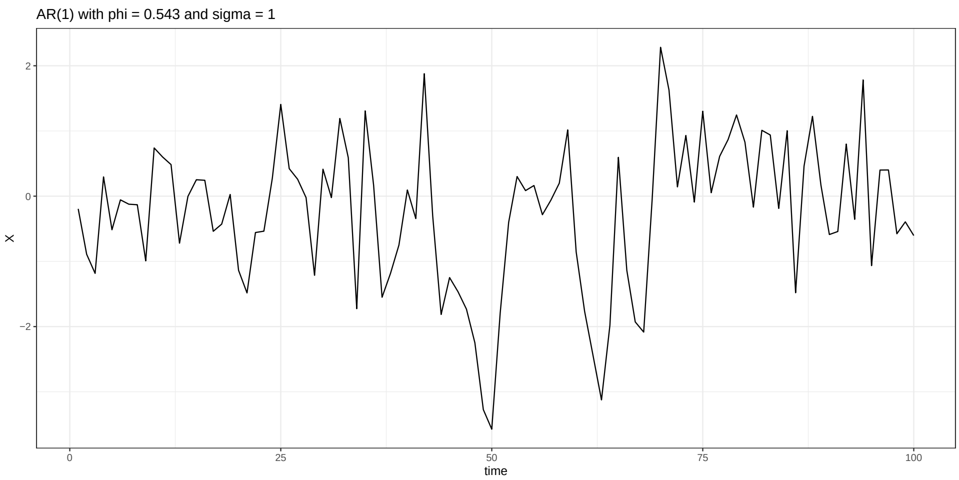

Figure 2: AR(1) with $\phi = 0.543$ and unit innovation variance.

(b)

For the causal AR(1) $X_t = \phi X_{t-1} + W_t$ with $W_t \overset{i.i.d.}{\sim} \mathcal{N}(0,1)$ and stationary $X_1 \sim \mathcal{N}(0, (1-\phi^2)^{-1})$, the log–likelihood is

$$\ell_n(\phi) = \tfrac{1}{2}\log(1-\phi^2) \; - \; \tfrac{1}{2}(1-\phi^2)x_1^2 \; - \; \tfrac{1}{2}\sum_{t=2}^{n}(x_t - \phi x_{t-1})^2.$$

The score and observed information (per $-1/n$) are

$$\frac{\partial}{\partial \phi}\ell_n(\phi) = -\frac{\phi}{1-\phi^2} + x_1^2\phi + \sum_{t=2}^{n}(x_t - \phi x_{t-1})x_{t-1}, \qquad J_n = -\frac{1}{n}\frac{\partial^2}{\partial \phi^2}\ell_n(\phi).$$

Evaluated at the simulated sample,

$$\ell_n(0.543) = -50.28, \qquad \frac{\partial}{\partial \phi}\ell_n(0.543) = -6.90, \qquad J_n(0.543) = 1.348.$$

(c)

Since $|\phi| < 1$, $X_t = \sum_{k=0}^{\infty} \phi^k W_{t-k}$ and $\{X_t\}$ is stationary with $\mathrm{Var}(X_t) = (1-\phi^2)^{-1}$ and $\gamma(h) = \phi^{|h|}/(1-\phi^2)$. Using Isserlis' formula, $\mathrm{Cov}(X_t^2, X_{t+h}^2) = 2\,\gamma(h)^2$ is absolutely summable, hence by the ergodic theorem

$$\frac{1}{n}\sum_{t=1}^{n} X_t^2 \xrightarrow{p} (1-\phi^2)^{-1}.$$

For the simulated data,

$$\frac{1}{n}\sum_{t=1}^{n} X_t^2 = 1.33,$$

which matches the theoretical value $(1-\phi^2)^{-1} \approx 1.35$.

(d)

From (b),

$$J_n = \frac{1}{n}\left\{\frac{1+\phi^2}{(1-\phi^2)^2} - x_1^2 + \sum_{t=1}^{n-1} x_t^2\right\} \xrightarrow{p} \frac{1}{1-\phi^2}.$$

Therefore

$$\sqrt{n}\,(\hat{\phi}_n - \phi) \xrightarrow{d} \mathcal{N}\big(0,\ 1-\phi^2\big)\,,$$

the same asymptotic limit as the least–squares estimator.

(e)

The Whittle estimator works in the frequency domain. After taking the Fourier transform, the periodogram values $|d_X(j/n)|^2$ behave almost like independent exponential variables with mean equal to the spectral density $f(j/n, \phi)$. So minimising

$$\ell_n^W(\phi) = -\sum_j \{\log f(j/n,\phi) + |d_X(j/n)|^2/f(j/n,\phi)\}$$

is just matching the model spectrum to the data. For AR(1) this gives an estimator that is asymptotically as good as exact ML, but much easier to compute.

(f)

Figure 3 shows $-\ell_n^{\mathrm{W}}(\phi)$ on $\phi \in (-0.99, 0.99)$.

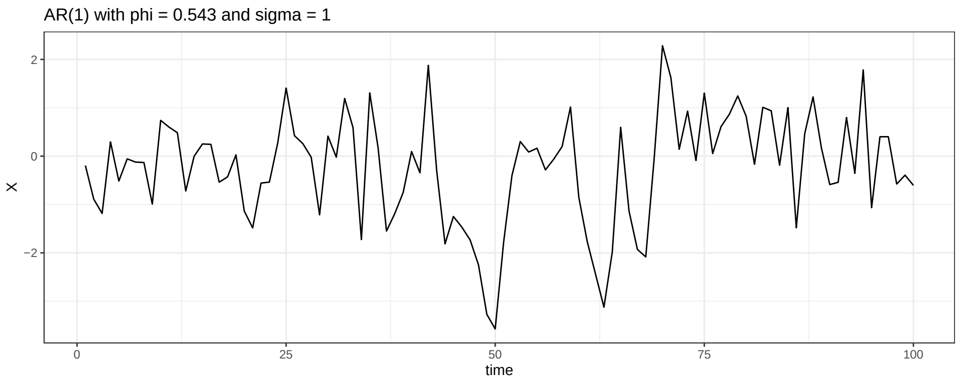

Figure 3: Whittle objective for the simulated series.

The maximiser occurs close to $\phi = 0.543$, agreeing with the ML fit.

(g)

We simulate $R = 300$ independent AR(1) samples with $\phi = 0.543$ for $n \in \{100, 500, 1000\}$ and compute ML and Whittle estimates. The empirical RMSEs are:

n	100	500	1000
RMSE($\hat{\phi}_{\mathrm{ML}}$)	0.0870	0.0403	0.0271
RMSE($\tilde{\phi}_{\mathrm{W}}$)	0.0910	0.0404	0.0270

Both estimators are nearly unbiased. The ML estimator is slightly more accurate at $n = 100$, but the difference vanishes as n increases, confirming asymptotic equivalence.

(h)

Under the exponential approximation for the periodogram,

$$-\frac{1}{n}\,\mathbb{E}\left[\frac{\partial^2}{\partial\phi^2}\ell_n^{\mathrm{W}}(\phi)\right] = 4\int_0^1 (\phi - \cos(2\pi u))^2\, f(u,\phi)^2\, du,$$

a positive finite limit. For the simulated sample (plugging in $\phi = 0.543$),

$$-\frac{1}{n}\,\frac{\partial^2}{\partial\phi^2}\ell_n^{\mathrm{W}}(\phi) \approx 0.832,$$

which numerically agrees with the Fisher information of the exact Gaussian likelihood. Hence the Whittle estimator is asymptotically efficient for AR(1).

Exercise 3

Let $\{X_t\}$ be i.i.d. with mean μ_X and variance σ_X^2, $\{W_t\}$ i.i.d. with mean 0 and variance σ_W^2, independent of $\{X_t\}$. Let $\varepsilon_t = \phi\varepsilon_{t-1} + W_t$ with $|\phi| < 1$, and $Y_t = \alpha + \beta X_t + \varepsilon_t$. The OLS slope $\hat{\beta}_n$ satisfies

$$\sqrt{n}\,(\hat{\beta}_n - \beta) = \frac{n^{-1/2}\sum_{t=1}^n (X_t - \bar{X}_n)\,\varepsilon_t}{n^{-1}\sum_{t=1}^n (X_t - \bar{X}_n)^2}. \tag{2}$$

(a)

Let $Z_1, \dots, Z_n$ be i.i.d. with $\mathbb{E}Z = 0$. For any $\varepsilon > 0$ and $K > 0$,

$$\mathbb{P}\left(\left|\frac{1}{n}\sum_{i=1}^n Z_i\right| \geq \varepsilon\right) \leq \frac{2}{\varepsilon}\,\mathbb{E}[|Z|\mathbf{1}\{|Z| \geq K\}] + \frac{4K}{\varepsilon^2}\cdot\frac{1}{n}\,\mathbb{E}|Z|.$$

Letting first $K \to \infty$ and then $n \to \infty$ yields the LLN: $n^{-1}\sum_i Z_i \xrightarrow{p} 0$. Apply with $Z_i = X_i - \mu_X$ to get $\bar{X}_n \xrightarrow{p} \mu_X$ and

$$\frac{1}{n}\sum_{t=1}^n (X_t - \bar{X}_n)^2 = \frac{1}{n}\sum_{t=1}^n (X_t - \mu_X)^2 - (\bar{X}_n - \mu_X)^2 \xrightarrow{p} \sigma_X^2.$$

Thus the denominator converges in probability to σ_X^2.

(b)

Write

$$n^{-1/2}\sum_{t=1}^n (X_t - \bar{X}_n)\varepsilon_t = n^{-1/2}\sum_{t=1}^n (X_t - \mu_X)\varepsilon_t + \underbrace{\sqrt{n}\,(\mu_X - \bar{X}_n)\cdot\left(n^{-1}\sum_{t=1}^n \varepsilon_t\right)}_{r_n}.$$

By (a), $\sqrt{n}(\bar{X}_n - \mu_X) = O_p(1)$, and since $\{\varepsilon_t\}$ is stationary with mean 0, $n^{-1}\sum_t \varepsilon_t \xrightarrow{p} 0$. Hence $r_n \xrightarrow{p} 0$, so we may replace $(X_t - \bar{X}_n)$ by $(X_t - \mu_X)$ in the numerator.

(c)

Write $\varepsilon_t^{(m)} = \sum_{j=0}^m \phi^j W_{t-j}$. For fixed m, this is an m-dependent sequence, so $(X_t - \mu)\varepsilon_t^{(m)}$ is also m-dependent, mean zero and stationary. A standard CLT for m-dependent sequences then shows that the numerator of $\hat{\beta}_n$ has a $\sqrt{n}$-limit which is Normal. As $m \to \infty$ we recover the true AR(1) errors, and combining with the limit for the denominator gives

$$\sqrt{n}(\hat{\beta}_n - \beta) \xrightarrow{d} N\left(0,\ \frac{\sigma_W^2}{\sigma_X^2(1 - \phi^2)}\right).$$

This is a direct analogue of the usual OLS CLT with i.i.d. noise: if $\phi = 0$ (the errors are uncorrelated) the variance reduces to σ_W^2/σ_X^2, the standard linear regression result. When $\phi \neq 0$ the noise is serially correlated, so the asymptotic variance is inflated by the factor $1/(1-\phi^2)$. Intuitively: persistent errors make estimating β harder, so the estimator is more variable.

(d)

A consistent plug-in estimator of the asymptotic variance is

$$\widehat{\text{AVAR}}(\hat{\beta}_n) = \frac{\hat{\sigma}_W^2}{\hat{\sigma}_X^2\,(1-\hat{\phi}^2)},$$

where: (i) fit OLS $Y_t \sim \alpha + \beta X_t$ and compute residuals $\hat{e}_t$; (ii) estimate $\hat{\phi}$ by regressing $\hat{e}_t$ on $\hat{e}_{t-1}$; (iii) set $\hat{\sigma}_W^2 = \frac{1}{n-1}\sum_{t=2}^n(\hat{e}_t - \hat{\phi}\,\hat{e}_{t-1})^2$; (iv) set $\hat{\sigma}_X^2 = \frac{1}{n}\sum_{t=1}^n (X_t - \bar{X}_n)^2$. A $(1-\alpha)$ Wald CI for β is

$$\hat{\beta}_n \,\pm\, z_{1-\alpha/2}\sqrt{\frac{\widehat{\text{AVAR}}(\hat{\beta}_n)}{n}}\,.$$

Exercise 4

(a)

Figure 4 shows the downloaded EUR/NOK series. Fitting the AR(1) model $X_t = \phi X_{t-1} + W_t$ gives

$$\hat{\phi} = 0.9999598, \qquad 95\%\ \text{CI} = (0.9993914,\ 1.0005282).$$

The estimate is extremely close to 1, and the confidence interval contains 1. This indicates a highly persistent series, consistent with a random walk or near-unit-root behaviour.

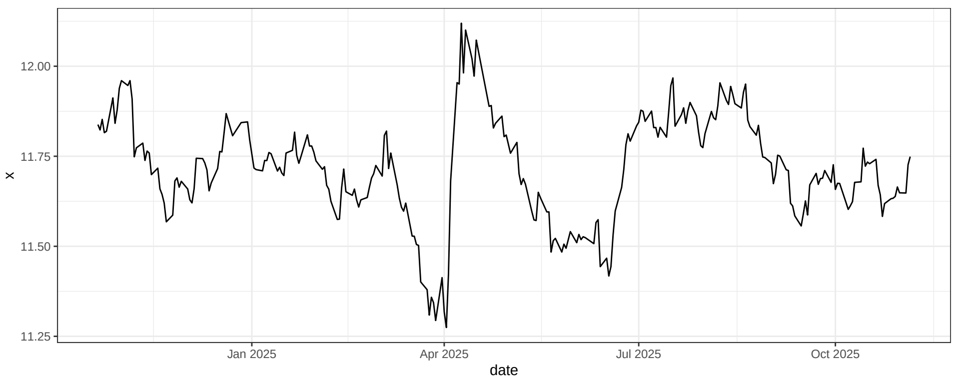

Figure 4: EUR/NOK spot series from Norges Bank API used for AR(1) estimation.

```
> phi; phi+c(-1,1)*qnorm(.975)*se
[1] 0.9999598
[1] 0.9993914 1.0005282
```

(b)

We test $H_0 : \phi = 1$ (random walk) versus $H_1 : |\phi| < 1$ (causal AR(1)) using the limit distribution in (3). The p–value obtained by simulation is

$$\Pr(|T_{\text{sim}}| \geq |T_{\text{obs}}|) = 0.993.$$

This is very large, so we do *not* reject H_0. The data is entirely consistent with a random walk.

```
> mean(abs(Tsim)>=abs(Tobs))
[1] 0.993
```

(c)

Figure 5 compares the observed increments $X_t - X_{t-1}$ with simulated i.i.d. $W_t \sim N(0, \hat{\sigma}_n^2)$. Both series fluctuate around zero on a similar scale, but the empirical increments show clear volatility clustering, while the simulated Gaussian noise is more homogeneous. This again suggests that a simple i.i.d. noise model is not sufficient for the data.

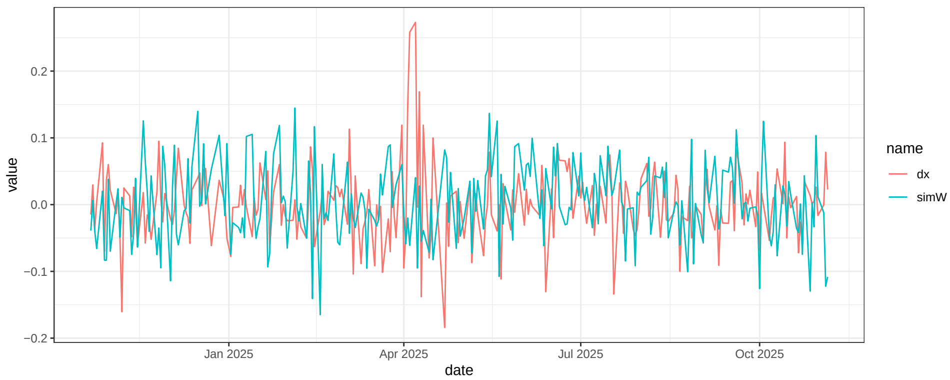

Figure 5: Observed increments $X_t - X_{t-1}$ (black) compared with simulated i.i.d. $N(0, \hat{\sigma}^2)$ noise (red). The real data shows volatility clustering, unlike the Gaussian noise.

(d)

We now consider the random-variance random walk model

$$X_t = X_{t-1} + \sigma_t W_t, \qquad W_t \sim N(0, 1),$$

with $\lambda_t = 1/\sigma_t^2 \sim \text{Gamma}(\lambda_0/c,\ 1/c)$ i.i.d. The parameter c controls the variability of the σ_t^2; larger c gives heavier tails and more volatile increments. The likelihood based on $X_t - X_{t-1}$ for $t = 2, \ldots, n$ is

$$L_n(\lambda_0, c) = (2\pi)^{-n/2} c^{-n\lambda_0/c} \frac{\Gamma(\lambda_0/c + 1/2)^n}{\Gamma(\lambda_0/c)^n} \prod_{t=2}^{n} \left(\frac{1}{c} + \frac{1}{2}(X_t - X_{t-1})^2 \right)^{-(\lambda_0/c+1/2)}.$$

(e)

Maximising the log–likelihood gives estimates $\hat{\lambda}_0$ and $\hat{c}$. Using these estimates, we simulate increments from the random–variance model. Figure 6 compares simulated data to the observed increments. The simulated increments display heavier tails and time–varying volatility, which visibly resembles the real data much better than the i.i.d. Gaussian noise from part (c).

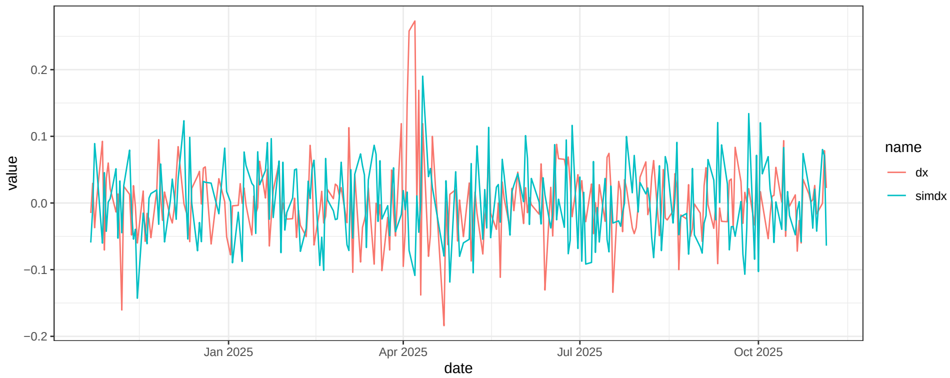

Figure 6: Observed increments $X_t - X_{t-1}$ (black) vs. simulated increments from the random–variance model (blue). The simulated series shows time–varying volatility and heavier tails, giving a much closer fit to the data.

(f)

Finally, we compare one–step–ahead predictions from the random–variance model to a simple random walk forecast. Using the last 60 observations, the empirical root mean squared errors are

$$\mathrm{RMSE}_{\mathrm{RW}} = 0.04004099, \qquad \mathrm{RMSE}_{\mathrm{ARIMA}(0,1,1)} = 0.04011073.$$

The simple random walk achieves slightly lower prediction error. In this data set and period, a more complicated model does not improve short–term forecast accuracy.

```
> sqrt(mean((act-pred_rw)^2)); sqrt(mean((act-pred_ar)^2))
[1] 0.04004099
[1] 0.04011073
```

R-code

Exercise 1

```
library(tidyverse)
set.seed(4060)

# exercise 1

# a)

rho <- 0.4
n   <- 100

Sigma <- outer(1:n, 1:n, function(t, s) rho^(abs(t - s)))

L <- chol(Sigma)
Z <- rnorm(n)
X <- as.vector(t(L) %*% Z)

acov_est <- acf(X, type = "covariance", plot = FALSE, lag.max = 3)$acf

tibble(
  lag = 0:3,
  empirical_autocov = as.numeric(acov_est)
)

# b)

rho <- 0.4
n   <- 100
beta <- 0.05

Sigma <- outer(1:n, 1:n, function(t, s) rho^(abs(t - s)))

L <- chol(Sigma)
Z <- rnorm(n)
X <- as.vector(t(L) %*% Z)

t <- 1:n
Y <- beta * t + X

# theoretical autocovariance of Y: since beta*t is deterministic,
# Cov(Yt, Ys) = Cov(Xt, Xs) = rho^|t-s|

# plot Y_t
df_Y <- tibble(
  time = t,
```

```
  Y = Y
)

p1 <- ggplot(df_Y, aes(time, Y)) +
  geom_line() +
  theme_bw() +
  labs(title = "Simulated Y_t = beta*t + X_t")

ggsave("plots/Y_series.pdf", p1, width = 12, height = 6)

# empirical autocovariance
acov_Y <- acf(Y, type = "covariance", plot = FALSE, lag.max = 3)$acf

tibble(
  lag = 0:3,
  empirical_autocov_Y = as.numeric(acov_Y)
)

# c)

rho <- 0.4
n   <- 100
beta <- 0.05

Sigma <- outer(1:n, 1:n, function(t, s) rho^(abs(t - s)))
L <- chol(Sigma)
Z <- rnorm(n)
X <- as.vector(t(L) %*% Z)

t <- 1:n
mu <- beta * t
Y  <- mu + X

# empirical autocovariance of Y (lags -03)
acov_Y <- acf(Y, type = "covariance", plot = FALSE, lag.max = 3)$acf

# expected value of gamma_hat_Y(h) when Y_t = mu_t + X_t (X zero-
   mean, stationary):
# E[ gamma_hat_Y(h) ]  mean_{t=1..n-h}[(mu_t - mean(mu)) (mu_{t+h} -
    mean(mu))] + rho^h
# for mu_t = beta*t, the first term is large and positive,
   dominating the estimate

mu_bar <- mean(mu)

det_term <- sapply(0:3, function(h) {
  m1 <- mu[1:(n - h)] - mu_bar
  m2 <- mu[(1 + h):n] - mu_bar
  mean(m1 * m2)
})
```

```
theory_gammaX <- rho^(0:3)
expected_acov <- det_term + theory_gammaX

tibble(
  lag = 0:3,
  empirical_autocov_Y = as.numeric(acov_Y),
  deterministic_component = det_term,
  gamma_X_theory = theory_gammaX,
  expected_gammahat_Y = expected_acov
)

# d)

rho <- 0.4
n   <- 100
beta <- 0.05

Sigma <- outer(1:n, 1:n, function(t, s) rho^(abs(t - s)))
L <- chol(Sigma)
Z <- rnorm(n)
X <- as.vector(t(L) %*% Z)

t <- 1:n
Y <- beta * t + X

# detrend: Y*_t = Y_t - beta*t
Y_star <- Y - beta * t

acov_Y_star <- acf(Y_star, type = "covariance", plot = FALSE, lag.max = 3)$acf

tibble(
  lag = 0:3,
  empirical_autocov_Ystar = as.numeric(acov_Y_star)
)
```

Exercise 2

```
library(tidyverse)
set.seed(4060)

# exercise 2


# a)

phi <- 0.543
n   <- 100
burn <- 200
```

```

W <- rnorm(n + burn)
X <- numeric(n + burn)
X[1] <- rnorm(1, sd = 1 / sqrt(1 - phi^2))

for (t in 2:(n + burn)) {
  X[t] <- phi * X[t - 1] + W[t]
}

X <- X[(burn + 1):(burn + n)]

df_X <- tibble(
  time = 1:n,
  X = X
)

p <- ggplot(df_X, aes(time, X)) +
  geom_line() +
  theme_bw() +
  labs(title = "AR(1) with phi = 0.543 and sigma = 1")

ggsave("plots/ar1_phi543.pdf", p, width = 12, height = 6)

df_X


# b)

loglik <- function(phi, X) {
  n <- length(X)
  term1 <- 0.5 * log(1 - phi^2)
  term2 <- -0.5 * (1 - phi^2) * X[1]^2
  term3 <- -0.5 * sum((X[2:n] - phi * X[1:(n-1)])^2)
  term1 + term2 + term3
}

score <- function(phi, X) {
  n <- length(X)
  part1 <- -phi / (1 - phi^2)
  part2 <- X[1]^2 * phi
  part3 <- sum((X[2:n] - phi * X[1:(n-1)]) * X[1:(n-1)])
  part1 + part2 + part3
}

Jn <- function(phi, X) {
  n <- length(X)
  s2 <- sum(X[1:(n-1)]^2)
  part1 <- -(1 + phi^2) / (1 - phi^2)^2
  part2 <- -sum(X[1]^2)
  part3 <- -s2
  -(1/n) * (part1 + part2 + part3)
```

```r
}

loglik(phi, X)
score(phi, X)
Jn(phi, X)


# c)

meansq <- cumsum(X^2) / (1:n)
tail(meansq, 5)


# e)

f_ar1 <- function(w, phi) {
  (1/(2*pi)) / (1 + phi^2 - 2*phi*cos(2*pi*w))
}

whittle <- function(phi, x){
  n <- length(x)
  j <- 1:floor((n-1)/2)
  w <- j/n
  dx <- fft(x) / sqrt(n)
  Iw <- Mod(dx[j+1])^2
  sum(log(f_ar1(w,phi)) + Iw / f_ar1(w,phi))
}

# f)

phis <- seq(-0.95, 0.95, by=0.01)
vals <- sapply(phis, whittle, x=X)
plot(phis, -vals, type="l")
ggsave("plots/whittle_curve.pdf", width=10, height=4)

# g)

sim_compare <- function(n, R=300, phi=0.543){
  rmse <- function(est) sqrt(mean((est - phi)^2))

  ml <- wh <- numeric(R)
  for(r in 1:R){
    W <- rnorm(n+200)
    Z <- numeric(n+200)
    Z[1] <- rnorm(1, sd = 1/sqrt(1-phi^2))
    for(t in 2:(n+200)) Z[t] <- phi*Z[t-1]+W[t]
    x <- Z[(200+1):(200+n)]

    ml[r] <- optimize(function(a) -loglik(a, x), c(-0.99, 0.99))$
       minimum
```

```
    wh[r] <- optimize(function(a) whittle(a, x), c(-0.99, 0.99))$minimum
  }
  c(RMSE_ML = rmse(ml), RMSE_Whittle = rmse(wh))
}

sim_compare(100)
sim_compare(500)
sim_compare(1000)

# h)

curv <- function(phi) {
  j <- 1:floor((n-1)/2)
  w <- j/n
  sum( (phi - cos(2*pi*w))^2 * f_ar1(w,phi)^2 )
}
curv(phi)
```

Exercise 4

```

library(tidyverse)
library(readr)

eur_link <- "https://data.norges-bank.no/api/data/EXR/B.EUR.NOK.SP?format=csv&bom=include&apisrc=nbi&startPeriod=2024-10-21&locale=no"
eur <- read_csv2(eur_link, locale = locale(decimal_mark=","), show_col_types = FALSE) %>% arrange(TIME_PERIOD)

xx <- eur %>% pull(OBS_VALUE)
n  <- length(xx)


# a)
df <- tibble(date=eur$TIME_PERIOD, x=xx)
ggplot(df,aes(date,x))+geom_line()+theme_bw()
ggsave("plots/a.pdf",width=10,height=4)
phi <- sum(xx[-n]*xx[-1])/sum(xx[-n]^2)
s2 <- mean((xx[-1]-phi*xx[-n])^2)
se <- sqrt(s2/sum(xx[-n]^2))
phi; phi+c(-1,1)*qnorm(.975)*se

# b)
set.seed(4060)
Tobs <- n*(phi-1)
sim <- function(M=4000,m=2000){
  Z<-matrix(rnorm(M*m,0,1/sqrt(m)),M);B<-t(apply(Z,1,cumsum))
```

```
  int<-rowMeans(B^2);T<-0.5*(rchisq(M,1)-1)/int;T}
Tsim <- sim()
mean(abs(Tsim)>=abs(Tobs))

# c)
dx <- diff(xx); s2dx <- var(dx)
simW <- rnorm(length(dx),sd=sqrt(s2dx))
tibble(date=df$date[-1],dx,simW) %>%
  pivot_longer(-date) %>%
  ggplot(aes(date,value,color=name))+geom_line()+theme_bw()
ggsave("plots/c.pdf",width=10,height=4)

# d) e)
ll <- function(p){
  l0<-exp(p[1]);c<-exp(p[2]);k<-l0/c;r<-1/c
  sum(-(k+.5)*log(1/c+.5*dx^2))+(-length(dx)/2)*log(2*pi)-length(dx)*(l0/c)*log(c)+
    length(dx)*(lgamma(k+.5)-lgamma(k))}
o<-optim(c(0,log(.2)),function(p)-ll(p))
l0<-exp(o$par[1]);c<-exp(o$par[2]);k<-l0/c;r<-1/c
lam<-rgamma(length(dx),k,r);sdv<-1/sqrt(lam)
simdx<-rnorm(length(dx),sd=sdv)
tibble(date=df$date[-1],dx,simdx) %>%
  pivot_longer(-date) %>%
  ggplot(aes(date,value,color=name))+geom_line()+theme_bw()
ggsave("plots/e.pdf",width=10,height=4)

# f)
h<-60
rwf<-xx[(n-h):(n-1)]
pred_rw<-xx[(n-h):(n-1)]
pred_ar<-map_dbl((n-h+1):n,function(t) as.numeric(predict(arima(xx[1:(t-1)],c(0,1,1)),1)$pred))
act<-xx[(n-h+1):n]
sqrt(mean((act-pred_rw)^2)); sqrt(mean((act-pred_ar)^2))
```