

### 13.3.3 Linear Regression with ARMA Errors

When residual analysis shows that the residuals are correlated, then one of the key assumptions of the linear model does not hold, and tests and confidence intervals based on this assumption are invalid and cannot be trusted. Fortunately, there is a solution to this problem: replace the assumption of independent noise by the weaker assumption that the noise process is stationary but possibly correlated. One could, for example, assume that the noise is an ARMA process. This is the strategy we will discuss in this section; this approach is referred to as an ARMAX model, in which the X indicates the inclusion of exogenous regression variables.

The linear regression model with ARMA errors combines the linear regression model (9.1) and the ARMA model (12.26) for the noise, so that

$$Y_t = \beta_0 + \beta_1 X_{t,1} + \cdots + \beta_p X_{t,p} + \epsilon_t, \quad (13.13)$$

where

$$(1 - \phi_1 B - \cdots - \phi_p B^p) \epsilon_t = (1 + \theta_1 B + \cdots + \theta_q B^q) u_t, \quad (13.14)$$

and  $u_1, \dots, u_n$  is white noise.

#### *Example 13.8. Demand for ice cream*

This example uses the data set `Icecream` in R's `Ecdat` package. The data are four-weekly observations from March 18, 1951, to July 11, 1953 on four variables, `cons` = U.S. consumption of ice cream per head in pints; `income` = average family income per week (in U.S. Dollars); `price` = price of ice cream (per pint); and `temp` = average temperature (in Fahrenheit). There is a total of 30 observations. Since there are 13 four-week periods per year, there are slightly over two years of data.

First, a linear model was fit with `cons` as the response and `income`, `price`, and `temp` as the predictor variables. One can see that `income` and `temp` are significant, especially `temp` (not surprisingly).

```
Call:
lm(formula = cons ~ income + price + temp, data = Icecream)

Residuals:
    Min      1Q  Median      3Q     Max 
-0.06530 -0.01187  0.00274  0.01595  0.07899 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 0.197315  0.270216   0.73   0.472    
income      0.003308  0.001171   2.82   0.009 **  
price       -1.044414  0.834357  -1.25   0.222    

```

```
temp          0.003458   0.000446    7.76  3.1e-08 ***
```

```
---
```

```
Residual standard error: 0.0368 on 26 degrees of freedom
Multiple R-squared: 0.719,      Adjusted R-squared: 0.687
F-statistic: 22.2 on 3 and 26 DF,  p-value: 2.45e-07
```

A Durbin–Watson test has a very small  $p$ -value, so we can reject the null hypothesis that the noise is uncorrelated.

```
28 options(digits=3)
29 library("car")
30 durbinWatsonTest(fit_ic_lm)

lag Autocorrelation D-W Statistic p-value
 1           0.33       1.02      0
Alternative hypothesis: rho != 0
```

Next, the linear regression model with AR(1) errors was fit and the AR(1) coefficient was over three times its standard error, indicating statistical significance. This was done using R’s `arima()` function, which specifies the regression model with the `xreg` argument. It is interesting to note that the coefficient of `income` is now nearly equal to 0 and no longer significant. The effect of `temp` is similar to that of the linear model fit, though its standard error is now larger.

```
Series: cons
ARIMA(1,0,0) with non-zero mean

Coefficients:
        ar1  intercept  income  price  temp
        0.732     0.538    0.000 -1.086  0.003
  s.e.  0.237     0.325    0.003   0.734  0.001

sigma^2 estimated as 0.00091:  log likelihood=62.1
AIC=-112  AICc=-109  BIC=-104
```

Finally, the linear regression model with MA(1) errors was fit and the MA(1) coefficient was also over three times its standard error, again indicating statistical significance. The model with AR(1) errors has a slightly better (smaller) AIC and BIC values than the model with MA(1), but there is not much of a difference between the models in terms of AIC or BIC. However, the two models imply rather different types of noise autocorrelation. The MA(1) model has no correlation beyond lag 1. The AR(1) model with coefficient 0.732 has autocorrelation persisting much longer. For example, the autocorrelation is  $0.732^2 = 0.536$  at lag 2,  $0.732^3 = 0.392$  at lag 3, and still  $0.732^4 = 0.287$  at lag 4.

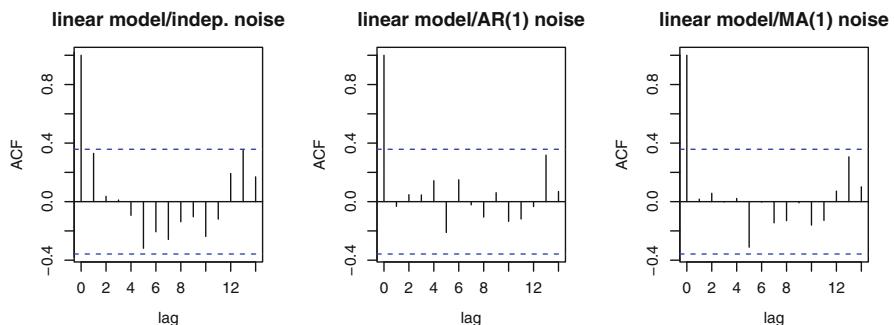
```
Series: cons
ARIMA(0,0,1) with non-zero mean
```

Coefficients:

	ma1	intercept	income	price	temp
	0.503	0.332	0.003	-1.398	0.003
s.e.	0.160	0.270	0.001	0.798	0.001

$\sigma^2$  estimated as 0.000957: log likelihood=61.6  
AIC=-111 AICc=-107 BIC=-103

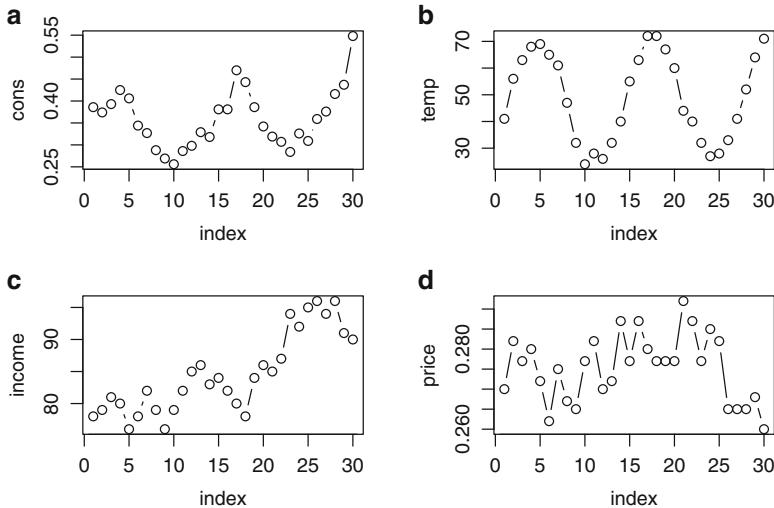
Interestingly, the estimated effect of `income` is larger and significant, much like its effect as estimated by the linear model with independent errors but unlike the result for the linear model with AR(1) errors.



**Fig. 13.9.** Ice cream consumption example. Residual ACF plots for the linear model with independent noise, the linear model with AR(1) noise, and the linear model with MA(1) noise.

The ACFs of the residuals from the linear model and from the linear models with AR(1) and MA(1) errors are shown in Fig. 13.9. The residuals from the linear model estimate  $\epsilon_1, \dots, \epsilon_n$  in (13.13), and show some autocorrelation. The residuals from the linear models with either AR(1) or MA(1) errors estimate  $u_1, \dots, u_n$  in (13.14), and show little autocorrelation. One concludes that the linear model with either AR(1) or MA(1) errors fits well and either an AR(1) or MA(1) term is needed.

Why is the effect of `income` larger and significant if the noise is assumed to be either independent or MA(1) but smaller and insignificant if the noise is AR(1)? To attempt an answer, time series plots of the four variables were examined. The plots are shown in Fig. 13.10. The strong seasonal trend in `temp` is obvious and `cons` follows this trend. There is a slightly increasing trend in `cons`, which appears to have two possible explanations. The trend might be explained by the increasing trend in `income`. However, with the strong residual autocorrelation implied by the AR(1) model, the trend in `cons` could also be explained by noise autocorrelation. One problem here is that we have a small sample size, only 30 observations. With more data it might be possible to separate the effects on ice cream consumption of income and noise autocorrelation.



**Fig. 13.10.** Time series plots for the ice cream consumption example and the variables used to predict consumption.

In summary, there is a strong seasonal component to ice cream consumption, with consumption increasing, as would be expected, with warmer temperatures. Ice cream consumption does not depend much, if at all, on `price`, though it should be noted that `price` has not varied much in this study; see Fig. 13.10. Greater variation in `price` might cause `cons` to depend more on `price`. Finally, it is uncertain whether ice cream consumption increases with family income.  $\square$

## 13.4 Multivariate Time Series

Suppose that for each  $t$ ,  $\mathbf{Y}_t = (Y_{1,t}, \dots, Y_{d,t})'$  is a  $d$ -dimensional random vector representing quantities that were measured at time  $t$ , e.g., returns on  $d$  equities. Then  $\mathbf{Y}_1, \mathbf{Y}_2, \dots$  is called a  $d$ -dimensional *multivariate time series*.

The definition of stationarity for multivariate time series is the same as given before for univariate time series. A multivariate time series is said to be *stationary* if for every  $n$  and  $m$ ,  $\mathbf{Y}_1, \dots, \mathbf{Y}_n$  and  $\mathbf{Y}_{1+m}, \dots, \mathbf{Y}_{n+m}$  have the same distributions.

### 13.4.1 The Cross-Correlation Function

Suppose that  $Y_j$  and  $Y_i$  are the two component series of a stationary multivariate time series. The *cross-correlation function* (CCF) between  $Y_j$  and  $Y_i$  is defined as

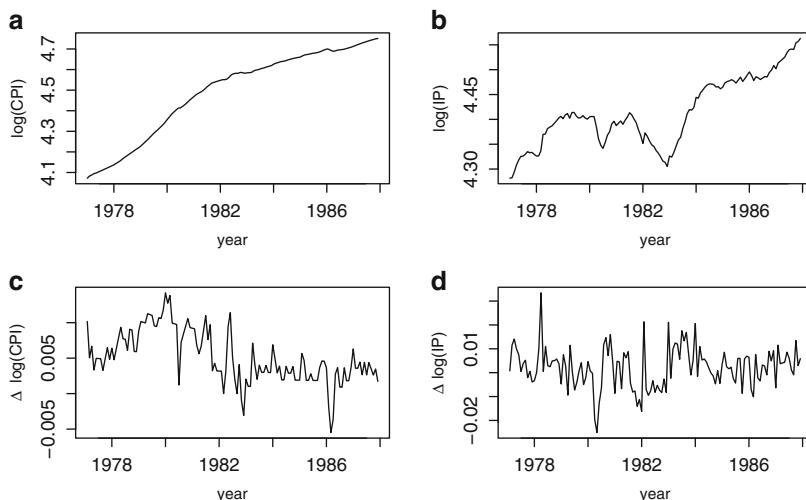
$$\rho_{Y_j, Y_i}(h) = \text{Corr}\{Y_j(t), Y_i(t-h)\} \quad (13.15)$$

and is the correlation between  $Y_j$  at a time  $t$  and  $Y_i$  at  $h$  time units earlier. As with autocorrelation,  $h$  is called the *lag*. However, unlike the ACF, the CCF is not symmetric in the lag variable  $h$ , that is,  $\rho_{Y_j, Y_i}(h) \neq \rho_{Y_j, Y_i}(-h)$ . Instead, as a direct consequence of definition (13.15), we have that  $\rho_{Y_j, Y_i}(h) = \rho_{Y_i, Y_j}(-h)$ .

The CCF can be defined for multivariate time series that are not stationary, but only weakly stationary. A multivariate time series  $\mathbf{Y}_1, \mathbf{Y}_2, \dots$  is said to be weakly stationary if the mean and covariance matrix of  $\mathbf{Y}_t$  are finite and do not depend on  $t$ , and if the right-hand side of (13.15) is independent of  $t$  for all  $j$ ,  $i$ , and  $h$ .

Cross-correlations can suggest how the component series might be influencing each other or might be influenced by a common factor. Like all correlations, cross-correlations only show statistical association, not causation, but a causal relationship might be deduced from other knowledge.

*Example 13.9. Cross-correlation between changes in CPI (consumer price index) and IP (industrial production)*



**Fig. 13.11.** (a) Time series plot of  $\log(\text{CPI})$  (b) Time series plot of  $\log(\text{IP})$  (c) Time series plot of changes in  $\log(\text{CPI})$  (d) Time series plot of changes in  $\log(\text{IP})$ .

Time series plots for the logarithm of CPI ( $cpi$ ), the logarithm of IP ( $ip$ ), and changes in  $cpi$  and  $ip$ , are shown in Fig. 13.11 panels (a)–(d), respectively. The cross-correlation function between changes in the logarithm of CPI ( $\Delta cpi$ ) and changes in the logarithm of IP ( $\Delta ip$ ) is shown in Fig. 13.12. It was created by the `ccf()` function in R.

<sup>31</sup> `CPI.dat = read.csv("CPI.dat.csv")`

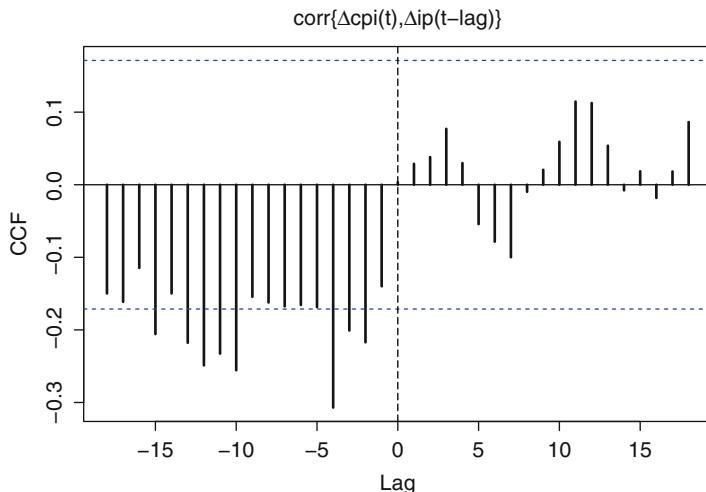
<sup>32</sup> `CPI_diff1 = diff(log(as.matrix(CPI.dat$cpi)[769:900,])) # 1977--1987`

```

33 IP.dat = read.csv("IP.dat.csv")
34 IP_diff1 = diff(log(as.matrix(IP.dat$IP)[697:828,]))      # 1977--1987
35 ccf(CPI_diff1, IP_diff1)

```

The largest absolute cross-correlations are at negative lags and these correlations are negative. This means that an above-average (below-average) change in  $cpi$  predicts a future change in  $ip$  that is below (above) average. As just emphasized, correlation does not imply causation, so we cannot say that changes in  $cpi$  cause opposite changes in future  $ip$ , but the two series behave as if this were happening. Correlation does imply predictive ability. Therefore, if we observe an above-average change in  $cpi$ , then we should predict future changes in  $ip$  that will be below average. In practice, we should use the currently observed changes in both  $cpi$  and  $ip$ , not just  $cpi$ , to predict future changes in  $ip$ . We will discuss prediction using two or more related time series in Sect. 13.4.5.  $\square$



**Fig. 13.12.** Sample CCF for  $\Delta cpi$  and  $\Delta ip$ . Note the negative correlation at negative lags, that is, between the  $cpi$  and future values of  $ip$ .

### 13.4.2 Multivariate White Noise

A  $d$ -dimensional multivariate time series  $\mathbf{Y}_1, \mathbf{Y}_2, \dots$  is a weak  $WN(\boldsymbol{\mu}, \boldsymbol{\Sigma})$  process if

1.  $E(\mathbf{Y}_t) = \boldsymbol{\mu}$  (constant and finite) for all  $t$ ;
2.  $\text{COV}(\mathbf{Y}_t) = \boldsymbol{\Sigma}$  (constant and finite) for all  $t$ ; and
3. for all  $t \neq s$ , all components of  $\mathbf{Y}_t$  are uncorrelated with all components of  $\mathbf{Y}_s$ .

Notice that if  $\Sigma$  is not diagonal, then there is cross-correlation between the components of  $\mathbf{Y}_t$  because  $\text{Corr}(Y_{j,t}, Y_{i,t}) = \Sigma_{j,i}$ ; in other words, there may be nonzero *contemporaneous* correlations. However, for all  $1 \leq j, i \leq d$ ,  $\text{Corr}(Y_{j,t}, Y_{i,s}) = 0$  if  $t \neq s$ .

Furthermore,  $\mathbf{Y}_1, \mathbf{Y}_2, \dots$  is an i.i.d.  $\text{WN}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$  process if, in addition to conditions 1–3,  $\mathbf{Y}_1, \mathbf{Y}_2, \dots$  are independent and identically distributed. If  $\mathbf{Y}_1, \mathbf{Y}_2, \dots$  are also multivariate normally distributed, then they are a Gaussian  $\text{WN}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$  process.

### 13.4.3 Multivariate ACF Plots and the Multivariate Ljung–Box Test

The ACF for multivariate time series includes the  $d$  marginal ACFs for each univariate series  $\{\rho_{Y_i}(h) : i = 1, \dots, d\}$ , and the  $d(d - 1)/2$  CCFs for all unordered pairs of the univariate series  $\{\rho_{Y_j, Y_i}(h) : 1 \leq j < i \leq d\}$ . It is sufficient to only consider the unordered pairs because  $\rho_{Y_j, Y_i}(h) = \rho_{Y_i, Y_j}(-h)$ .

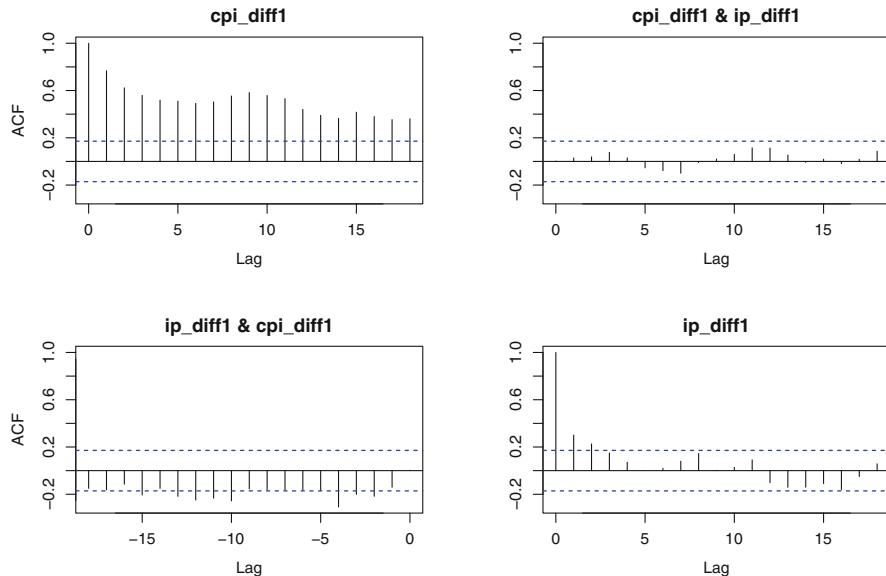
In R, if two (or more) univariate time series with matching time indices are stored as  $n \times 1$  vectors, the `cbind()` function can be used to make an  $n \times 2$  matrix consisting of the joint time series. The `acf()` function may be applied to such multivariate time series. The sample ACF for  $(\Delta cpi, \Delta ip)'$  is shown in Fig. 13.13, and generated by the following commands in R.

```
36 CPI_IP = cbind(CPI_diff1, IP_diff1)
37 acf(CPI_IP)
```

The marginal sample ACF for  $\Delta cpi$  and  $\Delta ip$  are shown in the first and second diagonal panels, respectively. Both show significant serial correlation, but there is much more persistence in the first. The sample CCF for  $\Delta cpi$  and  $\Delta ip$  has been split between the top right and bottom left panels by positive and negative lags, respectively. Notice that combining the off-diagonal panels in Fig. 13.13 reproduce the CCF shown in Fig. 13.12.

Each of the panels in Fig. 13.13 include *test bounds* to test the null hypothesis that an individual autocorrelation or lagged cross-correlation coefficient is 0. As in the univariate case, the usual level of the test is 0.05, and one can expect to see about 1 out of 20 sample correlations outside the test bounds simply by chance. Also, as in the univariate case, a simultaneous test is available.

Let  $\boldsymbol{\rho}(h)$  denote the  $d \times d$  lag- $h$  cross-correlation matrix for a  $d$ -dimensional multivariate time series. The null hypothesis of the multivariate Ljung–Box test is  $H_0 : \boldsymbol{\rho}(1) = \boldsymbol{\rho}(2) = \dots = \boldsymbol{\rho}(K) = \mathbf{0}$  for some  $K$ , say  $K = 5$  or 10. If the multivariate Ljung–Box test rejects, then we conclude that one or more of  $\boldsymbol{\rho}(1), \dots, \boldsymbol{\rho}(K)$  is nonzero. If, in fact, the lagged cross-correlation 1 to  $K$  are all zero, then there is only a 1 in 20 chance of falsely concluding that they are not all zero, assuming a level 0.05 test. In contrast, if the lagged cross-correlation are tested one at time, then there is a much higher chance of concluding that one or more is nonzero.



**Fig. 13.13.** Sample ACF for  $(\Delta \text{cpi}, \Delta \text{ip})'$ . The marginal sample ACF for  $\Delta \text{cpi}$  and  $\Delta \text{ip}$  are shown in the first and second diagonal panels, respectively; the sample CCF for  $\Delta \text{cpi}$  and  $\Delta \text{ip}$  has been split between the top right and bottom left panels by positive and negative lags, respectively.

The following commands will conduct the multivariate Ljung–Box test in R for the bivariate series  $(\Delta \text{cpi}, \Delta \text{ip})'$ .

```
38 source("SDAFE2.R")
39 mLjungBox(CPI_IP, lag = 10)

      K    Q(K) d.f. p-value
1 10  532.48   40      0
```

The multivariate Ljung–Box test statistic was 532.48, and the approximate  $p$ -value was 0, confirming that there is significant serial correlation in the first  $K = 10$  lags.

#### 13.4.4 Multivariate ARMA Processes

A  $d$ -dimensional multivariate time series  $\mathbf{Y}_1, \mathbf{Y}_2, \dots$  is a multivariate ARMA  $(p, q)$  process with mean  $\boldsymbol{\mu}$  if for  $d \times d$  matrices  $\boldsymbol{\Phi}_1, \dots, \boldsymbol{\Phi}_p$  and  $\boldsymbol{\Theta}_1, \dots, \boldsymbol{\Theta}_q$ ,

$$\mathbf{Y}_t - \boldsymbol{\mu} = \boldsymbol{\Phi}_1(\mathbf{Y}_{t-1} - \boldsymbol{\mu}) + \cdots + \boldsymbol{\Phi}_p(\mathbf{Y}_{t-p} - \boldsymbol{\mu}) + \boldsymbol{\epsilon}_t + \boldsymbol{\Theta}_1 \boldsymbol{\epsilon}_{t-1} + \cdots + \boldsymbol{\Theta}_q \boldsymbol{\epsilon}_{t-q}, \quad (13.16)$$

where  $\boldsymbol{\epsilon}_1, \dots, \boldsymbol{\epsilon}_n$  is a multivariate weak WN( $\mathbf{0}, \boldsymbol{\Sigma}$ ) process. Multivariate AR processes (the case  $q = 0$ ) are also called vector AR or VAR processes and are widely used in practice.

As an example, a bivariate AR(1) process can be written as

$$\begin{pmatrix} Y_{1,t} - \mu_1 \\ Y_{2,t} - \mu_2 \end{pmatrix} = \begin{pmatrix} \phi_{1,1} & \phi_{1,2} \\ \phi_{2,1} & \phi_{2,2} \end{pmatrix} \begin{pmatrix} Y_{1,t-1} - \mu_1 \\ Y_{2,t-1} - \mu_2 \end{pmatrix} + \begin{pmatrix} \epsilon_{1,t} \\ \epsilon_{2,t} \end{pmatrix},$$

where

$$\boldsymbol{\Phi} = \boldsymbol{\Phi}_1 = \begin{pmatrix} \phi_{1,1} & \phi_{1,2} \\ \phi_{2,1} & \phi_{2,2} \end{pmatrix}.$$

Therefore,

$$Y_{1,t} = \mu_1 + \phi_{1,1}(Y_{1,t-1} - \mu_1) + \phi_{1,2}(Y_{2,t-1} - \mu_2) + \epsilon_{1,t}$$

and

$$Y_{2,t} = \mu_2 + \phi_{2,1}(Y_{1,t-1} - \mu_1) + \phi_{2,2}(Y_{2,t-1} - \mu_2) + \epsilon_{2,t},$$

so that  $\phi_{i,j}$  is the amount of “influence” of  $Y_{j,t-1}$  on  $Y_{i,t}$ . Similarly, for a bivariate AR( $p$ ) process,  $\phi_{i,j}^k$  (the  $(i, j)$ th component of  $\boldsymbol{\Phi}_k$ ) is the influence of  $Y_{j,t-k}$  on  $Y_{i,t}$ ,  $k = 1, \dots, p$ .

For a  $d$ -dimensional AR(1), it follows from (13.16) with  $p = 1$  and  $\boldsymbol{\Phi} = \boldsymbol{\Phi}_1$  that

$$E(\mathbf{Y}_t | \mathbf{Y}_{t-1}, \dots, \mathbf{Y}_1) = E(\mathbf{Y}_t | \mathbf{Y}_{t-1}) = \boldsymbol{\mu} + \boldsymbol{\Phi}(\mathbf{Y}_{t-1} - \boldsymbol{\mu}). \quad (13.17)$$

How does  $E(\mathbf{Y}_t)$  depend on the more distant past, say on  $\mathbf{Y}_{t-2}$ ? To answer this question, we can generalize (13.17). To keep notation simple, assume that the mean has been subtracted from  $\mathbf{Y}_t$  so that  $\boldsymbol{\mu} = \mathbf{0}$ . Then

$$\mathbf{Y}_t = \boldsymbol{\Phi}\mathbf{Y}_{t-1} + \boldsymbol{\epsilon}_t = \boldsymbol{\Phi}\{\boldsymbol{\Phi}\mathbf{Y}_{t-2} + \boldsymbol{\epsilon}_{t-1}\} + \boldsymbol{\epsilon}_t$$

and, because  $E(\boldsymbol{\epsilon}_{t-1} | \mathbf{Y}_{t-2}) = \mathbf{0}$  and  $E(\boldsymbol{\epsilon}_t | \mathbf{Y}_{t-2}) = \mathbf{0}$ ,

$$E(\mathbf{Y}_t | \mathbf{Y}_{t-2}) = \boldsymbol{\Phi}^2\mathbf{Y}_{t-2}.$$

By similar calculations,

$$E(\mathbf{Y}_t | \mathbf{Y}_{t-k}) = \boldsymbol{\Phi}^k\mathbf{Y}_{t-k}, \text{ for all } k > 0. \quad (13.18)$$

It can be shown using (13.18), that the mean will explode if any of the eigenvectors of  $\boldsymbol{\Phi}$  are greater than 1 in magnitude. In fact, an AR(1) process is stationary if and only if all of the eigenvalues of  $\boldsymbol{\Phi}$  are less than 1 in absolute value. The `eigen()` function in R can be used to find the eigenvalues.

*Example 13.10. A bivariate AR model for  $\Delta \text{cpi}$  and  $\Delta \text{ip}$*

This example uses the CPI and IP data sets discussed in earlier examples ( $\text{cpi}$  and  $\text{ip}$  denote the log transformed series). Bivariate AR processes were fit to  $(\Delta \text{cpi}, \Delta \text{ip})'$  using R's function `ar()`. AIC as a function of  $p$  is shown

below. The two best-fitting models are AR(1) and AR(5), with the latter being slightly better by AIC. Although BIC is not part of `ar()`'s output, it can be calculated easily since  $BIC = AIC + \{\log(n) - 2\}p$ . Because  $\{\log(n) - 2\} = 2.9$  in this example, it is clear that BIC is much smaller for the AR(1) model than for the AR(5) model. For this reason and because the AR(1) model is so much simpler to analyze, we will use the AR(1) model.

```

40 CPI_IP = cbind(CPI_diff1,IP_diff1)
41 arFit = ar(CPI_IP,order.max=10)
42 options(digits=2)
43 arFit$aic

      0      1      2      3      4
P   127.99  0.17  1.29  5.05  3.40
AIC  0.00   6.87  9.33 10.83 13.19 14.11
      5      6      7      8      9      10
      0.00   6.87  9.33 10.83 13.19 14.11

```

The commands and results for fitting the bivariate AR(1) model are

```
44 arFit1 = ar(CPI_IP, order.max = 1) ; arFit1
```

with

$$\hat{\Phi} = \begin{pmatrix} 0.767 & 0.0112 \\ -0.330 & 0.3014 \end{pmatrix}$$

and

$$\hat{\Sigma} = \begin{pmatrix} 5.68e-06 & 3.33e-06 \\ 3.33e-06 & 6.73e-05 \end{pmatrix}. \quad (13.19)$$

The function `ar()` does not estimate  $\mu$ , but  $\mu$  can be estimated by the sample mean, which is  $(0.0052, 0.0021)'$ .

```
45 colMeans(CPI_IP)
```

It is useful to look at the two off-diagonals of  $\hat{\Phi}$ . Since  $\Phi_{1,2} = 0.01 \approx 0$ ,  $Y_{2,t-1}$  (lagged *ip*) has little influence on  $Y_{1,t}$  (*cpi*), and since  $\Phi_{2,1} = -0.330$ ,  $Y_{1,t-1}$  (lagged *cpi*) has a substantial negative effect on  $Y_{2,t}$  (*ip*), given the other variables in the model. It should be emphasized that “effect” means statistical association, not necessarily causation. This agrees with what we found when looking at the CCF for these series in Example 13.9.

How does *ip* depend on *cpi* further back in time? To answer this question we look at the (1, 2) elements of the following powers of  $\Phi$ :

```

46 bPhi = arFit1$ar[,] ; bPhi
47 bPhi2 = bPhi %*% bPhi ; bPhi2
48 bPhi3 = bPhi2 %*% bPhi ; bPhi3
49 bPhi4 = bPhi3 %*% bPhi ; bPhi4
50 bPhi5 = bPhi4 %*% bPhi ; bPhi5

```

$$\begin{aligned}\widehat{\boldsymbol{\Phi}}^2 &= \begin{pmatrix} 0.58 & 0.012 \\ -0.35 & 0.087 \end{pmatrix}, \quad \widehat{\boldsymbol{\Phi}}^3 = \begin{pmatrix} 0.44 & 0.010 \\ -0.30 & 0.022 \end{pmatrix}, \\ \widehat{\boldsymbol{\Phi}}^4 &= \begin{pmatrix} 0.34 & 0.0081 \\ -0.24 & 0.0034 \end{pmatrix}, \quad \text{and} \quad \widehat{\boldsymbol{\Phi}}^5 = \begin{pmatrix} 0.26 & 0.0062 \\ -0.18 & -0.0017 \end{pmatrix}.\end{aligned}$$

What is interesting here is that the (1,2) elements, that is,  $-0.35$ ,  $-0.30$ ,  $-0.24$ , and  $-0.18$ , decay to zero slowly, much like the CCF. This helps explain why the AR(1) model fits the data well. This behavior where the cross-correlations are all negative and decay only slowly to zero is quite different from the behavior of the ACF of a univariate AR(1) process. For the latter, the correlations either are all positive or else alternate in sign, and in either case, unless the lag-1 correlation is nearly equal to 1, the correlations decay rapidly to 0.

In contrast to these negative correlations between  $\Delta cpi$  and future  $\Delta ip$ , it follows from (13.19) that the white noise series has a positive, albeit small, correlation of  $3.33/\sqrt{(5.68)(67.3)} = 0.17$ . The white noise series represents unpredictable changes in the  $\Delta cpi$  and  $\Delta ip$  series, so we see that the unpredictable changes have positive correlation. In contrast, the negative correlations between  $\Delta cpi$  and future  $\Delta ip$  concern predictable changes.

Figure 13.14 shows the ACF of the  $\Delta cpi$  and  $\Delta ip$  residuals and the CCF of these residuals. There is little auto- or cross-correlation in the residuals at nonzero lags, indicating that the AR(1) has a satisfactory fit. Figure 13.14 was produced by the `acf()` function in R. When applied to a multivariate time series, `acf()` creates a matrix of plots. The univariate ACFs are on the main diagonal, the CCFs at positive lags are above the main diagonal, and the CCFs at negative values of lag are below the main diagonal.  $\square$

### 13.4.5 Prediction Using Multivariate AR Models

Forecasting with multivariate AR processes is much like forecasting with univariate AR processes. Given a multivariate AR( $p$ ) time series  $\mathbf{Y}_1, \dots, \mathbf{Y}_n$ , the forecast of  $\mathbf{Y}_{n+1}$  is

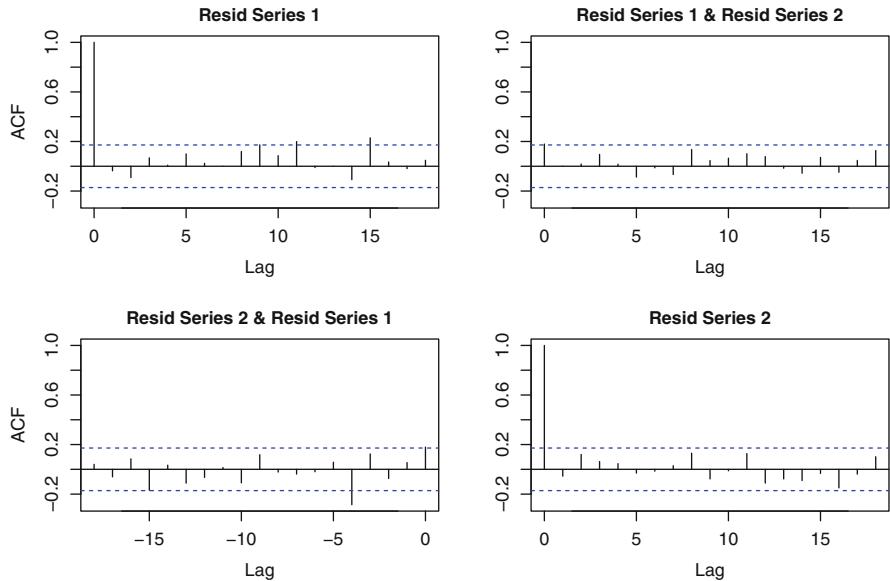
$$\widehat{\mathbf{Y}}_{n+1} = \widehat{\boldsymbol{\mu}} + \widehat{\boldsymbol{\Phi}}_1(\mathbf{Y}_n - \widehat{\boldsymbol{\mu}}) + \cdots + \widehat{\boldsymbol{\Phi}}_p(\mathbf{Y}_{n+1-p} - \widehat{\boldsymbol{\mu}}),$$

the forecast of  $\mathbf{Y}_{n+2}$  is

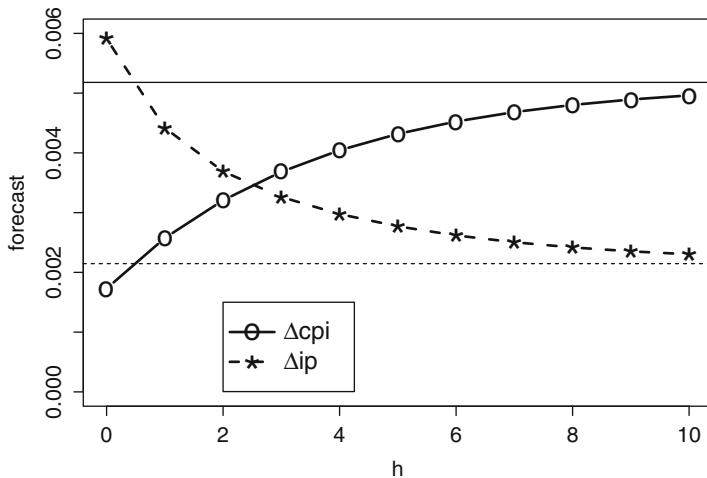
$$\widehat{\mathbf{Y}}_{n+2} = \widehat{\boldsymbol{\mu}} + \widehat{\boldsymbol{\Phi}}_1(\widehat{\mathbf{Y}}_{n+1} - \widehat{\boldsymbol{\mu}}) + \cdots + \widehat{\boldsymbol{\Phi}}_p(\mathbf{Y}_{n+2-p} - \widehat{\boldsymbol{\mu}}),$$

and so forth, so that for all  $h$ ,

$$\widehat{\mathbf{Y}}_{n+h} = \widehat{\boldsymbol{\mu}} + \widehat{\boldsymbol{\Phi}}_1(\widehat{\mathbf{Y}}_{n+h-1} - \widehat{\boldsymbol{\mu}}) + \cdots + \widehat{\boldsymbol{\Phi}}_p(\widehat{\mathbf{Y}}_{n+h-p} - \widehat{\boldsymbol{\mu}}), \quad (13.20)$$



**Fig. 13.14.** The ACF and CCF for the residuals when fitting a bivariate AR(1) model to  $(\Delta \text{cpi}, \Delta \text{ip})'$ . Top left: The ACF of  $\Delta \text{cpi}$  residuals. Top right: The CCF of  $\Delta \text{cpi}$  and  $\Delta \text{ip}$  residuals with positive values of lag. Bottom left: The CCF of  $\Delta \text{cpi}$  and  $\Delta \text{ip}$  residuals with negative values of lag. Bottom right: The ACF of  $\Delta \text{ip}$  residuals.



**Fig. 13.15.** Forecasts of  $\Delta \text{cpi}$  (solid) and  $\Delta \text{ip}$  (dashed) using a bivariate AR(1) model. The number of time units ahead is  $h$ . At  $h = 0$ , the last observed values of the time series are plotted. The two horizontal lines are at the means of the series, and the forecasts will asymptote to these lines as  $h \rightarrow \infty$  since this model is stationary.

where we use the convention that  $\widehat{\mathbf{Y}}_t = \mathbf{Y}_t$  if  $t \leq n$ . For an AR(1) model, repeated application of (13.20) shows that

$$\widehat{\mathbf{Y}}_{n+h} = \widehat{\boldsymbol{\mu}} + \widehat{\boldsymbol{\Phi}}_1^h (\mathbf{Y}_n - \widehat{\boldsymbol{\mu}}). \quad (13.21)$$

*Example 13.11.* Using a bivariate AR(1) model to predict CPI and IP

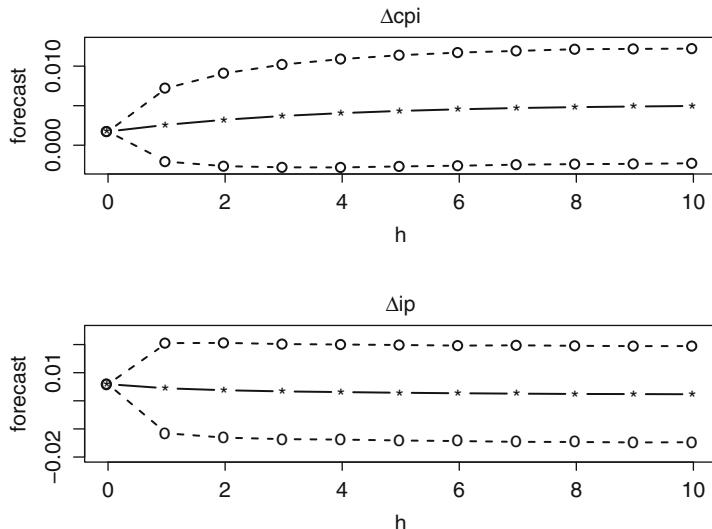
The  $\Delta\text{CPI}$  and  $\Delta\text{IP}$  series were forecast using (13.21) with estimates found in Example 13.10. Figure 13.15 shows forecasts up to 10 months ahead for both CPI and IP. Figure 13.16 shows forecast limits computed by simulation using the techniques described in Sect. 12.12.2 generalized to a multivariate time series.  $\square$

## 13.5 Long-Memory Processes

### 13.5.1 The Need for Long-Memory Stationary Models

In Chap. 12, ARMA processes were used to model stationary time series. Stationary ARMA processes have only short memories in that their auto-correlation functions decay to zero exponentially fast. That is, there exist a  $D > 0$  and  $r < 1$  such that

$$\rho(k) < D|r|^k$$



**Fig. 13.16.** Forecast limits (dashed) for  $\Delta\text{cpi}$  and  $\Delta\text{ip}$  computed by simulation, and forecasts (solid). At  $h = 0$ , the last observed changes are plotted so the widths of the forecast intervals are zero.

for all  $k$ . In contrast, many financial time series appear to have long memory since their ACFs decay at a (slow) polynomial rate rather than a (fast) geometric rate, that is,

$$\rho(k) \sim Dk^{-\alpha}$$

for some  $D$  and  $\alpha > 0$ . A polynomial rate of decay is sometimes called a hyperbolic rate. In this section, we will introduce the fractional ARIMA models, which include stationary processes with long memory.

### 13.5.2 Fractional Differencing

The most widely used models for stationary, long-memory processes use fractional differencing. For integer values of  $d$  we have

$$\Delta^d = (1 - B)^d = \sum_{k=0}^d \binom{d}{k} (-B)^k. \quad (13.22)$$

In this subsection, the definition of  $\Delta^d$  will be extended to noninteger values of  $d$ . The only restriction on  $d$  will be that  $d > -1$ .

We define

$$\binom{d}{k} = \frac{d(d-1)\cdots(d-k+1)}{k!} \quad (13.23)$$

for any  $d$  except negative integers and any integer  $k \geq 0$ , except if  $d$  is an integer and  $k > d$ , in which case  $d - k$  is a negative integer and  $(d - k)!$  is not defined. In the latter case, we define  $\binom{d}{k}$  to be 0, so  $\binom{d}{k}$  is defined for all  $d$  except negative integers and for all integer  $k \geq 0$ . Only values of  $d$  greater than  $-1$  are needed for modeling long-memory processes, so we will restrict attention to this case.

The function  $f(x) = (1 - x)^d$  has an infinite Taylor series expansion

$$(1 - x)^d = \sum_{k=0}^{\infty} \binom{d}{k} (-x)^k. \quad (13.24)$$

Since  $\binom{d}{k} = 0$  if  $k > d$  and  $d > -1$  is an integer, when  $d$  is an integer we have

$$(1 - x)^d = \sum_{k=0}^{\infty} \binom{d}{k} (-x)^k = \sum_{k=0}^d \binom{d}{k} (-x)^k. \quad (13.25)$$

The right-hand side of (13.25) is the usual finite binomial expansion for  $d$  a nonnegative integer, so (13.24) extends the binomial expansion to all  $d > -1$ . Since  $(1 - x)^d$  is defined for all  $d > -1$ , we can define  $\Delta^d = (1 - B)^d$  for any  $d > -1$ . In summary, if  $d > -1$ , then

$$\Delta^d Y_t = \sum_{k=0}^{\infty} \binom{d}{k} (-1)^k Y_{t-k}. \quad (13.26)$$

### 13.5.3 FARIMA Processes

A process  $Y_t$  is a fractional ARIMA( $p, d, q$ ) process, also called an ARFIMA or FARIMA( $p, d, q$ ) process, if  $\Delta^d Y_t$  is an ARMA( $p, q$ ) process. We say that  $Y_t$  is a fractionally integrated process of order  $d$  or, simply,  $I(d)$  process. This is, of course, the previous definition of an ARIMA process extended to noninteger values of  $d$ . Usually,  $d \geq 0$ , with  $d = 0$  being the ordinary ARMA case, but  $d$  could be negative. If  $-1/2 < d < 1/2$ , then the process is stationary. If  $0 < d < 1/2$ , then it is a long-memory stationary process.

If  $d > \frac{1}{2}$ , then  $Y_t$  can be differenced an integer number of times to become a stationary process, though perhaps with long-memory. For example, if  $\frac{1}{2} < d < 1\frac{1}{2}$ , then  $\Delta Y_t$  is fractionally integrated of order  $d - 1 \in (-\frac{1}{2}, \frac{1}{2})$  and  $\Delta Y_t$  has long-memory if  $1 < d < 1\frac{1}{2}$  so that  $d - 1 \in (0, \frac{1}{2})$ .

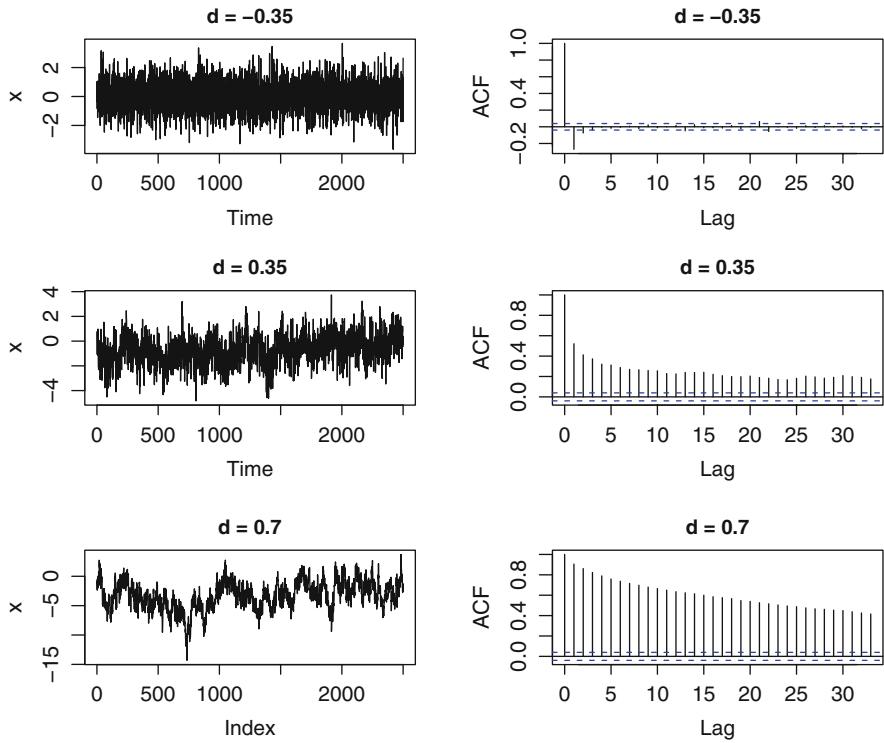
Figure 13.17 shows time series plots and sample ACFs for simulated FARIMA( $0, d, 0$ ) processes with  $n = 2,500$  and  $d = -0.35, 0.35$ , and  $0.7$ . The last case is nonstationary. The R function `simARMA0()` in the `longmemo` package was used to simulate the stationary series. For the case  $d = 0.7$ , `simARMA0()` was used to simulate a FARIMA( $0, -0.3, 0$ ) series and this was integrated to create a FARIMA( $0, d, 0$ ) with  $d = -0.3 + 1 = 0.7$ . As explained in Sect. 12.9, integration is implemented by taking partial sums, and this was done with R's function `cumsum()`.

The FARIMA( $0, 0.35, 0$ ) process has a sample ACF which drops below 0.5 almost immediately but then persists well beyond 30 lags. This behavior is typical of stationary processes with long memory. A short-memory stationary process would not have autocorrelations persisting that long, and a nonstationary processes would not have a sample ACF that dropped below 0.5 so quickly.

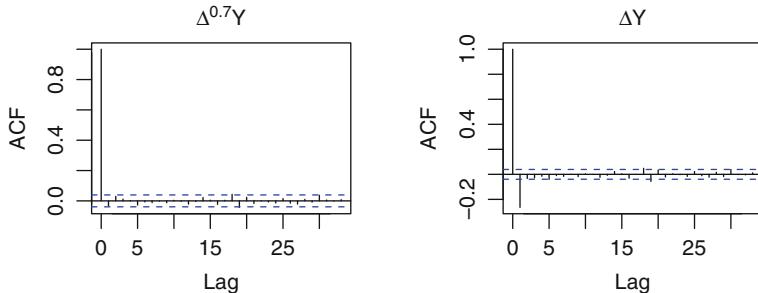
Note that the case  $d = -0.35$  in Fig. 13.17 has an ACF with a negative lag-1 autocorrelation and little additional autocorrelation. This type of ACF is often found when a time series is differenced once. After differencing, an MA term is needed to accommodate the negative lag-1 autocorrelated. A more parsimonious model can sometimes be used if the differencing is fractional. For example, consider the third series in Fig. 13.17. If it is differenced once, then a series with  $d = -0.3$  is the result. However, if it is differenced with  $d = 0.7$ , then white noise is the result. This can be seen in the ACF plots in Fig. 13.18.

#### *Example 13.12. Inflation rates—FARIMA modeling*

This example uses the inflation rates that have been studied already in Chap. 12. From the analysis in that chapter it was unclear whether to model the series as  $I(0)$  or  $I(1)$ . Perhaps it would be better to have a compromise



**Fig. 13.17.** Time series plots (left) and sample ACFs (right) for simulated  $\text{FARIMA}(0, d, 0)$ : the top series is stationary with short-term memory; the middle series is stationary with long-term memory; the bottom series is nonstationary.

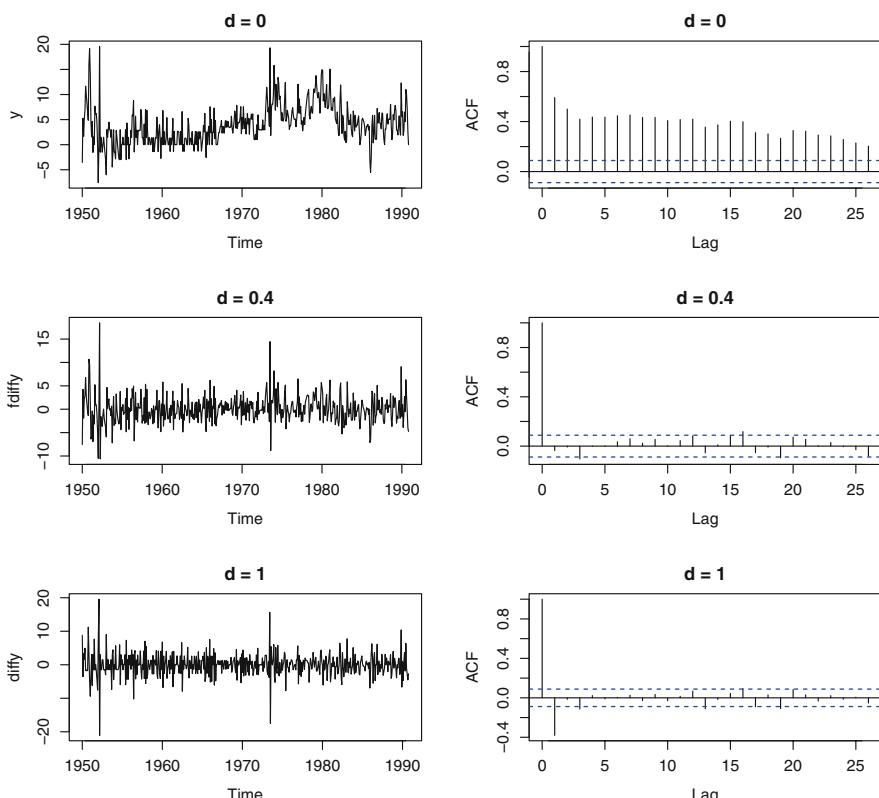


**Fig. 13.18.** Sample ACF plots for the simulated  $\text{FARIMA}(0, 0.7, 0)$  series in Figure 13.17 after differencing using  $d = 0.7$  and 1.

between these alternatives. Now, with the new tool of fractional integration, we can try differencing with  $d$  between 0 and 1. There is some reason to believe that fractional differencing is suitable for this example, since the ACF plot in Fig. 12.3 is similar to that of the  $d = 0.35$  plot in Fig. 13.17.

The function `fracdiff()` in R's `fracdiff` package will fit a FARIMA  $(p, d, q)$  process. The values of  $p$ ,  $d$ , and  $q$  must be input; we are not aware of any R function that will choose  $p$ ,  $d$ , and  $q$  automatically in the way this can be done for an ARIMA process (that is, with  $d$  restricted to be an integer) using `auto.arima()`. First, a trial value of  $d$  was chosen by using `fracdiff()` with  $p = q = 0$ , the default values. The estimate was  $\hat{d} = 0.378$ . Then, the inflation rates were fractionally differenced using this value of  $d$  and `auto.arima()` was applied to the fractionally differenced series. The result was that BIC selected  $p = q = d = 0$ . The value  $d = 0$  means that no further differencing is applied to the already fractionally differenced series. Fractional differencing was done with the `diffseries()` function in R's `fracdiff` package.

Figure 13.19 has sample ACF plots of the original series and the series differenced with  $d = 0$ , 0.4 (from rounding 0.378), and 1. The first series has a slowly decaying ACF typical of a long-memory process, the second series looks like white noise, and the third series has negative autocorrelation at lag-1 which indicates overdifferencing.



**Fig. 13.19.** Time series plots (left) and sample ACF plots (right) for the inflation rates series with differencing using  $d = 0$ , 0.4, and 1.

The conclusion is that a white noise process seems to be a suitable model for the fractionally differenced series and the original series can be model as FARIMA(0,0.378,0), or, perhaps, more simply as FARIMA(0,0.4,0).

Differencing a stationary process creates another stationary process, but the differenced process often has a more complex autocorrelation structure than the original process. Therefore, one should not *overdifference* a time series. However, if  $d$  is restricted to integer values, then often, as in this example, overdifferencing cannot be avoided.  $\square$

## 13.6 Bootstrapping Time Series

The resampling methods introduced in Chap. 6 are designed for i.i.d. univariate data but are easily extended to multivariate data. As discussed in Sect. 7.11, if  $\mathbf{Y}_1, \dots, \mathbf{Y}_n$  is a sample of vectors, then one resamples the  $\mathbf{Y}_i$  themselves, not their components, to maintain the covariance structure of the data in the resamples.

It is not immediately obvious whether one can resample a time series  $Y_1, \dots, Y_n$ . A time series is essentially a sample of size 1 from a stochastic process. Resampling a sample of size 1 in the usual way is a futile exercise—each resample is the original sample, so one learns nothing by resampling. Therefore, resampling of a time series requires new ideas.

Model-based resampling is easily adapted to time series. The resamples are obtained by simulating the time series model. For example, if the model is ARIMA( $p, 1, q$ ), then the resamples start with simulated samples of an ARMA( $p, q$ ) model with MLEs (from the differenced series) of the autoregressive and moving average coefficients and the noise variance. The resamples are the sequences of partial sums of the simulated ARMA( $p, q$ ) process.

Model-free resampling of a time series is accomplished by *block resampling*, also called the *block bootstrap*, which can be implemented using the `tsboot()` function in R's `boot` package. The idea is to break the time series into roughly equal-length blocks of consecutive observations, to resample the blocks with replacement, and then to paste the blocks together. For example, if the time series is of length 200 and one uses 10 blocks of length 20, then the blocks are the first 20 observations, the next 20, and so forth. A possible resample is the fourth block (observations 61 to 80), then the last block (observations 181 to 200), then the second block (observations 21 to 40), then the fourth block again, and so on until there are 10 blocks in the resample.

A major issue is how best to select the block length. The correlations in the original sample are preserved only within blocks, so a large block size is desirable. However, the number of possible resamples depends on the number of blocks, so a large number of blocks is also desirable. Obviously, there must be a tradeoff between the block size and the number of blocks. A full discussion of block bootstrapping is beyond the scope of this book, but see Sect. 13.7 for further reading.

## 13.7 Bibliographic Notes

Beran (1994) is a standard reference for long-memory processes, and Beran (1992) is a good introduction to this topic. Most of the time series textbooks listed in the “References” section discuss seasonal ARIMA models. For more details on HC and HAC covariance matrix estimators and the R package `sandwich` see Zeileis (2004). Enders (2004) has a section on bootstrapping time series and a chapter on multivariate time series. Reinsel (2003) is an in-depth treatment of multivariate time series; see also Hamilton (1994) for this topic. Transfer function models are another method for analyzing multivariate time series; see Box, Jenkins, and Reinsel (2008). Davison and Hinkley (1997) discuss both model-based and block resampling of time series and other types of dependent data. Lahiri (2003) provides an advanced and comprehensive account of block resampling. Bühlmann (2002) is a review article about bootstrapping time series.

## 13.8 R Lab

### 13.8.1 Seasonal ARIMA Models

This section uses seasonally non-adjusted quarterly data on income and consumption in the UK. Run the following code to load the data and plot the variable `consumption`.

```
1 library("Ecdat")
2 library("forecast")
3 data(IncomeUK)
4 consumption = IncomeUK[,2]
5 plot(consumption)
```

**Problem 1** *Describe the behavior of `consumption`. What types of differencing, seasonal, nonseasonal, or both, would you recommend? Do you recommend fitting a seasonal ARIMA model to the data with or without a log transformation? Consider also using ACF plots to help answer these questions.*

**Problem 2** *Regardless of your answers to Problem 1, find an ARIMA model that provides a good fit to `log(consumption)`. What order model did you select? (Give the orders of the nonseasonal and seasonal components.)*

**Problem 3** *Check the ACF of the residuals from the model you selected in Problem 2. Do you see any residual autocorrelation?*

**Problem 4** *Apply `auto.arima()` to `log(consumption)` using BIC. Which model is selected?*

**Problem 5** Forecast `log(consumption)` for the next eight quarters using the models you found in Problems 2 and 4. Plot the two sets of forecasts in side-by-side plots with the same limits on the x- and y-axes. Describe any differences between the two sets of forecasts.

Note: To predict an `arima` object (an object returned by the `arima()` function), use the `predict` function. To learn how the `predict()` function works on an `arima` object, use `?predict.Arima`. To forecast an object returned by `auto.arima()`, use the `forecast()` function in the `forecast` package. For example, the following code will forecast eight quarters ahead using the object returned by `auto.arima()` and then plot the forecasts.

```
6 logConsumption = log(consumption)
7 fitAutoArima = auto.arima(logConsumption, ic="bic")
8 foreAutoArima = forecast(fitAutoArima, h=8)
9 plot(foreAutoArima, xlim=c(1985.5,1987.5), ylim=c(10.7,11.2))
```

### 13.8.2 Regression with HAC Standard Errors

Run the following commands in R to compute the OLS estimates of the regression of the differenced one-month T-bill rates, `tb1_diff`, on the differenced three-month T-bill rates, `tb3_diff`.

```
1 data(Mishkin, package="Ecdat")
2 tb1_dif = diff(as.vector(Mishkin[,3]))
3 tb3_dif = diff(as.vector(Mishkin[,4]))
4 fit = lm(tb1_dif ~ tb3_dif )
5 round(summary(fit)$coef, 4)
6 acf(fit$resid)
```

**Problem 6** Is there evidence of significant autocorrelation among the residuals? Why?

Now run the following commands to compute the HC standard error estimates and their associated  $t$  values.

```
7 library(sandwich)
8 sqrt(diag(NeweyWest(fit, lag = 0, prewhite = F)))
9 coef(fit)/sqrt(diag(NeweyWest(fit, lag = 0, prewhite = F)))
```

**Problem 7** How do these  $t$  values compare to the  $t$  values from the OLS fit? Does the HC adjustment change the conclusions of the hypothesis tests?

**Problem 8** Run the commands again, but with `lag` equal to 1,2, and 3 to obtain the corresponding HAC  $t$  values. How do the  $t$  values vary with `lag`?

### 13.8.3 Regression with ARMA Noise

This section uses the `USMacroG` data set used earlier in Sect. 9.11.1. In the earlier analysis, we did not investigate residual correlation, but now we will. The model will be the regression of changes in `unemp` = unemployment rate on changes in `government` = real government expenditures and changes in `invest` = real investment by the private sector. Run the following R code to read the data, compute differences, and then fit a linear regression model with AR(1) errors.

```

1 library(AER)
2 data("USMacroG")
3 MacroDiff = as.data.frame(apply(USMacroG, 2, diff))
4 attach(MacroDiff)
5 fit1 = arima(unemp, order=c(1,0,0), xreg=cbind(invest, government))

```

**Problem 9** Fit a linear regression model using `lm()`, which assumes uncorrelated errors. Compare the two models by AIC and residual ACF plots. Which model fits better?

**Problem 10** What are the values of BIC for the model with uncorrelated errors and for the model with AR(1) errors? Does the conclusion in Problem 9 about which model fits better change if one uses BIC instead of AIC?

**Problem 11** Does the model with AR(2) noise or the model with ARMA(1,1) noise offer a better fit than the model with AR(1) noise?

### 13.8.4 VAR Models

This section uses data on the 91-day Treasury bill, the real GDP, and the inflation rate. Run the following R code to read the data, find the best-fitting multivariate AR to changes in the three series, and check the residual correlations.

```

1 TbGdpPi = read.csv("TbGdpPi.csv", header=TRUE)
2 # r = the 91-day treasury bill rate
3 # y = the log of real GDP
4 # pi = the inflation rate
5 TbGdpPi = ts(TbGdpPi, start = 1955, freq = 4)
6 del_dat = diff(TbGdpPi)
7 var1 = ar(del_dat, order.max=4, aic=T)
8 var1
9 acf(na.omit(var1$resid))

```

**Problem 12** For this problem, use the notation of Eq. (13.16) with  $q = 0$ .

- What is  $p$  and what are the estimates  $\Phi_1, \dots, \Phi_p$ ?
- What is the estimated covariance matrix of  $\epsilon_t$ ?
- If the model fits adequately, then there should be no residual auto- or cross-correlation. Do you believe that the model does fit adequately?

**Problem 13** The last three changes in  $r$ ,  $y$ , and  $\pi$  are given next. What are the predicted values of the next set of changes in these series?

*10 tail(TbGdpPi, n = 4)*

	r	y	pi
[233,]	0.07	9.7	1.38
[234,]	0.04	9.7	0.31
[235,]	0.02	9.7	0.28
[236,]	0.07	9.7	-0.47

Now fit a VAR(1) using the following commands.

*11 var1 = ar(del\_dat, order.max=1)*

Suppose we observe changes in  $r$ ,  $y$ , and  $\pi$  that are each 10% above the mean changes:

*12 yn = var1\$x.mean \* 1.1 ; yn*

**Problem 14** Compute the  $h$ -step forecasts for  $h = 1, 2$ , and 5 using  $yn$  as the most recent observation. How do these forecasts compare to the mean  $var1$x.mean$ ? For each  $h$ , compute ratios between the forecasts and the mean. How do these values compare to the starting value,  $yn/var1$x.mean = 1.1$ ? Are they closer to or farther from  $1.0 = var1$x.mean/var1$x.mean$ ? What does this suggest?

Using the fitted VAR(1) from above, examine the estimate of  $\hat{\Phi}$ :

*13 Phi\_hat = var1\$ar[, , ] ; Phi\_hat*

**Problem 15** What do the elements of  $\Phi_{\text{hat}}$  suggest about the relationships among the changes in  $r$ ,  $y$ , and  $\pi$ ?

A VAR(1) process is stationary provided that the eigenvalues of  $\Phi$  are less than one in magnitude. Compute the eigenvalues of  $\hat{\Phi}$ :

*14 eigen.values = eigen(Phi\_hat)\$values*

*15 abs(eigen.values)*

**Problem 16** Is the estimated process stationary? How does this result relate to the forecast calculations in Problem 14 above?

The dataset `MacroVars.csv` contains three US macroeconomic indicators from Quarter 1 of 1959 to Quarter 4 of 1997: Real Gross Domestic Product (a measure of economic activity), Consumer Price Index (a measure of inflation), and Federal Funds Rate (a proxy for monetary policy). Each series has been transformed to stationary based on the procedures suggested by Stock and Watson (2005).

```
16 MacroVars = read.csv("MacroVars.csv", head=TRUE)
```

**Problem 17** Fit a  $VAR(p)$  model using the `ar()` function in R using AIC (the default) to select lag order.

### Problem 18

By modifying the output of the `ar()` function as discussed in Example 13.10, use BIC to select the lag order. Comment on any differences.

#### 13.8.5 Long-Memory Processes

This section uses changes in the square root of the Consumer Price Index. The following code creates this time series.

```
1 data(Mishkin, package="Ecdat")
2 cpi = as.vector(Mishkin[,5])
3 DiffSqrtCpi = diff(sqrt(cpi))
```

**Problem 19** Plot `DiffSqrtCpi` and its ACF. Do you see any signs of long memory? If so, describe them.

Run the following code to estimate the amount of fractional differencing, fractionally difference `DiffSqrtCpi` appropriately, and check the ACF of the fractionally differenced series.

```
4 library("fracdiff")
5 fit.frac = fracdiff(DiffSqrtCpi,nar=0,nma=0)
6 fit.frac$d
7 fdiff = diffseries(DiffSqrtCpi,fit.frac$d)
8 acf(fdiff)
```

**Problem 20** Do you see any short- or long-term autocorrelation in the fractionally differenced series?

**Problem 21** Fit an ARIMA model to the fractionally differenced series using `auto.arima()`. Compare the models selected using AIC and BIC.

### 13.8.6 Model-Based Bootstrapping of an ARIMA Process

This exercise uses the price of frozen orange juice. Run the following code to fit an ARIMA model.

```

1 library(AER)
2 library(forecast)
3 data("FrozenJuice")
4 price = FrozenJuice[,1]
5 plot(price)
6 auto.arima(price, ic="bic")

```

The output from `auto.arima()`, which is needed for model-based bootstrapping, is

```

Series: price
ARIMA(2,1,0)

Coefficients:
      ar1      ar2
    0.2825  0.0570
  s.e.  0.0407  0.0408

sigma^2 estimated as 9.989:  log likelihood = -1570.11
AIC = 3146.23  AICc = 3146.27  BIC = 3159.47

```

Next, we will use the model-based bootstrap to investigate how well BIC selects the “correct” model, which is ARIMA(2,1,0). Since we will be looking at the output of each fitted model, only a small number of resamples will be used. Despite the small number of resamples, we will get some sense of how well BIC works in this context. To simulate 10 model-based resamples from the ARIMA(2,1,0) model, run the following commands.

```

7 n = length(price)
8 sink("priceBootstrap.txt")
9 set.seed(1998852)
10 for (iter in 1:10){
11   eps = rnorm(n+20)
12   y = rep(0,n+20)
13   for (t in 3:(n+20)){
14     y[t] = 0.2825*y[t-1] + 0.0570*y[t-2] + eps[t]
15   }
16   y = y[101:n+20]
17   y = cumsum(y)
18   y = ts(y, frequency=12)
19   fit = auto.arima(y, d=1, D=0, ic="bic")
20   print(fit)
21 }
22 sink()

```

The results will be sent to the file `priceBootstrap.txt`. The first two values of `y` are independent and are used to initialize the process. A burn-in period of 20 is used to remove the effect of initialization. Note the use of `cumsum()` to integrate the simulated AR(2) process and the use of `ts()` to convert a vector to a monthly time series.

**Problem 22** *How often is the “correct” AR(2) model selected?*

Now we will perform a bootstrap where the correct model AR(2) is known and study the accuracy of the estimators. Since the correct model is known, it can be fit by `arima()`. The estimates will be stored in a matrix called `estimates`. In contrast to earlier when model-selection was investigated by resampling, now a large number of bootstrap samples can be used, since `arima()` is fast and only the estimates are stored. Run the following:

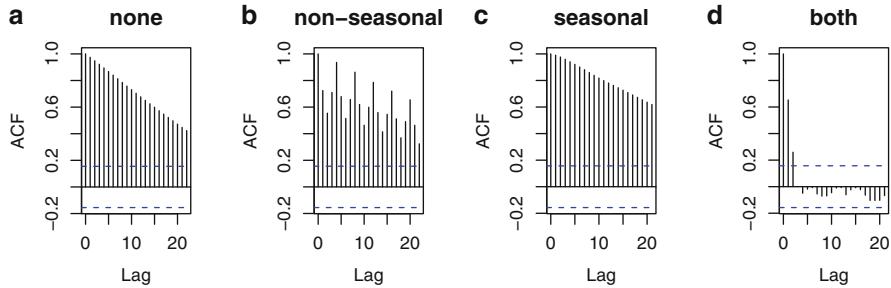
```

23 set.seed(1998852)
24 niter = 1000
25 estimates=matrix(0, nrow=niter, ncol=2)
26 for (iter in 1:niter){
27   eps = rnorm(n+20)
28   y = rep(0, n+20)
29   for (t in 3:(n+20)){
30     y[t] = .2825 *y[t-1] + 0.0570*y[t-2] + eps[t]
31   }
32   y = y[101:n+20]
33   y = cumsum(y)
34   y = ts(y, frequency=12)
35   fit=arima(y, order=c(2,1,0))
36   estimates[iter,] = fit$coef
37 }
```

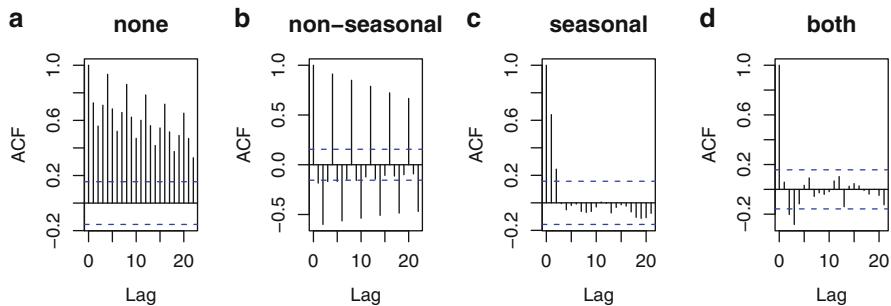
**Problem 23** *Find the biases, standard deviations, and MSEs of the estimators of the two coefficients.*

## 13.9 Exercises

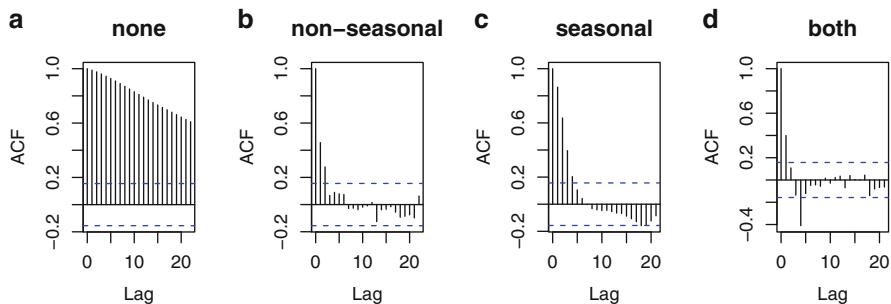
- Figure 13.20 contains ACF plots of 40 years of quarterly data, with all possible combinations of first-order seasonal and nonseasonal differencing. Which combination do you recommend in order to achieve stationarity?
- Figure 13.21 contains ACF plots of 40 years of quarterly data, with all possible combinations of first-order seasonal and nonseasonal differencing. Which combination do you recommend in order to achieve stationarity?



**Fig. 13.20.** ACF plots of quarterly data with no differencing, nonseasonal differencing, seasonal differencing, and both seasonal and nonseasonal differencing.



**Fig. 13.21.** ACF plots of quarterly data with no differencing, nonseasonal differencing, seasonal differencing, and both seasonal and nonseasonal differencing.



**Fig. 13.22.** ACF plots of quarterly data with no differencing, nonseasonal differencing, seasonal differencing, and both seasonal and nonseasonal differencing.

3. Figure 13.22 contains ACF plots of 40 years of quarterly data, with all possible combinations of first-order seasonal and nonseasonal differencing. Which combination do you recommend in order to achieve stationarity?
4. In Example 13.10, a bivariate AR(1) model was fit to  $(\Delta \text{cpi}, \Delta \text{ip})'$  and

$$\widehat{\Phi} = \begin{pmatrix} 0.767 & 0.0112 \\ -0.330 & 0.3014 \end{pmatrix}.$$

- The mean of  $(\Delta cpi, \Delta ip)'$  is  $(0.0052, 0.0021)'$  and the last observation of  $(\Delta cpi, \Delta ip)'$  is  $(0.0017, 0.0059)'$ . Forecast the next two values of  $\Delta ip$ . (The forecasts are shown in Fig. 13.15, but you should compute numerical values.)
5. Fit an ARIMA model to `income`, which is in the first column of the `IncomeUK` data set in the `Ecdat` package. Explain why you selected the model you did. Does your model exhibit any residual correlation?
  6. (a) Find an ARIMA model that provides a good fit to the variable `unemp` in the `USMacroG` data set in the `AER` package.  
 (b) Now perform a small model-based bootstrap to see how well `auto.arima()` can select the true model. To do this, simulate eight data sets from the ARIMA model selected in part (a) of this problem. Apply `auto.arima()` with BIC to each of these data sets. How often is the “correct” amount of differencing selected, that is,  $d$  and  $D$  are correctly selected? How often is the “correct” model selected? “Correct” means in agreement with the simulation model. “Correct model” means both the correct amount of differencing and the correct orders for all the seasonal and nonseasonal AR and MA components.
  7. This exercise uses the `TbGdpPi.csv` data set. In Sect. 12.15.1, nonseasonal models were fit. Now use `auto.arima()` to find a seasonal model. Which seasonal model is selected by AIC and by BIC? Do you feel that a seasonal model is needed, or is a nonseasonal model sufficient?

## References

- Beran, J. (1992) Statistical methods for data with long-range dependence. *Statistical Science*, **7**, 404–427.
- Beran, J. (1994) *Statistics for Long-Memory Processes*, Chapman & Hall, Boca Raton, FL.
- Box, G. E. P., Jenkins, G. M., and Reinsel, G. C. (2008) *Time Series Analysis: Forecasting and Control*, 4th ed., Wiley, Hoboken, NJ.
- Bühlmann, P. (2002) Bootstraps for time series. *Statistical Science*, **17**, 52–72.
- Davison, A. C. and Hinkley, D. V. (1997) *Bootstrap Methods and Their Applications*, Cambridge University Press, Cambridge.
- Enders, W. (2004) *Applied Econometric Time Series*, 2nd ed., Wiley, New York.
- Hamilton, J. D. (1994) *Time Series Analysis*, Princeton University Press, Princeton, NJ.
- Lahiri, S. N. (2003) *Resampling Methods for Dependent Data*, Springer, New York.
- Newey, W. and West, K. (1987) A simple, positive semidefinite, heteroscedasticity and autocorrelation consistent covariance matrix. *Econometrica*, **55**, 703–708.

- Reinsel, G. C. (2003) *Elements of Multivariate Time Series Analysis*, 2nd ed., Springer, New York.
- Stock, J. H. and Watson, M. W. (2005). *An empirical comparison of methods for forecasting using many predictors*, manuscript [http://www4.ncsu.edu/~arhall/bab\\_4.pdf](http://www4.ncsu.edu/~arhall/bab_4.pdf)
- White, H. (1980) A heteroscedasticity consistent covariance matrix estimator and a direct test for heteroscedasticity. *Econometrica*, **48**, 827–838.
- Zeileis, A. (2004) Econometric computing with HC and HAC covariance matrix estimators. *Journal of Statistical Software*, **11**(10), 1–17.

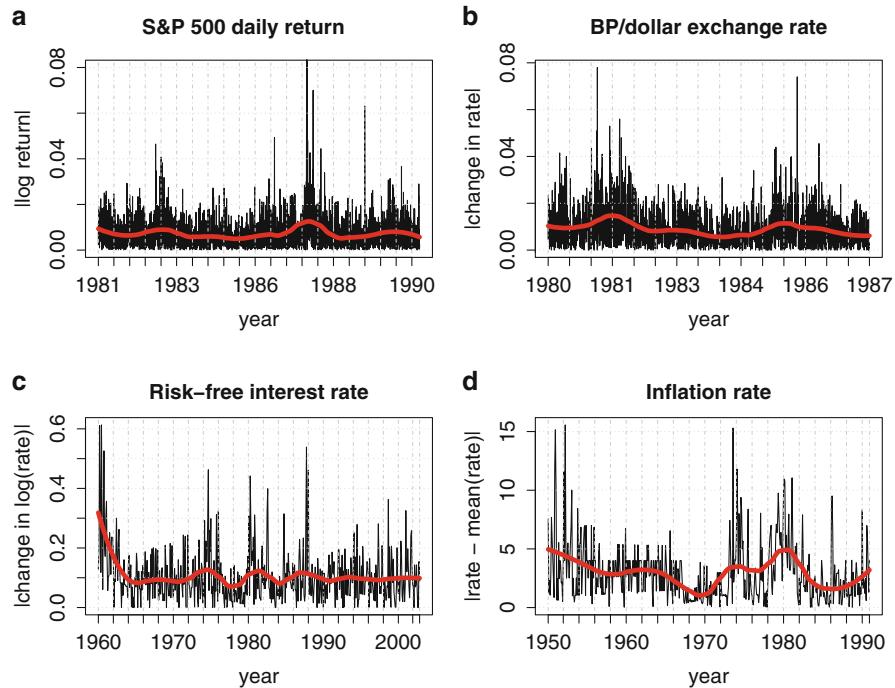
## GARCH Models

### 14.1 Introduction

As seen in earlier chapters, financial market data often exhibits volatility clustering, where time series show periods of high volatility and periods of low volatility; see, for example, Fig. 14.1. In fact, with economic and financial data, time-varying volatility is more common than constant volatility, and accurate modeling of time-varying volatility is of great importance in financial engineering.

As we saw in Chap. 12, ARMA models are used to model the conditional expectation of a process given the past, but in an ARMA model the conditional variance given the past is constant. What does this mean for, say, modeling stock returns? Suppose we have noticed that recent daily returns have been unusually volatile. We might expect that tomorrow's return is also more variable than usual. However, an ARMA model cannot capture this type of behavior because its conditional variance is constant. So we need better time series models if we want to model the nonconstant volatility. In this chapter we look at GARCH time series models that are becoming widely used in econometrics and finance because they have randomly varying volatility.

ARCH is an acronym meaning Auto-Regressive Conditional Heteroskedasticity. In ARCH models the conditional variance has a structure very similar to the structure of the conditional expectation in an AR model. We first study the first order ARCH(1) model, which is the simplest GARCH model, and analogous to an AR(1) model. Then we look at ARCH( $p$ ) models, which are analogous to AR( $p$ ) models, and GARCH (Generalized ARCH) models, which model conditional variances much as the conditional expectation is modeled by an ARMA model. Finally, we consider several multivariate GARCH processes.



**Fig. 14.1.** Examples of financial markets and economic data with time-varying volatility: (a) absolute values of S&P 500 log returns; (b) absolute values of changes in the BP/dollar exchange rate; (c) absolute values of changes in the log of the risk-free interest rate; (d) absolute deviations of the inflation rate from its mean. Loess (see Section 21.2) smooths have been added in red.

## 14.2 Estimating Conditional Means and Variances

Before looking at GARCH models, we study some general principles about modeling nonconstant conditional variance. Consider regression modeling with a *constant* conditional variance,  $\text{Var}(Y_t | X_{1,t}, \dots, X_{p,t}) = \sigma^2$ . Then the general form for the regression of  $Y_t$  on  $X_{1,t}, \dots, X_{p,t}$  is

$$Y_t = f(X_{1,t}, \dots, X_{p,t}) + \epsilon_t, \quad (14.1)$$

where  $\epsilon_t$  is independent of  $X_{1,t}, \dots, X_{p,t}$  and has expectation equal to 0 and a constant conditional variance  $\sigma_\epsilon^2$ . The function  $f(\cdot)$  is the conditional expectation of  $Y_t$  given  $X_{1,t}, \dots, X_{p,t}$ . Moreover, the conditional variance of  $Y_t$  is  $\sigma_\epsilon^2$ .

Equation (14.1) can be modified to allow conditional heteroskedasticity. Let  $\sigma^2(X_{1,t}, \dots, X_{p,t})$  be the conditional variance of  $Y_t$  given  $X_{1,t}, \dots, X_{p,t}$ . Then the model

$$Y_t = f(X_{1,t}, \dots, X_{p,t}) + \epsilon_t \sigma(X_{1,t}, \dots, X_{p,t}), \quad (14.2)$$

where  $\epsilon_t$  has conditional (given  $X_{1,t}, \dots, X_{p,t}$ ) mean equal to 0 and conditional variance equal to 1, gives the correct conditional mean and variance of  $Y_t$ .

The function  $\sigma(X_{1,t}, \dots, X_{p,t})$  should be nonnegative since it is a standard deviation. If the function  $\sigma(\cdot)$  is linear, then its coefficients must be constrained to ensure nonnegativity. Such constraints are cumbersome to implement, so nonlinear nonnegative functions are usually used instead. Models for conditional variances are often called *variance function models*. The GARCH models of this chapter are an important class of variance function models.

## 14.3 ARCH(1) Processes

Suppose for now that  $\epsilon_1, \epsilon_2, \dots$  is Gaussian white noise with unit variance. Later we will allow the noise to be i.i.d. white noise with a possibly non-normal distribution, such as, a standardized  $t$ -distribution. Then

$$E(\epsilon_t | \epsilon_{t-1}, \dots) = 0,$$

and

$$\text{Var}(\epsilon_t | \epsilon_{t-1}, \dots) = 1. \quad (14.3)$$

Property (14.3) is called *conditional homoskedasticity*.

The process  $a_t$  is an ARCH(1) process under the model

$$a_t = \epsilon_t \sqrt{\omega + \alpha a_{t-1}^2}, \quad (14.4)$$

which is a special case of (14.2) with  $f$  equal to 0 and  $\sigma$  equal to  $\sqrt{\omega + \alpha a_{t-1}^2}$ . We require that  $\omega > 0$  and  $\alpha \geq 0$  so that  $\omega + \alpha a_{t-1}^2 > 0$  for all  $t$ . It is also required that  $\alpha < 1$  in order for  $\{a_t\}$  to be stationary with a finite variance. Equation (14.4) can be written as

$$a_t^2 = \epsilon_t^2 (\omega + \alpha a_{t-1}^2),$$

which is similar to an AR(1), but in  $a_t^2$ , not  $a_t$ , and with multiplicative noise with a mean of 1 rather than additive noise with a mean of 0. In fact, the ARCH(1) model induces an ACF for  $a_t^2$  that is the same as an AR(1)'s ACF, as we will see from the calculations below.

Define

$$\sigma_t^2 = \text{Var}(a_t | a_{t-1}, \dots)$$

to be the conditional variance of  $a_t$  given past values. Since  $\epsilon_t$  is independent of  $a_{t-1}$  and  $E(\epsilon_t^2) = \text{Var}(\epsilon_t) = 1$ , we have

$$E(a_t | a_{t-1}, \dots) = 0, \quad (14.5)$$

and

$$\begin{aligned}\sigma_t^2 &= E\{(\omega + \alpha a_{t-1}^2) \epsilon_t^2 | a_{t-1}, a_{t-2}, \dots\} \\ &= (\omega + \alpha a_{t-1}^2) E\{\epsilon_t^2 | a_{t-1}, a_{t-2}, \dots\} \\ &= \omega + \alpha a_{t-1}^2.\end{aligned}\tag{14.6}$$

Equation (14.6) is crucial to understanding how GARCH processes work. If  $a_{t-1}$  has an unusually large absolute value, then  $\sigma_t$  is larger than usual and so  $a_t$  is also expected to have an unusually large magnitude. This volatility propagates since when  $a_t$  has a large magnitude that makes  $\sigma_{t+1}^2$  large, then  $a_{t+1}$  tends to be large in magnitude, and so on. Similarly, if  $a_{t-1}^2$  is unusually small, then  $\sigma_t^2$  is small, and  $a_t^2$  is also expected to be small, and so forth. Because of this behavior, unusual volatility in  $a_t$  tends to persist, though not forever. The conditional variance tends to revert to the unconditional variance provided that  $\alpha < 1$ , so that the process is stationary with a finite variance.

The unconditional, that is, marginal, variance of  $a_t$  denoted by  $\gamma_a(0)$  is obtained by taking expectations in (14.6), which gives us

$$\gamma_a(0) = \omega + \alpha \gamma_a(0)$$

for a stationary model. This equation has a positive solution if  $\alpha < 1$ :

$$\gamma_a(0) = \omega / (1 - \alpha).$$

If  $\alpha = 1$ , then  $\gamma_a(0)$  is infinite, but  $a_t$  is stationary nonetheless and is called an integrated GARCH (I-GARCH) model.

Straightforward calculations using (14.5) show that the ACF of  $a_t$  is

$$\rho_a(h) = 0 \text{ if } h \neq 0.$$

In fact, any process in which the conditional expectation of the present observation given the past is constant is an uncorrelated process.

In introductory statistics courses, it is often mentioned that independence implies zero correlation but not vice versa. A process, such as a GARCH process, in which the conditional mean is constant but the conditional variance is nonconstant is an example of an uncorrelated but dependent process. The dependence of the conditional variance on the past causes the process to be dependent. The independence of the conditional mean on the past is the reason that the process is uncorrelated.

Although  $a_t$  is an uncorrelated process, the process  $a_t^2$  has a more interesting ACF. If  $\alpha < 1$ , then

$$\rho_{a^2}(h) = \alpha^{|h|}, \quad \forall h.$$

If  $\alpha \geq 1$ , then  $a_t^2$  either is nonstationary or has an infinite variance, so it does not have an ACF. This geometric decay in the ACF of  $a_t^2$  for an ARCH(1)

process is analogous to the geometric decay in the ACF of an AR(1) process. To complete the analogy, define  $\eta_t = a_t^2 - \sigma_t^2$ , and note that  $\{\eta_t\}$  is a mean zero weak white noise process, but not an i.i.d. white noise process. Adding  $\eta_t$  to both sides of (14.6) and simplifying we have

$$\sigma_t^2 + \eta_t = a_t^2 = \omega + \alpha a_{t-1}^2 + \eta_t, \quad (14.7)$$

which is a direct representation of  $\{a_t^2\}$  as an AR(1) process.

## 14.4 The AR(1)+ARCH(1) Model

As we have seen, an AR(1) process has a nonconstant conditional mean but a constant conditional variance, while an ARCH(1) process is just the opposite. If both the conditional mean and variance of the data depend on the past, then we can combine the two models. In fact, we can combine any ARMA model with any of the GARCH models in Sect. 14.6. In this section we combine an AR(1) model with an ARCH(1) model.

Let  $a_t$  be an ARCH(1) process so that  $a_t = \epsilon_t \sqrt{\omega + \alpha a_{t-1}^2}$ , where  $\epsilon_t$  is i.i.d.  $N(0, 1)$ , and suppose that

$$y_t - \mu = \phi(y_{t-1} - \mu) + a_t.$$

The process  $y_t$  is an AR(1) process, except that the noise term ( $a_t$ ) is not i.i.d. white noise, but rather an ARCH(1) process which is only weak white noise.

Because  $a_t$  is an uncorrelated process, it has the same ACF as independent white noise, and therefore,  $y_t$  has the same ACF as an AR(1) process with independent white noise

$$\rho_y(h) = \phi^{|h|} \quad \forall h,$$

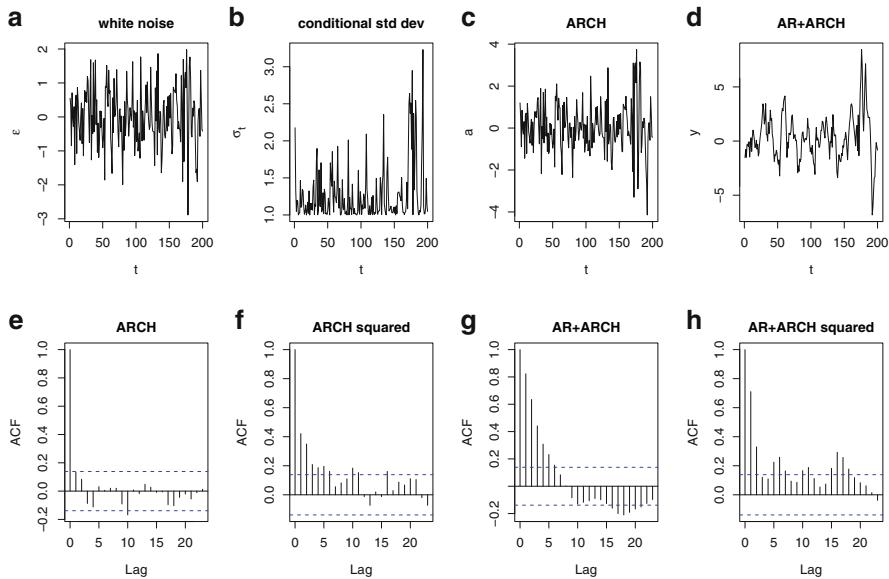
in the stationary case. Moreover,  $a_t^2$  has the ARCH(1) ACF:

$$\rho_{a^2}(h) = \alpha^{|h|} \quad \forall h.$$

The ACF of  $y_t^2$  also decays with  $|h|$  at a geometric rate in the stationary case, provided some additional assumptions hold, however, the exact expressions are more complicated (see Palma and Zevallos, 2004). We need to assume that both  $|\phi| < 1$  and  $\alpha < 1$  in order for  $y_t$  to be stationary with a finite variance. Of course,  $\omega > 0$  and  $\alpha \geq 0$  are also assumed for positiveness of the conditional variance process  $\sigma_t^2$ . The process  $y_t$  is such that its conditional mean and variance, given the past, are both nonconstant, so a wide variety of time series can be modeled.

*Example 14.1.* A simulated ARCH(1) process and AR(1)+ARCH(1) process

A simulated ARCH(1) process is shown in Fig. 14.2. Panel (a) shows the i.i.d. white noise process  $\epsilon_t$ , (b) shows  $\sigma_t = \sqrt{1 + 0.55a_{t-1}^2}$ , the conditional standard deviation process, and (c) shows  $a_t = \sigma_t\epsilon_t$ , the ARCH(1) process. As discussed in the previous section, an ARCH(1) process can be used as the noise term of an AR(1) process. This process is shown in panel (d). The AR(1) parameters are  $\mu = 0.1$  and  $\phi = 0.8$ . The unconditional variance of  $a_t$  is  $\gamma_a(0) = 1/(1 - 0.55) = 2.22$ , so the unconditional standard deviation is  $\sqrt{2.22} = 1.49$ . Panels (e)–(h) are sample ACF plots of the ARCH and AR+ARCH processes and squared processes. Notice that for the ARCH series, the process is uncorrelated but the squared series has autocorrelation. Also notice that for the AR(1)+ARCH(1) series the ACFs of the process and the squared process, panels (g) and (h), both show autocorrelation. While the true ACFs have an exact geometric decay, this is only approximately true for the sample ACFs in panels (f)–(h); similarly, negative values are not present in the true ACFs, but the sample ACF has sampling error and may result in negative values. The processes were all started at 0 and simulated for 10,200 observations. The first 10,000 observations were treated as a burn-in period and discarded.  $\square$



**Fig. 14.2.** Simulation of 200 observations from an ARCH(1) process and an AR(1)+ARCH(1) process. The parameters are  $\omega = 1$ ,  $\alpha = 0.55$ ,  $\mu = 0.1$ , and  $\phi = 0.8$ . Sample ACF plots of the ARCH and AR+ARCH processes and squared processes are shown in the bottom row.

## 14.5 ARCH( $p$ ) Models

As before, let  $\epsilon_t$  be Gaussian white noise with unit variance. Then  $a_t$  is an ARCH( $p$ ) process if

$$a_t = \sigma_t \epsilon_t,$$

where

$$\sigma_t = \sqrt{\omega + \sum_{i=1}^p \alpha_i a_{t-i}^2}$$

is the conditional standard deviation of  $a_t$  given the past values  $a_{t-1}, a_{t-2}, \dots$  of this process. Like an ARCH(1) process, an ARCH( $p$ ) process is uncorrelated and has a constant mean (both conditional and unconditional) and a constant unconditional variance, but its conditional variance is nonconstant. In fact, the ACF of  $a_t^2$  has the same structure as the ACF of an AR( $p$ ) process; see Sect. 14.9.

## 14.6 ARIMA( $p_M, d, q_M$ ) + GARCH( $p_V, q_V$ ) Models

A deficiency of ARCH( $p$ ) models is that the conditional standard deviation process has high-frequency oscillations with high volatility coming in short bursts. This behavior can be seen in Fig. 14.2b. GARCH models permit a wider range of behavior, in particular, more persistent volatility. The GARCH( $p, q$ ) model is

$$a_t = \sigma_t \epsilon_t,$$

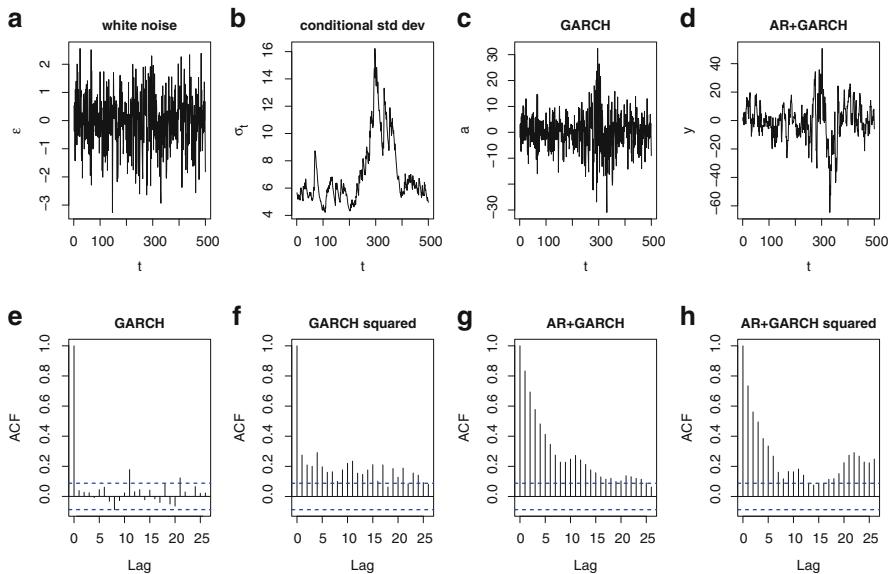
in which

$$\sigma_t = \sqrt{\omega + \sum_{i=1}^p \alpha_i a_{t-i}^2 + \sum_{j=1}^q \beta_j \sigma_{t-j}^2}. \quad (14.8)$$

Because past values of the  $\sigma_t$  process are fed back into the present value (with nonnegative coefficients  $\beta_j$ ), the conditional standard deviation can exhibit more persistent periods of high or low volatility than seen in an ARCH process. In the stationary case, the process  $a_t$  is uncorrelated with a constant unconditional mean and variance and  $a_t^2$  has an ACF like an ARMA process (see Sect. 14.9). GARCH models include ARCH models as a special case, and we use the term “GARCH” to refer to both ARCH and GARCH models.

A very general time series model lets  $a_t$  be GARCH( $p_V, q_V$ ) and uses  $a_t$  as the noise term in an ARIMA( $p_M, d, q_M$ ) model. The subscripts on  $p$  and  $q$  distinguish between the conditional variance (V) or GARCH parameters and the conditional mean (M) or ARIMA parameters. We will call such a process an ARIMA( $p_M, d, q_M$ ) + GARCH( $p_V, q_V$ ) model.

Figure 14.3 is a simulation of 500 observations from a GARCH(1,1) process and from a AR(1)+GARCH(1,1) process. The GARCH parameters are  $\omega = 1$ ,  $\alpha = 0.08$ , and  $\beta = 0.9$ . The large value of  $\beta$  causes  $\sigma_t$  to be highly correlated with  $\sigma_{t-1}$  and gives the conditional standard deviation process a relatively long-term persistence, at least compared to its behavior under an ARCH model. In particular, notice that the conditional standard deviation is less “bursty” than for the ARCH(1) process in Fig. 14.2.



**Fig. 14.3.** Simulation of GARCH(1,1) and AR(1)+GARCH(1,1) processes. The parameters are  $\omega = 1$ ,  $\alpha = 0.08$ ,  $\beta = 0.9$ , and  $\phi = 0.8$ .

#### 14.6.1 Residuals for ARIMA( $p_M, d, q_M$ )+GARCH( $p_V, q_V$ ) Models

When one fits an ARIMA( $p_M, d, q_M$ )+GARCH( $p_V, q_V$ ) model to a time series  $Y_t$ , there are two types of residuals. The ordinary residual, denoted  $\hat{a}_t$ , is the difference between  $Y_t$  and its conditional expectation. As the notation implies,  $\hat{a}_t$  estimates  $a_t$ . A standardized residual, denoted  $\hat{\epsilon}_t$ , is an ordinary residual  $\hat{a}_t$  divided by its estimated conditional standard deviation  $\hat{\sigma}_t$ . A standardized residual estimates  $\epsilon_t$ . The standardized residuals should be used for model checking. If the model fits well, then neither  $\hat{\epsilon}_t$  nor  $\hat{\epsilon}_t^2$  should exhibit serial correlation. Moreover, if  $\epsilon_t$  has been assumed to have a normal distribution, then this assumption can be checked by a normal plot of the standardized residuals  $\hat{\epsilon}_t$ . The  $\hat{a}_t$  are the residuals of the ARIMA process and are used when forecasting via the methods in Sect. 12.12.

## 14.7 GARCH Processes Have Heavy Tails

Researchers have long noticed that stock returns have “heavy-tailed” or “outlier-prone” probability distributions, and we have seen this ourselves in earlier chapters. One reason for outliers may be that the conditional variance is not constant, and the outliers occur when the variance is large, as in the normal mixture example of Sect. 5.5. In fact, GARCH processes exhibit heavy tails even if  $\{\epsilon_t\}$  is Gaussian. Therefore, when we use GARCH models, we can model both the conditional heteroskedasticity and the heavy-tailed distributions of financial market data. Nonetheless, many financial time series have tails that are heavier than implied by a GARCH process with Gaussian  $\{\epsilon_t\}$ . To handle such data, one can assume that, instead of being Gaussian white noise,  $\{\epsilon_t\}$  is an i.i.d. white noise process with a heavy-tailed distribution.

## 14.8 Fitting ARMA+GARCH Models

*Example 14.2. AR(1)+GARCH(1,1) model fit to daily BMW stock log returns*

This example uses the daily BMW stock log returns. The `ugarchfit()` function from R’s `rugarch` package is used to fit an AR(1)+GARCH(1,1) model to this series. Although `ugarchfit()` allows the white noise to have a nonGaussian distribution, we begin this example using Gaussian white noise (the default). First the model is specified using the `ugarchspec()` function; for an AR(1)+GARCH(1,1) model we specify `armaOrder=c(1,0)` and `garchOrder=c(1,1)`. The commands and abbreviated output are below.

```

1 library(rugarch)
2 data(bmw, package="evir")
3 arma.garch.norm = ugarchspec(mean.model=list(armaOrder=c(1,0)),
4                               variance.model=list(garchOrder=c(1,1)))
5 bmw.garch.norm = ugarchfit(data=bmw, spec=arma.garch.norm)
6 show(bmw.garch.norm)

GARCH Model : sGARCH(1,1)
Mean Model : ARFIMA(1,0,0)
Distribution : norm

Optimal Parameters
-----
      Estimate Std. Error t value Pr(>|t|)
mu     0.000453   0.000175  2.5938 0.009493
ar1     0.098135   0.014261  6.8813 0.000000
omega   0.000009   0.000000 23.0613 0.000000
alpha1   0.099399   0.005593 17.7730 0.000000
beta1    0.863672   0.006283 137.4591 0.000000

```

```
LogLikelihood : 17752
```

#### Information Criteria

Akaike	-5.7751
Bayes	-5.7696
Shibata	-5.7751
Hannan-Quinn	-5.7732

In the output,  $\hat{\phi}_1$  is denoted by `ar1`, the estimated mean  $\hat{\mu}$  is `mean`, and  $\hat{\omega}$  is called `omega`. Note that  $\hat{\phi}_1 = 0.0981$  and is statistically significant, implying that there is a small amount of positive autocorrelation. Both  $\alpha_1$  and  $\beta_1$  are highly significant and  $\hat{\beta}_1 = 0.8636$ , which implies rather persistent volatility clustering. There are two additional information criteria reported, Shibata's information criterion and Hannan–Quinn information criterion (HQIC). These are less widely used than AIC and BIC and will not be discussed here.

In the output from `ugarchfit()`, the AIC and BIC values have been normalized by dividing by  $n$ , so these values should be multiplied by  $n = 6146$  to have their usual values. In particular, AIC and BIC will not be so close to each other after multiplication by 6146. The daily BMW stock log return series  $Y_t$ , with two estimated conditional standard deviations superimposed, and the estimated conditional standard deviation series  $\hat{\sigma}_t$  (vs. the absolute value of the log return series  $|Y_t|$ ) are shown in the top row of Fig. 14.4.

The output also includes the following tests applied to the standardized and squared standardized residuals.

#### Weighted Ljung-Box Test on Standardized Residuals

	statistic p-value	
Lag[1]	0.7786	0.3776
Lag[2*(p+q)+(p+q)-1] [2]	0.9158	0.7892
Lag[4*(p+q)+(p+q)-1] [5]	3.3270	0.3536
d.o.f=1		

H0 : No serial correlation

#### Weighted Ljung-Box Test on Standardized Squared Residuals

	statistic p-value	
Lag[1]	0.277	0.5987
Lag[2*(p+q)+(p+q)-1] [5]	1.026	0.8537
Lag[4*(p+q)+(p+q)-1] [9]	1.721	0.9356
d.o.f=2		

#### Weighted ARCH LM Tests

	Statistic	Shape	Scale	P-Value
ARCH Lag[3]	0.1922	0.500	2.000	0.6611
ARCH Lag[5]	1.1094	1.440	1.667	0.7008
ARCH Lag[7]	1.2290	2.315	1.543	0.8737

Adjusted Pearson Goodness-of-Fit Test:

```
-----  
group statistic p-value(g-1)  
1    20     493.1   1.563e-92  
2    30     513.4   5.068e-90  
3    40     559.3   2.545e-93  
4    50     585.6   5.446e-93
```

Weighted versions of the Ljung-Box (and ARCH-LM) test statistics<sup>1</sup> and their approximate  $p$ -values all indicate that the estimated model for the conditional mean and variance are adequate for removing serial correlation from the series and squared series, respectively. The sample ACF of the standardized residuals  $\hat{\epsilon}_t$ , and the squared standardized residuals  $\hat{\epsilon}_t^2$  are shown in the middle row of Fig. 14.4. The Goodness-of-Fit tests<sup>2</sup> compare the empirical distribution of the standardized residuals with the theoretical ones from the specified density, which is Gaussian by default. The small  $p$ -values strongly reject the null hypothesis that the white noise standardized innovation process  $\{\epsilon_t\}$  is Gaussian. Empirical density estimates and a normal quantile plot of the standardized residuals  $\hat{\epsilon}_t$  are shown in the bottom row of Fig. 14.4.

Figure 14.5 shows a  $t$ -plot with 4 df for the standardized residuals  $\hat{\epsilon}_t$ . Unlike the normal quantile plot in the last panel of Fig. 14.4, this plot is nearly a straight line except for four outliers in the left tail. The sample size is 6146, so the outliers are a very small fraction of the data. Thus, it seems like a  $t$ -distribution would be suitable for the innovation process  $\epsilon_t$ . A  $t$ -distribution was fit to the standardized residuals by maximum likelihood using the `fitdistr()` function from the R package MASS.

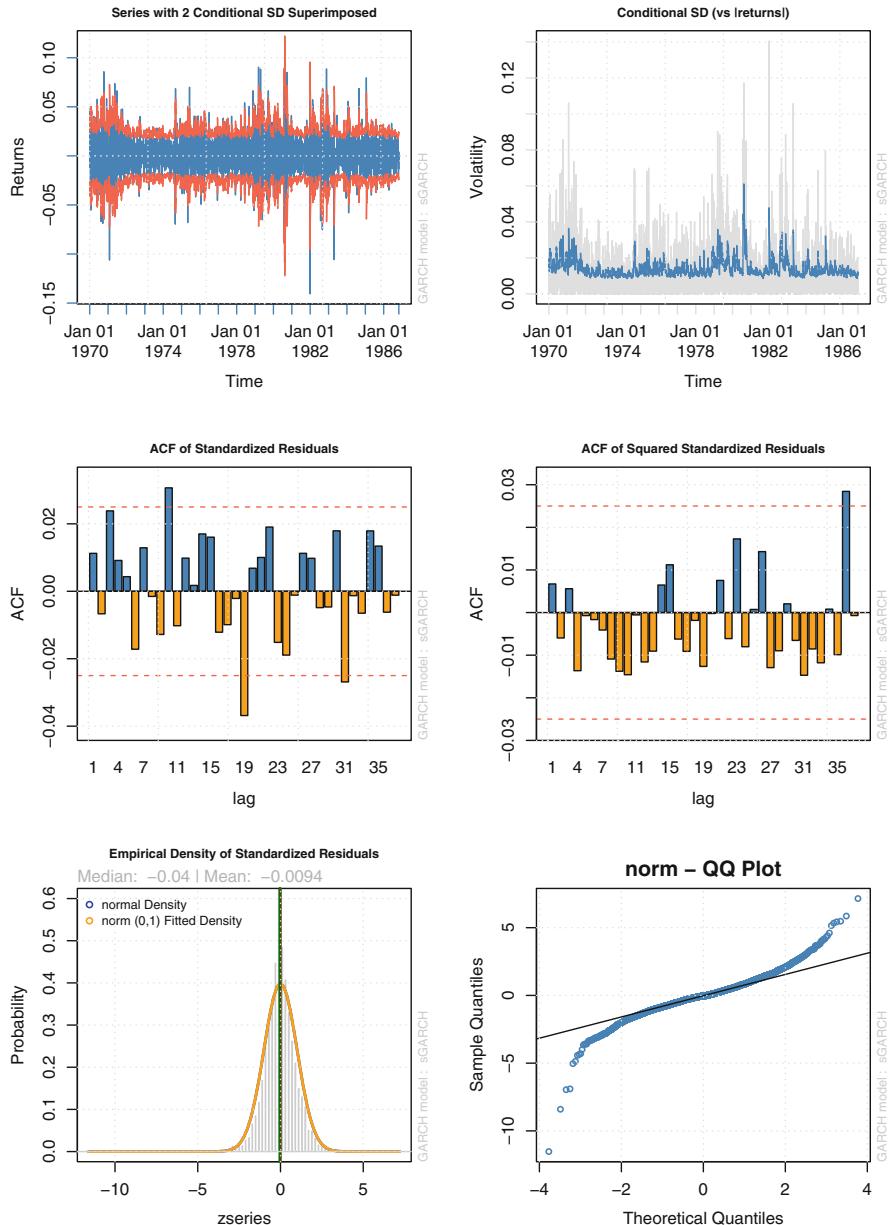
```
7 library(MASS)  
8 e = residuals(bmw.garch.norm, standardize=TRUE)  
9 fitdistr(e,"t")  
  
m           s           df  
-0.0243    0.7269    4.1096  
( 0.0109) ( 0.0121) ( 0.2359)
```

The MLE of the degrees-of-freedom parameter was 4.1. This confirms the good fit by this distribution seen in Fig. 14.5. The AR(1)+GARCH(1,1) model was refit assuming  $t$ -distributed errors, so `distribution.model = "std"` in `ugarchspec()`. The commands and abbreviated results are below.

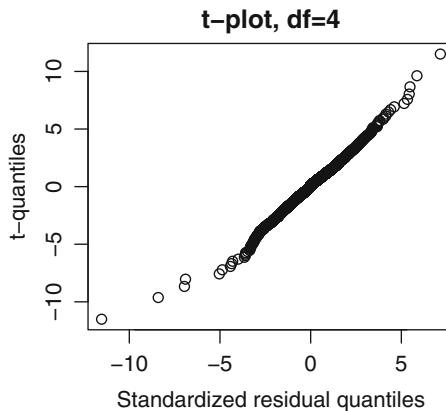
```
10 arma.garch.t = ugarchspec(mean.model=list(armaOrder=c(1,0)),  
11                           variance.model=list(garchOrder=c(1,1)),  
12                           distribution.model = "std")  
13 bmw.garch.t = ugarchfit(data=bmw, spec=arma.garch.t)  
14 show(bmw.garch.t)
```

<sup>1</sup> Weighted Ljung-Box and ARCH-LM statistics of Fisher and Gallagher (2012) are provided by the `ugarchfit()` function to better account for the distribution of the statistics when applied to residuals from a fitted model; their use and interpretation remains unchanged.

<sup>2</sup> These Chi-squared tests are based on the tests of Palm (1996); `group` indicates the number of bins used in the implementation.



**Fig. 14.4.** The daily BMW stock log return series  $Y_t$ , with two estimated conditional standard deviations superimposed; the estimated conditional standard deviation  $\hat{\sigma}_t$  series (vs. the absolute value of the log return series  $|Y_t|$ ); the sample ACF of the standardized residuals  $\hat{\epsilon}_t$  and the squared standardized residuals  $\hat{\epsilon}_t^2$ ; empirical density estimates of the standardized residuals  $\hat{\epsilon}_t$ ; and a normal quantile plot of the standardized residuals  $\hat{\epsilon}_t$ .



**Fig. 14.5.** A *t*-plot with 4 df for the standardized residuals  $\hat{\epsilon}_t$  from an AR(1)+GARCH(1,1) model fit to daily BMW stock log return; the reference lines go through the first and third quartiles.

```
GARCH Model : sGARCH(1,1)
Mean Model : ARFIMA(1,0,0)
Distribution : std
```

#### Optimal Parameters

	Estimate	Std. Error	t value	Pr(> t )
mu	0.000135	0.000144	0.93978	0.347333
ar1	0.063911	0.012521	5.10436	0.000000
omega	0.000006	0.000003	1.69915	0.089291
alpha1	0.090592	0.012479	7.25936	0.000000
beta1	0.889887	0.014636	60.80228	0.000000
shape	4.070078	0.301306	13.50813	0.000000

```
LogLikelihood : 18152
```

#### Information Criteria

Akaike	-5.9048
Bayes	-5.8983
Shibata	-5.9048
Hannan-Quinn	-5.9026

#### Weighted Ljung-Box Test on Standardized Residuals

	statistic	p-value
Lag[1]	9.640	1.904e-03
Lag[2*(p+q)+(p+q)-1] [2]	9.653	3.367e-09
Lag[4*(p+q)+(p+q)-1] [5]	11.983	1.455e-04

```

d.o.f=1
H0 : No serial correlation

Weighted Ljung-Box Test on Standardized Squared Residuals
-----
              statistic p-value
Lag[1]          0.5641  0.4526
Lag[2*(p+q)+(p+q)-1] [5]  1.2964  0.7898
Lag[4*(p+q)+(p+q)-1] [9]  2.0148  0.9032
d.o.f=2

Adjusted Pearson Goodness-of-Fit Test:
-----
      group statistic p-value(g-1)
1     20      229.0   5.460e-38
2     30      279.6   8.428e-43
3     40      313.8   1.230e-44
4     50      374.6   1.037e-51

```

The weighted Ljung–Box tests for the residuals have small  $p$ -values. These are due to small autocorrelations that should not be of practical importance. The sample size here is 6146 so, not surprisingly, small autocorrelations are statistically significant. The goodness-of-fit test statistics are much smaller but still significant; the large sample size again makes rejection likely even when the discrepancies are negligible from a practical standpoint. However, both AIC and BIC decreased substantially, and the refit model with a  $t$  conditional distribution offers an improvement over the original fit with a Gaussian conditional distribution.  $\square$

## 14.9 GARCH Models as ARMA Models

The similarities seen in this chapter between GARCH and ARMA models are not a coincidence. If  $a_t$  is a GARCH process, then  $a_t^2$  is an ARMA process, but with weak white noise, not i.i.d. white noise. To show this, we will start with the GARCH(1,1) model, where  $a_t = \sigma_t \epsilon_t$ . Here  $\epsilon_t$  is i.i.d. white noise and

$$E(a_t^2 | \mathcal{F}_{t-1}) = \sigma_t^2 = \omega + \alpha a_{t-1}^2 + \beta \sigma_{t-1}^2, \quad (14.9)$$

where  $\mathcal{F}_{t-1}$  is the information set at time  $t - 1$ . Define  $\eta_t = a_t^2 - \sigma_t^2$ . Since  $E(\eta_t | \mathcal{F}_{t-1}) = E(a_t^2 | \mathcal{F}_{t-1}) - \sigma_t^2 = 0$  by (A.33),  $\eta_t$  is an uncorrelated process, that is, a weak white noise process. The conditional heteroskedasticity of  $a_t$  is inherited by  $\eta_t$ , so  $\eta_t$  is not i.i.d. white noise.

Simple algebra shows that

$$\sigma_t^2 = \omega + (\alpha + \beta)a_{t-1}^2 - \beta\eta_{t-1} \quad (14.10)$$

and therefore

$$a_t^2 = \sigma_t^2 + \eta_t = \omega + (\alpha + \beta)a_{t-1}^2 - \beta\eta_{t-1} + \eta_t. \quad (14.11)$$

Assume that  $\alpha + \beta < 1$ . If  $v = \omega/\{1 - (\alpha + \beta)\}$ , then

$$a_t^2 - v = (\alpha + \beta)(a_{t-1}^2 - v) + \beta\eta_{t-1} + \eta_t. \quad (14.12)$$

From (14.12) one sees that  $a_t^2$  is an ARMA(1,1). Using the notation of (12.25), the mean is  $\mu = v$ , the AR(1) coefficient is  $\phi = \alpha + \beta$  and the MA(1) coefficient is  $\theta = -\beta$ .

For the general case, assume that  $\sigma_t$  follows (14.8) such that

$$\sigma_t^2 = \omega + \sum_{i=1}^p \alpha_i a_{t-i}^2 + \sum_{j=1}^q \beta_j \sigma_{t-j}^2. \quad (14.13)$$

To simplify notation, if  $q > p$ , then define  $\alpha_i = 0$  for  $i = p+1, \dots, q$ . Similarly, if  $p > q$ , then define  $\beta_j = 0$  for  $j = q+1, \dots, p$ . Define  $v = \omega/\{1 - \sum_{i=1}^{\max(p,q)} (\alpha_i + \beta_i)\}$ . Straightforward algebra similar to the GARCH(1,1) case shows that

$$a_t^2 - v = \sum_{i=1}^{\max(p,q)} (\alpha_i + \beta_i)(a_{t-i}^2 - v) - \sum_{j=1}^q \beta_j \eta_{t-j} + \eta_t, \quad (14.14)$$

so that  $a_t^2$  is an ARMA( $\max(p, q)$ ,  $q$ ) process with mean  $\mu = v$ , AR coefficients  $\phi_i = \alpha_i + \beta_i$  and MA coefficients  $\theta_j = -\beta_j$ . As a byproduct of these calculations, we obtain a necessary condition for  $a_t$  to be stationary:

$$\sum_{i=1}^{\max(p,q)} (\alpha_i + \beta_i) < 1. \quad (14.15)$$

## 14.10 GARCH(1,1) Processes

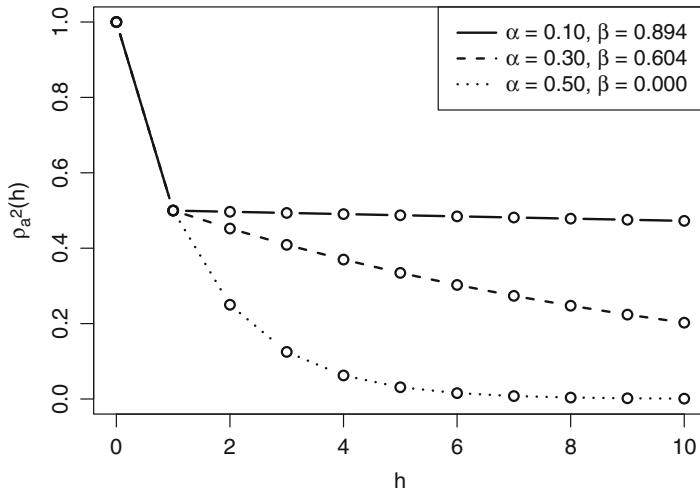
The GARCH(1,1) is the most widely used GARCH process, so it is worthwhile to study it in some detail. If  $a_t$  is GARCH(1,1), then as we have just seen,  $a_t^2$  is ARMA(1,1). Therefore, the ACF of  $a_t^2$  can be obtained from formulas (12.31) and (12.32). After some algebra, one finds that

$$\rho_{a^2}(1) = \frac{\alpha(1 - \alpha\beta - \beta^2)}{1 - 2\alpha\beta - \beta^2} \quad (14.16)$$

and

$$\rho_{a^2}(h) = (\alpha + \beta)^{h-1} \rho_{a^2}(1), \quad h \geq 2. \quad (14.17)$$

These formulas also hold in an AR(1)+GARCH(1,1) model, and the ACF of  $y_t^2$  also decays with  $h \geq 2$  at a geometric rate in the stationary case, provided some additional assumptions hold, however, the exact expressions are more complicated (see Palma and Zevallos, 2004).



**Fig. 14.6.** ACFs of three GARCH(1,1) processes with  $\rho_{a^2}(1) = 0.5$ .

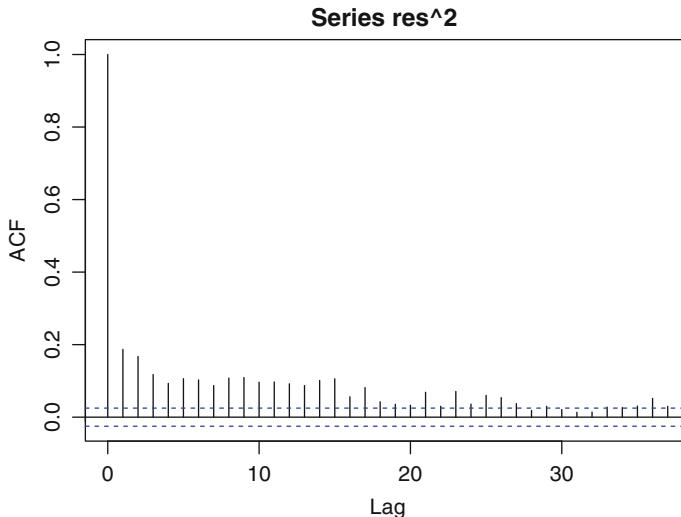
By (14.16), there are infinitely many values of  $(\alpha, \beta)$  with the same value of  $\rho_{a^2}(1)$ . By (14.17), a higher value of  $\alpha + \beta$  means a slower decay of  $\rho_{a^2}(\cdot)$  after the first lag. This behavior is illustrated in Fig. 14.6, which contains the ACF of  $a_t^2$  for three GARCH(1,1) processes with a lag-1 autocorrelation of 0.5. The solid curve has the highest value of  $\alpha + \beta$  and the ACF decays very slowly. The dotted curve is a pure ARCH(1) process and has the most rapid decay.

In Example 14.2, an AR(1)+GARCH(1,1) model was fit to the BMW daily log returns. The GARCH parameters were estimated to be  $\hat{\alpha} = 0.10$  and  $\hat{\beta} = 0.86$ . By (14.16) the  $\hat{\rho}_{a^2}(1) = 0.197$  for this process and the high value of  $\hat{\beta}$  suggests slow decay. The sample ACF of the squared residuals [from an AR(1) model] is plotted in Fig. 14.7. In that figure, we see the lag-1 autocorrelation is slightly below 0.2 and after one lag the ACF decays slowly, exactly as expected.

The capability of the GARCH(1,1) model to fit the lag-1 autocorrelation and the subsequent rate of decay separately is important in practice. It appears to be the main reason that the GARCH(1,1) model fits so many financial time series.

## 14.11 APARCH Models

In some financial time series, large negative returns appear to increase volatility more than do positive returns of the same magnitude. This is called the



**Fig. 14.7.** ACF of the squared residuals from an AR(1) fit to the BMW log returns.

*leverage effect.* Standard GARCH models, that is, the models given by (14.8), cannot model the leverage effect because they model  $\sigma_t$  as a function of past values of  $a_t^2$ —whether the past values of  $a_t$  are positive or negative is not taken into account. The problem here is that the square function  $x^2$  is symmetric in  $x$ . The solution is to replace the square function with a flexible class of nonnegative functions that include asymmetric functions. The APARCH (asymmetric power ARCH) models do this. They also offer more flexibility than GARCH models by modeling  $\sigma_t^\delta$ , where  $\delta > 0$  is another parameter.

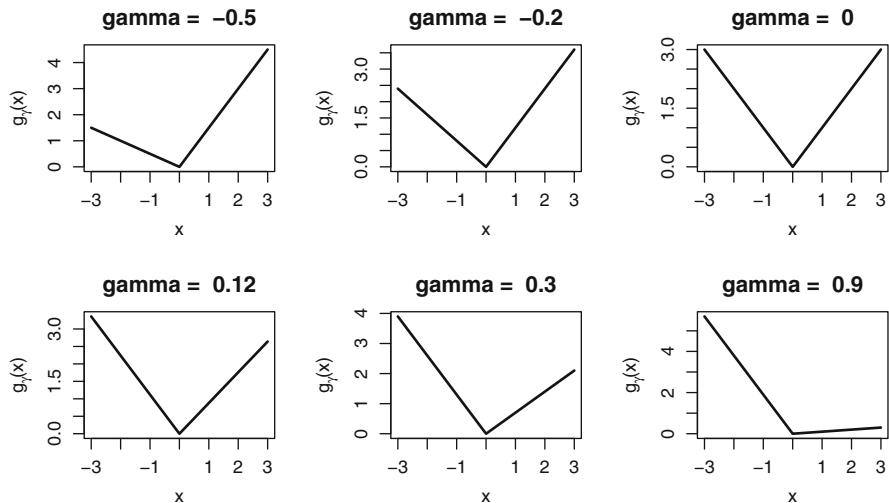
The APARCH( $p, q$ ) model for the conditional standard deviation is

$$\sigma_t^\delta = \omega + \sum_{i=1}^p \alpha_i (|a_{t-i}| - \gamma_i a_{t-i})^\delta + \sum_{j=1}^q \beta_j \sigma_{t-j}^\delta, \quad (14.18)$$

where  $\delta > 0$  and  $-1 < \gamma_i < 1$ ,  $i = 1, \dots, p$ . Note that  $\delta = 2$  and  $\gamma = \dots = \gamma_p = 0$  give a standard GARCH model.

The effect of  $a_{t-i}$  upon  $\sigma_t$  is through the function  $g_{\gamma_i}$ , where  $g_\gamma(x) = |x| - \gamma x$ . Figure 14.8 shows  $g_\gamma(x)$  for several values of  $\gamma$ . When  $\gamma > 0$ ,  $g_\gamma(-x) > g_\gamma(x)$  for any  $x > 0$ , so there is a leverage effect. If  $\gamma < 0$ , then there is a leverage effect in the opposite direction to what is expected—positive past values of  $a_t$  increase volatility more than negative past values of the same magnitude.

*Example 14.3.* AR(1)+APARCH(1,1) fit to daily BMW stock log returns



**Fig. 14.8.** Plots of  $g_\gamma(x)$  for various values of  $\gamma$ .

In this example, an AR(1)+APARCH(1,1) model with  $t$ -distributed errors is fit to the BMW log returns. The commands and abbreviated output from `ugarchfit()` is below. The estimate of  $\delta$  is 1.48 with a standard error of 0.14, so there is strong evidence that  $\delta$  is not 2, the value under a standard GARCH model. Also,  $\hat{\gamma}_1$  is 0.12 with a standard error of 0.045, so there is a statistically significant leverage effect, since we reject the null hypothesis that  $\gamma_1 = 0$ . However, the leverage effect is small, as can be seen in the plot in Fig. 14.8 with  $\gamma = 0.12$ . The leverage might not be of practical importance.

```

15 arma.aparch.t = ugarchspec(mean.model=list(armaOrder=c(1,0)),
16                             variance.model=list(model="apARCH",
17                               garchOrder=c(1,1)),
18                             distribution.model = "std")
19 bmw.aparch.t = ugarchfit(data=bmw, spec=arma.aparch.t)
20 show(bmw.aparch.t)

GARCH Model : apARCH(1,1)
Mean Model : ARFIMA(1,0,0)
Distribution : std
Optimal Parameters
-----
          Estimate Std. Error t value Pr(>|t|)
mu      0.000048   0.000147   0.3255 0.744801
ar1     0.063666   0.012352   5.1543 0.000000

```

```

omega  0.000050  0.000032  1.5541 0.120158
alpha1  0.098839  0.012741  7.7574 0.000000
beta1   0.899506  0.013565  66.3105 0.000000
gamma1  0.121947  0.044664  2.7303 0.006327
delta   1.476643  0.142442  10.3666 0.000000
shape   4.073809  0.234417  17.3784 0.000000

```

LogLikelihood : 18161

#### Information Criteria

```

-----  

Akaike      -5.9073  

Bayes       -5.8985  

Shibata     -5.9073  

Hannan-Quinn -5.9042

```

#### Weighted Ljung-Box Test on Standardized Residuals

```

-----  

statistic  p-value  

Lag[1]        9.824 1.723e-03  

Lag[2*(p+q)+(p+q)-1] [2]    9.849 2.003e-09  

Lag[4*(p+q)+(p+q)-1] [5]    12.253 1.100e-04  

d.o.f=1  

H0 : No serial correlation

```

#### Weighted Ljung-Box Test on Standardized Squared Residuals

```

-----  

statistic p-value  

Lag[1]        1.456 0.2276  

Lag[2*(p+q)+(p+q)-1] [5]    2.363 0.5354  

Lag[4*(p+q)+(p+q)-1] [9]    3.258 0.7157  

d.o.f=2

```

As mentioned earlier, in the output from `ugarchfit()`, the **Information Criteria** values have been normalized by dividing by  $n$ , though this is not noted in the output.

The normalized BIC for this model ( $-5.8985$ ) is very nearly the same as the normalized BIC for the GARCH model with  $t$ -distributed errors ( $-5.8983$ ), but after multiplying by  $n = 6146$ , the difference in the BIC values is 1.23. The difference between the two normalized AIC values,  $-5.9073$  and  $-5.9048$ , is even larger, 15.4, after multiplication by  $n$ . Therefore, AIC and BIC support using the APARCH model instead of the GARCH model.

ACF plots (not shown) for the standardized residuals and their squares showed little correlation, so the AR(1) model for the conditional mean and the APARCH(1,1) model for the conditional variance fit well. Finally, `shape` is the estimated degrees of freedom of the  $t$ -distribution and is 4.07 with a small standard error, so there is very strong evidence that the conditional distribution is heavy-tailed.  $\square$

## 14.12 Linear Regression with ARMA+GARCH Errors

When using time series regression, one often observes autocorrelated residuals. For this reason, linear regression with ARMA disturbances was introduced in Sect. 13.3.3. The model considered was

$$Y_t = \beta_0 + \beta_1 X_{t,1} + \cdots + \beta_p X_{t,p} + e_t, \quad (14.19)$$

where

$$(1 - \phi_1 B - \cdots - \phi_p B^p)(e_t - \mu) = (1 + \theta_1 B + \cdots + \theta_q B^q)a_t, \quad (14.20)$$

and  $\{a_t\}$  is i.i.d. white noise. This model is sufficient for serially correlated errors, but it does not accommodate volatility clustering, which is often found in the residuals.

One solution is to model the noise as an ARMA+GARCH process. Therefore, we will now assume that, instead of being i.i.d. white noise,  $\{a_t\}$  is a GARCH process so that

$$a_t = \sigma_t \epsilon_t, \quad (14.21)$$

where

$$\sigma_t = \sqrt{\omega + \sum_{i=1}^p \alpha_i a_{t-i}^2 + \sum_{j=1}^q \beta_j \sigma_{t-j}^2}, \quad (14.22)$$

and  $\{\epsilon_t\}$  is i.i.d. white noise. The model given by (14.19)–(14.22) is a *linear regression model with ARMA+GARCH disturbances*.

Some software, including the `ugarchfit()` function from R's `rugarch` package, can fit the linear regression model with ARMA+GARCH disturbances in one step. Another solution is to adjust or correct the estimated covariance matrix of the regression coefficients, via the HAC estimator from Sect. 13.3.2, by using the `NeweyWest()` function from the R package `sandwich`. However, if such software is not available, then a three-step estimation method is the following:

1. estimate the parameters in (14.19) by ordinary least-squares;
2. fit model (14.20)–(14.22) to the ordinary least-squares residuals;
3. reestimate the parameters in (14.19) by weighted least-squares with weights equal to the reciprocals of the conditional variances from step 2.

*Example 14.4. Regression analysis with ARMA+GARCH errors of the Nelson-Plosser data*

In Example 9.9, we saw that a parsimonious model for the yearly log returns on the stock index `diff(log(sp))` used `diff(log(ip))` and `diff(bnd)` as predictors. Figure 14.9 contains ACF plots of the residuals [panel (a)] and

squared residuals [panel (b)]. Externally studentized residuals were used, but the plots for the raw residuals are similar. There is some autocorrelation in both the residuals and squared residuals.

```

21 nelsonplosser = read.csv("nelsonplosser.csv", header = TRUE)
22 new_np = na.omit(nelsonplosser)
23 attach(new_np)
24 fit.lm1 = lm(diff(log(sp)) ~ diff(log(ip)) + diff(bnd))
25 summary(fit.lm1)

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)    0.01657   0.02100   0.789  0.433316
diff(log(ip))  0.69748   0.16834   4.143  0.000113 ***
diff(bnd)     -0.13224   0.06225  -2.124  0.037920 *
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 0.1509 on 58 degrees of freedom
Multiple R-squared:  0.3087 , Adjusted R-squared:  0.2848
F-statistic: 12.95 on 2 and 58 DF,  p-value: 2.244e-05

```

The `auto.arima()` function from R's `forecast` package selected an MA(1) model [i.e., ARIMA(0,0,1)] for the residuals. Next an MA(1)+ARCH(1) model was fit to the regression model's raw residuals. Sample ACF plots of the standardized residuals from the MA(1)+ARCH(1) model are in Fig. 14.9c and d. One sees essentially no short-term autocorrelation in the ARMA+GARCH standardized or squared standardized residuals, which indicates that the ARMA+GARCH model accounts for the observed dependence in the regression residuals satisfactorily. A normal plot showed that the standardized residuals are close to normally distributed, which is not unexpected for yearly log returns.

Finally, the linear model was refit with the reciprocals of the conditional variances as weights. The estimated regression coefficients are given below along with their standard errors and *p*-values.

```

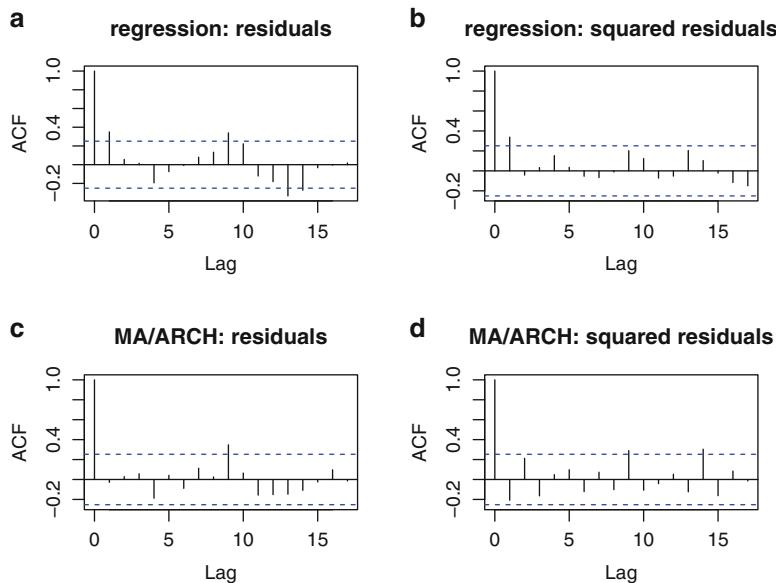
26 fit.lm3 = lm(diff(log(sp)) ~ diff(log(ip)) + diff(bnd),
27                 weights = 1/sigma.arch^2)
28 summary(fit.lm3)

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)    0.03216   0.02052   1.567  0.12263
diff(log(ip))  0.55464   0.16942   3.274  0.00181 **
diff(bnd)     -0.12215   0.05827  -2.096  0.04051 *
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 1.071 on 57 degrees of freedom
Multiple R-squared:  0.2416 , Adjusted R-squared:  0.2149
F-statistic: 9.077 on 2 and 57 DF,  p-value: 0.0003783

```

There are no striking differences between these results and the unweighted fit in Example 9.9. In this situation, the main reason for using the GARCH



**Fig. 14.9.** (a) Sample ACF of the externally studentized residuals and (b) their squared values, from a linear model; (c) Sample ACF of the standardized residuals and (d) their squared values, from an MA(1)+ARCH(1) fit to the regression residuals.

model for the residuals would be in providing more accurate prediction intervals if the model were to be used for forecasting; see Sect. 14.13.  $\square$

### 14.13 Forecasting ARMA+GARCH Processes

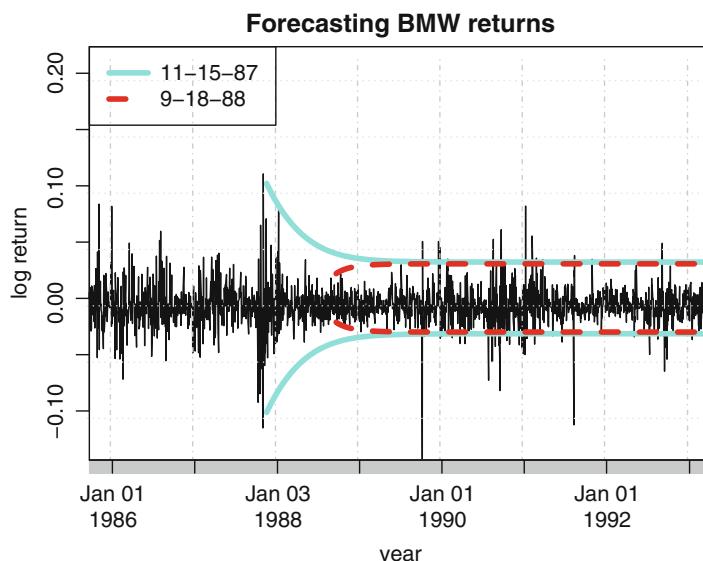
Forecasting ARMA+GARCH processes is in one way similar to forecasting ARMA processes—point estimates, e.g., forecasts of the conditional mean, are the same because a GARCH process is weak white noise. What differs between forecasting ARMA+GARCH and ARMA processes is the behavior of the prediction intervals. In times of high volatility, prediction intervals using an ARMA+GARCH model will widen to take into account the higher amount of uncertainty. Similarly, the prediction intervals will narrow in times of lower volatility. Prediction intervals using an ARMA model without conditional heteroskedasticity cannot adapt in this way.

To illustrate, we will compare the prediction of a Gaussian white noise process and the prediction of a GARCH(1,1) process with Gaussian innovations.

Both have an ARMA(0,0) model for the conditional mean so their forecasts are equal to the marginal mean, which will be called  $\mu$ . For Gaussian white noise, the prediction limits are  $\mu \pm z_{\alpha/2}\sigma$ , where  $\sigma$  is the marginal standard deviation. For a GARCH(1,1) process  $\{Y_t\}$ , the prediction limits at time origin  $n$  for  $h$ -steps ahead forecasting are  $\mu \pm z_{\alpha/2}\sigma_{n+h|n}$  where  $\sigma_{n+h|n}$  is the conditional standard deviation of  $Y_{n+h}$  given the information available at time  $n$ . As  $h$  increases,  $\sigma_{n+h|n}$  converges to  $\sigma$ , so for long lead times the prediction intervals for the two models are similar. For shorter lead times, however, the prediction limits can be quite different.

*Example 14.5. Forecasting BMW log returns*

In this example, we will return to the daily BMW stock log returns used in several earlier examples. We have seen in Example 14.2 that an AR(1)+GARCH(1,1) model fits the returns well. Also, the estimated AR(1) coefficient is small, less than 0.1. Therefore, it is reasonable to use a GARCH (1,1) model for forecasting.



**Fig. 14.10.** Prediction limits for forecasting daily BMW stock log returns from two different time origins.

Figure 14.10 plots the returns from 1986 until 1992. Forecast limits are also shown for two time origins, November 15, 1987 and September 18, 1988. At the first time origin, which is soon after Black Monday, the markets were very volatile. The forecast limits are wide initially but narrow as the conditional standard deviation converges downward to the marginal standard deviation. At the second time origin, the markets were less volatile than usual and the

prediction intervals are narrow initially but then widen. In theory, both sets of prediction limits should converge to the same values,  $\mu \pm z_{\alpha/2}\sigma$  where  $\sigma$  is the marginal standard deviation for a stationary process. In this example, they do not quite converge to each other because the estimates of  $\sigma$  differ between the two time origins.  $\square$

## 14.14 Multivariate GARCH Processes

Financial asset returns tend to move together over time, as do their respective volatilities, across both assets and markets. Modeling a time-varying conditional covariance matrix, or volatility matrix, is important in many financial applications, including asset pricing, hedging, portfolio selection, and risk management.

Multivariate volatility modeling has major challenges to overcome. First, the curse of dimensionality; there are  $d(d+1)/2$  variances and covariances for a  $d$ -dimensional process, e.g., 45 for  $d = 9$ , all of which may vary over time. Further, unlike returns, all of these variances and covariances are unobserved, or latent. Many parameterizations for the evolution of the volatility matrix use such a large number of parameters that estimation becomes infeasible for  $d > 10$ . In addition to empirical adequacy (i.e., goodness of fit of the model to the data), ease and feasibility of estimation are important considerations.

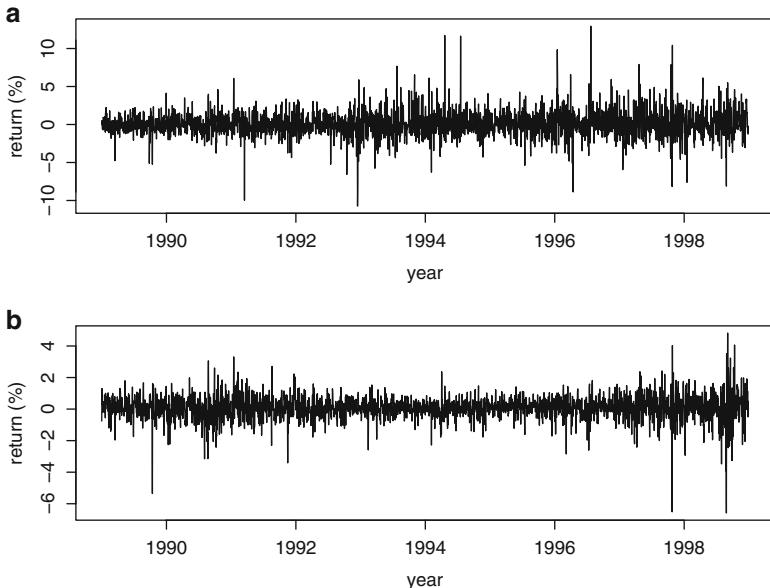
Analogous to positivity constraints in univariate GARCH models, a well-defined multivariate volatility matrix process must be positive-definite at each time point, and model-based forecasts should as well. From a practical perspective, a well-defined inverse of a volatility matrix is frequently needed in applications. Additionally, a positive conditional variance estimate for a portfolio's return, which are a linear combination of asset returns, is essential; fortunately, this is guaranteed by positive definiteness.

### 14.14.1 Multivariate Conditional Heteroscedasticity

Figures 14.11a and b are time series plots of daily returns (in percentage) for IBM stock and the Center for Research in Security Prices (CRSP) value-weighted index, including dividends, from January 3, 1989 to December 31, 1998, respectively. The data are from the `Ecdat` package in R. Each series clearly exhibits volatility clustering. Let  $\mathbf{Y}_t$  denote the vector time series of these returns.

```

29 data(CRSPday, package="Ecdat")
30 CRSPday = ts(CRSPday, start = c(1989, 1), frequency = 253)
31 ibm = CRSPday[,5] * 100
32 crsp = CRSPday[,7] * 100
33 Y = cbind(ibm, crsp)
34 par(mfrow = c(2,1))
35 plot(Y[,1], type='l', xlab="year", ylab="return (%)", main="(a)")
36 plot(Y[,2], type='l', xlab="year", ylab="return (%)", main="(b)")
```



**Fig. 14.11.** Daily returns (in percentage) for (a) IBM stock and (b) the CRSP value-weighted index, including dividends.

Figures 14.12a and b are the sample ACF plots for the IBM stock and CRSP index returns, respectively. There is some evidence of minor serial correlation at low lags. Next, we consider the lead-lag linear relationship between pairs of returns. Figure 14.12c is the sample cross-correlation function (CCF) between IBM and CRSP. The lag zero estimate for contemporaneous correlation is approximately 0.49. There is also some evidence of minor cross-correlation at low lags.

```

37 layout(rbind(c(1,2), c(3,3)), widths=c(1,1,2), heights=c(1,1))
38 acf(as.numeric(Y[,1]), ylim=c(-0.1,0.1), main="(a)")
39 acf(as.numeric(Y[,2]), ylim=c(-0.1,0.1), main="(b)")
40 ccf(as.numeric(Y[,1]),as.numeric(Y[,2]),
41      type=c("correlation"), main="(c)", ylab="CCF", lag=20)
42 cor(ibm, crsp)

[1] 0.4863639

```

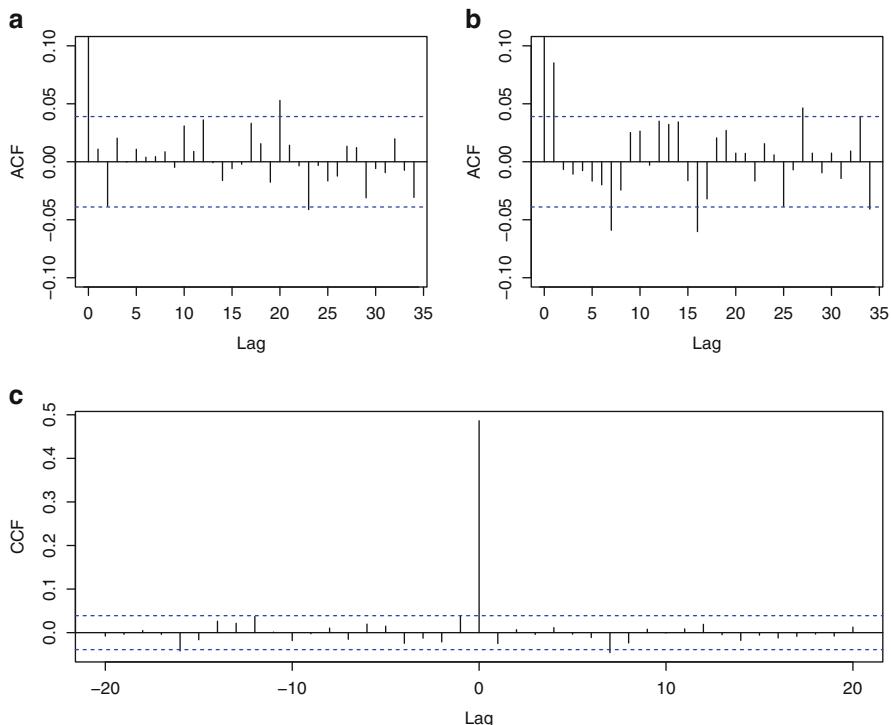
The multivariate Ljung-Box test (see Sect. 13.4.3) is applied to simultaneously test that the first  $K$  auto-correlations, as well as the lagged cross-correlations, are all zero. The multivariate Ljung-Box test statistic at lag five is 50.15. The associate  $p$ -value is very close to zero, which provides strong evidence to reject the null hypothesis and indicates there is significant serial correlation in the vector process.

```

43 source("SDAFE2.R")
44 mLjungBox(Y, 5)

      K   Q(K) d.f. p-value
1 5  50.15    20      0

```



**Fig. 14.12.** ACFs for (a) the IBM stock and (b) CRSP index returns; (c) CCF between IBM and CRSP returns.

For simplicity, we use ordinary least squares to fit a VAR(1) model (see Sect. 13.4.4) to remove the minor serial correlation and focus on the conditional variance and covariance. Let  $\hat{a}_t$  denote the estimated residuals from the regression;  $\hat{a}_t$  is an estimate of the innovation process  $a_t$ , which is described more fully below. The multivariate Ljung-Box test statistic at lag five is now 16.21, which has an approximate  $p$ -value of 0.704, indicating there is no significant serial correlation in the vector residual process.

```

45 fit.AR1 = ar(Y, aic = FALSE, order.max=1)
46 A = fit.AR1$resid[-1,]
47 mLjungBox(A, 5)

      K   Q(K) d.f. p-value
1 5  16.21    20      0.704

```

Although the residual series  $\mathbf{a}_t$  is serially uncorrelated, Fig. 14.13 shows it is not an independent process. Figures 14.13a and b are sample ACF plots for the squared residual series  $\hat{a}_{it}^2$ . They both show substantial positive autocorrelation because of the volatility clustering. Figure 14.13c is the sample CCF for the squared series; this figure shows there is a dynamic relationship between the squared series at low lags. Figure 14.13d is the sample ACF for the product series  $\hat{a}_{1t}\hat{a}_{2t}$  and shows that there is also evidence of positive autocorrelation in the conditional covariance series. The multivariate volatility models described below attempt to account for these forms of dependence exhibited in the vector residual series.

#### 14.14.2 Basic Setting

Let  $\mathbf{Y}_t = (Y_{1,t}, \dots, Y_{d,t})'$  denote a  $d$ -dimensional vector process and let  $\mathcal{F}_t$  denote the information set at time index  $t$ , generated by  $\mathbf{Y}_t, \mathbf{Y}_{t-1}, \dots$ . We may partition the process as

$$\mathbf{Y}_t = \boldsymbol{\mu}_t + \mathbf{a}_t, \quad (14.23)$$

in which  $\boldsymbol{\mu}_t = \mathbb{E}(\mathbf{Y}_t | \mathcal{F}_{t-1})$  is the conditional mean vector at time index  $t$ , and  $\{\mathbf{a}_t\}$  is the mean zero weak white noise innovation vector process with unconditional covariance matrix  $\boldsymbol{\Sigma}_a = \text{Cov}(\mathbf{a}_t)$ . Let

$$\boldsymbol{\Sigma}_t = \text{Cov}(\mathbf{a}_t | \mathcal{F}_{t-1}) = \text{Cov}(\mathbf{Y}_t | \mathcal{F}_{t-1}) \quad (14.24)$$

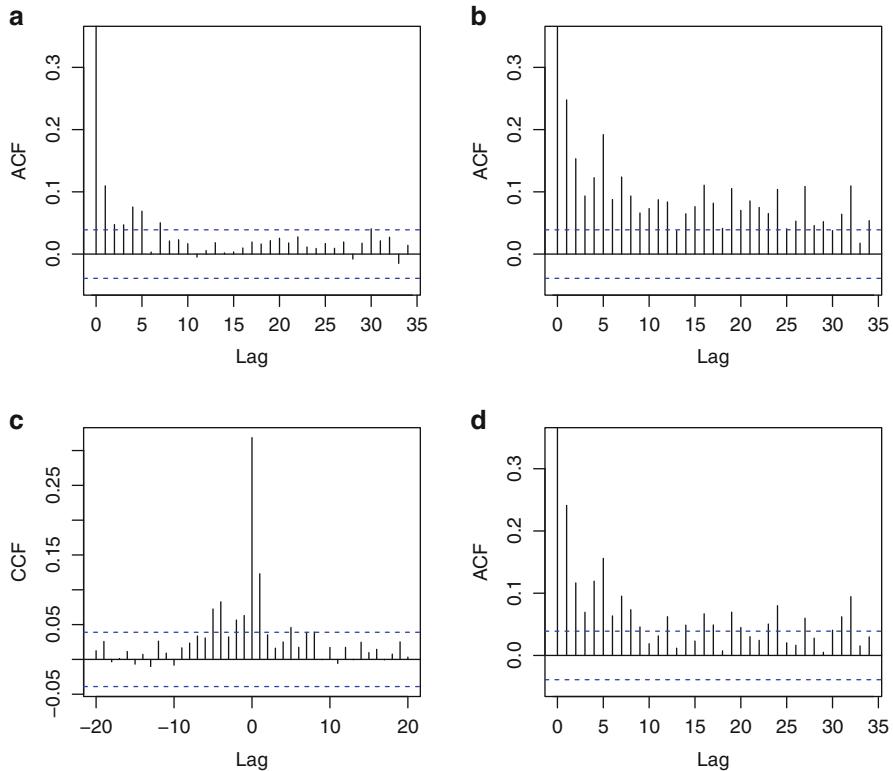
denote the conditional covariance or volatility matrix at time index  $t$ . Multivariate time series modeling is primarily concerned with the time evolutions of  $\boldsymbol{\mu}_t$  and  $\boldsymbol{\Sigma}_t$ , the conditional mean and conditional covariance matrix. For a stationary process, the unconditional mean and unconditional covariance matrix are constant, even though the conditional mean and conditional covariance matrix may be time-varying.

Throughout this section we assume that  $\boldsymbol{\mu}_t$  follows a stationary VAR( $p$ ) model with  $\boldsymbol{\mu}_t = \boldsymbol{\mu} + \sum_{\ell=1}^p \boldsymbol{\Phi}_\ell (\mathbf{Y}_{t-\ell} - \boldsymbol{\mu})$ , where  $p$  is a non-negative integer,  $\boldsymbol{\mu}$  is the  $d \times 1$  unconditional mean vector, and the  $\boldsymbol{\Phi}_\ell$  are  $d \times d$  coefficient matrices, respectively. Recall, the residual series considered in Fig. 14.13 were from a VAR model with  $p = 1$ .

The relationship between the innovation process and the volatility process is defined by

$$\mathbf{a}_t = \boldsymbol{\Sigma}_t^{1/2} \boldsymbol{\epsilon}_t, \quad \boldsymbol{\epsilon}_t \stackrel{iid}{\sim} F(\mathbf{0}, \mathbf{I}_d), \quad (14.25)$$

in which  $\boldsymbol{\Sigma}_t^{1/2}$  is a symmetric *matrix square-root* of  $\boldsymbol{\Sigma}_t$ , such that  $\boldsymbol{\Sigma}_t^{1/2} \boldsymbol{\Sigma}_t^{1/2} = \boldsymbol{\Sigma}_t$ . The iid white noise  $\boldsymbol{\epsilon}_t$  are *standardized* innovations from a multivariate distribution  $F$  with mean zero and a covariance matrix equal to the identity. The models detailed below describe dynamic evolutions for the volatility matrix  $\boldsymbol{\Sigma}_t$ .



**Fig. 14.13.** ACFs of squared residuals from a VAR(1) model for (a) IBM and (b) CRSP; (c) CCF between the squared residuals; (d) ACF for the product of the residuals.

#### 14.14.3 Exponentially Weighted Moving Average (EWMA) Model

The simplest matrix generalization of a univariate volatility model is the exponentially weighted moving average (EWMA) model. It is indexed by a single parameter  $\lambda \in (0, 1)$ , and is defined by the recursion

$$\begin{aligned}\boldsymbol{\Sigma}_t &= (1 - \lambda)\mathbf{a}_{t-1}\mathbf{a}'_{t-1} + \lambda\boldsymbol{\Sigma}_{t-1} \\ &= (1 - \lambda)\sum_{\ell=1}^{\infty} \lambda^{\ell-1}\mathbf{a}_{t-\ell}\mathbf{a}'_{t-\ell}.\end{aligned}\tag{14.26}$$

When the recursion in (14.26) is initialized with a positive-definite (p.d.) matrix the sequence remains p.d. This single parameter model is simple to estimate regardless of the dimension, with large values of  $\lambda$  indicating high persistence in the volatility process. However, the dynamics can be too restrictive in practice, since the component-wise evolutions all have the same discounting factor (i.e., persistence parameter)  $\lambda$ .

Figure 14.14 shows the in-sample fitted EWMA model for  $\hat{a}_t$  assuming a multivariate standard normal distribution for  $\epsilon_t$  and using conditional maximum likelihood estimation. The estimated conditional standard deviations are shown in (a) and (d), and the conditional covariances and implied conditional correlations are shown in (b) and (c), respectively. The persistence parameter  $\lambda$  was estimated as 0.985. Estimation and Fig. 14.14 were calculated using the following commands in R.

```

48 source("SDAFE2.R")
49 EWMA.param = est.ewma(lambda.0=0.95, innov=A)
50 EWMA.Sigma = sigma.ewma(lambda=EWMA.param$lambda.hat, innov=A)
51 par(mfrow = c(2,2))
52 plot(ts(EWMA.Sigma[1,1]^0.5, start = c(1989, 1), frequency = 253),
53       type = 'l', xlab = "year", ylab = NULL,
54       main = expression(paste("(a) ", hat(sigma)[1,t])))
55 plot(ts(EWMA.Sigma[1,2], start = c(1989, 1), frequency = 253),
56       type = 'l', xlab = "year", ylab = NULL,
57       main = expression(paste("(b) ", hat(sigma)[12,t])))
58 plot(ts(EWMA.Sigma[1,2]/(sqrt(EWMA.Sigma[1,1]* EWMA.Sigma[2,2])), 
59       start = c(1989, 1), frequency = 253),
60       type = 'l', xlab = "year", ylab = NULL,
61       main = expression(paste("(c) ", hat(rho)[12,t])))
62 points(ts(mvwindow.cor(A[,1],A[,2], win = 126)$correlation,
63          start = c(1989, 1), frequency = 253),
64          type = 'l', col = 2, lty = 2, lwd=2)
65 plot(ts(EWMA.Sigma[2,2]^0.5, start = c(1989, 1), frequency = 253),
66       type = 'l', xlab = "year", ylab = NULL,
67       main = expression(paste("(d) ", hat(sigma)[2,t])))
68 EWMA.param$lambda.hat

```

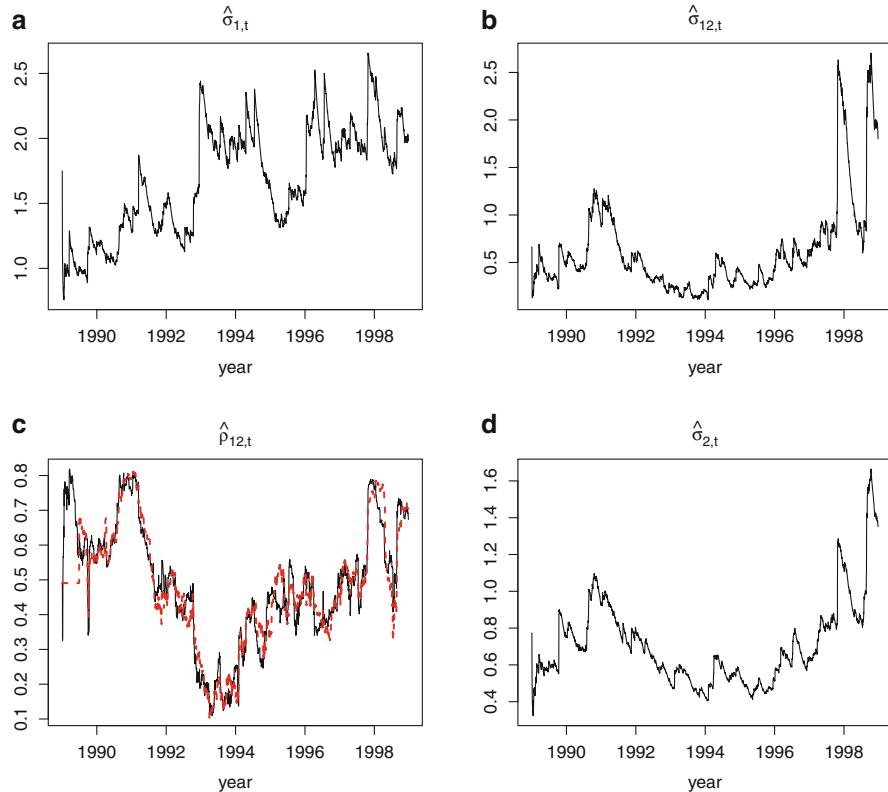
[1] 0.985046

#### 14.14.4 Orthogonal GARCH Models

Several factor and orthogonal models have been proposed to reduce the number of parameters and parameter constraints by imposing a common dynamic structure on the elements of the volatility matrix. The orthogonal GARCH (O-GARCH) model of Alexander (2001) is among the most popular because of its simplicity. It is assumed that the innovations  $a_t$  can be decomposed into orthogonal components  $z_t$  via a linear transformation  $U$ . This is done in conjunction with principal component analysis (PCA, see Sect. 18.2) as follows. Let  $O$  be the matrix of eigenvectors and  $\Lambda$  the diagonal matrix of the corresponding eigenvalues of  $\Sigma_a$ . Then, take  $U = \Lambda^{-1/2}O'$ , and let

$$z_t = U a_t.$$

The components are constructed such that  $\text{Cov}(z_t) = I_d$ . The sample estimate of  $\Sigma_a$  is typically used to estimate  $U$ .

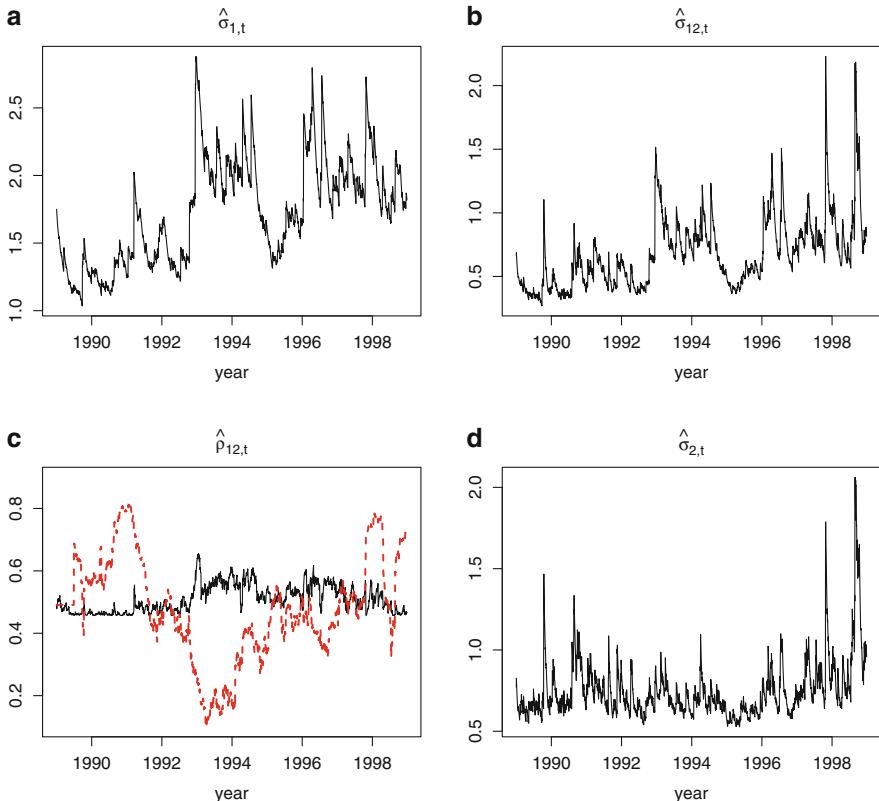


**Fig. 14.14.** A fitted EWMA model with  $\lambda = 0.985$ . The red line in (c) is the sample correlation estimate over the previous six months for comparison.

Next, univariate GARCH(1,1) models are individually fit to each orthogonal component to estimate the *conditional* covariance  $\mathbf{V}_t = \text{Cov}(\mathbf{z}_t | \mathcal{F}_{t-1})$ . Let

$$\begin{aligned} v_{it}^2 &= \omega_i + \alpha_i z_{i,t-1}^2 + \beta_i v_{i,t-1}^2 \\ \mathbf{V}_t &= \text{diag}\{v_{1,t}^2, \dots, v_{d,t}^2\} \\ \boldsymbol{\Sigma}_t &= U^{-1} \mathbf{V}_t U^{-1'}. \end{aligned}$$

In summary, a linear transformation  $U$  is estimated, using PCA, such that the components of  $\mathbf{z}_t = U\mathbf{a}_t$  have unconditional correlation approximately equal to zero. It is then also *assumed* that the *conditional* correlations of  $\mathbf{z}_t$  are also zero; however, this is not at all assured to be true. Under this additional stronger assumption,  $\mathbf{V}_t$ , the conditional covariance matrix for  $\mathbf{z}_t$ , is diagonal. For simplicity, univariate models are then fit to model the conditional variance  $v_{it}^2$  for each component of  $\mathbf{z}_t$ .



**Fig. 14.15.** A fitted first order orthogonal GARCH model with  $(\omega_1, \alpha_1, \beta_1)' = (0.0038, 0.0212, 0.9758)'$ ,  $(\omega_2, \alpha_2, \beta_2)' = (0.0375, 0.0711, 0.8913)'$ , and  $U^{-1} = ((1.7278, 0.2706)', (0.2706, 0.7241)')$ . The red line in (c) is the sample correlation estimate over the previous six months for comparison.

The main drawback of this model is that the orthogonal components are uncorrelated unconditionally, but they may still be conditionally correlated. The O-GARCH model implicitly assumes the conditional correlations for  $\mathbf{z}_t$  are zero. Figure 14.15 shows a fitted O-GARCH model for  $\hat{\mathbf{a}}_t$  using PCA followed by univariate conditional maximum likelihood estimation. The estimated conditional standard deviations are shown in (a) and (d), and the conditional covariances and conditional correlations are shown in (b) and (c), respectively. The implied conditional correlations do not appear adequate for this fitted model compared to the sample correlation estimate over the previous six months (used as a proxy for the conditional correlation process).

### 14.14.5 Dynamic Orthogonal Component (DOC) Models

To properly apply univariate modeling after estimating a linear transformation in the spirit of the O-GARCH model above, the resulting component processes must not only be orthogonal contemporaneously, the conditional correlations must also be zero. Additionally, the lagged cross-correlations for the squared components must also be zero. In Matteson and Tsay (2011), if the components of a time series  $\mathbf{s}_t$  satisfy these conditions, then they are called dynamic orthogonal components (DOCs) in volatility.

Let  $\mathbf{s}_t = (s_{1,t}, \dots, s_{d,t})'$  denote a vector time series of DOCs. Without loss of generality,  $\mathbf{s}_t$  is assumed to be standardized such that  $E(s_{i,t}) = 0$  and  $\text{Var}(s_{i,t}) = 1$  for  $i = 1, \dots, d$ . A Ljung-Box type statistic, defined below, is used to test for the existence of DOCs in volatility. Including lag zero in the test implies that the pairwise product processes among stationary DOCs  $s_{i,t}s_{j,t}$  has zero serial correlation since the Cauchy-Schwarz inequality gives

$$|\text{Cov}(s_{i,t}s_{j,t}, s_{i,t-h}s_{j,t-h})| \leq \text{Var}(s_{i,t}s_{j,t}) = E(s_i^2 s_j^2), \quad (14.27)$$

and  $E(s_{i,t}s_{j,t}) = E(s_{i,t-h}s_{j,t-h}) = 0$  by the assumption of a DOC model.

Let  $\rho_{s_i^2, s_j^2}(h) = \text{Corr}(s_{i,t}^2, s_{j,t-h}^2)$ . The joint lag- $K$  null and alternative hypotheses to test for the existence of DOCs in volatility are

$$\begin{aligned} H_0 : \rho_{s_i^2, s_j^2}(h) &= 0 \text{ for all } i \neq j, h = 0, \dots, K \\ H_A : \rho_{s_i^2, s_j^2}(h) &\neq 0 \text{ for some } i \neq j, h = 0, \dots, K. \end{aligned}$$

The corresponding Ljung-Box type test statistic is

$$Q_d^0(\mathbf{s}^2; K) = n \sum_{i < j} \rho_{s_i^2, s_j^2}(0)^2 + n(n+2) \sum_{h=1}^K \sum_{i \neq j} \rho_{s_i^2, s_j^2}(h)^2 / (n-h). \quad (14.28)$$

Under  $H_0$ ,  $Q_d^0(\mathbf{s}^2; K)$  is asymptotically distributed as Chi-squared with  $d(d-1)/2 + Kd(d-1)$  degrees of freedom. The null hypothesis is rejected for a large value of  $Q_d^0$ . When  $H_0$  is rejected, one must seek an alternative modeling procedure.

As expected from Fig. 14.13, the DOCs in volatility hypothesis is rejected for the VAR(1) residuals. The test statistic is  $Q_2^0(\hat{\mathbf{a}}^2, 5) = 356.926$  with a  $p$ -value near zero. DOCs in volatility is also rejected for the principal components used in the O-GARCH model, the test statistic is  $Q_2^0(\mathbf{z}^2, 5) = 135.492$  with a  $p$ -value near zero. Starting with the uncorrelated principal components  $\mathbf{z}_t$ , Matteson and Tsay (2011) propose estimating an orthogonal matrix  $\mathbf{W}$  such that the components  $\mathbf{s}_t = \mathbf{W}\mathbf{z}_t$  are as close to DOCs in volatility as possible. This is done by minimizing a reweighted version of the Ljung-Box type test statistic (14.28), with respect to the separating matrix  $\mathbf{W}$ . The null hypothesis of DOCs in volatility is accepted for the estimated components  $\mathbf{s}_t$ , with  $Q_2^0(\mathbf{s}^2, 5) = 7.845$  which has a  $p$ -value approximately equal to 0.727.

After DOCs are identified, a univariate volatility model is considered for each process  $v_{i,t}^2 = \text{Var}(s_{i,t} | \mathcal{F}_{t-1})$ . For example, the following model was fit

$$\begin{aligned}\mathbf{a}_t &= \mathbf{M}\mathbf{s}_t = \mathbf{M}\mathbf{V}_t^{1/2}\boldsymbol{\epsilon}_t, \\ \mathbf{V}_t &= \text{diag}\{v_{1,t}^2, \dots, v_{d,t}^2\}, \quad \boldsymbol{\epsilon}_{it} \stackrel{iid}{\sim} t_{\nu_i}(0, 1) \\ v_{i,t}^2 &= \omega_i + \alpha_i s_{i,t-1}^2 + \beta_i v_{i,t-1}^2 \\ \boldsymbol{\Sigma}_t &= \mathbf{M}\mathbf{V}_t\mathbf{M}',\end{aligned}$$

in which  $t_{\nu_i}(0, 1)$  denotes the standardized Student- $t$  distribution with tail-index  $\nu_i$ . Each  $\boldsymbol{\Sigma}_t$  is positive-definite if  $v_{i,t}^2 > 0$  for all components. The fundamental motivation is that empirically the dynamics of  $\mathbf{a}_t$  can often be well approximated by an invertible linear combination of DOCs  $\mathbf{a}_t = \mathbf{M}\mathbf{s}_t$ , in which  $\mathbf{M} = U^{-1}\mathbf{W}'$  by definition.

In summary,  $U$  is estimated by PCA to uncorrelate  $\mathbf{a}_t$ ,  $\mathbf{W}$  is estimated to minimize a reweighted version of (14.28) defined above (giving more weight to lower lags). The matrices  $U$  and  $\mathbf{W}$  are combined to estimate DOCs  $s_t$ , of which univariate volatility modeling may then be appropriately applied. This approach allows modeling of a  $d$ -dimensional multivariate volatility process with  $d$  univariate volatility models, while greatly reducing both the number of parameters and the computational cost of estimation, and at the same time maintaining adequate empirical performance.

Figure 14.16 shows a fitted DOCs in volatility GARCH model for  $\hat{\mathbf{a}}_t$  using generalized decorrelation followed by univariate conditional maximum likelihood estimation. The estimated conditional standard deviations are shown in (a) and (d), and the conditional covariances and implied correlations are shown in (b) and (c), respectively. Unlike the O-GARCH fit, the implied conditional correlations appear adequate compared to the rolling estimator. Estimation of the O-GARCH and DOC models and Figs. 14.15 and 14.16 were calculated using the following commands in R.

```
69 source("SDAFE2.R")
70 DOC.fit = doc.garch(E = A, L = 4., c = 2.25, theta.ini = NULL)

71 par(mfrow = c(2,2)) # O-GARCH
72 plot(ts(DOC.fit$Sigma.pca[1,1,]^ .5, start=c(1989,1), frequency=253),
73       type = 'l', xlab = "year", ylab = NULL,
74       main = expression(paste("(a) ", hat(sigma)[["1,t"]])))
75 plot(ts(DOC.fit$Sigma.pca[2,1,], start=c(1989,1), frequency=253),
76       type = 'l', xlab = "year", ylab = NULL,
77       main = expression(paste("(b) ", hat(sigma)[["12,t"]])))
78 plot(ts(DOC.fit$Sigma.pca[2,1,]/(sqrt(DOC.fit$Sigma.pca[1,1,]*

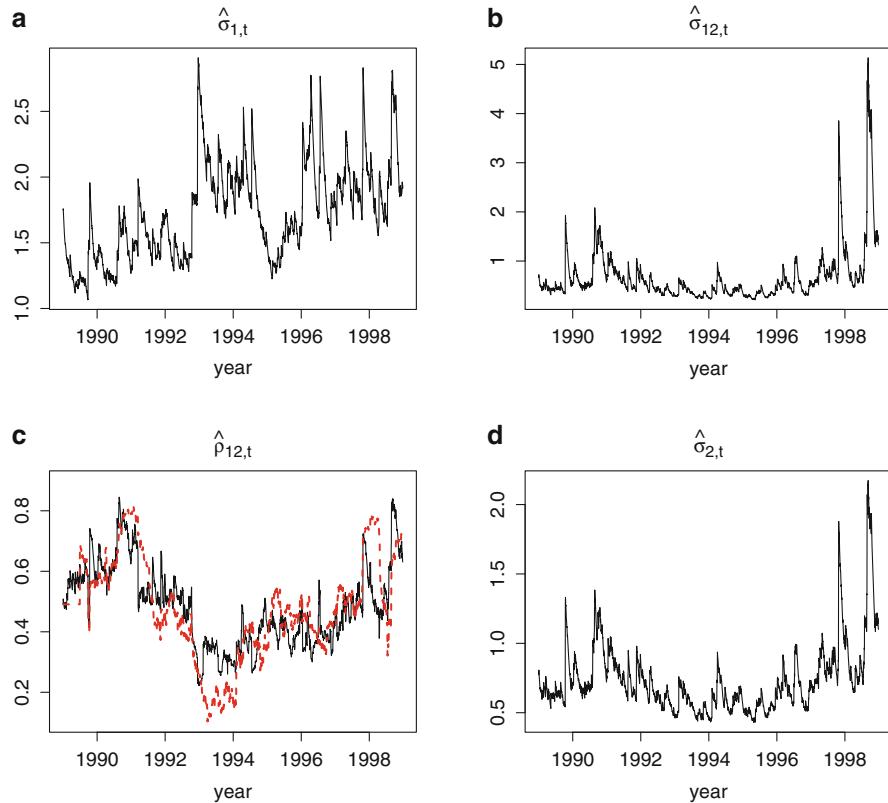
79                               DOC.fit$Sigma.pca[2,2,])),          

80       start=c(1989,1), frequency=253),
81       type = 'l', xlab = "year", ylab = NULL, ylim = c(0.1,0.9),
82       main = expression(paste("(c) ", hat(rho)[["12,t"]])))
83 points(ts(mvwindow.cor(A[,1],A[,2], win = 126)$correlation,
84          start = c(1989, 1), frequency = 253),
```

```

85     type = 'l', col = 2, lty = 2, lwd = 2)
86 plot(ts(DOC.fit$Sigma.pca[2,2,]^ .5, start=c(1989,1), frequency=253),
87       type = 'l', xlab = "year", ylab = NULL,
88       main = expression(paste("(d) ", hat(sigma)[ "2,t"])))

```



**Fig. 14.16.** A fitted first order DOCs in volatility GARCH model with  $(\omega_1, \alpha_1, \beta_1, \nu_1)' = (0.0049, 0.0256, 0.9703, 4.3131)', (\omega_2, \alpha_2, \beta_2, \nu_2)' = (0.0091, 0.0475, 0.9446, 5.0297)',$  and  $\mathbf{M} = ((1.5350, 0.838)', (0.0103, 0.7730)')$ . The red line in (c) is the sample correlation estimate over the previous six months for comparison.

```

89 par(mfrow = c(2,2)) # DOCs in volatility
90 plot(ts(DOC.fit$Sigma.doc[1,1,]^ .5, start=c(1989,1), frequency=253),
91       type = 'l', xlab = "year", ylab = NULL,
92       main = expression(paste("(a) ", hat(sigma)[ "1,t"])))
93 plot(ts(DOC.fit$Sigma.doc[2,1,], start=c(1989,1), frequency=253),
94       type = 'l', xlab = "year", ylab = NULL,
95       main = expression(paste("(b) ", hat(sigma)[ "12,t"])))
96 plot(ts(DOC.fit$Sigma.doc[2,1,]/(sqrt(DOC.fit$Sigma.doc[1,1,]*
```

```

97                               DOC.fit$Sigma.doc[2,2,]), ,
98   start=c(1989,1), frequency=253),
99   type = 'l', xlab = "year", ylab = NULL, ylim = c(0.1,0.9),
100  main = expression(paste("(c) ", hat(rho)["12,t"])))
101 points(ts(mvwindow.cor(A[,1],A[,2], win = 126)$correlation,
102         start=c(1989,1), frequency=253),
103         type = 'l', col = 2, lty = 2,lwd=2)
104 plot(ts(DOC.fit$Sigma.doc[2,2,]^5, start=c(1989,1), frequency=253),
105       type = 'l', xlab = "year", ylab = NULL,
106       main = expression(paste("(d) ", hat(sigma)["2,t"])))
107 DOC.fit$coef.pca
108
      omega      alpha1      beta1
[1,] 0.003845283 0.02118369 0.9758129
[2,] 0.037473820 0.07101731 0.8913321
109 DOC.fit$coef.doc
110
      omega      alpha1      beta1      shape
[1,] 0.004874403 0.02560464 0.9702966 4.313164
[2,] 0.009092705 0.04740792 0.9446408 5.030019
111 DOC.fit$W.hat
112
      [,1]      [,2]
[1,] 0.9412834 -0.3376174
[2,] 0.3376174  0.9412834
113 DOC.fit$U.hat
114
      [,1]      [,2]
[1,] 0.6147515 -0.2297417
[2,] -0.2297417  1.4669516
115 DOC.fit$M.hat
116
      [,1]      [,2]
[1,] 1.53499088 0.8380397
[2,] 0.01024854 0.7729063
117 solve(DOC.fit$U.hat)
118
      [,1]      [,2]
[1,] 1.7277983 0.2705934
[2,] 0.2705934 0.7240638

```

#### 14.14.6 Dynamic Conditional Correlation (DCC) Models

Nonlinear combinations of univariate volatility models have been proposed to allow for time-varying correlations, a feature that is prevalent in many financial applications. Both Tse and Tsui (2002) and Engle (2002) generalize the constant correlation model of Bollerslev (1990) to allow for such dynamic conditional correlations (DCC).

Analogously to the GARCH(1,1) model, the first order form of the DCC model in Engle (2002) may be represented by the following equations

$$\begin{aligned}\sigma_{i,t}^2 &= \omega_i + \alpha_i a_{i,t-1}^2 + \beta_i \sigma_{i,t-1}^2, \\ \mathbf{D}_t &= \text{diag}\{\sigma_{1,t}, \dots, \sigma_{d,t}\}, \\ \boldsymbol{\varepsilon}_t &= \mathbf{D}_t^{-1} \mathbf{a}_t, \\ \mathbf{Q}_t &= (1 - \lambda) \boldsymbol{\varepsilon}_{t-1} \boldsymbol{\varepsilon}'_{t-1} + \lambda \mathbf{Q}_{t-1}, \\ \mathbf{R}_t &= \text{diag}\{\mathbf{Q}_t\}^{-\frac{1}{2}} \mathbf{Q}_t \text{diag}\{\mathbf{Q}_t\}^{-\frac{1}{2}}, \\ \boldsymbol{\Sigma}_t &= \mathbf{D}_t \mathbf{R}_t \mathbf{D}_t.\end{aligned}$$

The main idea is to first model the conditional variance of each individual series  $\sigma_{it}^2$  with a univariate volatility model, estimate the *scaled* innovations  $\boldsymbol{\varepsilon}_t$  (not to be confused with *standardized* innovations  $\boldsymbol{\epsilon}_t$ ) from these models, then focus on modeling the conditional correlation matrix  $\mathbf{R}_t$ . These are then combined at each time point  $t$  to estimate the volatility matrix  $\boldsymbol{\Sigma}_t$ . The recursion  $\mathbf{Q}_t$  is an EWMA model applied to the scaled innovations  $\boldsymbol{\varepsilon}_t$ . It is indexed by a single parameter  $\lambda \in (0, 1)$ . The matrix  $\mathbf{Q}_t$  needs to be rescaled to form a proper correlation matrix  $\mathbf{R}_t$  with the value 1 for all elements on the main diagonal.

The DCC model parameters can be estimated consistently in two stages using quasi-maximum likelihood. First, a univariate GARCH(1,1) model is fit to each series to estimate  $\sigma_{it}^2$ . Then, given  $\boldsymbol{\varepsilon}_t$ ,  $\lambda$  is estimated by maximizing the components of the quasi-likelihood that only depend on the correlations. This is justified since the squared residuals do not depend on the correlation parameters.

In the form above, the variance components only condition on their own individual lagged returns and not the joint returns. Also, the dynamics for each of the conditional correlations are constrained to have equal persistence parameters, similar to the EWMA model. An explicit parameterization of the conditional correlation matrix  $\mathbf{R}_t$ , with flexible dynamics, is just as difficult to estimate in high dimensions as  $\boldsymbol{\Sigma}_t$  itself. Figure 14.17 shows a fitted DCC model for  $\hat{\boldsymbol{\alpha}}_t$  using quasi-maximum likelihood estimation. The estimated conditional standard deviations are shown in (a) and (d), and the conditional covariances and conditional correlations are shown in (b) and (c), respectively. Estimation and Fig. 14.17 were calculated using the following commands in R.

```
113 source("SDAFE2.R")
114 DCCe.fit = fit.DCCe(theta.0=0.95, innov=A)
115 DCCe.fit$coeff

$$\begin{array}{llll} \text{omega} & \text{alpha1} & \text{beta1} \\ [1,] & 0.07435095 & 0.05528162 & 0.9231251 \\ [2,] & 0.02064808 & 0.08341755 & 0.8822517 \end{array}$$

116 DCCe.fit$lambda
```

```
[1] 0.9876297

117 par(mfrow = c(2,2))
118 plot(ts(DCCe.fit$Sigma.t[1,1,]^0.5, start=c(1989, 1), frequency=253),
119       type = 'l', xlab = "year", ylab = NULL,
120       main = expression(paste("(a) ", hat(sigma)[1,t])))
121 plot(ts(DCCe.fit$Sigma.t[2,1,], start=c(1989, 1), frequency=253),
122       type = 'l', xlab = "year", ylab = NULL,
123       main = expression(paste("(b) ", hat(sigma)[12,t])))
124 plot(ts(DCCe.fit$R.t[2,1,], start=c(1989, 1), frequency=253),
125       type = 'l', xlab = "year", ylab = NULL,
126       main = expression(paste("(c) ", hat(rho)[12,t])))
127 points(ts(mvwindow.cor(A[,1],A[,2], win = 126)$correlation,
128          start=c(1989, 1), frequency=253),
129          type = 'l', col = 2, lty = 2, lwd=2)
130 plot(ts(DCCe.fit$Sigma.t[2,2,]^0.5, start=c(1989, 1), frequency=253),
131       type = 'l', xlab = "year", ylab = NULL,
132       main = expression(paste("(d) ", hat(sigma)[2,t])))
```

#### 14.14.7 Model Checking

For a fitted volatility sequence  $\hat{\Sigma}_t$ , the *standardized* residuals are defined as

$$\hat{\epsilon}_t = \hat{\Sigma}_t^{-1/2} \mathbf{a}_t, \quad (14.29)$$

in which  $\hat{\Sigma}_t^{-1/2}$  denotes the inverse of the matrix  $\hat{\Sigma}_t^{1/2}$ . To verify the adequacy of a fitted volatility model, lagged cross-correlations of the squared standardized residuals should be zero. The product process  $\hat{\epsilon}_{it}\hat{\epsilon}_{jt}$  should also have no serial correlation. Additional diagnostic checks for time series are considered in Li (2003). Since the standardized residuals are estimated and not observed, all *p*-values given in this section are only approximate.

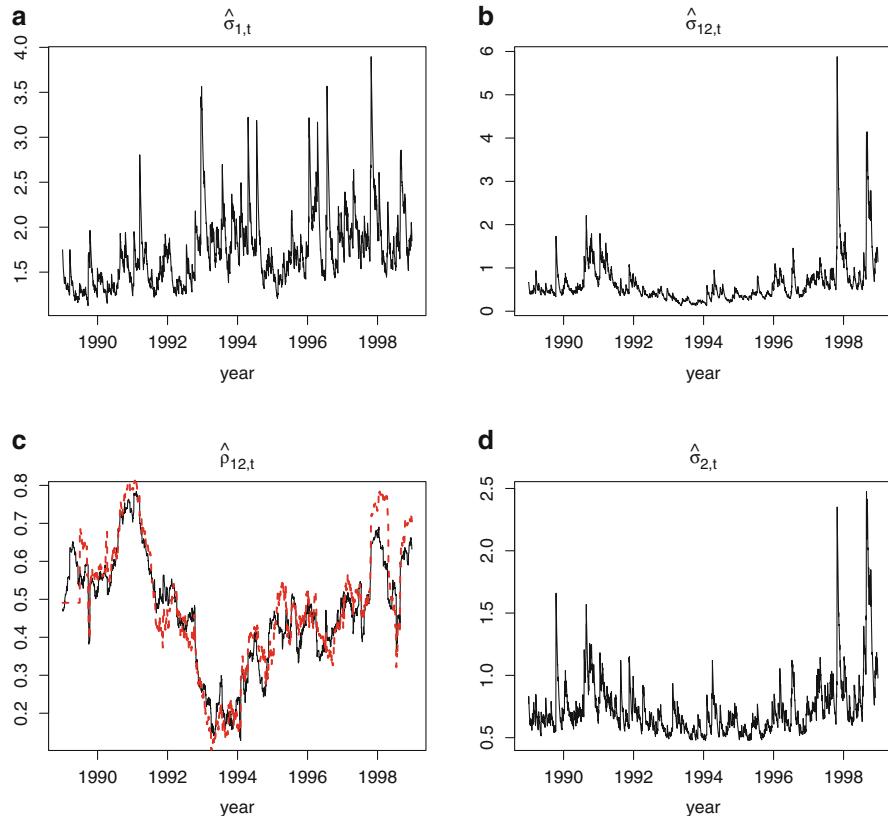
To check the first condition we can apply a multivariate Ljung-Box test to the squared standardized residuals. For the EWMA model,  $Q_2(\hat{\epsilon}_t^2, 5) = 26.40$  with a *p*-value of 0.153, implying no significant serial correlation. For the O-GARCH model,  $Q_2(\hat{\epsilon}_t^2, 5) = 30.77$  with a *p*-value 0.058. In this case, there is some minor evidence of serial correlation. For the DOC in volatility model,  $Q_2(\hat{\epsilon}_t^2, 5) = 18.68$  with a *p*-value 0.543, implying no significant serial correlation. For the DCC model,  $Q_2(\hat{\epsilon}_t^2, 5) = 10.54$  with a *p*-value 0.957, implying no significant serial correlation.

```
133 n = dim(A)[1] ; d = dim(A)[2]
134 stdResid.EWMA = matrix(0,n,d)
135 stdResid.PCA = matrix(0,n,d)
136 stdResid.DOC = matrix(0,n,d)
137 stdResid.DCCe = matrix(0,n,d)
138 for(t in 1:n){
139   stdResid.EWMA[t,] = A[t,] %*% matrix.sqrt.inv(EWMA.Sigma[,,t])
140   stdResid.PCA[t,] = A[t,] %*% matrix.sqrt.inv(DOC.fit$Sigma.pca[,,t])
141   stdResid.DOC[t,] = A[t,] %*% matrix.sqrt.inv(DOC.fit$Sigma.doc[,,t])
```

```

142 stdResid.DCCe[t,] = A[t,] %*% matrix.sqrt.inv(DCCe.fit$Sigma.t[, , t])
143 }
144 mLjungBox(stdResid.EWMA^2, lag=5)
145 mLjungBox(stdResid.PCA^2, lag=5)
146 mLjungBox(stdResid.DOC^2, lag=5)
147 mLjungBox(stdResid.DCCe^2, lag=5)

```



**Fig. 14.17.** A fitted first order DCC model with  $(\omega_1, \alpha_1, \beta_1)' = (0.0741, 0.0552, 0.9233)'$ ,  $(\omega_2, \alpha_2, \beta_2)' = (0.0206, 0.0834, 0.8823)'$ , and  $\lambda = 0.9876$ . The red line in (c) is the sample correlation estimate over the previous six months for comparison.

The multivariate Ljung-Box test for the squared standardized residuals is not sensitive to misspecification of the conditional correlation structure. To check this condition, we apply univariate Ljung-Box tests to the product of each pair of standardized residuals. For the EWMA model,  $Q(\hat{\epsilon}_{1t}\hat{\epsilon}_{2t}, 5) = 16.45$  with a  $p$ -value of 0.006. This model has not adequately accounted for the time-varying conditional correlation. For the O-GARCH model,  $Q(\hat{\epsilon}_{1t}\hat{\epsilon}_{2t}, 5) = 63.09$  with a  $p$ -value near zero. This model also fails to

account for the observed time-varying conditional correlation. For the DOC in volatility model,  $Q(\hat{\epsilon}_{1t}\hat{\epsilon}_{2t}, 5) = 9.07$  with a  $p$ -value of 0.106, implying no significant serial correlation. For the DCC model,  $Q(\hat{\epsilon}_{1t}\hat{\epsilon}_{2t}, 5) = 8.37$  with a  $p$ -value of 0.137, also implying no significant serial correlation.

```
148 mLjungBox(stdResid.EWMA[,1] * stdResid.EWMA[,2], lag=5)
149 mLjungBox(stdResid.PCA[,1] * stdResid.PCA[,2], lag=5)
150 mLjungBox(stdResid.DOC[,1] * stdResid.DOC[,2], lag=5)
151 mLjungBox(stdResid.DCCe[,1] * stdResid.DCCe[,2], lag=5)
```

## 14.15 Bibliographic Notes

Modeling nonconstant conditional variances in regression is treated in depth in the book by Carroll and Ruppert (1988).

There is a vast literature on GARCH processes beginning with Engle (1982), where ARCH models were introduced. Hamilton (1994), Enders (2004), Pindyck and Rubinfeld (1998), Gourieroux and Jasiak (2001), Alexander (2001), and Tsay (2005) have chapters on GARCH models. There are many review articles, including Bollerslev (1986), Bera and Higgins (1993), Bollerslev, Engle, and Nelson (1994), and Bollerslev, Chou, and Kroner (1992). Jarrow (1998) and Rossi (1996) contain a number of papers on volatility in financial markets. Duan (1995), Ritchken and Trevor (1999), Heston and Nandi (2000), Hsieh and Ritchken (2000), Duan and Simonato (2001), and many other authors study the effects of GARCH errors on options pricing, and Bollerslev, Engle, and Wooldridge (1988) use GARCH models in the CAPM.

For a thorough review of multivariate GARCH modeling see Bauwens, Laurent, and Rombouts (2006), and Silvennoinen and Teräsvirta (2009).

## 14.16 R Lab

### 14.16.1 Fitting GARCH Models

Run the following code to load the data set `TbGdpPi.csv`, which has three variables: the 91-day T-bill rate, the log of real GDP, and the inflation rate. In this lab you will use only the T-bill rate.

```
1 TbGdpPi = read.csv("TbGdpPi.csv", header=TRUE)
2 # r = the 91-day treasury bill rate
3 # y = the log of real GDP
4 # pi = the inflation rate
5 TbGdpPi = ts(TbGdpPi, start = 1955, freq = 4)
6 Tbill = TbGdpPi[,1]
7 Tbill.diff = diff(Tbill)
```

**Problem 1** Plot both Tbill and Tbill.diff. Use both time series and ACF plots. Also, perform ADF and KPSS tests on both series. Which series do you think are stationary? Why? What types of heteroskedasticity can you see in the Tbill.diff series?

In the following code, the variable Tbill can be used if you believe that series is stationary. Otherwise, replace Tbill by Tbill.diff. This code will fit an ARMA+GARCH model to the series.

```

8 library(rugarch)
9 arma.garch.norm = ugarchspec(mean.model=list(armaOrder=c(1,0)),
10                               variance.model=list(garchOrder=c(1,1)))
11 Tbill.arma.garch.norm = ugarchfit(data=Tbill, spec=arma.garch.norm)
12 show(Tbill.arma.garch.norm)

```

**Problem 2 (a)** Which ARMA+GARCH model is being fit? Write down the model using the same parameter names as in the R output.

(b) What are the estimates of each of the parameters in the model?

Next, plot the residuals (ordinary or raw) and standardized residuals in various ways using the code below. The standardized residuals are best for checking the model, but the residuals are useful to see if there are GARCH effects in the series.

```

13 res = ts(residuals(Tbill.arma.garch.norm, standardize=FALSE),
14           start = 1955, freq = 4)
15 res.std = ts(residuals(Tbill.arma.garch.norm, standardize=TRUE),
16               start = 1955, freq = 4)
17 par(mfrow=c(2,3))
18 plot(res)
19 acf(res)
20 acf(res^2)
21 plot(res.std)
22 acf(res.std)
23 acf(res.std^2)

```

**Problem 3 (a)** Describe what is plotted by acf(res). What, if anything, does the plot tell you about the fit of the model?

(b) Describe what is plotted by acf(res^2). What, if anything, does the plot tell you about the fit of the model?

(c) Describe what is plotted by acf(res\_std^2). What, if anything, does the plot tell you about the fit of the model?

(d) Is there anything noteworthy in the figure produced by the command plot(res.std) ?

**Problem 4** Now find an ARMA+GARCH model for the series `diff.log.Tbill`, which we will define as `diff(log(Tbill))`. Do you see any advantages of working with the differences of the logarithms of the T-bill rate, rather than with the difference of `Tbill` as was done earlier?

### 14.16.2 The GARCH-in-Mean (GARCH-M) Model

A GARCH-in-Mean or *GARCH-M* model takes the form

$$\begin{aligned} Y_t &= \mu + \delta\sigma_t + a_t \\ a_t &= \sigma_t \epsilon_t \\ \sigma_t^2 &= \omega + \alpha a_{t-1}^2 + \beta \sigma_{t-1}^2 \end{aligned}$$

in which  $\epsilon_t \stackrel{iid}{\sim} (0, 1)$ . The GARCH-M model directly incorporates volatility as a regression variable. The parameter  $\delta$  represents the *risk premium*, or reward for additional risk. Modern portfolio theory dictates that increased volatility leads to increased risk, requiring larger expected returns. The presence of volatility as a statistically significant predictor of returns is one of the primary contributors to serial correlation in historic return series. The data set `GPRO.csv()` contains the adjusted daily closing price of GoPro stock from June 26, 2014 to January 28, 2015.

Run the following R commands to fit a GARCH-M model to the GoPro stock returns.

```
1 library(rugarch)
2 GPRO = read.table("GPRO.csv")
3 garchm = ugarchspec(mean.model=list(armaOrder=c(0,0),
4                                     archm=T, archpow=1),
5                     variance.model=list(garchOrder=c(1,1)))
6 GPRO.garchm = ugarchfit(garchm, data=GPRO)
7 show(GPRO.garchm)
```

**Problem 5** Write out the fitted model. The parameter  $\delta$  is equal to `archm` in the R output.

**Problem 6** Test the one-sided hypothesis that  $\delta > 0$  verses the alternative that  $\delta = 0$ . Is the risk premium significant?

### 14.16.3 Fitting Multivariate GARCH Models

Run the following code to again load the data set `TbGdpPi.csv`, which has three variables: the 91-day T-bill rate, the log of real GDP, and the inflation rate. In this lab you will now use the first and third series after taking first differences.

```

1 TbGdpPi = read.csv("TbGdpPi.csv", header=TRUE)
2 TbPi.diff = ts(apply(TbGdpPi[,-2], 2, diff), start=c(1955,2), freq=4)
3 plot(TbPi.diff)
4 acf(TbPi.diff^2)
5 source("SDAFE2.R")
6 mLjungBox(TbPi.diff^2, lag=8)

```

**Problem 7** Does the joint series exhibit conditional heteroskedasticity? Why?

Now fit and plot a EWMA model with the following R commands.

```

7 EWMA.param = est.ewma(lambda.0=0.95, innov=TbPi.diff)
8 EWMA.param$lambda.hat
9 EWMA.Sigma=sigma.ewma(lambda=EWMA.param$lambda.hat, innov=TbPi.diff)
10 par(mfrow = c(2,2))
11 plot(ts(EWMA.Sigma[1,1]^0.5, start = c(1955, 2), frequency = 4),
12       type = 'l', xlab = "year", ylab = NULL,
13       main = expression(paste("(a) ", hat(sigma)[1,t])))
14 plot(ts(EWMA.Sigma[1,2], start = c(1955, 2), frequency = 4),
15       type = 'l', xlab = "year", ylab = NULL,
16       main = expression(paste("(b) ", hat(sigma)[12,t])))
17 plot(ts(EWMA.Sigma[1,2]/(sqrt(EWMA.Sigma[1,1]* EWMA.Sigma[2,2])), 
18       start = c(1955, 2), frequency = 4),
19       type = 'l', xlab = "year", ylab = NULL,
20       main = expression(paste("(c) ", hat(rho)[12,t])))
21 plot(ts(EWMA.Sigma[2,2]^0.5, start = c(1955, 2), frequency = 4),
22       type = 'l', xlab = "year", ylab = NULL,
23       main = expression(paste("(d) ", hat(sigma)[2,t])))

```

**Problem 8** What is the estimated persistence parameter  $\lambda$ ?

Now estimate standardized residuals and check whether they exhibit any conditional heteroskedasticity

```

24 n = dim(TbPi.diff)[1]
25 d = dim(TbPi.diff)[2]
26 stdResid.EWMA = matrix(0,n,d)
27 for(t in 1:n){
28   stdResid.EWMA[t,] = TbPi.diff[t,] %*% matrix.sqrt.inv
29   (EWMA.Sigma[,t])
30 }
31 mLjungBox(stdResid.EWMA^2, lag=8)

```

**Problem 9** Based on the output of the Ljung-Box test for the squared standardized residuals, is the EWMA model adequate?

Run the following command in R to determine whether the joint series are DOCs in volatility.

```
32 DOC.test(TbPi.diff^2, 8)
```

**Problem 10** Is the null hypothesis rejected? Based on this conclusion, how should the conditional heteroskedasticity in the bivariate series be modeled, jointly or separately?

## 14.17 Exercises

- Let  $Z$  have an  $N(0, 1)$  distribution. Show that

$$E(|Z|) = \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}} |z| e^{-z^2/2} dz = 2 \int_0^{\infty} \frac{1}{\sqrt{2\pi}} z e^{-z^2/2} dz = \sqrt{\frac{2}{\pi}}.$$

Hint:  $\frac{d}{dz} e^{-z^2/2} = -ze^{-z^2/2}$ .

- Suppose that  $f_X(x) = 1/4$  if  $|x| < 1$  and  $f_X(x) = 1/(4x^2)$  if  $|x| \geq 1$ . Show that

$$\int_{-\infty}^{\infty} f_X(x) dx = 1,$$

so that  $f_X$  really is a density, but that

$$\int_{-\infty}^0 x f_X(x) dx = -\infty$$

and

$$\int_0^{\infty} x f_X(x) dx = \infty,$$

so that a random variable with this density does not have an expected value.

- Suppose that  $\epsilon_t$  is an i.i.d.  $WN(0, 1)$  process, that

$$a_t = \epsilon_t \sqrt{1 + 0.35a_{t-1}^2},$$

and that

$$y_t = 3 + 0.72y_{t-1} + a_t.$$

- (a) Find the mean of  $y_t$ .
  - (b) Find the variance of  $y_t$ .
  - (c) Find the autocorrelation function of  $y_t$ .
  - (d) Find the autocorrelation function of  $a_t^2$ .
- Let  $y_t$  be the AR(1)+ARCH(1) model

$$a_t = \epsilon_t \sqrt{\omega + \alpha a_{t-1}^2},$$

$$(y_t - \mu) = \phi(y_{t-1} - \mu) + a_t,$$

where  $\epsilon_t$  is i.i.d.  $WN(0, 1)$ . Suppose that  $\mu = 0.4$ ,  $\phi = 0.45$ ,  $\omega = 1$ , and  $\alpha_1 = 0.3$ .

- (a) Find  $E(y_2|y_1 = 1, y_0 = 0.2)$ .  
 (b) Find  $\text{Var}(y_2|y_1 = 1, y_0 = 0.2)$ .
5. Suppose that  $\epsilon_t$  is white noise with mean 0 and variance 1, that  $a_t = \epsilon_t \sqrt{7 + a_{t-1}^2}/2$ , and that  $Y_t = 2 + 0.67Y_{t-1} + a_t$ .
- (a) What is the mean of  $Y_t$ ?  
 (b) What is the ACF of  $Y_t$ ?  
 (c) What is the ACF of  $a_t$ ?  
 (d) What is the ACF of  $a_t^2$ ?
6. Let  $Y_t$  be a stock's return in time period  $t$  and let  $X_t$  be the inflation rate during this time period. Assume the model

$$Y_t = \beta_0 + \beta_1 X_t + \delta \sigma_t + a_t, \quad (14.30)$$

where

$$a_t = \epsilon_t \sqrt{1 + 0.5a_{t-1}^2}. \quad (14.31)$$

Here the  $\epsilon_t$  are independent  $N(0, 1)$  random variables. Model (14.30)–(14.31) is called a *GARCH-in-mean* model or a GARCH-M model.

Assume that  $\beta_0 = 0.06$ ,  $\beta_1 = 0.35$ , and  $\delta = 0.22$ .

- (a) What is  $E(Y_t|X_t = 0.1 \text{ and } a_{t-1} = 0.6)$ ?  
 (b) What is  $\text{Var}(Y_t|X_t = 0.1 \text{ and } a_{t-1} = 0.6)$ ?  
 (c) Is the conditional distribution of  $Y_t$  given  $X_t$  and  $a_{t-1}$  normal? Why or why not?  
 (d) Is the marginal distribution of  $Y_t$  normal? Why or why not?
7. Suppose that  $\epsilon_1, \epsilon_2, \dots$  is a Gaussian white noise process with mean 0 and variance 1, and  $a_t$  and  $y_t$  are stationary processes such that

$$a_t = \sigma_t \epsilon_t \quad \text{where} \quad \sigma_t^2 = 2 + 0.3a_{t-1}^2,$$

and

$$y_t = 2 + 0.6y_{t-1} + a_t.$$

- (a) What type of process is  $a_t$ ?  
 (b) What type of process is  $y_t$ ?  
 (c) Is  $a_t$  Gaussian? If not, does it have heavy or lighter tails than a Gaussian distribution?  
 (d) What is the ACF of  $a_t$ ?  
 (e) What is the ACF of  $a_t^2$ ?  
 (f) What is the ACF of  $y_t$ ?
8. On Black Monday, the return on the S&P 500 was  $-22.8\%$ . Ouch! This exercise attempts to answer the question, “what was the conditional probability of a return this small or smaller on Black Monday?” “Conditional” means given the information available the previous trading day. Run the following R code:

```

1 library(rugarch)
2 library(Ecdat)
3 data(SP500, package="Ecdat")
4 returnBlMon = SP500$r500[1805] ; returnBlMon
5 x = SP500$r500[(1804-2*253+1):1804]
6 ts.plot(c(x,returnBlMon))
7 spec = ugarchspec(mean.model=list(armaOrder=c(1,0)),
8                   variance.model=list(garchOrder=c(1,1)),
9                   distribution.model = "std")
10 fit = ugarchfit(data=x, spec=spec)
11 dfhat = coef(fit)[6]
12 forecast = ugarchforecast(fit, data=x, n.ahead=1)

```

The S&P 500 returns are in the data set SP500 in the Ecdat package. The returns are the variable r500 (this is the only variable in this data set). Black Monday is the 1805th return in this data set. This code fits an AR(1)+GARCH(1,1) model to the last two years of data before Black Monday, assuming 253 trading days/year. The conditional distribution of the white noise is the  $t$ -distribution (called "std" in ugarchspec()). The code also plots the returns during these two years and on Black Monday. From the plot you can see that Black Monday was highly unusual. The parameter estimates are in coef(fit) and the sixth parameter is the degrees of freedom of the  $t$ -distribution. The ugarchforecast() function is used to predict one-step ahead, that is, to predict the return on Black Monday; the input variable n.ahead specifies how many days ahead to forecast, so n.ahead=5 would forecast the next five days. The object forecast will contain fitted(forecast), which is the conditional expected return on Black Monday, and sigma(forecast), which is the conditional standard deviation of the return on Black Monday.

- (a) Use the information above to calculate the conditional probability of a return less than or equal to  $-0.228$  on Black Monday.
  - (b) Compute and plot the standardized residuals. Also plot the ACF of the standardized residuals and their squares. Include all three plots with your work. Do the standardized residuals indicate that the AR(1)+GARCH(1,1) model fits adequately?
  - (c) Would an AR(1)+ARCH(1) model provide an adequate fit?
  - (d) Does an AR(1) model with a Gaussian conditional distribution provide an adequate fit? Use the arima() function to fit the AR(1) model. This function only allows a Gaussian conditional distribution.
9. This problem uses monthly observations of the two-month yield, that is,  $Y_T$  with  $T$  equal to two months, in the data set Irates in the Ecdat package. The rates are log-transformed to stabilize the variance. To fit a GARCH model to the changes in the log rates, run the following R code.

```

13 library(rugarch)
14 library(Ecdat)
15 data(Irates)

```

```

16 r = as.numeric(log(Irates[,2]))
17 n = length(r)
18 lagr = r[1:(n-1)]
19 diffrr = r[2:n] - lagr
20 spec = ugarchspec(mean.model=list(armaOrder=c(1,0)),
21                   variance.model=list(garchOrder=c(1,1)),
22                   distribution.model = "std")
23 fit = ugarchfit(data=diffrr, spec=spec)
24 plot(fit, which="all")

```

- (a) What model is being fit to the changes in  $r$ ? Describe the model in detail.
- (b) What are the estimates of the parameters of the model?
- (c) What is the estimated ACF of  $\Delta r_t$ ?
- (d) What is the estimated ACF of  $a_t$ ?
- (e) What is the estimated ACF of  $a_t^2$ ?
10. Consider the daily log returns on the S&P 500 index (GSPC). Begin by running the following commands in R, then answer the questions below for the series  $y$ .
- ```

25 library(rugarch)
26 library(quantmod)
27 getSymbols("^GSPC", from="2005-01-01", to="2014-12-31")
28 head(GSPC)
29 sp500 = xts( diff( log( GSPC[,6] ) )[-1] )
30 plot(sp500)
31 y = as.numeric(sp500)

```
- (a) Is there any serial correlation in the log returns of S&P 500 index? Why?
- (b) Is there any ARCH effect (evidence of conditional heteroskedasticity) in the log returns of S&P 500 index? Why?
- (c) Specify and fit an ARCH model to the log returns of S&P 500 index. Write down the fitted model.
- (d) Is your fitted ARCH model stationary? Why?
- (e) Fit a GARCH(1,1) model for the log returns on the S&P 500 index using the Gaussian distribution for the innovations. Write down the fitted model.
- (f) Perform model checking to ensure that the model is adequate using 20 lags in a Ljung-Box test of the standardized residuals and the squared standardized residuals.
- (g) Is the fitted GARCH model stationary? Why?
- (h) Make a Normal quantile plot for the standardized residuals. Use `qqnorm()` and `qqline()` in R. Is the Gaussian distribution appropriate for the standardized innovations?
- (i) Plot the fitted conditional standard deviation process  $\hat{\sigma}_t$  and comment.

- (j) Calculate the 1–10 step ahead forecasts from the end of the series for both the process  $y_t$  and the conditional variance using the `ugarchforecast()` function.

## References

- Alexander, C. (2001) *Market Models: A Guide to Financial Data Analysis*, Wiley, Chichester.
- Bauwens, L., Laurent, S., and Rombouts, J. V. (2006) Multivariate GARCH models: a survey. *Journal of Applied Econometrics*, **21**(1), 79–109.
- Bera, A. K., and Higgins, M. L. (1993) A survey of Arch models. *Journal of Economic Surveys*, **7**, 305–366. [Reprinted in Jarrow (1998).]
- Bollerslev, T. (1986) Generalized autoregressive conditional heteroskedasticity. *Journal of Econometrics*, **31**, 307–327.
- Bollerslev, T. (1990) Modelling the coherence in short-run nominal exchange rates: a multivariate generalized ARCH model. *The Review of Economics and Statistics*, **72**(3), 498–505.
- Bollerslev, T., Chou, R. Y., and Kroner, K. F. (1992) ARCH modelling in finance. *Journal of Econometrics*, **52**, 5–59. [Reprinted in Jarrow (1998)]
- Bollerslev, T., Engle, R. F., and Nelson, D. B. (1994) ARCH models, In *Handbook of Econometrics, Vol IV*, Engle, R.F., and McFadden, D.L., Elsevier, Amsterdam.
- Bollerslev, T., Engle, R. F., and Wooldridge, J. M. (1988) A capital asset pricing model with time-varying covariances. *Journal of Political Economy*, **96**, 116–131.
- Carroll, R. J., and Ruppert, D. (1988) *Transformation and Weighting in Regression*, Chapman & Hall, New York.
- Duan, J.-C. (1995) The GARCH option pricing model. *Mathematical Finance*, **5**, 13–32. [Reprinted in Jarrow (1998).]
- Duan, J.-C., and Simonato, J. G. (2001) American option pricing under GARCH by a Markov chain approximation. *Journal of Economic Dynamics and Control*, **25**, 1689–1718.
- Enders, W. (2004) *Applied Econometric Time Series*, 2nd ed., Wiley, New York.
- Engle, R. F. (1982) Autoregressive conditional heteroskedasticity with estimates of variance of U.K. inflation. *Econometrica*, **50**, 987–1008.
- Engle, R. F. (2002) Dynamic conditional correlation: A simple class of multivariate generalized autoregressive conditional heteroskedasticity models. *Journal of Business & Economic Statistics*, **20**(3), 339–350.
- Fisher, T.J., and Gallagher, C.M. (2012) New weighted portmanteau statistics for time series goodness of fit testing. *Journal of the American Statistical Association*, **107**(498), 777–787.
- Gouriéroux, C. and Jasiak, J. (2001) *Financial Econometrics*, Princeton University Press, Princeton, NJ.

- Hamilton, J. D. (1994) *Time Series Analysis*, Princeton University Press, Princeton, NJ.
- Heston, S. and Nandi, S. (2000) A closed form GARCH option pricing model. *The Review of Financial Studies*, **13**, 585–625.
- Hsieh, K. C. and Ritchken, P. (2000) An empirical comparison of GARCH option pricing models. working paper.
- Jarrow, R. (1998) *Volatility: New Estimation Techniques for Pricing Derivatives*, Risk Books, London. (This is a collection of articles, many on GARCH models or on stochastic volatility models, which are related to GARCH models.)
- Li, W. K. (2003) *Diagnostic checks in time series*, CRC Press.
- Matteson, D. S. and Tsay, R. S. (2011) Dynamic orthogonal components for multivariate time series. *Journal of the American Statistical Association*, **106**(496), 1450–1463.
- Palm, F.C. (1996) GARCH models of volatility. *Handbook of Statistics*, **14**, 209–240.
- Palma, W. and Zevallos, M. (2004). Analysis of the correlation structure of square time series. *Journal of Time Series Analysis*, **25**(4), 529–550.
- Pindyck, R. S. and Rubinfeld, D. L. (1998) *Econometric Models and Economic Forecasts*, Irwin/McGraw Hill, Boston.
- Ritchken, P. and Trevor, R. (1999) Pricing options under generalized GARCH and stochastic volatility processes. *Journal of Finance*, **54**, 377–402.
- Rossi, P. E. (1996) *Modelling Stock Market Volatility*, Academic Press, San Diego.
- Silvennoinen, A. and Teräsvirta, T (2009) Multivariate GARCH models. In *Handbook of Financial Time Series*, 201–229, Springer, Berlin.
- Tsay, R. S. (2005) *Analysis of Financial Time Series*, 2nd ed., Wiley, New York.
- Tse, Y. K. and Tsui, A. K. C. (2002) A multivariate generalized autoregressive conditional heteroscedasticity model with time-varying correlations. *Journal of Business & Economic Statistics*, **20**(3), 351–362.

## Cointegration

### 15.1 Introduction

Cointegration analysis is a technique that is frequently applied in econometrics. In finance it can be used to find trading strategies based on mean-reversion.

Suppose one could find a stock whose price (or log-price) series was stationary and therefore mean-reverting. This would be a wonderful investment opportunity. Whenever the price was below the mean, one could buy the stock and realize a profit when the price returned to the mean. Similarly, one could realize profits by selling short whenever the price was above the mean. Alas, returns are stationary but not prices. We have seen that log-prices are integrated. However, not all is lost. Sometimes one can find two or more assets with prices so closely connected that a linear combination of their prices is stationary. Then, a portfolio with weights assigned by the *cointegrating vector*, which is the vector of coefficients of this linear combination, will have a stationary price. Cointegration analysis is a means for finding cointegration vectors.

Two time series,  $Y_{1,t}$  and  $Y_{2,t}$ , are cointegrated if each is  $I(1)$  but there exists a  $\lambda$  such that  $Y_{1,t} - \lambda Y_{2,t}$  is stationary. For example, the common trends model is that

$$Y_{1,t} = \beta_1 W_t + \epsilon_{1,t},$$

$$Y_{2,t} = \beta_2 W_t + \epsilon_{2,t},$$

where  $\beta_1$  and  $\beta_2$  are nonzero, the trend  $W_t$  common to both series is  $I(1)$ , and the noise processes  $\epsilon_{1,t}$  and  $\epsilon_{2,t}$  are  $I(0)$ . Because of the common trend,

$Y_{1,t}$  and  $Y_{2,t}$  are nonstationary but there is a linear combination of these two series that is free of the trend, so they are cointegrated. To see this, note that if  $\lambda = \beta_1/\beta_2$ , then

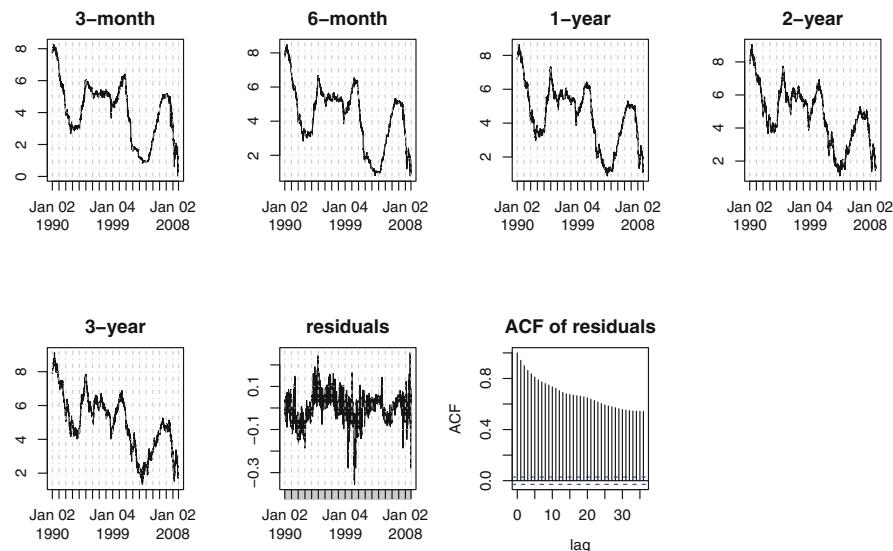
$$\beta_2(Y_{1,t} - \lambda Y_{2,t}) = \beta_2 Y_{1,t} - \beta_1 Y_{2,t} = \beta_2 \epsilon_{1,t} - \beta_1 \epsilon_{2,t} \quad (15.1)$$

is free of the trend  $W_t$ , and therefore is  $I(0)$ .

The definition of cointegration extends to more than two time series. A  $d$ -dimensional multivariate time series is cointegrated of order  $r$  if the component series are  $I(1)$  but  $r$  independent linear combinations of the components are  $I(0)$  for some  $r$ ,  $0 < r \leq d$ . Somewhat different definitions of cointegration exist, but this one is best for our purposes.

In Sect. 13.3.1 we saw the danger of spurious regression when the residuals are integrated. This problem should make one cautious about regression with nonstationary time series. However, if  $Y_t$  is regressed on  $X_t$  and the two series are cointegrated, then the residuals will be  $I(0)$  so that the least-squares estimator will be consistent.

The Phillips–Ouliaris cointegration test regresses one integrated series on others and applies the Phillips–Perron unit root test to the residuals. The null hypothesis is that the residuals are unit root nonstationary, which implies that the series are *not* cointegrated. Therefore, a small  $p$ -value implies that the series *are* cointegrated and therefore suitable for regression analysis. The residuals will still be correlated and so they should be modeled as such; see Sect. 13.3.3.



**Fig. 15.1.** Time series plots of the five yields and the residuals from a regression of the 1-year yields on the other four yields. Also, a sample ACF plot of the residuals.

*Example 15.1. Phillips–Ouliaris test on bond yields*

This example uses three-month, six-month, one-year, two-year, and three-year bond yields recorded daily from January 2, 1990 to October 31, 2008, for a total of 4,714 observations. The five yield series are plotted in Fig. 15.1, and one can see that they track each other somewhat closely. This suggests that the five series may be cointegrated. The one-year yields were regressed on the four others and the residuals and their ACF are also plotted in Fig. 15.1. The two residual plots are ambiguous about whether the residuals are stationary, so a test of cointegration would be helpful.

```

1 library(forecast)
2 library(tseries)
3 library(urca)
4 library(xts)
5 yieldDat = read.table("treasury_yields.txt", header=T)
6 date = as.Date(yieldDat[,1], format = "%m/%d/%y")
7 dat = as.xts(yieldDat[,3:7], date)
8 res = residuals(lm(dat[,3]~dat[,1]+dat[,2]+dat[,4]+dat[,5]))
```

Next, the Phillips–Ouliaris test was run using the R function `po.test()` in the `tseries` package.

```

9 po.test(dat[,c(3,1,2,4,5)])
Phillips-Ouliaris Cointegration Test

data:  dat[, c(3, 1, 2, 4, 5)]
Phillips-Ouliaris demeaned = -323.546, Truncation lag
parameter = 47, p-value = 0.01

Warning message:
In po.test(dat[, c(3, 1, 2, 4, 5)]) : p-value smaller
than printed p-value
```

The  $p$ -value is computed by interpolation if it is within the range of a table in Phillips and Ouliaris (1990). In this example, the  $p$ -value is outside the range and we know only that it is below 0.01, the lower limit of the table. The small  $p$ -value leads to the conclusion that the residuals are stationary and so the five series are cointegrated.

Though stationary, the residuals have a large amount of autocorrelation and may have long-term memory. They take a long time to revert to their mean of zero. Devising a profitable trading strategy from these yields seems problematic.  $\square$

## 15.2 Vector Error Correction Models

The regression approach to cointegration is somewhat unsatisfactory, since one series must be chosen as the dependent variable, and this choice must be

somewhat arbitrary. In Example 15.1, the middle yield, ordered by maturity, was used but for no compelling reason. Moreover, regression will find only one cointegration vector, but there could be more than one.

An alternative approach to cointegration that treats the series symmetrically uses a *vector error correction model* (VECM). In these models, the deviation from the mean is called the “error” and whenever the stationary linear combination deviates from its mean, it is subsequently pushed back toward its mean (the error is “corrected”).

The idea behind error correction is simplest when there are only two series,  $Y_{1,t}$  and  $Y_{2,t}$ . In this case, the error correction model is

$$\Delta Y_{1,t} = \phi_1(Y_{1,t-1} - \lambda Y_{2,t-1}) + \epsilon_{1,t}, \quad (15.2)$$

$$\Delta Y_{2,t} = \phi_2(Y_{1,t-1} - \lambda Y_{2,t-1}) + \epsilon_{2,t}, \quad (15.3)$$

where  $\epsilon_{1,t}$  and  $\epsilon_{2,t}$  are white noise. Subtracting  $\lambda$  times (15.3) from (15.2) gives

$$\Delta(Y_{1,t} - \lambda Y_{2,t}) = (\phi_1 - \lambda\phi_2)(Y_{1,t-1} - \lambda Y_{2,t-1}) + (\epsilon_{1,t} - \lambda\epsilon_{2,t}). \quad (15.4)$$

Let  $\mathcal{F}_t$  denote the information set at time  $t$ . If  $(\phi_1 - \lambda\phi_2) < 0$ , then  $E\{\Delta(Y_{1,t} - \lambda Y_{2,t})|\mathcal{F}_{t-1}\}$  is opposite in sign to  $Y_{1,t-1} - \lambda Y_{2,t-1}$ . This causes error correction because whenever  $Y_{1,t-1} - \lambda Y_{2,t-1}$  is positive, its expected change is negative and vice versa.

A rearrangement of (15.4) shows that  $Y_{1,t-1} - \lambda Y_{2,t-1}$  is an AR(1) process with coefficient  $1 + \phi_1 - \lambda\phi_2$ . Therefore, the series  $Y_{1,t} - \lambda Y_{2,t}$  is  $I(0)$ , unit-root nonstationary, or an explosive series in the cases where  $|1 + \phi_1 - \lambda\phi_2|$  is less than 1, equal to 1, and greater than 1, respectively.

- If  $\phi_1 - \lambda\phi_2 > 0$ , then  $1 + \phi_1 - \lambda\phi_2 > 1$  and  $Y_{1,t} - \lambda Y_{2,t}$  is explosive.
- If  $\phi_1 - \lambda\phi_2 = 0$ , then  $1 + \phi_1 - \lambda\phi_2 = 1$  and  $Y_{1,t} - \lambda Y_{2,t}$  is a random walk.
- If  $\phi_1 - \lambda\phi_2 < 0$ , then  $1 + \phi_1 - \lambda\phi_2 < 1$  and  $Y_{1,t} - \lambda Y_{2,t}$  is stationary, unless  $\phi_1 - \lambda\phi_2 \leq -2$ , so that  $1 + \phi_1 - \lambda\phi_2 \leq -1$ .

The case  $\phi_1 - \lambda\phi_2 \leq -2$  is “over-correction.” The change in  $Y_{1,t} - \lambda Y_{2,t}$  is in the correct direction but too large, so the series oscillates in sign but diverges to  $\infty$  in magnitude.

### *Example 15.2. Simulation of an error correction model*

Model (15.2)–(15.3) was simulated with  $\phi_1 = 0.5$ ,  $\phi_2 = 0.55$ , and  $\lambda = 1$ . A total of 5,000 observations were simulated, but, for visual clarity, only every 10th observation is plotted in Fig. 15.2. Neither  $Y_{1,t}$  nor  $Y_{2,t}$  is stationary, but  $Y_{1,t} - \lambda Y_{2,t}$  is stationary. Notice how closely  $Y_{1,t}$  and  $Y_{2,t}$  track one another.

```
10 n = 5000
11 set.seed(12345)
12 a1 = 0.5
```

```

13 a2 = 0.55
14 lambda = 1
15 y1 = rep(0,n)
16 y2 = y1
17 e1 = rnorm(n)
18 e2 = rnorm(n)
19 for (i in 2:n){
20   y1[i] = y1[i-1] + a1 * (y1[i-1] - lambda*y2[i-1]) + e1[i]
21   y2[i] = y2[i-1] + a2 * (y1[i-1] - lambda*y2[i-1]) + e2[i]
22 }

```

□

To see how to generalize error correction to more than two series, it is useful to rewrite Eqs. (15.2) and (15.3) in vector form. Let  $\mathbf{Y}_t = (Y_{1,t}, Y_{2,t})'$  and  $\boldsymbol{\epsilon}_t = (\epsilon_{1,t}, \epsilon_{2,t})'$ . Then

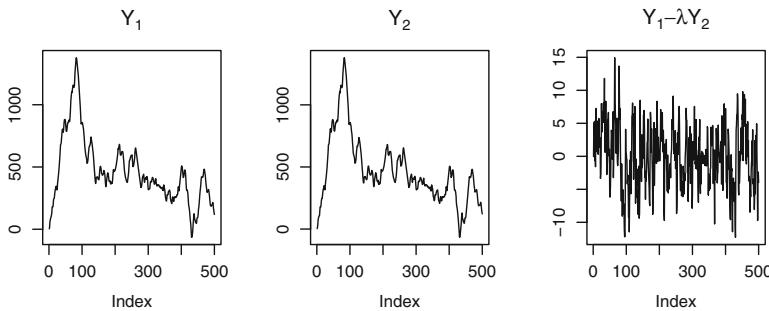
$$\Delta \mathbf{Y}_t = \boldsymbol{\alpha} \boldsymbol{\beta}' \mathbf{Y}_{t-1} + \boldsymbol{\epsilon}_t, \quad (15.5)$$

where

$$\boldsymbol{\alpha} = \begin{pmatrix} \phi_1 \\ \phi_2 \end{pmatrix} \quad \text{and} \quad \boldsymbol{\beta} = \begin{pmatrix} 1 \\ -\lambda \end{pmatrix}, \quad (15.6)$$

so that  $\boldsymbol{\beta}$  is the cointegration vector, and  $\boldsymbol{\alpha}$  specifies the speed of mean-reversion and is called the *loading matrix* or *adjustment matrix*.

Model (15.5) also applies when there are  $d$  series such that  $\mathbf{Y}_t$  and  $\boldsymbol{\epsilon}_t$  are  $d$ -dimensional. In this case  $\boldsymbol{\beta}$  and  $\boldsymbol{\alpha}$  are each full-rank  $d \times r$  matrices for



**Fig. 15.2.** Simulation of an error correction model. 5,000 observations were simulated but only every 10th is plotted.

some  $r \leq d$  which is the number of linearly independent cointegration vectors. The columns of  $\boldsymbol{\beta}$  are the cointegration vectors.

Model (15.5) is a vector AR(1) [that is, VAR(1)] model but, for added flexibility, can be extended to a VAR( $p$ ) model, and there are several ways to do this. We will use the notation and the second of two forms of the VECM from the function `ca.jo()` in R's `urca` package. This VECM is

$$\Delta \mathbf{Y}_t = \boldsymbol{\Gamma}_1 \Delta \mathbf{Y}_{t-1} + \cdots + \boldsymbol{\Gamma}_{p-1} \Delta \mathbf{Y}_{t-p+1} + \boldsymbol{\Pi} \mathbf{Y}_{t-1} + \boldsymbol{\mu} + \boldsymbol{\Phi} \mathbf{D}_t + \boldsymbol{\epsilon}_t, \quad (15.7)$$

where  $\boldsymbol{\mu}$  is a mean vector,  $\mathbf{D}_t$  is a vector of nonstochastic regressors, and

$$\boldsymbol{\Pi} = \boldsymbol{\alpha} \boldsymbol{\beta}' . \quad (15.8)$$

As before,  $\boldsymbol{\beta}$  and  $\boldsymbol{\alpha}$  are each full-rank  $d \times r$  matrices and  $\boldsymbol{\alpha}$  is called the loading matrix.

It is easy to show that the columns of  $\boldsymbol{\beta}$  are the cointegration vectors. Since  $\mathbf{Y}_t$  is  $I(1)$ ,  $\Delta \mathbf{Y}_t$  on the left-hand side of (15.7) is  $I(0)$  and therefore  $\boldsymbol{\Pi} \mathbf{Y}_{t-1} = \boldsymbol{\alpha} \boldsymbol{\beta}' \mathbf{Y}_{t-1}$  on the right-hand side of (15.7) is also  $I(0)$ . It follows that each of the  $r$  components of  $\boldsymbol{\beta}' \mathbf{Y}_{t-1}$  is  $I(0)$ .

### *Example 15.3. VECM test on bond yields*

A VECM was fit to the bond yields using R's `ca.jo()` function. The output is below. The eigenvalues are used to test null hypotheses of the form  $H_0: r \leq r_0$ . The values of the test statistics and critical values (for 1%, 5%, and 10% level tests) are listed below the eigenvalues. The null hypothesis is rejected when the test statistic exceeds the critical level. In this case, regardless of whether one uses a 1%, 5%, or 10% level test, one accepts that  $r$  is less than or equal to 3 but rejects that  $r$  is less than or equal to 2, so one concludes that  $r = 3$ . Although five cointegration vectors are printed, only the first three would be meaningful. The cointegration vectors are the columns of the matrix labeled "Eigenvectors, normalized to first column." The cointegration vectors are determined only up to multiplication by a nonzero scalar and so can be normalized so that their first element is 1.

```
#####
# Johansen-Procedure #
#####

Test type: maximal eigenvalue statistic (lambda max),
with linear trend

Eigenvalues (lambda):
[1] 0.03436 0.02377 0.01470 0.00140 0.00055

Values of test statistic and critical values of test:

            test 10pct 5pct 1pct
r <= 4 |   2.59   6.5  8.18 11.6
r <= 3 |   6.62  12.9 14.90 19.2
r <= 2 |  69.77  18.9 21.07 25.8
r <= 1 | 113.36  24.8 27.14 32.1
r = 0  | 164.75  30.8 33.32 38.8

Eigenvectors, normalised to first column:
```

(These are the cointegration relations)

|         | X3mo.12 | X6mo.12 | X1yr.12 | X2yr.12 | X3yr.12 |
|---------|---------|---------|---------|---------|---------|
| X3mo.12 | 1.000   | 1.00    | 1.00    | 1.0000  | 1.000   |
| X6mo.12 | -1.951  | 2.46    | 1.07    | 0.0592  | 0.897   |
| X1yr.12 | 1.056   | 14.25   | -3.95   | -2.5433 | -1.585  |
| X2yr.12 | 0.304   | -46.53  | 3.51    | -3.4774 | -0.118  |
| X3yr.12 | -0.412  | 30.12   | -1.71   | 5.2322  | 1.938   |

Weights W:

(This is the loading matrix)

|        | X3mo.12  | X6mo.12   | X1yr.12   | X2yr.12   | X3yr.12   |
|--------|----------|-----------|-----------|-----------|-----------|
| X3mo.d | -0.03441 | -0.002440 | -0.011528 | -0.000178 | -0.000104 |
| X6mo.d | 0.01596  | -0.002090 | -0.007066 | 0.000267  | -0.000170 |
| X1yr.d | -0.00585 | -0.001661 | -0.001255 | 0.000358  | -0.000289 |
| X2yr.d | 0.00585  | -0.000579 | -0.003673 | -0.000072 | -0.000412 |
| X3yr.d | 0.01208  | -0.000985 | -0.000217 | -0.000431 | -0.000407 |

□

## 15.3 Trading Strategies

As discussed previously, price series that are cointegrated can be used in *statistical arbitrage*. Unlike pure arbitrage, statistical arbitrage means an opportunity where a profit is only likely, not guaranteed. Pairs trading uses pairs of cointegrated asset prices and has been a popular statistical arbitrage technique. Pairs trading requires the trader to find cointegrated pairs of assets, to select from these the pairs that can be traded profitably after accounting for transaction costs, and finally to design a trading strategy which includes the buy and sell signals. A full discussion of statistical arbitrage is outside the scope of this book, but see Sect. 15.4 for further reading.

Although many firms have been very successful using statistical arbitrage, one should be mindful of the risks. One is model risk; the error-correction model may be incorrect. Even if the model is correct, one must use estimates based on past data and the parameters might change, perhaps rapidly. If statistical arbitrage opportunities exist, then it is possible that other traders have discovered them and their trading activity is one reason to expect parameters to change. Another risk is that one can go bankrupt before a stationary process reverts to its mean. This risk is especially large because firms engaging in statistical arbitrage are likely to be heavily leveraged. High leverage will magnify a small loss caused when a process diverges even farther from its mean before reverting. See Sects. 2.4.2 and 15.5.4.

## 15.4 Bibliographic Notes

Alexander (2001), Enders (2004), and Hamilton (1994) contain useful discussions of cointegration. Pfaff (2006) is a good introduction to the analysis of cointegrated time series using R.

The MLEs and likelihood ratio tests of the parameters in (15.7) were developed by Johansen (1991), Johansen (1995) and Johansen and Juselius (1990).

The applications of cointegration theory in statistical arbitrage are discussed by Vidyamurthy (2004) and Alexander, Giblin, and Weddington (2001). Pole (2007) is a less technical introduction to statistical arbitrage.

## 15.5 R Lab

### 15.5.1 Cointegration Analysis of Midcap Prices

The data set `midcapD.ts.csv` has daily log returns on 20 midcap stocks in columns 2–21. Columns 1 and 22 contain the date and market returns, respectively. In this section, we will use returns on the first 10 stocks. To find the stock prices from the returns, we use the relationship

$$P_t = P_0 \exp(r_1 + \cdots + r_t),$$

where  $P_t$  and  $r_t$  are the price and log return at time  $t$ . The returns will be used as approximations to the log returns. The prices at time 0 are unknown, so we will use  $P_0 = 1$  for each stock. This means that the price series we use will be off by multiplicative factors. This does not affect the number of cointegration vectors. If we find that there are cointegration relationships, then it would be necessary to get the price data to investigate trading strategies.

Johansen's cointegration analysis will be applied to the prices with the `ca.jo()` function in the `urca` package. Run

```

1 library(urca)
2 midcapD.ts = read.csv("midcapD.ts.csv", header=T)
3 x = midcapD.ts[,2:11]
4 prices= exp(apply(x,2,cumsum))
5 options(digits=3)
6 summary(ca.jo(prices))
```

**Problem 1** How many cointegration vectors were found?

### 15.5.2 Cointegration Analysis of Yields

This example is similar to Example 15.3 but uses different yield data. The data are in the `mk.zero2.csv` data set. There are 55 maturities and they are in the vector `mk.maturity`. We will use only the first 10 yields. Run the following commands in R.

```

1 library(urca)
2 mk.maturity = read.csv("mk.zero2.csv", header=T)
3 summary(ca.jo(mk.maturity[,2:11]))

```

**Problem 2** What maturities are being used? Are they short-, medium-, or long-term, or a mixture of short- and long-term maturities?

**Problem 3** How many cointegration vectors were found? Use 1 % level tests.

### 15.5.3 Cointegration Analysis of Daily Stock Prices

The CokePepsi.csv data set contains the adjusted daily closing prices of Coke and Pepsi stock from January 2007 to November 2012. Run the following commands in R.

```

1 CokePepsi = read.table("CokePepsi.csv", header=T)
2 ts.plot(CokePepsi)

```

**Problem 4** Do these two series appear cointegrated from the time series plot? Why?

Now make a time series plot of the difference between the two prices.

```
3 ts.plot(CokePepsi[,2] - CokePepsi[,1])
```

**Problem 5** Does this difference series appear stationary? Why?

Run the following commands to conduct Johansen's cointegration test.

```

4 library(urca)
5 summary(ca.jo(CokePepsi))

```

**Problem 6** Are these two series cointegrated? Why?

Now consider the daily adjusted closing prices for 10 company stocks from January 2, 1987 to September 1, 2006 from the Stock\_FX\_Bond.csv dataset.

```

6 Stock_FX_Bond = read.csv("Stock_FX_Bond.csv", header=T)
7 adjClose = Stock_FX_Bond[,seq(from=3, to=21, by=2)]
8 ts.plot(adjClose)
9 summary(ca.jo(adjClose))

```

**Problem 7** Are these 10 stock price series cointegrated? If so, what is the rank of the cointegrating matrix, and what are the cointegrating vectors?

Rerun the Johansen's cointegration test with lag K = 8.

```
10 summary(ca.jo(adjClose, K=8))
```

**Problem 8** Are these 10 stock price series cointegrated if Johansen's cointegration test is conducted with lag K = 8? If so, has the estimated rank of the cointegrating matrix or the cointegrating vectors changed?

### 15.5.4 Simulation

In this section, you will run simulations similar to those in Sect. 2.4.2. The difference is that now the price process is mean-reverting.

Suppose a hedge fund owns a \$1,000,000 position in a portfolio and used \$50,000 of its own capital and \$950,000 in borrowed money for the purchase. If the value of the portfolio falls below \$950,000 at the end of any trading day, then the hedge fund must liquidate and repay the loan.

The portfolio was selected by cointegration analysis and its price is an AR(1) process,

$$(P_t - \mu) = \phi(P_{t-1} - \mu) + \epsilon_t,$$

where  $P_t$  is the price of the portfolio at the end of trading day  $t$ ,  $\mu = \$1,030,000$ ,  $\phi = 0.99$ , and the standard deviation of  $\epsilon_t$  is \$5000. The hedge fund knows that the price will eventually revert to \$1,030,000 (assuming that the model is correct and, of course, this is a big assumption). It has decided to liquidate its position on day  $t$  if  $P_t \geq \$1,020,000$ . This will yield a profit of at least \$20,000. However, if the price falls below \$950,000, then it must liquidate and lose its entire \$50,000 investment plus the difference between \$950,000 and the price at liquidation.

In summary, the hedge fund will liquidate at the end of the first day such that the price is either above \$1,020,000 or below \$950,000. In the first case, it will achieve a profit of at least \$20,000 and in the second case it will suffer a loss of at least \$50,000. Presumably, the probability of a loss is small, and we will see how small by simulation.

Run a simulation experiment similar to the one in Sect. 2.4.2 to answer the following questions. Use 10,000 simulations.

**Problem 9** *What is the expected profit?*

**Problem 10** *What is the probability that the hedge fund will need to liquidate for a loss?*

**Problem 11** *What is the expected waiting time until the portfolio is liquidated?*

**Problem 12** *What is the expected yearly return on the \$50,000 investment?*

## 15.6 Exercises

1. Show that (15.4) implies that  $Y_{1,t-1} - \lambda Y_{2,t-1}$  is an AR(1) process with coefficient  $1 + \phi_1 - \lambda\phi_2$ .

2. In (15.2) and (15.3) there are no constants, so that  $Y_{1,t} - \lambda Y_{2,t}$  is a stationary process with mean zero. Introduce constants into (15.2) and (15.3) and show how they determine the mean of  $Y_{1,t} - \lambda Y_{2,t}$ .
3. Verify that in Example 15.2  $Y_{1,t} - \lambda Y_{2,t}$  is stationary.
4. Suppose that  $\mathbf{Y}_t = (Y_{1,t}, Y_{2,t})'$  is the bivariate AR(1) process in Example 15.2. Is  $\mathbf{Y}_t$  stationary? (Hint: See Sect. 13.4.4.)

## References

- Alexander, C. (2001) *Market Models: A Guide to Financial Data Analysis*, Wiley, Chichester.
- Alexander, C., Giblin, I., and Weddington, W. III (2001) *Cointegration and Asset Allocation: A New Hedge Fund*, ISMA Discussion Centre Discussion Papers in Finance 2001–2003.
- Enders, W. (2004) *Applied Econometric Time Series*, 2nd ed., Wiley, New York.
- Hamilton, J. D. (1994) *Time Series Analysis*, Princeton University Press, Princeton, NJ.
- Johansen, S. (1991) Estimation and hypothesis testing of cointegration vectors in gaussian vector autoregressive models. *Econometrica*, **59**, 1551–1580.
- Johansen, S. (1995) *Likelihood-Based Inference in Cointegrated Vector Autoregressive Models*, Oxford University Press, New York.
- Johansen, S., and Juselius, K. (1990) Maximum likelihood estimation and inference on cointegration — With applications to the demand for money. *Oxford Bulletin of Economics and Statistics*, **52**, 2, 169–210.
- Pfaff, B. (2006) *Analysis of Integrated and Cointegrated Time Series with R*, Springer, New York.
- Phillips, P. C. B., and Ouliaris, S. (1990) Asymptotic properties of residual based tests for cointegration. *Econometrica*, **58**, 165–193.
- Pole, A. (2007) *Statistical Arbitrage*, Wiley, Hoboken, NJ.
- Vidyamurthy, G. (2004) *Pairs Trading*, Wiley, Hoboken, NJ.

## Portfolio Selection

### 16.1 Trading Off Expected Return and Risk

How should we invest our wealth? Portfolio theory provides an answer to this question based upon two principles:

- we want to maximize the expected return; and
- we want to minimize the risk, which we define in this chapter to be the standard deviation of the return, though we may ultimately be concerned with the probabilities of large losses.

These goals are somewhat at odds because riskier assets generally have a higher expected return, since investors demand a reward for bearing risk. The difference between the expected return of a risky asset and the risk-free rate of return is called the *risk premium*. Without risk premiums, few investors would invest in risky assets.

Nonetheless, there are optimal compromises between expected return and risk. In this chapter we show how to maximize expected return subject to an upper bound on the risk, or to minimize the risk subject to a lower bound on the expected return. One key concept that we discuss is reduction of risk by diversifying the portfolio.

### 16.2 One Risky Asset and One Risk-Free Asset

We start with a simple example with one risky asset, which could be a portfolio, for example, a mutual fund. Assume that the expected return is 0.15 and the standard deviation of the return is 0.25. Assume that there is a *risk-free asset*, such as, a 90-day T-bill, and the risk-free rate is 6 %, so the return on the risk-free asset is 6 %, or 0.06. The standard deviation of the return on the

risk-free asset is 0 by definition of “risk-free.” The rates and returns here are annual, though all that is necessary is that they be in the same time units.

We are faced with the problem of constructing an investment portfolio that we will hold for one time period, which is called the *holding period* and which could be a day, a month, a quarter, a year, 10 years, and so forth. At the end of the holding period we might want to readjust the portfolio, so for now we are only looking at returns over one time period. Suppose that a fraction  $w$  of our wealth is invested in the risky asset and the remaining fraction  $1 - w$  is invested in the risk-free asset. Then the expected return is

$$E(R) = w(0.15) + (1 - w)(0.06) = 0.06 + 0.09w, \quad (16.1)$$

the variance of the return is

$$\sigma_R^2 = w^2 (0.25)^2 + (1 - w)^2 (0)^2 = w^2(0.25)^2,$$

and the standard deviation of the return is

$$\sigma_R = 0.25 |w|. \quad (16.2)$$

As will be discussed later,  $w$  is negative if the risky asset is sold short, so we have  $|w|$  rather than  $w$  in (16.2).

To decide what proportion  $w$  of one’s wealth to invest in the risky asset, one chooses either the expected return  $E(R)$  one wants or the amount of risk  $\sigma_R$  with which one is willing to live. Once either  $E(R)$  or  $\sigma_R$  is chosen,  $w$  can be determined.

Although  $\sigma$  is a measure of risk, a more direct measure of risk is actual monetary loss. In the next example,  $w$  is chosen to control the maximum size of the loss.

### *Example 16.1. Finding $w$ to achieve a targeted value-at-risk*

Suppose that a firm is planning to invest \$1,000,000 and has capital reserves that could cover a loss of \$150,000 but no more. Therefore, the firm would like to be certain that, if there is a loss, then it is no more than 15%, that is, that  $R$  is greater than  $-0.15$ . Suppose that  $R$  is normally distributed. Then the only way to guarantee that  $R$  is greater than  $-0.15$  with probability equal to 1 is to invest entirely in the risk-free asset. The firm might instead be more modest and require only that  $P(R < -0.15)$  be small, for example, 0.01. Therefore, the firm should find the value of  $w$  such that

$$P(R < -0.15) = \Phi\left(\frac{-0.15 - (0.06 + 0.09w)}{0.25w}\right) = 0.01.$$

The solution is

$$w = \frac{-0.21}{0.25\Phi^{-1}(0.01) + 0.9} = 0.4264.$$

The value of  $\Phi^{-1}(0.01)$  is calculated by `qnorm(0.01)` and is  $-2.33$ .

In Chap. 19, \$150,000 is called the value-at-risk (= VaR) and  $1 - 0.01 = 0.99$  is called the confidence coefficient. What was done in this example is to find the portfolio that has a VaR of \$150,000 with 0.99 confidence.

We saw in Chap. 5 that the distributions of stock returns usually have much heavier tails than a normal distribution. In Chap. 19, VaR is estimated under more realistic assumptions, e.g., that the returns are  $t$ -distributed.  $\square$

More generally, if the expected returns on the risky and risk-free assets are  $\mu_1$  and  $\mu_f$  and if the standard deviation of the risky asset is  $\sigma_1$ , then the expected return on the portfolio is  $w\mu_1 + (1 - w)\mu_f$  while the standard deviation of the portfolio's return is  $|w|\sigma_1$ .

This model is simple but not as useless as it might seem at first. As discussed later, finding an optimal portfolio can be achieved in two steps:

1. finding the “optimal” portfolio of risky assets, called the “tangency portfolio,” and
2. finding the appropriate mix of the risk-free asset and the tangency portfolio.

So we now know how to do the second step. What we still need to learn is how find the tangency portfolio.

### 16.2.1 Estimating $E(R)$ and $\sigma_R$

The value of the risk-free rate,  $\mu_f$ , will be known since Treasury bill rates are published in sources providing financial information.

What should we use as the values of  $E(R)$  and  $\sigma_R$ ? If returns on the asset are assumed to be stationary, then we can take a time series of past returns and use the sample mean and standard deviation. Whether the stationarity assumption is realistic is always debatable. If we think that  $E(R)$  and  $\sigma_R$  will be different from the past, we could subjectively adjust these estimates upward or downward according to our opinions, but we must live with the consequences if our opinions prove to be incorrect. Also, the sample mean and standard deviation are not particularly accurate and could be replaced by estimates from a factor model such as the CAPM or the Fama-French model; see Chaps. 17, 18, and 20.

Another question is how long a time series to use, that is, how far back in time one should gather data. A long series, say 10 or 20 years, will give much less variable estimates. However, if the series is not stationary but rather has slowly drifting parameters, then a shorter series (maybe 1 or 2 years) will be more representative of the future. Almost every time series of returns is nearly stationary over short enough time periods.

Even if the time series is stationary, it is likely to exhibit volatility clustering. In that case one might use a GARCH estimate of the conditional standard deviation of the return over the holding period. See Chap. 14.

## 16.3 Two Risky Assets

### 16.3.1 Risk Versus Expected Return

The mathematics of mixing risky assets is most easily understood when there are only two risky assets. This is where we start.

Suppose the two risky assets have returns  $R_1$  and  $R_2$  and that we mix them in proportions  $w$  and  $1 - w$ , respectively. The return on the portfolio is  $R_p = wR_1 + (1 - w)R_2$ . The expected return on the portfolio is  $E(R_p) = w\mu_1 + (1 - w)\mu_2$ . Let  $\rho_{12}$  be the correlation between the returns on the two risky assets. The variance of the return on the portfolio is

$$\sigma_p^2 = w^2\sigma_1^2 + (1 - w)^2\sigma_2^2 + 2w(1 - w)\rho_{12}\sigma_1\sigma_2. \quad (16.3)$$

Note that  $\rho_{12}\sigma_1\sigma_2 = \sigma_{R_1,R_2}$ .

*Example 16.2. The expectation and variance of the return on a portfolio with two risky assets*

Suppose that  $\mu_1 = 0.14$ ,  $\mu_2 = 0.08$ ,  $\sigma_1 = 0.2$ ,  $\sigma_2 = 0.15$ , and  $\rho_{12} = 0$ . Then

$$E(R_p) = 0.08 + 0.06w,$$

and because  $\rho_{12} = 0$  in this example,

$$\sigma_{R_p}^2 = (0.2)^2 w^2 + (0.15)^2 (1 - w)^2.$$

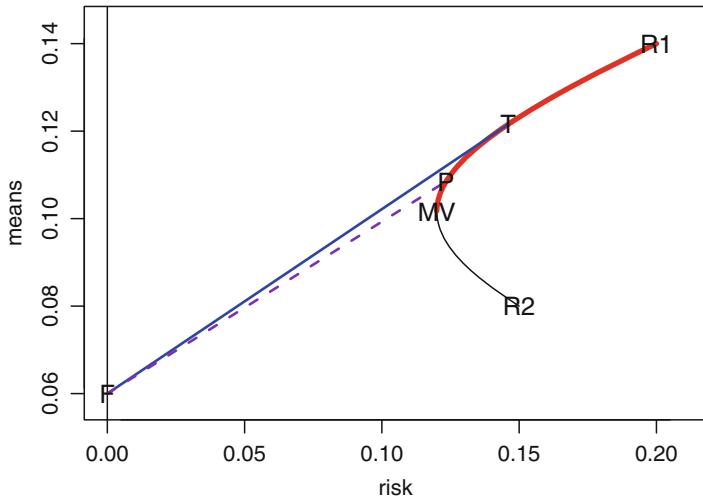
Using differential calculus, one can easily show that the portfolio with the minimum risk is  $w = 0.045/0.125 = 0.36$ . For this portfolio  $E(R_p) = 0.08 + (0.06)(0.36) = 0.1016$  and  $\sigma_{R_p} = \sqrt{(0.2)^2(0.36)^2 + (0.15)^2(0.64)^2} = 0.12$ .

The somewhat parabolic curve<sup>1</sup> in Fig. 16.1 is the locus of values of  $(\sigma_R, E(R))$  when  $0 \leq w \leq 1$ . The leftmost point on this locus achieves the minimum value of the risk and is called the *minimum variance portfolio*. The points on this locus that have an expected return at least as large as the minimum variance portfolio are called the *efficient frontier*. Portfolios on the efficient frontier are called *efficient portfolios* or, more precisely, *mean-variance efficient portfolios*.<sup>2</sup> The points labeled  $R_1$  and  $R_2$  correspond to  $w = 1$  and  $w = 0$ , respectively. The other features of this figure are explained in Sect. 16.4.  $\square$

---

<sup>1</sup> In fact, the curve would be parabolic if  $\sigma_p^2$  were plotted on the  $x$ -axis instead of  $\sigma_R$ .

<sup>2</sup> When a risk-free asset is available, then the efficient portfolios are no longer those on the efficient frontier but rather are characterized by Result 16.1 ahead.



**Fig. 16.1.** Expected return versus risk for Example 16.2.  $F$  = risk-free asset.  $T$  = tangency portfolio.  $R_1$  is the first risky asset.  $R_2$  is the second risky asset.  $MV$  is the minimum variance portfolio. The efficient frontier is the red curve. All points on the curve connecting  $R_2$  and  $R_1$  are attainable with  $0 \leq w \leq 1$ , but the ones on the black curve are suboptimal.  $P$  is a typical portfolio on the efficient frontier.

In practice, the mean and standard deviations of the returns can be estimated as discussed in Sect. 16.2.1 and the correlation coefficient can be estimated by the sample correlation coefficient. Alternatively, in Chaps. 18 and 20 factor models are used to estimate expected returns and the covariance matrix of returns.

## 16.4 Combining Two Risky Assets with a Risk-Free Asset

Our ultimate goal is to find optimal portfolios combining many risky assets with a risk-free asset. However, many of the concepts needed for this task can be first understood most easily when there are only two risky assets.

### 16.4.1 Tangency Portfolio with Two Risky Assets

As mentioned in Sect. 16.3.1, each point on the efficient frontier in Fig. 16.1 is  $(\sigma_{R_p}, E(R_p))$  for some value of  $w$  between 0 and 1. If we fix  $w$ , then we have a fixed portfolio of the two risky assets. Now let us mix that portfolio of risky assets with the risk-free asset. The point  $F$  in Fig. 16.1 gives  $(\sigma_{R_P}, E(R))$  for the risk-free asset; of course,  $\sigma_{R_P} = 0$  at  $F$ . The possible values of  $(\sigma_{R_P}, E(R_p))$  for a portfolio consisting of the fixed portfolio of two

risky assets and the risk-free asset is a line connecting the point F with a point on the efficient frontier, for example, the dashed purple line.

Notice that the solid blue line connecting F with the point labeled T lies above the dashed purple line connecting F and the typical portfolio. This means that for any value of  $\sigma_{R_P}$ , the solid blue line gives a higher expected return than the dashed purple line. The slope of each line is called its *Sharpe's ratio*, named after William Sharpe, whom we will meet again in Chap. 17. If  $E(R_P)$  and  $\sigma_{R_P}$  are the expected return and standard deviation of the return on a portfolio and  $\mu_f$  is the risk-free rate, then

$$\frac{E(R_P) - \mu_f}{\sigma_{R_P}} \quad (16.4)$$

is Sharpe's ratio of the portfolio. Sharpe's ratio can be thought of as a “reward-to-risk” ratio. It is the ratio of the reward quantified by the excess expected return<sup>3</sup> to the risk as measured by the standard deviation.

A line with a larger slope gives a higher expected return for a given level of risk, so the larger Sharpe's ratio, the better regardless of what level of risk one is willing to accept. The point T on the efficient frontier is the portfolio with the highest Sharpe's ratio. It is the optimal portfolio for the purpose of mixing with the risk-free asset. This portfolio is called the *tangency portfolio* since its line is tangent to the efficient frontier.

**Result 16.1** *The optimal or efficient portfolios mix the tangency portfolio with the risk-free asset. Each efficient portfolio has two properties:*

- *it has a higher expected return than any other portfolio with the same or smaller risk, and*
- *it has a smaller risk than any other portfolio with the same or higher expected return.*

Thus we can only improve (reduce) the risk of an efficient portfolio by accepting a worse (smaller) expected return, and we can only improve (increase) the expected return of an efficient portfolio by accepting worse (higher) risk.

Note that all efficient portfolios use the same mix of the two risky assets, namely, the tangency portfolio. Only the proportion allocated to the tangency portfolio and the proportion allocated to the risk-free asset vary.

Given the importance of the tangency portfolio, you may be wondering “how do we find it?” Again, let  $\mu_1$ ,  $\mu_2$ , and  $\mu_f$  be the expected returns on the two risky assets and the return on the risk-free asset. Let  $\sigma_1$  and  $\sigma_2$  be the standard deviations of the returns on the two risky assets and let  $\rho_{12}$  be the correlation between the returns on the risky assets.

---

<sup>3</sup> Here “excess” means in excess of the risk-free rate.

Define  $V_1 = \mu_1 - \mu_f$  and  $V_2 = \mu_2 - \mu_f$ , the excess expected returns. Then the tangency portfolio uses weight

$$w_T = \frac{V_1\sigma_2^2 - V_2\rho_{12}\sigma_1\sigma_2}{V_1\sigma_2^2 + V_2\sigma_1^2 - (V_1 + V_2)\rho_{12}\sigma_1\sigma_2} \quad (16.5)$$

for the first risky asset and weight  $(1 - w_T)$  for the second.

Let  $R_T$ ,  $E(R_T)$ , and  $\sigma_T$  be the return, expected return, and standard deviation of the return on the tangency portfolio. Then  $E(R_T)$  and  $\sigma_T$  can be found by first finding  $w_T$  using (16.5) and then using the formulas

$$E(R_T) = w_T\mu_1 + (1 - w_T)\mu_2$$

and

$$\sigma_T = \sqrt{w_T^2\sigma_1^2 + (1 - w_T)^2\sigma_2^2 + 2w_T(1 - w_T)\rho_{12}\sigma_1\sigma_2}.$$

*Example 16.3. The tangency portfolio with two risky assets*

Suppose as before that  $\mu_1 = 0.14$ ,  $\mu_2 = 0.08$ ,  $\sigma_1 = 0.2$ ,  $\sigma_2 = 0.15$ , and  $\rho_{12} = 0$ . Suppose as well that  $\mu_f = 0.06$ . Then  $V_1 = 0.14 - 0.06 = 0.08$  and  $V_2 = 0.08 - 0.06 = 0.02$ . Plugging these values into formula (16.5), we get  $w_T = 0.693$  and  $1 - w_t = 0.307$ . Therefore,

$$E(R_T) = (0.693)(0.14) + (0.307)(0.08) = 0.122,$$

and

$$\sigma_T = \sqrt{(0.693)^2(0.2)^2 + (0.307)^2(0.15)^2} = 0.146.$$

□

### 16.4.2 Combining the Tangency Portfolio with the Risk-Free Asset

Let  $R_p$  be the return on the portfolio that allocates a fraction  $\omega$  of the investment to the tangency portfolio and  $1 - \omega$  to the risk-free asset. Then  $R_p = \omega R_T + (1 - \omega)\mu_f = \mu_f + \omega(R_T - R_f)$ , so that

$$E(R_p) = \mu_f + \omega\{E(R_T) - \mu_f\} \quad \text{and} \quad \sigma_{R_p} = \omega\sigma_T.$$

*Example 16.4. (Continuation of Example 16.2 and 16.3)*

In this example, we will find the optimal investment with  $\sigma_{R_p} = 0.05$ .

The maximum expected return with  $\sigma_{R_p} = 0.05$  mixes the tangency portfolio and the risk-free asset such that  $\sigma_{R_p} = 0.05$ . Since  $\sigma_T = 0.146$ , we have that  $0.05 = \sigma_{R_p} = \omega \sigma_T = 0.146 \omega$ , so that  $\omega = 0.05/0.146 = 0.343$  and  $1 - \omega = 0.657$ .

So 65.7 % of the portfolio should be in the risk-free asset, and 34.3 % should be in the tangency portfolio. Thus  $(0.343)(69.3\%) = 23.7\%$  should be in the first risky asset and  $(0.343)(30.7\%) = 10.5\%$  should be in the second risky asset. The total is not quite 100 % because of rounding.  $\square$

### *Example 16.5. (Continuation of Examples 16.2–16.4)*

Now suppose that you want a 10 % expected return. In this example we will compare

- the best portfolio of only risky assets, and
- The best portfolio of the risky assets and the risk-free asset.

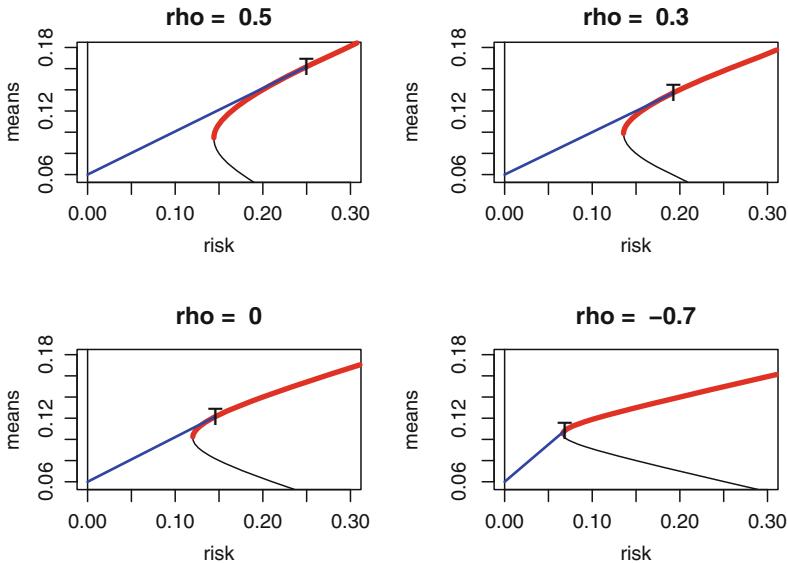
The best portfolio of only risky assets uses  $w$  solving  $0.1 = w(0.14) + (1-w)(0.08)$ , which implies that  $w = 1/3$ . This is the *only* portfolio of risky assets with  $E(R_p) = 0.1$ , so by default it is best. Then

$$\sigma_{R_p} = \sqrt{w^2(0.2)^2 + (1-w)^2(0.15)^2} = \sqrt{(1/9)(0.2)^2 + 4/9(0.15)^2} = 0.120.$$

The best portfolio of the two risky assets and the risk-free asset can be found as follows. First,  $0.1 = E(R) = \mu_f + \omega\{E(R_T) - \mu_f\} = 0.06 + 0.062\omega = 0.06 + 0.425\sigma_R$ , since  $\sigma_{R_p} = \omega\sigma_T$  or  $\omega = \sigma_{R_p}/\sigma_T = \sigma_{R_p}/0.146$ . This implies that  $\sigma_{R_p} = 0.04/0.425 = 0.094$  and  $\omega = 0.04/0.062 = 0.645$ . So combining the risk-free asset with the two risky assets reduces  $\sigma_{R_p}$  from 0.120 to 0.094 while maintaining  $E(R_p)$  at 0.1. The reduction in risk is  $(0.120 - 0.094)/0.094 = 28\%$ , which is substantial.  $\square$

#### 16.4.3 Effect of $\rho_{12}$

Positive correlation between the two risky assets increases risk. With positive correlation, the two assets tend to move together which increases the volatility of the portfolio. Conversely, negative correlation is beneficial since it decreases risk. If the assets are negatively correlated, a negative return of one tends to occur with a positive return of the other so the volatility of the portfolio decreases. Figure 16.2 shows the efficient frontier and tangency portfolio when  $\mu_1 = 0.14$ ,  $\mu_2 = 0.09$ ,  $\sigma_1 = 0.2$ ,  $\sigma_2 = 0.15$ , and  $\mu_f = 0.03$ . The value of  $\rho_{12}$  is varied from 0.5 to  $-0.7$ . Notice that Sharpe's ratio of the tangency portfolio returns increases as  $\rho_{12}$  decreases. This means that when  $\rho_{12}$  is small, then efficient portfolios have less risk for a given expected return compared to when  $\rho_{12}$  is large.



**Fig. 16.2.** Efficient frontier (red) and tangency portfolio ( $T$ ) when  $\mu_1 = 0.14$ ,  $\mu_2 = 0.09$ ,  $\sigma_1 = 0.2$ ,  $\sigma_2 = 0.15$ , and  $\mu_f = 0.03$ . The value of  $\rho_{12}$  is varied from 0.5 to  $-0.7$ .

## 16.5 Selling Short

Often some of the weights in an efficient portfolio are negative. A negative weight on an asset means that this asset is sold short. *Selling short* is a way to profit if a stock price goes *down*. To sell a stock short, one sells the stock without owning it. The stock must be borrowed from a broker or another customer of the broker. At a later point in time, one buys the stock and gives it back to the lender. This closes the short position.

Suppose a stock is selling at \$25/share and you sell 100 shares short. This gives you \$2500. If the stock goes down to \$17/share, you can buy the 100 shares for \$1700 and close out your short position. You made a profit of \$800 (ignoring transaction costs) because the stock went down 8 points. If the stock had gone up, then you would have had a loss.

Suppose now that you have \$100 and there are two risky assets. With your money you could buy \$150 worth of risky asset 1 and sell \$50 short of risky asset 2. The net cost would be exactly \$100. If  $R_1$  and  $R_2$  are the returns on risky assets 1 and 2, then the return on your portfolio would be

$$\frac{3}{2}R_1 + \left(-\frac{1}{2}\right)R_2.$$

Your portfolio weights are  $w_1 = 3/2$  and  $w_2 = -1/2$ . Thus, you hope that risky asset 1 rises in price and risky asset 2 falls in price. Here, again, we have ignored transaction costs.

If one sells a stock short, one is said to have a *short position* in that stock, and owning the stock is called a *long position*.

## 16.6 Risk-Efficient Portfolios with $N$ Risky Assets

In this section, we use quadratic programming to find efficient portfolios with an arbitrary number of assets. An advantage of quadratic programming is that it allows one to impose constraints such as limiting short sales. With no constraints on the allocation vector  $\mathbf{w}$ , analytic formulas for the tangency portfolio can be derived using Lagrange multipliers, but this approach does not generalize to constrained  $\mathbf{w}$ .

Assume that we have  $N$  risky assets and that the return on the  $i$ th risky asset is  $R_i$  and has expected value  $\mu_i$ . Define

$$\mathbf{R} = \begin{pmatrix} R_1 \\ \vdots \\ R_N \end{pmatrix}$$

to be the random vector of returns,

$$E(\mathbf{R}) = \boldsymbol{\mu} = \begin{pmatrix} \mu_1 \\ \vdots \\ \mu_N \end{pmatrix},$$

and  $\boldsymbol{\Sigma}$  to be the covariance matrix of  $\mathbf{R}$ .

Let

$$\mathbf{w} = \begin{pmatrix} w_1 \\ \vdots \\ w_N \end{pmatrix}$$

be a vector of portfolio weights so that  $w_1 + \cdots + w_N = \mathbf{1}^\top \mathbf{w} = 1$ , where

$$\mathbf{1} = \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix}$$

is a column of  $N$  ones. The expected return on the portfolio is

$$\sum_{i=1}^N w_i \mu_i = \mathbf{w}^\top \boldsymbol{\mu}. \quad (16.6)$$

Suppose there is a target value,  $\mu_P$ , of the expected return on the portfolio. When  $N = 2$ , the target expected returns is achieved by only one portfolio and its  $w_1$ -value solves  $\mu_P = w_1 \mu_1 + w_2 \mu_2 = \mu_2 + w_1(\mu_1 - \mu_2)$ . For  $N \geq 3$ ,

there will be an infinite number of portfolios achieving the target  $\mu_P$ . The one with the smallest variance is called the “efficient” portfolio. Our goal is to find the efficient portfolio.

The variance of the return on the portfolio with weights  $\mathbf{w}$  is

$$\mathbf{w}^\top \boldsymbol{\Sigma} \mathbf{w}. \quad (16.7)$$

Thus, given a target  $\mu_P$ , the efficient portfolio minimizes (16.7) subject to

$$\mathbf{w}^\top \boldsymbol{\mu} = \mu_P \quad (16.8)$$

and

$$\mathbf{w}^\top \mathbf{1} = 1. \quad (16.9)$$

*Quadratic programming* is used to minimize a quadratic objective function subject to linear constraints. In applications to portfolio optimization, the objective function is the variance of the portfolio return. The objective function is a function of  $N$  variables, such as the weights of  $N$  assets, that are denoted by an  $N \times 1$  vector  $\mathbf{x}$ . Suppose that the quadratic objective function to be minimized is

$$\frac{1}{2} \mathbf{x}^\top \mathbf{D} \mathbf{x} - \mathbf{d}^\top \mathbf{x}, \quad (16.10)$$

where  $\mathbf{D}$  is an  $N \times N$  matrix and  $\mathbf{d}$  is an  $N \times 1$  vector. The factor of  $1/2$  is not essential but is used here to keep our notation consistent with R. There are two types of linear constraints on  $\mathbf{x}$ , inequality and equality constraints. The linear inequality constraints are

$$\mathbf{A}_{\text{neq}}^\top \mathbf{x} \geq \mathbf{b}_{\text{neq}}, \quad (16.11)$$

where  $\mathbf{A}_{\text{neq}}$  is an  $m \times N$  matrix,  $\mathbf{b}_{\text{neq}}$  is an  $m \times 1$  vector, and  $m$  is the number of inequality constraints. The equality constraints are

$$\mathbf{A}_{\text{eq}}^\top \mathbf{x} = \mathbf{b}_{\text{eq}}, \quad (16.12)$$

where  $\mathbf{A}_{\text{eq}}$  is an  $n \times N$  matrix,  $\mathbf{b}_{\text{eq}}$  is an  $n \times 1$  vector, and  $n$  is the number of equality constraints. Quadratic programming minimizes the quadratic objective function (16.10) subject to linear inequality constraints (16.11) and linear equality constraints (16.12).

To apply quadratic programming to find an efficient portfolio, we use  $\mathbf{x} = \mathbf{w}$ ,  $\mathbf{D} = 2\boldsymbol{\Sigma}$ , and  $\mathbf{d}$  equal to an  $N \times 1$  vector of zeros so that (16.10) is  $\mathbf{w}^\top \boldsymbol{\Sigma} \mathbf{w}$ , the return variance of the portfolio. There are two equality constraints, one that the weights sum to 1 and the other that the portfolio return is a specified target  $\mu_P$ . Therefore, we define

$$\mathbf{A}_{\text{eq}}^\top = \begin{pmatrix} \mathbf{1}^\top \\ \boldsymbol{\mu}^\top \end{pmatrix}$$

and

$$\mathbf{b}_{\text{eq}} = \begin{pmatrix} 1 \\ \mu_P \end{pmatrix},$$

so that (16.12) becomes

$$\begin{pmatrix} \mathbf{1}^T \mathbf{w} \\ \boldsymbol{\mu}^T \mathbf{w} \end{pmatrix} = \begin{pmatrix} 1 \\ \mu_P \end{pmatrix},$$

which is the same as constraints (16.8) and (16.9). So far, inequality constraints have not been used.

Investors often wish to impose additional inequality constraints. If an investor cannot or does not wish to sell short, then the constraint

$$\mathbf{w} \geq \mathbf{0}$$

can be used. Here  $\mathbf{0}$  is a vector of  $N$  zeros. In this case  $\mathbf{A}_{\text{neq}}$  is the  $N \times N$  identical matrix and  $\mathbf{b}_{\text{neq}} = \mathbf{0}$ .

To avoid concentrating the portfolio in just one or a few stocks, an investor may wish to constrain the portfolio so that no  $w_i$  exceeds a bound  $\lambda$ , for example,  $\lambda = 1/4$  means that no more than 1/4 of the portfolio can be in any single stock. In this case,  $\mathbf{w} \leq \lambda \mathbf{1}$  or equivalently  $-\mathbf{w} \geq -\lambda \mathbf{1}$ , so that  $\mathbf{A}_{\text{neq}}$  is minus the  $N \times N$  identity matrix and  $\mathbf{b}_{\text{neq}} = -\lambda \mathbf{1}$ . One can combine these constraints with those that prohibit short selling.

To find the efficient frontier, one uses a grid of values of  $\mu_P$  and finds the corresponding efficient portfolios. For each portfolio,  $\sigma_P^2$ , which is the minimized value of the objective function, can be calculated. Then one can find the minimum variance portfolio by finding the portfolio with the smallest value of the  $\sigma_P^2$ . The efficient frontier is the set of efficient portfolios with expected return above the expected return of the minimum variance portfolio. One can also compute Sharpe's ratio for each portfolio on the efficient frontier and the tangency portfolio is the one maximizing Sharpe's ratio.

*Example 16.6. Finding the efficient frontier, tangency portfolio, and minimum variance portfolio using quadratic programming*

The following R program uses the returns on three stocks, GE, IBM, and Mobil, in the `CRSPday` data set in the `Ecdat` package. The function `solve.QP()` in the `quadprog` package is used for quadratic programming. `solve.QP()` combines  $\mathbf{A}_{\text{eq}}^T$  and  $\mathbf{A}_{\text{neq}}^T$  into a single matrix `Amat` by stacking  $\mathbf{A}_{\text{eq}}^T$  on top of  $\mathbf{A}_{\text{neq}}^T$ . The parameter `meq` is the number of rows of  $\mathbf{A}_{\text{eq}}^T$ .  $\mathbf{b}_{\text{eq}}$  and  $\mathbf{b}_{\text{neq}}$  are handled analogously. In this example, there are no inequality constraints, so  $\mathbf{A}_{\text{neq}}^T$  and  $\mathbf{b}_{\text{neq}}$  are not needed, but they are used in the next example.

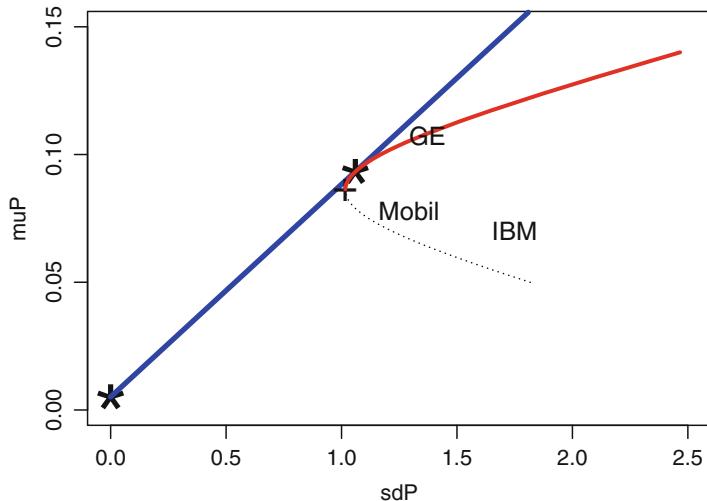
The efficient portfolio is found for each of 300 target values of  $\mu_P$  between 0.05 and 0.14. For each portfolio, Sharpe's ratio is found at line 28 and the

logical vector `ind` at line 29 indicates which portfolio is the tangency portfolio maximizing Sharpe's ratio. Similarly, `ind2` at line 34 indicates the minimum variance portfolio. Also, `ind3` at line 36 indicates the points on the efficient frontier. It is assumed that the risk-free rate is 1.3%/year; see line 26.

```

1 llibrary(Ecdat)
2 library(quadprog)
3 data(CRSPday)
4 R = 100*CRSPday[,4:6] # convert to percentages
5 mean_vect = apply(R, 2, mean)
6 cov_mat = cov(R)
7 sd_vect = sqrt(diag(cov_mat))
8 Amat = cbind(rep(1, 3), mean_vect) # set the constraints matrix
9 muP = seq(0.05, 0.14, length = 300) # target portfolio means
10 # for the expect portfolio return
11 sdP = muP # set up storage for std dev's of portfolio returns
12 weights = matrix(0, nrow = 300, ncol = 3) # storage for weights
13 for (i in 1:length(muP)) # find the optimal portfolios
14 {
15   bvec = c(1, muP[i]) # constraint vector
16   result =
17   solve.QP(Dmat = 2 * cov_mat, dvec = rep(0, 3),
18             Amat = Amat, bvec = bvec, meq = 2)
19   sdP[i] = sqrt(result$value)
20   weights[i,] = result$solution
21 }
22 pdf("quad_prog_plot.pdf", width = 6, height = 5)
23 plot(sdP, muP, type = "l", xlim = c(0, 2.5),
24       ylim = c(0, 0.15), lty = 3) # plot efficient frontier (and
25                               # inefficient portfolios below the min var portfolio)
26 mufree = 1.3 / 253 # input value of risk-free interest rate
27 points(0, mufree, cex = 4, pch = "*") # show risk-free asset
28 sharpe = (muP - mufree) / sdP # compute Sharpe's ratios
29 ind = (sharpe == max(sharpe)) # Find maximum Sharpe's ratio
30 weights[ind, ] # print the weights of the tangency portfolio
31 lines(c(0, 2), mufree + c(0, 2) * (muP[ind] - mufree) / sdP[ind],
32        lwd = 4, lty = 1, col = "blue") # show line of optimal portfolios
33 points(sdP[ind], muP[ind], cex = 4, pch = "*") # tangency portfolio
34 ind2 = (sdP == min(sdP)) # find minimum variance portfolio
35 points(sdP[ind2], muP[ind2], cex = 2, pch = "+") # min var portfolio
36 ind3 = (muP > muP[ind2])
37 lines(sdP[ind3], muP[ind3], type = "l", xlim = c(0, 0.25),
38        ylim = c(0, 0.3), lwd = 3, col = "red") # plot efficient frontier
39 text(sd_vect[1], mean_vect[1], "GE", cex = 1.15)
40 text(sd_vect[2], mean_vect[2], "IBM", cex = 1.15)
41 text(sd_vect[3], mean_vect[3], "Mobil", cex = 1.15)
42 graphics.off()

```



**Fig. 16.3.** Efficient frontier (solid), line of efficient portfolios (dashed) connecting the risk-free asset and tangency portfolio (asterisks), and the minimum variance portfolio (plus) with three stocks (GE, IBM, and Mobil). The three stocks are also shown on reward-risk space.

The plot produced by this program is Fig. 16.3. The program prints the weights of the tangency portfolio, which are

```
> weights[ind,] # Find tangency portfolio
[1] 0.5512 0.0844 0.3645
```

□

*Example 16.7. Finding the efficient frontier, tangency portfolio, and minimum variance portfolio with no short selling using quadratic programming*

In this example, Example 16.6 is modified so that short sales are not allowed. Only three lines of code need to be changed. When short sales are prohibited, the target expected return on the portfolio must lie between the smallest and largest expected returns on the stocks. To prevent numerical errors, the target expected returns will start 0.0001 above the smallest expected stock return and end 0.0001 below the largest expected stock return. This is enforced by the following change:

```
muP = seq(min(mean_vect) + 0.0001, max(mean_vect) - 0.0001,
           length = 300)
```

To enforce no short sales, an  $A_{\text{neq}}$  matrix is needed and is set equal to a  $3 \times 3$  identity matrix:

```
Amat = cbind(rep(1, 3), mean_vect, diag(1, nrow = 3))
```

Also,  $b_{\text{neq}}$  is set equal to a three-dimensional vector of zeros:

```
bvec = c(1, muP[i], rep(0, 3))
```

The new plot is shown in Fig. 16.4. Since the tangency portfolio in Example 16.6 had all weights positive, the tangency portfolio is unchanged by the prohibition of short sales. The efficient frontier is changed since without short sales, it is impossible to have expected returns greater than the expected return of GE, the stock with the highest expected return. In contrast, when short sales are allowed, there is no upper bound on the expected return (or on the risk). In Fig. 16.4 the red curve is the entire efficient frontier, but in Fig. 16.3 the efficient frontier is the red curve extended to  $(+\infty, +\infty)$ .  $\square$

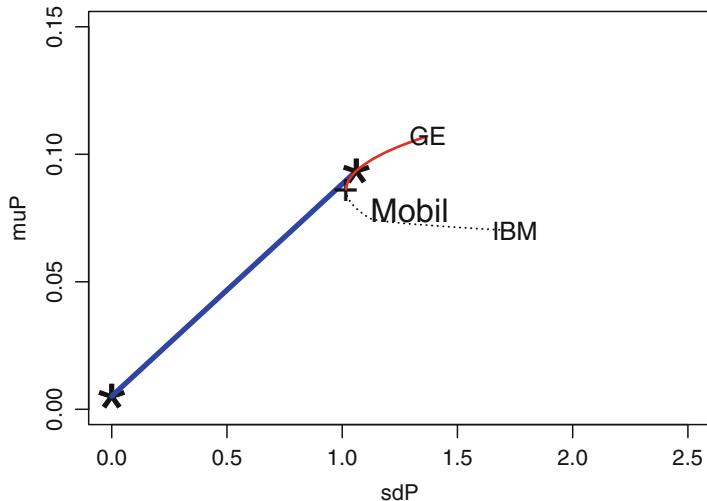
## 16.7 Resampling and Efficient Portfolios

The theory of portfolio optimization assumes that the expected returns and the covariance matrix of the returns is known. In practice, one must replace these quantities with estimates as in the previous examples. However, the effects of estimation error, especially with smaller values of  $N$ , can result in portfolios that only appear efficient. This problem will be investigated in this section using the bootstrap to quantify the effects of estimation error.

### *Example 16.8. The global asset allocation problem*

One application of optimal portfolio selection is allocation of capital to different market segments. For example, Michaud (1998) discusses a global asset allocation problem where capital must be allocated to “U.S. stocks and government/corporate bonds, euros, and the Canadian, French, German, Japanese, and U.K. equity markets.” Here we look at a similar example where we allocate capital to the equity markets of 10 different countries. Monthly returns for these markets were calculated from MSCI Hong Kong, MSCI Singapore, MSCI Brazil, MSCI Argentina, MSCI UK, MSCI Germany, MSCI Canada, MSCI France, MSCI Japan, and the S&P 500. “MSCI” means “Morgan Stanley Capital Index.” The data are from January 1988 to January 2002, inclusive, so there are 169 months of data.

Assume that we want to find the tangency portfolio that maximizes Sharpe’s ratio. The tangency portfolio was estimated using sample means and the sample covariance as in Example 16.6, and its Sharpe’s ratio is estimated to be 0.3681. However, we should suspect that 0.3681 must be an overestimate since this portfolio only maximizes Sharpe’s ratio using estimated parameters,

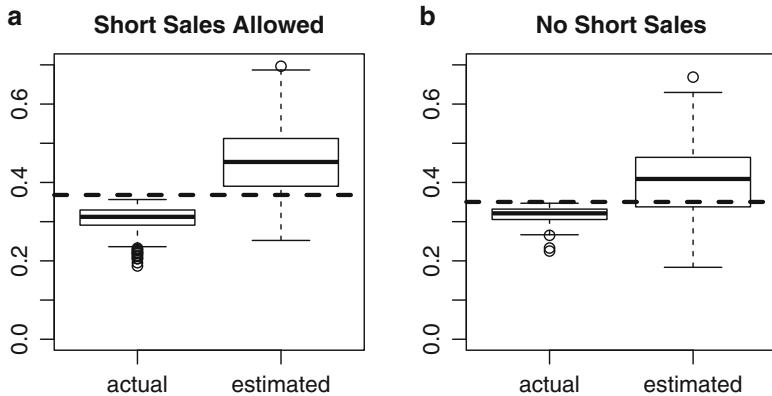


**Fig. 16.4.** Efficient frontier (solid), line of efficient portfolios (dashed) connecting the risk-free asset and tangency portfolio (asterisks), and the minimum variance portfolio (plus) with three stocks (GE, IBM, and Mobil) with short sales prohibited.

not the true means and covariance matrix. To evaluate the possible amount of overestimation, one can use the bootstrap. As discussed in Chap. 6, in the bootstrap simulation experiment, the sample is the “true population” so that the sample mean and covariance matrix are the “true parameters,” and the resamples mimic the sampling process. Actual Sharpe’s ratios are calculated with the sample means and covariance matrix, while estimated Sharpe’s ratio use the means and covariance matrix of the resamples.

First, 250 resamples were taken and for each the tangency portfolio was estimated. Resampling was done by sampling rows of the data matrix as discussed in Sect. 7.11. For each of the 250 tangency portfolios estimated from the resamples, the actual and estimated Sharpe’s ratios were calculated. Boxplots of the 250 actual and 250 estimated Sharpe’s ratios of the estimated tangency portfolios are in Fig. 16.5a. “Estimated” means calculated from the resample and “true” means calculated from the sample. In this figure, there is a dashed horizontal line at height 0.3681, the actual Sharpe’s ratio of the true tangency portfolio. One can see that all 250 estimated tangency portfolios have actual Sharpe’s ratios below this value, as they must since the actual Sharpe’s ratio is maximized by the true tangency portfolio, not the estimated tangency portfolios.

From the boxplot on the right-hand side of (a), one can see that the estimated Sharpe’s ratios overestimate not only the actual Sharpe’s ratios of the estimated tangency portfolios but also the somewhat larger (and unattainable) actual Sharpe’s ratio of the true (but unknowable) tangency portfolio.  $\square$



**Fig. 16.5.** Bootstrapping estimation of the tangency portfolio and its Sharpe's ratio. (a) Short sales allowed. The left-hand boxplot is of the actual Sharpe's ratios of the estimated tangency portfolios for 250 resamples. The right-hand boxplot contains the estimated Sharpe's ratios for these portfolios. The horizontal dashed line indicates Sharpe's ratio of the true tangency portfolio. (b) Same as (a) but with short sales not allowed.

There are several ways to alleviate the problems caused by estimation error when attempting to find a tangency portfolio. One can try to find more accurate estimators; the factor models of Chap. 18 and Bayes estimators of Chap. 20 (see especially Example 20.12) do this. Another possibility is to restrict short sales.

Portfolios with short sales aggressively attempt to maximize Sharpe's ratio by selling short those stocks with the smallest estimated mean returns and having large long positions in those stocks with the highest estimated mean returns. The weakness with this approach is that it is particularly sensitive to estimation error. Unfortunately, expected returns are estimated with relatively large uncertainty. This problem can be seen in Table 16.1, which has 95 % confidence intervals for the mean returns. The percentile method is used for the confidence intervals, so the endpoints are the 2.5 and 97.5 bootstrap percentiles. Notice for Singapore and Japan, the confidence intervals include both positive and negative values. In the table, the returns are expressed as percentage returns.

#### *Example 16.9. The global asset allocation problem: short sales prohibited*

This example repeats the bootstrap experimentation of Example 16.8 with short sales prohibited by using inequality constraints such as in Example 16.7. With short sales not allowed, the actual Sharpe's ratio of the true tangency portfolio is 0.3503, which is only slightly less than when short sales are allowed.

**Table 16.1.** 95 % percentile-method bootstrap confidence intervals for the mean returns of the 10 countries.

| Country   | 2.5 %  | 97.5 % |
|-----------|--------|--------|
| Hong Kong | 0.186  | 2.709  |
| Singapore | -0.229 | 2.003  |
| Brazil    | 0.232  | 5.136  |
| Argentina | 0.196  | 6.548  |
| UK        | 0.071  | 1.530  |
| Germany   | 0.120  | 1.769  |
| Canada    | 0.062  | 1.580  |
| France    | 0.243  | 2.028  |
| Japan     | -0.884 | 0.874  |
| U.S.      | 0.636  | 1.690  |

Boxplots of actual and apparent Sharpe's ratios are in Fig. 16.5b. Comparing Fig. 16.5a and b, one sees that prohibiting short sales has two beneficial effects—Sharpe's ratios actually achieved are slightly higher with no short sales allowed compared to having no constraints on short sales. In fact, the mean of the 250 actual Sharpe's ratios is 0.3060 with short sales allowed and 0.3169 with short sales prohibited. Moreover, the overestimation of Sharpe's ratio is reduced by prohibiting short sales—the mean apparent Sharpe's ratio is 0.4524 [with estimation error  $(0.4524 - 0.3681) = 0.0843$ ] with short sales allowed but only 0.4038 [with estimation error  $(0.4038 - 0.3503) = 0.0535$ ] with short sales prohibited. However, these effects, though positive, are only modest and do not entirely solve the problem of overestimation of Sharpe's ratio.  $\square$

#### *Example 16.10. The global asset allocation problem: Shrinkage estimation and short sales prohibited*

In Example 16.9, we saw that prohibiting short sales can increase Sharpe's ratio of the estimated tangency portfolio, but the improvement is only modest. Further improvement requires more accurate estimation of the mean vector or the covariance matrix of the returns.

This example investigates possible improvements from shrinking the 10 estimated means toward each other. Specifically, if  $\bar{Y}_i$  is the sample mean of the  $i$ th country,  $\bar{Y} = (\sum_{i=1}^{10} \bar{Y}_i)/10$  is the grand mean (mean of the means), and  $\alpha$  is a tuning parameter between 0 and 1, then the estimated mean return for the  $i$ th country is

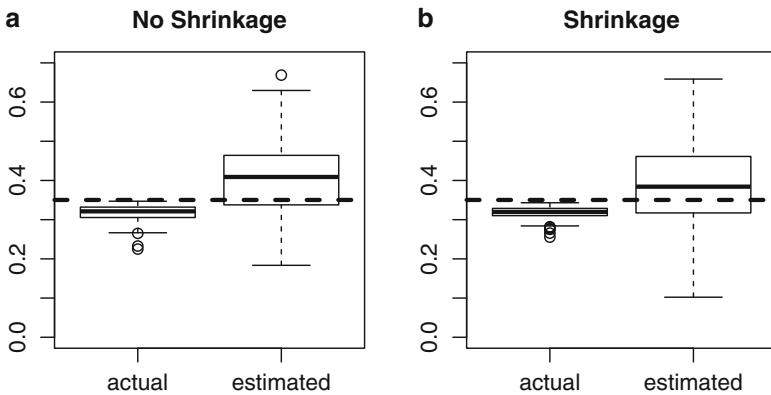
$$\hat{\mu}_i = \alpha \bar{Y}_i + (1 - \alpha) \bar{Y}. \quad (16.13)$$

The purpose of shrinkage is to reduce the variance of the estimator, though the reduced variance comes at the expense of some bias. Since it is the mean of

10 means,  $\bar{Y}$  is much less variable than any of  $\bar{Y}_1, \dots, \bar{Y}_{10}$ . Therefore,  $\text{Var}(\hat{\mu}_i)$  decreases as  $\alpha$  is decreased toward 0. However,

$$E(\hat{\mu}_i) = \alpha\mu_i + \frac{1-\alpha}{10} \sum_{i=1}^{10} \mu_i \quad (16.14)$$

so that, for any  $\alpha \neq 1$ ,  $\hat{\mu}_i$  is biased, except under the very likely circumstance that  $\mu_1 = \dots = \mu_{10}$ . The parameter  $\alpha$  controls the bias–variance tradeoff. In this example,  $\alpha = 1/2$  will be used for illustration and short sales will not be allowed.



**Fig. 16.6.** Bootstrapping estimation of the tangency portfolio and its Sharpe's ratio. Short sales not allowed. (a) No shrinkage. The left-hand boxplot is of the actual Sharpe's ratios of the estimated tangency portfolios for 250 resamples. The right-hand boxplot contains the estimated Sharpe's ratios for these portfolios. The horizontal dashed line indicates Sharpe's ratio of the true tangency portfolio. (b) Same as (a) but with shrinkage.

Figure 16.6 compares the performance of shrinkage versus no shrinkage. Panel (a) contains the boxplots that we saw in panel (b) of Fig. 16.5 where  $\alpha = 1$ . Panel (b) has the boxplots when the tangency portfolio is estimated using  $\alpha = 1/2$ . Compared to panel (a), in panel (b) the actual Sharpe's ratios are somewhat closer to the dashed line indicating Sharpe's ratio of the true tangency portfolio; the means of the actual Sharpe's ratios are 0.317 and 0.318 with and without shrinkage, respectively. These values should be compared with the Sharpe ratio of the true (but unknown) tangency portfolio of 0.34.

Moreover, the estimated Sharpe's ratios in (b) are smaller and closer to the true Sharpe's ratios, so there is less overoptimization—shrinkage has helped in two ways. The mean estimated Sharpe's ratios are 0.390 and 0.404 with and without shrinkage.

The next step might be selection of  $\alpha$  to optimize performance of shrinkage estimation. Doing this need not be difficult, since different values of  $\alpha$  can be compared by bootstrapping.  $\square$

There are other methods for improving the estimation of the mean vector and estimation of the covariance matrix can be improved as well, for example, by using the factor models in Chap. 18 or Bayesian estimation as in Chap. 20. Moreover, one need not focus on the tangency portfolio but could, for example, estimate the minimum variance portfolio. Whatever the focus of estimation, the bootstrap can be used to compare various strategies for improving the estimation of the optimal portfolio.

## 16.8 Utility

Economists generally do not model economic decisions in terms of the mean and variance of the return but rather by using a *utility function*. The utility of an amount  $X$  of money is said to be  $U(X)$  where the utility function  $U$  generally has the properties:

1.  $U(0) = 0$ ;
2.  $U$  is strictly increasing;
3. the first derivative  $U'(X)$  is strictly decreasing.

Assumption 1 is not necessary but is reasonable and states that the utility of 0 dollars is 0. Assumption 2 merely states that more money is better than less. Assumption 3 implies that the more money we have the less we value an extra dollar and is called *risk aversion*. Assumption 3 implies that we would decline a bet that pays  $\pm \Delta$  with equal probabilities. In fact, Assumption 3 implies that we would decline any bet with a payoff that is symmetrically distributed about 0, because the expected utility of our wealth would be reduced if we accepted the bet. Mathematically, Assumption 3 implies that  $U$  is strictly concave. If the second derivative  $U''$  exists then Assumption 3 is equivalent to the assumption that  $U''(X) < 0$  for all  $X$ .

It is assumed that a rational person will make investment decisions so as to maximize

$$E\{U(X)\} = E[U\{X_0(1 + R)\}] \quad (16.15)$$

where  $X$  is that person's final wealth,  $X_0$  is the person's initial wealth, and  $R$  is the return from the investments. In economics this is almost a part of the definition of a rational person, with another component of the definition being that a rational person will update probabilities using Bayes' law (see Chap. 20). Each individual is assumed to have his or her own utility function and two different rational people may make different decisions because they have different utility functions.

How different are mean-variance efficient portfolios and portfolios that maximize expected utility? In the case that returns are normally distributed, this question can be answered.

**Result 16.2** *If returns on all portfolios are normally distributed and if  $U$  satisfies Assumption 3, then the portfolio that maximizes expected utility is on the efficient frontier.*

So, if one chose a portfolio to maximize expected utility, then a mean-variance efficient portfolio would be selected. Exactly which portfolio on the efficient frontier one chooses would depend on one's utility function.

### *Proof of Result 16.2*

This result can be proven by proving the following fact: if  $R_1$  and  $R_2$  are normally distributed with the same means and with standard deviations  $\sigma_1$  and  $\sigma_2$  such that  $\sigma_1 < \sigma_2$ , then  $E\{U(R_1)\} > E\{U(R_2)\}$ . We will show that this follows from Jensen's inequality which states that if  $U$  is concave function and  $X$  is any random variable, then  $E\{U(X)\} \leq U\{E(X)\}$ . The inequality is strict if  $U$  is strictly convex and  $X$  is nondegenerate.<sup>4</sup>

Let  $X = R_1 + e$  where  $e$  is independent of  $R_1$  and normally distributed with mean 0 and variance  $\sigma_2^2 - \sigma_1^2$ . Then  $X$  has the same distribution as  $R_2$ ,  $e$  is nondegenerate, and, using the law of iterated expectations and then Jensen's inequality and Assumption 3, we have

$$E\{U(R_2)\} = E\{U(X)\} = E[E\{U(X)|R_1\}] < E[\{U\{E(X|R_1)\}\}] = E\{U(R_1)\}, \quad (16.16)$$

since  $E(X|R_1) = R_1$ . □

The assumption in Result 16.2 that the returns are normally distributed can be weakened to the more realistic assumption that the vector of returns on the assets is a multivariate scale mixture, e.g., has a multivariate  $t$ -distribution. To prove this extension, one conditions on the mixing variable so that the returns have a conditional multivariate normal distribution. Then (16.16) holds conditionally for all values of the mixing variable and therefore holds unconditionally.

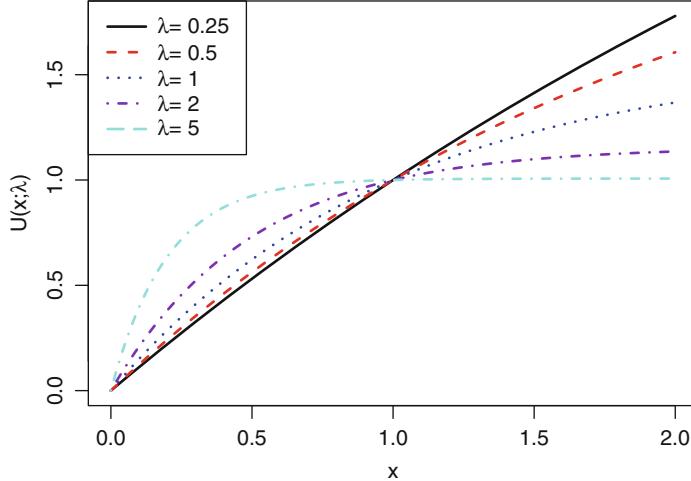
A common class of utility functions is

$$U(x; \lambda) = 1 - \exp(-\lambda x), \quad (16.17)$$

where  $\lambda > 0$  determines the amount of risk aversion. Note that  $U'(x; \lambda) = \lambda \exp(-\lambda x)$  and  $U''(x; \lambda) = -\lambda^2 \exp(-\lambda x)$  (differentiation is with respect to  $x$ ).

---

<sup>4</sup> Jensen's inequality is usually stated for convex functions with the inequality reversed. If  $U$  is concave then  $-U$  is convex so that the two forms of Jensen's inequality are equivalent. A random variable  $X$  is degenerate if there is a constant  $a$  such that  $P(X = a) = 1$ . Otherwise, it is nondegenerate.



**Fig. 16.7.** The utility functions  $\tilde{U}(x; \lambda) = U(x; \lambda)/U(0; \lambda)$  where  $U(x; \lambda) = 1 - \exp(-\lambda x)$ .

Thus,  $U''(x; \lambda)$  is negative for all  $x$  so Assumption 3 is met; it is easy to see that Assumptions 1 and 2 also hold. As  $x \rightarrow \infty$ ,  $U(x; \lambda) \rightarrow 1$ .

Multiplying a utility function by a positive constant will not affect which decision maximizes utility and can standardize utility functions to make them more comparable. In Fig. 16.7,  $\tilde{U}(x; \lambda) := U(x; \lambda)/U(0; \lambda)$  is plotted for  $\lambda = 0.25, 0.5, 1, 2$  and  $5$ . Since  $\tilde{U}(1; \lambda) = 1$  for all  $\lambda$ , these utility functions have been standardized so that the utility corresponding to a return of 0 is always 1. Stated differently, a return equal to 0 has the same utility for all degrees of risk aversion.

Adding a constant to the utility function does not effect the optimal decision, so one could work with the slightly simpler utility function  $-\exp(-\lambda x)$  instead of  $U(x; \lambda)$  or  $\tilde{U}(x; \lambda)$ .

When the utility function is given by (16.17) and  $R$  is normally distributed, then (16.15) becomes

$$1 - \exp \left[ -\lambda X_0 \{1 + E(R)\} + (\lambda X_0)^2 \frac{\text{var}(R)}{2} \right] \quad (16.18)$$

by properties of the lognormal distribution; see Appendix A.9.4. For given values of  $\lambda$  and  $X_0$ , the expected utility is maximized by maximizing

$$E(R) - (\lambda X_0) \frac{\text{var}(R)}{2}. \quad (16.19)$$

Therefore, using the notation of Sect. 16.6, one selects the allocation vector  $\mathbf{w}$  of the portfolio to maximize

$$\mathbf{w}^\top \boldsymbol{\mu} - (\lambda X_0) \frac{\mathbf{w}^\top \boldsymbol{\Sigma} \mathbf{w}}{2} \quad (16.20)$$

subject to  $\mathbf{w}^\top \mathbf{1} = 1$ .

Maximizing (16.20) subject to linear constraints is a quadratic programming problem. As  $\lambda \rightarrow 0$ , the expected return and standard deviation of the return converge to  $\infty$ . Conversely, as  $\lambda \rightarrow \infty$ , the solution converges to the minimum variance portfolio. Therefore, as  $\lambda$  is varied from  $\infty$  to 0, one finds all of the portfolios on the efficient frontier from left to right. This behavior is illustrated in the next example; see Fig. 16.8.

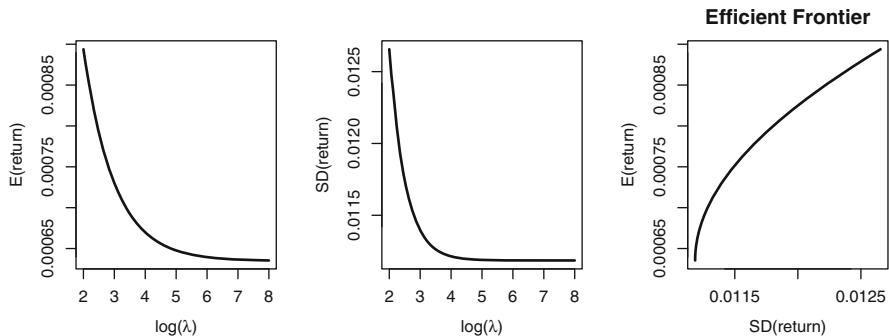
*Example 16.11. Finding portfolios the maximize expected utility*

We will use stock price data in the file `Stock_Bond.csv`. This data set was discussed in Sect. 2.4.1. For simplicity of notation, we subsume  $X_0$  into  $\lambda$ . The R code below solves the quadratic program (16.20) for 250 values of  $\log(\lambda)$  equally spaced from 2 to 8; this range was selected by trial-and-error.

```

1 library(quadprog)
2 dat = read.csv("Stock_Bond.csv")
3 y = dat[, c(3, 5, 7, 9, 11, 13, 15, 17, 19, 21)]
4 n = dim(y)[1]
5 m = dim(y)[2] - 1
6 r = y[-1,] / y[-n,] - 1
7 mean_vect = as.matrix(colMeans(r))
8 cov_mat = cov(r)
9 nlambda = 250
10 loglambda_vect = seq(2, 8, length = nlambda)
11 w_matrix = matrix(nrow = nlambda, ncol = 10)
12 mu_vect = matrix(nrow = nlambda, ncol = 1)
13 sd_vect = mu_vect
14 ExUtil_vect = mu_vect
15 conv_vect = mu_vect
16 for (i in 1:nlambda)
17 {
18   lambda = exp(loglambda_vect[i])
19   opt = solve.QP(Dmat = as.matrix(lambda^2 * cov_mat),
20                 dvec = lambda * mean_vect, Amat = as.matrix(rep(1,10)),
21                 bvec = 1, meq = 1)
22   w = opt$solution
23   mu_vect[i] = w %*% mean_vect
24   sd_vect[i] = sqrt(w %*% cov_mat %*% w)
25   w_matrix[i,] = w
26   ExUtil_vect[i] = opt$value
27 }
```

Next, the expected return and the standard deviation of the return are plotted against  $\lambda$  and then the efficient frontier is drawn by plotting the expect return again the standard deviation of the return. The plots are in Fig. 16.8.  $\square$



**Fig. 16.8.** The expected portfolio return versus  $\log(\lambda)$  (left), the standard deviation of the return versus  $\log(\lambda)$  (center) and the efficient frontier (right).

## 16.9 Bibliographic Notes

Markowitz (1952) was the original paper on portfolio theory and was expanded into the book Markowitz (1959). Bodie and Merton (2000) provide an elementary introduction to portfolio selection theory. Bodie, Kane, and Marcus (1999) and Sharpe, Alexander, and Bailey (1999) give a more comprehensive treatment. See also Merton (1972). Formula (16.5) is derived in Example 5.10 of Ruppert (2004).

Jobson and Korkie (1980) and Britten-Jones (1999) discuss the statistical issue of estimating the efficient frontier; see the latter for additional recent references. Britten-Jones (1999) shows that the tangency portfolio can be estimated by regression analysis and hypotheses about the tangency portfolio can be tested by regression  $F$ -tests. Jagannathan and Ma (2003) discuss how imposing constraints such as no short sales can reduce risk.

## 16.10 R Lab

### 16.10.1 Efficient Equity Portfolios

This section uses daily stock prices in the data set `Stock_Bond.csv` that is posted on the book's website and in which any variable whose name ends with “AC” is an adjusted closing price. As the name suggests, these prices have been adjusted for dividends and stock splits, so that returns can be calculated without further adjustments. Run the following code which will read the data, compute the returns for six stocks, create a scatterplot matrix of these returns, and compute the mean vector, covariance matrix, and vector of standard deviations of the returns. Note that returns will be percentages.

```

dat = read.csv("Stock_Bond.csv", header = T)
prices = cbind(dat$GM_AC, dat$F_AC, dat$CAT_AC, dat$UTX_AC,
               dat$MRK_AC, dat$IBM_AC)
n = dim(prices)[1]
returns = 100 * (prices[2:n, ] / prices[1:(n-1), ] - 1)
pairs(returns)
mean_vect = colMeans(returns)
cov_mat = cov(returns)
sd_vect = sqrt(diag(cov_mat))

```

**Problem 1** Write an R program to find the efficient frontier, the tangency portfolio, and the minimum variance portfolio, and plot on “reward-risk space” the location of each of the six stocks, the efficient frontier, the tangency portfolio, and the line of efficient portfolios. Use the constraints that  $-0.1 \leq w_j \leq 0.5$  for each stock. The first constraint limits short sales but does not rule them out completely. The second constraint prohibits more than 50 % of the investment in any single stock. Assume that the annual risk-free rate is 3 % and convert this to a daily rate by dividing by 365, since interest is earned on trading as well as nontrading days.

**Problem 2** If an investor wants an efficient portfolio with an expected daily return of 0.07 %, how should the investor allocate his or her capital to the six stocks and to the risk-free asset? Assume that the investor wishes to use the tangency portfolio computed with the constraints  $-0.1 \leq w_j \leq 0.5$ , not the unconstrained tangency portfolio.

**Problem 3** Does this data set include Black Monday?

### 16.10.2 Efficient Portfolios with Apple, Exxon-Mobil, Target, and McDonald’s Stock

This section constructs portfolios with stocks from four companies: Apple Inc. (AAPL), Exxon-Mobil (XOM), Target Corp. (TGT), and McDonalds (MCD). Run the following code to get 2013 returns in terms of percentage for each of the 4 companies:

```

dat = read.csv("FourStocks_Daily2013.csv", header = TRUE)
head(dat)
prices = dat[,-1]
n = dim(prices)[1]
returns = 100*(prices[-1,] / prices[-n,] - 1)

```

**Problem 4** Write an R program to plot the efficient frontier and to find the allocation weight vector  $\mathbf{w}$  corresponding to the tangency portfolio. Use the sample mean vector and sample covariance matrix of the returns to estimate  $\boldsymbol{\mu}$  and  $\boldsymbol{\Sigma}$ . Assume that the annual risk free rate is 1.3 %. Use the constraints that no  $w_j$  can be less than  $-0.5$  or greater than  $0.5$ . Let  $\mu_P$  range from 0.045 to 0.06 %. Report both the Sharpe's Ratio and  $\mathbf{w}$  for the tangency portfolio.

**Problem 5** Write an R program to minimize

$$\mathbf{w}^\top \boldsymbol{\mu} - \lambda \frac{\mathbf{w}^\top \boldsymbol{\Sigma} \mathbf{w}}{2} \quad (16.21)$$

over  $\mathbf{w}$ , subject to  $\mathbf{w}^\top \mathbf{1} = 1$ , for each  $\lambda$  on a log-spaced grid. Plot the expected return and standard deviation of the return for the portfolios found this way and show that the curve coincides with the efficient frontier found in Problem 4. Select the range of the grid of log- $\lambda$  values by trial and error to cover an interesting range of the efficient frontier. What value of  $\lambda$  yields a portfolio with  $\mu_P = 0.046$ ? What value of  $\lambda$  yields to the tangency portfolio? What value of  $\lambda$  yields to the minimum variance portfolio?

### 16.10.3 Finding the Set of Possible Expected Returns

In Sect. 16.6 when we found the efficient frontier by quadratic programming, it was necessary to set up a grid of possible values of the expected returns on the portfolios. When there are no constraints on the allocation vector  $\mathbf{w}$  except that its elements sum to 1, any expected return is feasible.<sup>5</sup> We saw in Example 16.7, that if short sales are prohibited by the constraints  $0 \leq w_i \leq 1$  for all  $i$ , the the feasible expected portfolio returns lie between the smallest and largest expected returns on the individual assets.

When more complex constraints are placed on the  $w_i$ , the set of feasible expected portfolio returns can be found by linear programming. In this section, we use the same data as used in Sect. 16.10.1. We will impose the constraints that  $w_i \leq B1$  and  $-B2 \leq w_i$  for all  $i$ .

The function `solveLP()` in the `linprog` package minimizes (or maximizes) over  $N$ -dimensional variable  $\mathbf{x}$  the objection function  $\mathbf{c}^\top \mathbf{x}$  subject to  $\mathbf{Ax} \leq \mathbf{b}$  and  $\mathbf{x} \geq \mathbf{0}$ . Here  $\mathbf{c}$  is an  $N \times 1$  constant vector,  $\mathbf{A}$  is an  $N \times k$  constant matrix, and  $\mathbf{b}$  is a  $k \times 1$  constant vector for some integers  $N$  and  $k$ . Also,  $\mathbf{0}$  is a  $k$ -dimensional zero vector.

Since  $\mathbf{x} \geq \mathbf{1}$  we cannot let  $\mathbf{w}$  be  $\mathbf{x}$  unless we are prohibiting short sale. When short sales are allowed, we can instead let  $\mathbf{w}$  equal  $\mathbf{x}_1 - \mathbf{x}_2$  and  $\mathbf{x}^\top = (\mathbf{x}_1^\top, \mathbf{x}_2^\top)$ . Then the constraints are that each element of  $\mathbf{x}_1$  is at most  $B1$  and each element of  $\mathbf{x}_2$  is at most  $B2$ . The objective function  $\mathbf{w}^\top \boldsymbol{\mu}$  is equal to  $(\boldsymbol{\mu}^\top, -\boldsymbol{\mu}^\top) \mathbf{x}$ .

<sup>5</sup> “Feasible” means that there exists a vector  $\mathbf{w}$  achieving that expected return.

The constraints in  $\mathbf{A}\mathbf{x} \leq \mathbf{b}$  can be a mixture of equality and inequality constraints. The argument `const.dir` specifies the directions of the constraints; see line 17. In the program below, there are  $2M + 1$  constraints where  $M$  is the number of assets. The first  $M$  constraints are that  $w_i \leq B_1$  for all  $i$ , the next  $M$  constraints are that  $-B_2 \leq w_i$  for all  $i$ , and the last constraint is that  $\mathbf{w}^\top \mathbf{1} = 1$ .

The function `solveLP()` is used twice, once at lines 18 and 19 to find the smallest feasible expected portfolio return and then at lines 20 and 21 to find the largest possible expected return.

```

1 dat = read.csv("Stock_Bond.csv", header = T)
2 prices = cbind(dat$GM_AC, dat$F_AC, dat$CAT_AC, dat$UTX_AC,
3     dat$MRK_AC, dat$IBM_AC)
4 n = dim(prices)[1]
5 returns = 100 * (prices[2:n, ] / prices[1:(n-1), ] - 1)
6 mean_vect = colMeans(returns)
7 M = length(mean_vect)
8 B1 = 0.3
9 B2 = 0.1
10 library(linprog)
11 AmatLP1 = cbind(diag(1, nrow = M), matrix(0, nrow = M, ncol = M))
12 AmatLP2 = cbind(matrix(0, nrow = M, ncol = M), diag(1, nrow = M))
13 AmatLP3 = c(rep(1, M), rep(-1, M))
14 AmatLP = rbind(AmatLP1, AmatLP2, AmatLP3)
15 bvecLP = c(rep(B1, M), rep(B2, M), 1)
16 cLP = c(mean_vect, -mean_vect)
17 const.dir = c(rep("<=", 2 * M), "=")
18 resultLP_min = solveLP(cvec = cLP, bvec = bvecLP, Amat = AmatLP,
19     lpSolve=T, const.dir = const.dir, maximum = FALSE)
20 resultLP_max = solveLP(cvec = cLP, bvec = bvecLP,
21     Amat = AmatLP, lpSolve = TRUE, maximum = TRUE)
```

**Problem 6** What is the set of feasible expected portfolio returns when  $-0.1 \leq w_i \leq 0.3$  for all  $i$ ? What allocation vector  $\mathbf{w}$  achieve the smallest possible expected portfolio return? What allocation vector  $\mathbf{w}$  achieve the largest possible expected portfolio return?

**Problem 7** Would it be possible to use  $B_1 = 0.15$  and  $B_2 = 0.15$ ? Explain your answer.

## 16.11 Exercises

- Suppose that there are two risky assets, A and B, with expected returns equal to 2.3 % and 4.5 %, respectively. Suppose that the standard deviations of the returns are  $\sqrt{6}\%$  and  $\sqrt{11}\%$  and that the returns on the assets have a correlation of 0.17.

- (a) What portfolio of A and B achieves a 3% rate of expected return?  
 (b) What portfolios of A and B achieve a  $\sqrt{5.5}\%$  standard deviation of return? Among these, which has the largest expected return?
2. Suppose there are two risky assets, C and D, the tangency portfolio is 65% C and 35% D, and the expected return and standard deviation of the return on the tangency portfolio are 5% and 7%, respectively. Suppose also that the risk-free rate of return is 1.5%. If you want the standard deviation of your return to be 5%, what proportions of your capital should be in the risk-free asset, asset C, and asset D?
3. (a) Suppose that stock A shares sell at \$75 and stock B shares at \$115. A portfolio has 300 shares of stock A and 100 of stock B. What are the weights  $w$  and  $1 - w$  of stocks A and B in this portfolio?  
 (b) More generally, if a portfolio has  $N$  stocks, if the price per share of the  $j$ th stock is  $P_j$ , and if the portfolio has  $n_j$  shares of stock  $j$ , then find a formula for  $w_j$  as a function of  $n_1, \dots, n_N$  and  $P_1, \dots, P_N$ .
4. Let  $\mathcal{R}_P$  be a return of some type on a portfolio and let  $\mathcal{R}_1, \dots, \mathcal{R}_N$  be the same type of returns on the assets in this portfolio. Is

$$\mathcal{R}_P = w_1 \mathcal{R}_1 + \cdots + w_N \mathcal{R}_N$$

true if  $\mathcal{R}_P$  is a net return? Is this equation true if  $\mathcal{R}_P$  is a gross return? Is it true if  $\mathcal{R}_P$  is a log return? Justify your answers.

5. Suppose one has a sample of monthly log returns on two stocks with sample means of 0.0032 and 0.0074, sample variances of 0.017 and 0.025, and a sample covariance of 0.0059. For purposes of resampling, consider these to be the “true population values.” A bootstrap resample has sample means of 0.0047 and 0.0065, sample variances of 0.0125 and 0.023, and a sample covariance of 0.0058.
- (a) Using the resample, estimate the efficient portfolio of these two stocks that has an expected return of 0.005; that is, give the two portfolio weights.  
 (b) What is the estimated variance of the return of the portfolio in part (a) using the resample variances and covariances?  
 (c) What are the actual expected return and variance of return for the portfolio in (a) when calculated with the true population values (e.g., with using the original sample means, variances, and covariance)?
6. Stocks 1 and 2 are selling for \$100 and \$125, respectively. You own 200 shares of stock 1 and 100 shares of stock 2. The weekly returns on these stocks have means of 0.001 and 0.0015, respectively, and standard deviations of 0.03 and 0.04, respectively. Their weekly returns have a correlation of 0.35. Find the correlation matrix of the weekly returns on the two stocks and the mean and standard deviation of the weekly returns on the portfolio.

## References

- Bodie, Z., and Merton, R. C. (2000) *Finance*, Prentice-Hall, Upper Saddle River, NJ.
- Bodie, Z., Kane, A., and Marcus, A. (1999) *Investments*, 4th ed., Irwin/McGraw-Hill, Boston.
- Britten-Jones, M. (1999) The sampling error in estimates of mean-variance efficient portfolio weights. *Journal of Finance*, **54**, 655–671.
- Jagannathan, R. and Ma, T. (2003) Risk reduction in large portfolios: Why imposing the wrong constraints helps. *Journal of Finance*, **58**, 1651–1683.
- Jobson, J. D., and Korkie, B. (1980) Estimation for Markowitz efficient portfolios. *Journal of the American Statistical Association*, **75**, 544–554.
- Markowitz, H. (1952) Portfolio Selection. *Journal of Finance*, **7**, 77–91.
- Markowitz, H. (1959) *Portfolio Selection: Efficient Diversification of Investment*, Wiley, New York.
- Merton, R. C. (1972) An analytic derivation of the efficient portfolio frontier. *Journal of Financial and Quantitative Analysis*, **7**, 1851–1872.
- Michaud, R. O. (1998) *Efficient Asset Management: A Practical Guide to Stock Portfolio Optimization and Asset Allocation*, Harvard Business School Press, Boston.
- Ruppert, D. (2004) *Statistics and Finance: An Introduction*, Springer, New York.
- Sharpe, W. F., Alexander, G. J., and Bailey, J. V. (1999) *Investments*, 6th ed., Prentice-Hall, Upper Saddle River, NJ.

## The Capital Asset Pricing Model

### 17.1 Introduction to the CAPM

The *CAPM* (*capital asset pricing model*) has a variety of uses. It provides a theoretical justification for the widespread practice of passive investing by holding *index funds*.<sup>1</sup> The CAPM can provide estimates of expected rates of return on individual investments and can establish “fair” rates of return on invested capital in regulated firms or in firms working on a cost-plus basis.<sup>2</sup>

The CAPM starts with the question, what would be the risk premiums on securities if the following assumptions were true?

1. The market prices are “in equilibrium.” In particular, for each asset, supply equals demand.
2. Everyone has the same forecasts of expected returns and risks.
3. All investors choose portfolios optimally according to the principles of efficient diversification discussed in Chap. 16. This implies that everyone holds a tangency portfolio of risky assets as well as the risk-free asset.
4. The market rewards people for assuming unavoidable risk, but there is no reward for needless risks due to inefficient portfolio selection. Therefore, the risk premium on a single security is not due to its “standalone” risk, but rather to its contribution to the risk of the tangency portfolio. The various components of risk are discussed in Sect. 17.4.

---

<sup>1</sup> An index fund holds the same portfolio as some index. For example, an S&P 500 index fund holds all 500 stocks on the S&P 500 in the same proportions as in the index. Some funds do not replicate an index exactly, but are designed to track the index, for instance, by being cointegrated with the index.

<sup>2</sup> See Bodie and Merton (2000).

Assumption 3 implies that the market portfolio is equal to the tangency portfolio. Therefore, a broad index fund that mimics the market portfolio can be used as an approximation to the tangency portfolio.

The validity of the CAPM can only be guaranteed if all of these assumptions are true, and certainly no one believes that any of them are exactly true. Assumption 3 is at best an idealization. Moreover, some of the conclusions of the CAPM are contradicted by the behavior of financial markets; see Sect. 18.4.1 for an example. Despite its shortcomings, the CAPM is widely used in finance and it is essential for a student of finance to understand the CAPM. Many of its concepts such as the beta of an asset and systematic and diversifiable risks are of great importance, and the CAPM has been generalized to the widely used factor models introduced in Chap. 18.

## 17.2 The Capital Market Line (CML)

The *capital market line* (CML) relates the excess expected return on an efficient portfolio to its risk. *Excess expected return* is the expected return minus the risk-free rate and is also called the risk premium. The CML is

$$\mu_R = \mu_f + \frac{\mu_M - \mu_f}{\sigma_M} \sigma_R, \quad (17.1)$$

where  $R$  is the return on a given efficient portfolio (mixture of the market portfolio [= tangency portfolio] and the risk-free asset),  $\mu_R = E(R)$ ,  $\mu_f$  is the risk-free rate,  $R_M$  is the return on the market portfolio,  $\mu_M = E(R_M)$ ,  $\sigma_M$  is the standard deviation of  $R_M$ , and  $\sigma_R$  is the standard deviation of  $R$ . The risk premium of  $R$  is  $\mu_R - \mu_f$  and the risk premium of the market portfolio is  $\mu_M - \mu_f$ .

In (17.1)  $\mu_f$ ,  $\mu_M$ , and  $\sigma_M$  are constant. What varies are  $\sigma_R$  and  $\mu_R$ . These vary as we change the efficient portfolio  $R$ . Think of the CML as showing how  $\mu_R$  depends on  $\sigma_R$ .

The slope of the CML is, of course,

$$\frac{\mu_M - \mu_f}{\sigma_M},$$

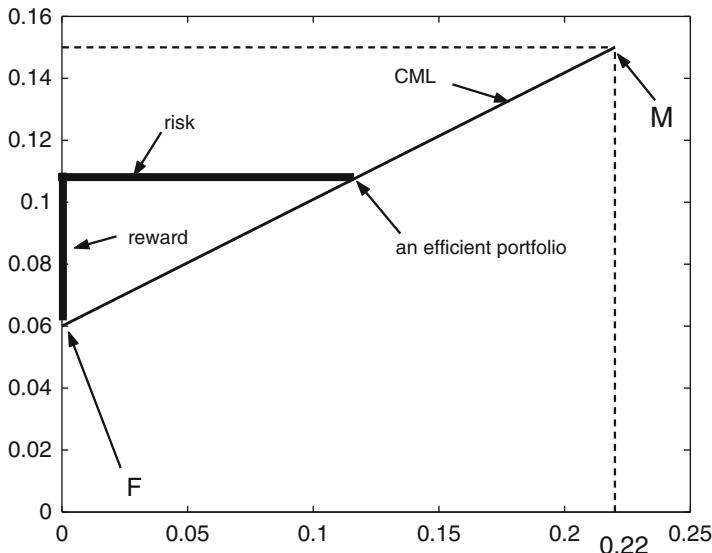
which can be interpreted as the ratio of the risk premium to the standard deviation of the market portfolio. This is Sharpe's famous "reward-to-risk ratio," which is widely used in finance. Equation (17.1) can be rewritten as

$$\frac{\mu_R - \mu_f}{\sigma_R} = \frac{\mu_M - \mu_f}{\sigma_M},$$

which says that the reward-to-risk ratio for any efficient portfolio equals that ratio for the market portfolio—all efficient portfolios have the same Sharpe's ratio as the market portfolio.

*Example 17.1. The CML*

Suppose that the risk-free rate of interest is  $\mu_f = 0.06$ , the expected return on the market portfolio is  $\mu_M = 0.15$ , and the risk of the market portfolio is  $\sigma_M = 0.22$ . Then the slope of the CML is  $(0.15 - 0.06)/0.22 = 9/22$ . The CML of this example is illustrated in Fig. 17.1.  $\square$



**Fig. 17.1.** *CML when  $\mu_f = 0.06$ ,  $\mu_M = 0.15$ , and  $\sigma_M = 0.22$ . All efficient portfolios are on the line connecting the risk-free asset ( $F$ ) and the market portfolio ( $M$ ). Therefore, the reward-to-risk ratio is the same for all efficient portfolios, including the market portfolio. This fact is illustrated by the thick lines, whose lengths are the risk and reward for a typical efficient portfolio.*

The CML is easy to derive. Consider an efficient portfolio that allocates a proportion  $w$  of its assets to the market portfolio and  $(1 - w)$  to the risk-free asset. Then

$$R = wR_M + (1 - w)\mu_f = \mu_f + w(R_M - \mu_f). \quad (17.2)$$

Therefore, taking expectations in (17.2),

$$\mu_R = \mu_f + w(\mu_M - \mu_f). \quad (17.3)$$

Also, from (17.2),

$$\sigma_R = w\sigma_M, \quad (17.4)$$

or

$$w = \frac{\sigma_R}{\sigma_M}. \quad (17.5)$$

Substituting (17.5) into (17.3) gives the CML.

The CAPM says that the optimal way to invest is to

1. decide on the risk  $\sigma_R$  that you can tolerate,  $0 \leq \sigma_R \leq \sigma_M$ <sup>3</sup>;
2. calculate  $w = \sigma_R/\sigma_M$ ;
3. invest  $w$  proportion of your investment in a market index fund, that is, a fund that tracks the market as a whole;
4. invest  $1 - w$  proportion of your investment in risk-free Treasury bills, or a money-market fund.

Alternatively,

1. choose the reward  $\mu_R - \mu_f$  that you want; the only constraint is that  $\mu_f \leq \mu_R \leq \mu_M$  so that  $0 \leq w \leq 1$ <sup>4</sup>;
2. calculate

$$w = \frac{\mu_R - \mu_f}{\mu_M - \mu_f};$$

3. do steps 3 and 4 as above.

Instead of specifying the expected return or standard deviation of return, as in Example 16.1 one can find the portfolio with the highest expected return subject to a guarantee that with confidence  $1 - \alpha$  the maximum loss is below a prescribed bound  $M$  determined, say, by a firm's capital reserves. If the firm invests an amount  $C$ , then for the loss to be greater than  $M$  the return must be less than  $-M/C$ . If we assume that the return is normally distributed, then by (A.11), (17.3), and (17.4),

$$P\left(R < -\frac{M}{C}\right) = \Phi\left(\frac{-M/C - \{\mu_f + w(\mu_M - \mu_f)\}}{w\sigma_M}\right). \quad (17.6)$$

Thus, we solve the following equation for  $w$ :

$$\Phi^{-1}(\alpha) = \frac{-M/C - \{\mu_f + w(\mu_M - \mu_f)\}}{w\sigma_M}.$$

One can view  $w = \sigma_R/\sigma_M$  as an index of the risk aversion of the investor. The smaller the value of  $w$  the more risk-averse the investor. If an investor has  $w$  equal to 0, then that investor is 100% in risk-free assets. Similarly, an investor with  $w = 1$  is totally invested in the tangency portfolio of risky assets.<sup>5</sup>

---

<sup>3</sup> In fact,  $\sigma_R > \sigma_M$  is possible by borrowing money to buy risky assets on margin.

<sup>4</sup> This constraint can be relaxed if one is permitted to buy assets on margin.

<sup>5</sup> An investor with  $w > 1$  is buying the market portfolio on margin, that is, borrowing money to buy the market portfolio.

## 17.3 Betas and the Security Market Line

The *security market line* (SML) relates the excess return on an asset to the slope of its regression on the market portfolio. The SML differs from the CML in that the SML applies to all assets while the CML applies only to efficient portfolios.

Suppose that there are many securities indexed by  $j$ . Define

$\sigma_{jM}$  = covariance between the returns on the  $j$ th security  
and the market portfolio.

Also, define

$$\beta_j = \frac{\sigma_{jM}}{\sigma_M^2}. \quad (17.7)$$

It follows from the theory of best linear prediction in Sect. 11.9.1 that  $\beta_j$  is the slope of the best linear predictor of the  $j$ th security's returns using returns of the market portfolio as the predictor variable. This fact follows from equation (11.37) for the slope of a best linear prediction equation. In fact, the best linear predictor of  $R_j$  based on  $R_M$  is

$$\hat{R}_j = \beta_{0,j} + \beta_j R_M, \quad (17.8)$$

where  $\beta_j$  in (17.8) is the same as in (17.7). Also,  $\beta_{0,j}$  is the intercept that can be calculated by taking expectations in (17.8) and solving to obtain  $\beta_{0,j} = E(R_j) - \beta_j E(R_M)$ .

Another way to appreciate the significance of  $\beta_j$  uses linear regression. As discussed in Sect. 11.9, linear regression is a method for estimating the coefficients of the best linear predictor based upon data. To apply linear regression, suppose that we have a bivariate time series  $(R_{j,t}, R_{M,t})_{t=1}^n$  of returns on the  $j$ th asset and the market portfolio. Then, the estimated slope of the linear regression of  $R_{j,t}$  on  $R_{M,t}$  is

$$\hat{\beta}_j = \frac{\sum_{t=1}^n (R_{j,t} - \bar{R}_j)(R_{M,t} - \bar{R}_M)}{\sum_{t=1}^n (R_{M,t} - \bar{R}_M)^2}, \quad (17.9)$$

which, after multiplying the numerator and denominator by the same factor  $n^{-1}$ , becomes an estimate of  $\sigma_{jM}$  divided by an estimate of  $\sigma_M^2$  and therefore by (17.7) an estimate of  $\beta_j$ .

Let  $\mu_j$  be the expected return on the  $j$ th security. Then  $\mu_j - \mu_f$  is the *risk premium* (or *reward for risk* or *excess expected return*) for that security. Using the CAPM, it can be shown that

$$\mu_j - \mu_f = \beta_j(\mu_M - \mu_f). \quad (17.10)$$

This equation, which is called the security market line (SML), is derived in Sect. 17.5.2. In (17.10)  $\beta_j$  is a variable in the linear equation, not the slope;

more precisely,  $\mu_j$  is a linear function of  $\beta_j$  with slope  $\mu_M - \mu_f$ . This point is worth remembering. Otherwise, there could be some confusion since  $\beta_j$  was defined earlier as a slope of a regression model. In other words,  $\beta_j$  is a slope in one context but is the independent variable in the different context of the SML. One can estimate  $\beta_j$  using (17.9) and then plug this estimate into (17.10).

The SML says that the risk premium of the  $j$ th asset is the product of its beta ( $\beta_j$ ) and the risk premium of the market portfolio ( $\mu_M - \mu_f$ ). Therefore,  $\beta_j$  measures both the riskiness of the  $j$ th asset and the reward for assuming that riskiness. Consequently,  $\beta_j$  is a measure of how “aggressive” the  $j$ th asset is. By definition, the beta for the market portfolio is 1; i.e.,  $\beta_M = 1$ . This suggest the rules-of-thumb

$$\begin{aligned}\beta_j > 1 &\Rightarrow \text{“aggressive,”} \\ \beta_j = 1 &\Rightarrow \text{“average risk,”} \\ \beta_j < 1 &\Rightarrow \text{“not aggressive.”}\end{aligned}$$

Figure 17.2 illustrates the SML and an asset J that is not on the SML. This asset contradicts the CAPM, because according to the CAPM all assets are on the SML so no such asset exists.

Consider what would happen if an asset like J did exist. Investors would not want to buy it because, since it is below the SML, its risk premium is too low for the risk given by its beta. They would invest less in J and more in other securities. Therefore, the price of J would decline and *after* this decline its expected return would increase. After that increase, the asset J would be on the SML, or so the theory predicts.

### 17.3.1 Examples of Betas

Table 17.1 has some “five-year betas” taken from the Salomon, Smith, Barney website between February 27 and March 5, 2001. The beta for the S&P 500 is given as 1.00; why?

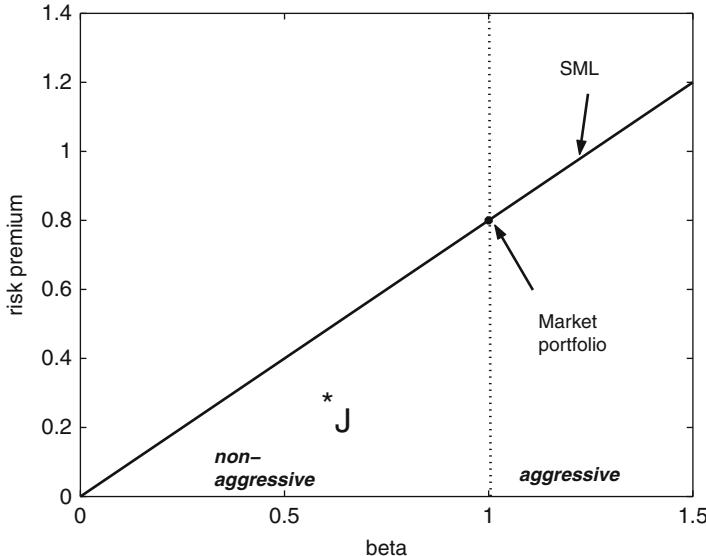
### 17.3.2 Comparison of the CML with the SML

The CML applies only to the return  $R$  of an efficient portfolio. It can be arranged so as to relate the excess expected return of that portfolio to the excess expected return of the market portfolio:

$$\mu_R - \mu_f = \left( \frac{\sigma_R}{\sigma_M} \right) (\mu_M - \mu_f). \quad (17.11)$$

The SML applies to *any* asset and like the CML relates its excess expected return to the excess expected return of the market portfolio:

$$\mu_j - \mu_f = \beta_j (\mu_M - \mu_f). \quad (17.12)$$



**Fig. 17.2.** Security market line (SML) showing that the risk premium of an asset is a linear function of the asset's beta.  $J$  is a security not on the line and a contradiction to the CAPM. Theory predicts that the price of  $J$  decreases until  $J$  is on the SML. The vertical dotted line separates the nonaggressive and aggressive regions.

If we take an efficient portfolio and consider it as an asset, then  $\mu_R$  and  $\mu_j$  both denote the expected return on that portfolio/asset. Both (17.11) and (17.12) hold so that

$$\frac{\sigma_R}{\sigma_M} = \beta_R.$$

## 17.4 The Security Characteristic Line

Let  $R_{j,t}$  be the return at time  $t$  on the  $j$ th asset. Similarly, let  $R_{M,t}$  and  $\mu_{f,t}$  be the return on the market portfolio and the risk-free return at time  $t$ . The *security characteristic line* (sometimes shortened to the characteristic line) is a regression model:

$$R_{j,t} = \mu_{f,t} + \beta_j(R_{M,t} - \mu_{f,t}) + \epsilon_{j,t}, \quad (17.13)$$

where  $\epsilon_{j,t}$  is  $N(0, \sigma_{\epsilon,j}^2)$ . It is often assumed that the  $\epsilon_{j,t}$ s are uncorrelated across assets, that is, that  $\epsilon_{j,t}$  is uncorrelated with  $\epsilon_{j',t}$  for  $j \neq j'$ . This assumption has important ramifications for risk reduction by diversification; see Sect. 17.4.1.

**Table 17.1.** Selected stocks and in which industries they are. Betas are given for each stock (Stock's  $\beta$ ) and its industry (Ind's  $\beta$ ). Betas taken from the Salomon, Smith, Barney website between February 27 and March 5, 2001.

| Stock (symbol)        | Industry             | Stock's $\beta$ | Ind's $\beta$ |
|-----------------------|----------------------|-----------------|---------------|
| Celanese (CZ)         | Synthetics           | 0.13            | 0.86          |
| General Mills (GIS)   | Food—major diversif  | 0.29            | 0.39          |
| Kellogg (K)           | Food—major, diversif | 0.30            | 0.39          |
| Proctor & Gamble (PG) | Cleaning prod        | 0.35            | 0.40          |
| Exxon-Mobil (XOM)     | Oil/gas              | 0.39            | 0.56          |
| 7-Eleven (SE)         | Grocery stores       | 0.55            | 0.38          |
| Merck (Mrk)           | Major drug manuf     | 0.56            | 0.62          |
| McDonalds (MCD)       | Restaurants          | 0.71            | 0.63          |
| McGraw-Hill (MHP)     | Pub—books            | 0.87            | 0.77          |
| Ford (F)              | Auto                 | 0.89            | 1.00          |
| Aetna (AET)           | Health care plans    | 1.11            | 0.98          |
| General Motors (GM)   | Major auto manuf     | 1.11            | 1.09          |
| AT&T (T)              | Long dist carrier    | 1.19            | 1.34          |
| General Electric (GE) | Conglomerates        | 1.22            | 0.99          |
| Genentech (DNA)       | Biotech              | 1.43            | 0.69          |
| Microsoft (MSFT)      | Software applic.     | 1.77            | 1.72          |
| Cree (Cree)           | Semicond equip       | 2.16            | 2.30          |
| Amazon (AMZN)         | Net soft & serv      | 2.99            | 2.46          |
| DoubleClick (Dclk)    | Net soft & serv      | 4.06            | 2.46          |

Let  $\mu_{j,t} = E(R_{j,t})$  and  $\mu_{M,t} = E(R_{M,t})$ . Taking expectations in (17.13) we get,

$$\mu_{j,t} = \mu_{f,t} + \beta_j(\mu_{M,t} - \mu_{f,t}),$$

which is equation (17.10), the SML, though in (17.10) it is not shown explicitly that the expected returns can depend on  $t$ . The SML gives us information about expected returns, but not about the variance of the returns. For the latter we need the characteristic line. The characteristic line is said to be a *return-generating process* since it gives us a probability model of the returns, not just a model of their expected values.

An analogy to the distinction between the SML and characteristic line is this. The regression line  $E(Y|X) = \beta_0 + \beta_1 X$  gives the expected value of  $Y$  given  $X$  but not the conditional probability distribution of  $Y$  given  $X$ . The regression model

$$Y_t = \beta_0 + \beta_1 X_t + \epsilon_t \quad \text{and} \quad \epsilon_t \sim N(0, \sigma^2)$$

does give us this conditional probability distribution.

The characteristic line implies that

$$\sigma_j^2 = \beta_j^2 \sigma_M^2 + \sigma_{\epsilon,j}^2,$$

that

$$\sigma_{jj'} = \beta_j \beta_{j'} \sigma_M^2 \quad (17.14)$$

for  $j \neq j'$ , and that

$$\sigma_{Mj} = \beta_j \sigma_M^2.$$

For (17.14) to hold,  $\epsilon_{j,t}$  and  $\epsilon_{j',t}$  must be uncorrelated. The total risk of the  $j$ th asset is

$$\sigma_j = \sqrt{\beta_j^2 \sigma_M^2 + \sigma_{\epsilon,j}^2}.$$

The squared risk has two components:  $\beta_j^2 \sigma_M^2$  is called the *market* or *systematic component of risk* and  $\sigma_{\epsilon,j}^2$  is called the *unique, nonmarket*, or *unsystematic component of risk*.

#### 17.4.1 Reducing Unique Risk by Diversification

The market component of risk cannot be reduced by diversification, but the unique component can be reduced or even eliminated by sufficient diversification.

Suppose that there are  $N$  assets with returns  $R_{1,t}, \dots, R_{N,t}$  for holding period  $t$ . If we form a portfolio with weights  $w_1, \dots, w_N$ , then the return of the portfolio is

$$R_{P,t} = w_1 R_{1,t} + \dots + w_N R_{N,t}.$$

Let  $R_{M,t}$  be the return on the market portfolio. According to the characteristic line model  $R_{j,t} = \mu_{f,t} + \beta_j(R_{M,t} - \mu_{f,t}) + \epsilon_{j,t}$ , so that

$$R_{P,t} = \mu_{f,t} + \left( \sum_{j=1}^N \beta_j w_j \right) (R_{M,t} - \mu_{f,t}) + \sum_{j=1}^N w_j \epsilon_{j,t}.$$

Therefore, the portfolio beta is

$$\beta_P = \sum_{j=1}^N w_j \beta_j,$$

and the “epsilon” for the portfolio is

$$\epsilon_{P,t} = \sum_{j=1}^N w_j \epsilon_{j,t}.$$

We now assume that  $\epsilon_{1,t}, \dots, \epsilon_{N,t}$  are uncorrelated. Therefore, by equation (7.11),

$$\sigma_{\epsilon,P}^2 = \sum_{j=1}^N w_j^2 \sigma_{\epsilon,j}^2.$$

*Example 17.2. Reduction in risk by diversification*

Suppose the assets in the portfolio are equally weighted; that is,  $w_j = 1/N$  for all  $j$ . Then

$$\beta_P = \frac{\sum_{j=1}^N \beta_j}{N},$$

and

$$\sigma_{\epsilon,P}^2 = \frac{N^{-1} \sum_{j=1}^N \sigma_{\epsilon,j}^2}{N} = \frac{\bar{\sigma}_{\epsilon}^2}{N},$$

where  $\bar{\sigma}_{\epsilon}^2$  is the average of the  $\sigma_{\epsilon,j}^2$ .

As an illustration, if we assume the simple case where  $\sigma_{\epsilon,j}^2$  is a constant, say  $\sigma_{\epsilon}^2$ , for all  $j$ , then

$$\sigma_{\epsilon,P} = \frac{\sigma_{\epsilon}}{\sqrt{N}}. \quad (17.15)$$

For example, suppose that  $\sigma_{\epsilon}$  is 5%. If  $N = 20$ , then by (17.15)  $\sigma_{\epsilon,P}$  is 1.12%. If  $N = 100$ , then  $\sigma_{\epsilon,P}$  is 0.5%. There are approximately 1600 stocks on the NYSE; if  $N = 1600$ , then  $\sigma_{\epsilon,P} = 0.125\%$ , a remarkable reduction from 5%.

□

#### 17.4.2 Are the Assumptions Sensible?

A key assumption that allows nonmarket risk to be removed by diversification is that  $\epsilon_{1,t}, \dots, \epsilon_{N,t}$  are uncorrelated. This assumption implies that *all* correlation among the cross-section<sup>6</sup> of asset returns is due to a single cause and that cause is measured by the market index. For this reason, the characteristic line is a “single-factor” or “single-index” model with  $R_{M,t}$  being the “factor.”

This assumption of uncorrelated  $\epsilon_{jt}$  would not be valid if, for example, two energy stocks are correlated over and beyond their correlation due to the market index. In this case, unique risk could not be eliminated by holding a large portfolio of all energy stocks. However, if there are many market sectors and the sectors are uncorrelated, then one could eliminate nonmarket risk by diversifying across all sectors. All that is needed is to treat the sectors themselves as the underlying assets and then apply the CAPM theory.

Correlation among the stocks in a market sector can be modeled using a factor model; see Chap. 18.

## 17.5 Some More Portfolio Theory

In this section we use portfolio theory to show that  $\sigma_{j,M}$  quantifies the contribution of the  $j$ th asset to the risk of the market portfolio. Also, we derive the SML.

---

<sup>6</sup> “Cross-section” of returns means returns across assets within a *single* holding period.

### 17.5.1 Contributions to the Market Portfolio's Risk

Suppose that the market consists of  $N$  risky assets and that  $w_{1,M}, \dots, w_{N,M}$  are the weights of these assets in the market portfolio. Then

$$R_{M,t} = \sum_{i=1}^N w_{i,M} R_{i,t},$$

which implies that the covariance between the return on the  $j$ th asset and the return on the market portfolio is

$$\sigma_{j,M} = \text{Cov} \left( R_{j,t}, \sum_{i=1}^N w_{i,M} R_{i,t} \right) = \sum_{i=1}^N w_{i,M} \sigma_{i,j}. \quad (17.16)$$

Therefore,

$$\sigma_M^2 = \sum_{j=1}^N \sum_{i=1}^N w_{j,M} w_{i,M} \sigma_{i,j} = \sum_{j=1}^N w_{j,M} \left( \sum_{i=1}^N w_{i,M} \sigma_{i,j} \right) = \sum_{j=1}^N w_{j,M} \sigma_{j,M}. \quad (17.17)$$

Equation (17.17) shows that the contribution of the  $j$ th asset to the risk of the market portfolio is  $w_{j,M} \sigma_{j,M}$ , where  $w_{j,M}$  is the weight of the  $j$ th asset in the market portfolio and  $\sigma_{j,M}$  is the covariance between the return on the  $j$ th asset and the return on the market portfolio.

### 17.5.2 Derivation of the SML

The derivation of the SML is a nice application of portfolio theory, calculus, and geometric reasoning. It is based on a clever idea of putting together a portfolio with two assets, the market portfolio and the  $i$ th risky asset, and then looking at the locus in reward-risk space as the portfolio weight assigned to the  $i$ th risky asset varies.

Consider a portfolio  $P$  with weight  $w_i$  given to the  $i$ th risky asset and weight  $(1 - w_i)$  given to the market (tangency) portfolio. The return on this portfolio is

$$R_{P,t} = w_i R_{i,t} + (1 - w_i) R_{M,t}.$$

The expected return is

$$\mu_P = w_i \mu_i + (1 - w_i) \mu_M, \quad (17.18)$$

and the risk is

$$\sigma_P = \sqrt{w_i^2 \sigma_i^2 + (1 - w_i)^2 \sigma_M^2 + 2w_i(1 - w_i)\sigma_{i,M}}. \quad (17.19)$$

As we vary  $w_i$ , we get the locus of points on  $(\sigma, \mu)$  space that is shown as a dashed curve in Fig. 17.3, which uses the same returns as in Fig. 16.3 and Mobil stock as asset  $i$ .

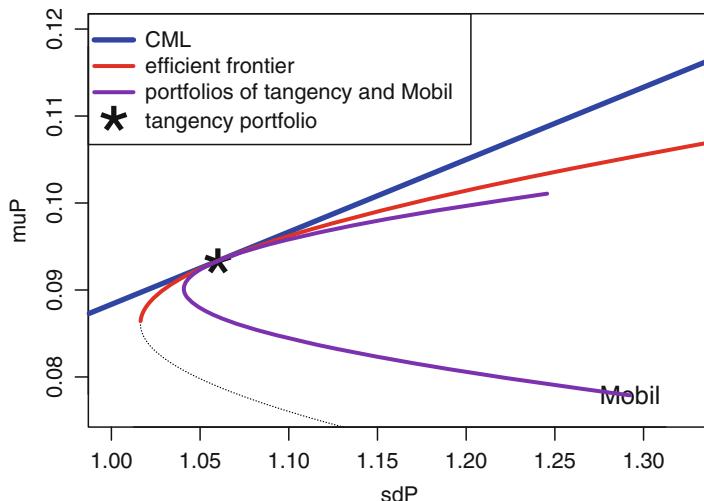
It is easy to see geometrically that the derivative of this locus of points evaluated at the tangency portfolio (which is the point where  $w_i = 0$ ) is equal to the slope of the CML. We can calculate this derivative and equate it to the slope of the CML to see what we get. We will see that the result is the SML.

We have from (17.18)

$$\frac{d\mu_P}{d w_i} = \mu_i - \mu_M,$$

and from (17.19) that

$$\frac{d\sigma_P}{d w_i} = \frac{1}{2}\sigma_P^{-1} \left\{ 2w_i\sigma_i^2 - 2(1-w_i)\sigma_M^2 + 2(1-2w_i)\sigma_{i,M} \right\}.$$



**Fig. 17.3.** Derivation of the SML. The purple curve is the locus of portfolios combining Mobil stock and the tangency portfolio (asterisk). The purple curve is to the right of the efficient frontier (red) and intersects the efficient frontier at the tangency portfolio. Therefore, the derivative of the purple curve at the tangency portfolio is equal to the slope of the CML (blue), since the purple curve is tangent to the CML at the tangency portfolio.

Therefore,

$$\frac{d\mu_P}{d\sigma_P} = \frac{d\mu_P/dw_i}{d\sigma_P/dw_i} = \frac{(\mu_i - \mu_M)\sigma_P}{w_i\sigma_i^2 - \sigma_M^2 + w_i\sigma_M^2 + \sigma_{i,M} - 2w_i\sigma_{i,M}}.$$

Next,

$$\frac{d\mu_P}{d\sigma_P} \Big|_{w_i=0} = \frac{(\mu_i - \mu_M)\sigma_M}{\sigma_{i,M} - \sigma_M^2}.$$

Recall that  $w_i = 0$  is the tangency portfolio, the point in Fig. 17.3 where the dashed locus is tangent to the CML. Therefore,

$$\frac{d\mu_P}{d\sigma_P} \Big|_{w_i=0}$$

must equal the slope of the CML, which is  $(\mu_M - \mu_f)/\sigma_M$ . Therefore,

$$\frac{(\mu_i - \mu_M)\sigma_M}{\sigma_{i,M}^2 - \sigma_M^2} = \frac{\mu_M - \mu_f}{\sigma_M},$$

which, after some algebra, gives us

$$\mu_i - \mu_f = \frac{\sigma_{i,M}}{\sigma_M^2}(\mu_M - \mu_f) = \beta_i(\mu_M - \mu_f),$$

which is the SML given in equation (17.10).

## 17.6 Estimation of Beta and Testing the CAPM

### 17.6.1 Estimation Using Regression

Recall the security characteristic line

$$R_{j,t} = \mu_{f,t} + \beta_j(R_{M,t} - \mu_{f,t}) + \epsilon_{j,t}. \quad (17.20)$$

Let  $R_{j,t}^* = R_{j,t} - \mu_{f,t}$  be the excess return on the  $j$ th security and let  $R_{M,t}^* = R_{M,t} - \mu_{f,t}$ , be the excess return on the market portfolio. Then (17.20) can be written as

$$R_{j,t}^* = \beta_j R_{M,t}^* + \epsilon_{j,t}. \quad (17.21)$$

Equation (17.21) is a regression model without an intercept and with  $\beta_j$  as the slope. A more elaborate model is

$$R_{j,t}^* = \alpha_j + \beta_j R_{M,t}^* + \epsilon_{j,t}, \quad (17.22)$$

which includes an intercept. The CAPM says that  $\alpha_j = 0$  but by allowing  $\alpha_j \neq 0$ , we recognize the possibility of mispricing.

Given time series  $R_{j,t}$ ,  $R_{M,t}$ , and  $\mu_{f,t}$  for  $t = 1, \dots, n$ , we can calculate  $R_{j,t}^*$  and  $R_{M,t}^*$  and regress  $R_{j,t}^*$  on  $R_{M,t}^*$  to estimate  $\alpha_j$ ,  $\beta_j$ , and  $\sigma_{\epsilon,j}^2$ . By testing the null hypothesis that  $\alpha_j = 0$ , we are testing whether the  $j$ th asset is mispriced according to the CAPM.

As discussed in Sect. 9.2.2, when fitting model (17.21) or (17.22) one should use daily data if available, rather than weekly or monthly data. A more difficult question to answer is how long a time series to use. Longer time series give more data, of course, but models (17.21) and (17.22) assume that  $\beta_j$  is constant and this might not be true over a long time period.

*Example 17.3. Estimation of  $\alpha$  and  $\beta$  for Microsoft*

As an example, daily closing prices on Microsoft and the S&P 500 index from November 1, 1993, to April 3, 2003, were used. The S&P 500 was taken as the market price. Three-month T-bill rates were used as the risk-free returns.<sup>7</sup> The excess returns are the returns minus the T-bill rates. The code is

```
dat = read.csv("capm.csv", header = TRUE)
attach(dat)
n = dim(dat)[1]
EX_R_sp500 = Close.sp500[2:n] / Close.sp500[1:(n-1)]
- 1 - Close.tbill[2:n] / (100 * 253)
EX_R_msft = Close.msft[2:n] / Close.msft[1:(n-1)]
- 1 - Close.tbill[2:n] / (100 * 253)
fit = lm(EX_R_msft ~ EX_R_sp500)
options(digits = 3)
summary(fit)
```

and the output is

```
Call:
lm(formula = EX_R_msft ~ EX_R_sp500)

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 0.000914   0.000409    2.23   0.026 *
EX_R_sp500  1.247978   0.035425   35.23  <2e-16 ***
---
Residual standard error: 0.0199 on 2360 degrees of freedom
Multiple R-squared:  0.345, Adjusted R-squared:  0.344
F-statistic: 1.24e+03 on 1 and 2360 DF,  p-value: <2e-16
```

For Microsoft, we find that  $\hat{\beta} = 1.25$  and  $\hat{\alpha} = 0.0009$ . The estimate of  $\alpha$  is very small and, although the  $p$ -value for  $\alpha$  is 0.026, we can conclude that for practical purposes,  $\alpha$  is essentially 0. This is another example of an effect being statistically significant according to a test of the hypothesis of no effect but not practically significant. Very small effects are often statistically significant when the sample size is large. In this example, we have nearly 10 years of daily data and the sample size is quite large for a hypothesis testing problem, 2363.

The estimate of  $\sigma_\epsilon$  is the root MSE which equals 0.0199. Notice that the  $R^2$  (R-sq) value for the regression is 34.5 %. The interpretation of  $R^2$  is the percent of the variance in the excess returns on Microsoft that is due to excess returns on the market. In other words, 34.5 % of the squared risk is due to

---

<sup>7</sup> Interest rates are return rates. Thus, we use the T-bill rates themselves as the risk-free returns. One does *not* take logs and difference the T-bill rates as if they were prices. However, the T-bill rates were divided by 100 to convert from a percentage and then by 253 to convert to a daily rate.

systematic or market risk ( $\beta_j^2 \sigma_M^2$ ). The remaining 65.5% is due to unique or nonmarket risk ( $\sigma_\epsilon^2$ ).

If we assume that  $\alpha = 0$ , then we can refit the model using a no-intercept model. The code for fitting the model is changed to

```
fit_NoInt = lm(EX_R_msft ~ EX_R_sp500 - 1)
options(digits = 3)
summary(fit_NoInt)
```

Notice the “ $-1$ ” in the formula. The “ $1$ ” represents the intercept so “ $-1$ ” indicates that the intercept is removed. The output changes to

```
Call:
lm(formula = EX_R_msft ~ EX_R_sp500 - 1)

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
EX_R_sp500   1.2491    0.0355   35.2 <2e-16 ***
---
Residual standard error: 0.0199 on 2361 degrees of freedom
Multiple R-squared:  0.345, Adjusted R-squared:  0.344
F-statistic: 1.24e+03 on 1 and 2361 DF,  p-value: <2e-16
```

With no intercept  $\hat{\beta}$ ,  $\hat{\sigma}_\epsilon$  and  $R^2$  are nearly the same as before—forcing a nearly zero intercept to be exactly zero has little effect.  $\square$

## 17.6.2 Testing the CAPM

Testing that  $\alpha$  equals 0 tests only one of the conclusions of the CAPM. Accepting this null hypothesis only means that the CAPM has passed one test, not that we should now accept it as true.<sup>8</sup> To fully test the CAPM, its other conclusions should also be tested. The factor models in Sect. 18.3 have been used to test the CAPM and fairly strong evidence against the CAPM has been found. Fortunately, these factor models do provide a generalization of the CAPM that is likely to be useful for financial decision making.

Often, as an alternative to regression using excess returns, the returns on the asset are regressed on the returns on the market. When this is done, an intercept model should be used. In the Microsoft data when using returns instead of excess returns, the estimate of beta changed hardly at all.

## 17.6.3 Interpretation of Alpha

If  $\alpha$  is nonzero, then the security is mispriced, at least according to the CAPM. If  $\alpha > 0$  then the security is underpriced; the returns are too large on average.

---

<sup>8</sup> In fact, acceptance of a null hypothesis should never be interpreted as proof that the null hypothesis is true.

This is an indication of an asset worth purchasing. Of course, one must be careful. If we reject the null hypothesis that  $\alpha = 0$ , all we have done is to show that the security was mispriced *in the past*.

*Warning:* If we use returns rather than excess returns, then the intercept of the regression equation does *not* estimate  $\alpha$ , so one cannot test whether  $\alpha$  is zero by testing the intercept.

## 17.7 Using the CAPM in Portfolio Analysis

Suppose we have estimated beta and  $\sigma_\epsilon^2$  for each asset in a portfolio and also estimated  $\sigma_M^2$  and  $\mu_M$  for the market. Then, since  $\mu_f$  is also known, we can compute the expectations, variances, and covariances of all asset returns by the formulas

$$\begin{aligned}\mu_j &= \mu_f + \beta_j(\mu_M - \mu_f), \\ \sigma_j^2 &= \beta_j^2\sigma_M^2 + \sigma_{\epsilon,j}^2, \\ \sigma_{jj'} &= \beta_j\beta_{j'}\sigma_M^2 \text{ for } j \neq j'.\end{aligned}$$

There is a noteworthy danger here: These estimates depend heavily on the validity of the CAPM assumptions. Any or all of the quantities beta,  $\sigma_\epsilon^2$ ,  $\sigma_M^2$ ,  $\mu_M$ , and  $\mu_f$  could depend on time  $t$ . However, it is generally assumed that the betas and  $\sigma_\epsilon^2$ s of the assets as well as  $\sigma_M^2$  and  $\mu_M$  of the market are independent of  $t$  so that these parameters can be estimated assuming stationarity of the time series of returns.

## 17.8 Bibliographic Notes

The CAPM was developed by Sharpe (1964), Lintner (1965a,b), and Mossin (1966). Introductions to the CAPM can be found in Bodie, Kane, and Marcus (1999), Bodie and Merton (2000), and Sharpe, Alexander, and Bailey (1999). I first learned about the CAPM from these three textbooks. Campbell, Lo, and MacKinlay (1997) discuss empirical testing of the CAPM. The derivation of the SML in Sect. 17.5.2 was adapted from Sharpe, Alexander, and Bailey (1999). Discussion of factor models can be found in Sharpe, Alexander, and Bailey (1999), Bodie, Kane, and Marcus (1999), and Campbell, Lo, and MacKinlay (1997).

## 17.9 R Lab

In this lab, you will fit model (17.20). The S&P 500 index will be a proxy for the market portfolio and the 90-day Treasury rate will serve as the risk-free rate.

This lab uses the data set `Stock_Bond_2004_to_2006.csv`, which is available on the book's website. This data set contains a subset of the data in the data set `Stock_Bond.csv` used elsewhere.

The R commands needed to fit model (17.20) will be given in small groups so that they can be explained better. First run the following commands to read the data, extract the prices, and find the number of observations:

```
dat = read.csv("Stock_Bond_2004_to_2006.csv", header = TRUE)
prices = dat[ , c(5, 7, 9, 11, 13, 15, 17, 24)]
n = dim(prices)[1]
```

Next, run these commands to convert the risk-free rate to a daily rate, compute net returns, extract the Treasury rate, and compute excess returns for the market and for seven stocks. The risk-free rate is given as a percentage so the returns are also computed as percentages.

```
dat2 = as.matrix(cbind(dat[(2:n), 3] / 365,
  100 * (prices[2:n, ] / prices[1:(n-1), ] - 1)))
names(dat2)[1] = "treasury"
risk_free = dat2[,1]
ExRet = dat2[ ,2:9] - risk_free
market = ExRet[ ,8]
stockExRet = ExRet[ ,1:7]
```

Now fit model (17.20) to each stock, compute the residuals, look at a scatter-plot matrix of the residuals, and extract the estimated betas.

```
fit_reg = lm(stockExRet ~ market)
summary(fit_reg)
res = residuals(fit_reg)
pairs(res)
options(digits = 3)
betas = fit_reg$coeff[2, ]
```

**Problem 1** Would you reject the null hypothesis that alpha is zero for any of the seven stocks? Why or why not?

**Problem 2** Use model (17.20) to estimate the expected excess return for all seven stocks. Compare these results to using the sample means of the excess returns to estimate these parameters. Assume for the remainder of this lab that all alphas are zero. (Note: Because of this assumption, one might consider reestimating the betas and the residuals with a no-intercept model. However, since the estimated alphas were close to zero, forcing the alphas to be exactly zero will not change the estimates of the betas or the residuals by much. Therefore, for simplicity, do not reestimate.)

**Problem 3** Compute the correlation matrix of the residuals. Do any of the residual correlations seem large? Could you suggest a reason why the large correlations might be large? (Information about the companies in this data set is available at Yahoo Finance and other Internet sites.)

**Problem 4** Use model (17.20) to estimate the covariance matrix of the excess returns for the seven companies.

**Problem 5** What percentage of the excess return variance for UTX is due to the market?

**Problem 6** An analyst predicts that the expected excess return on the market next year will be 4 %. Assume that the betas estimated here using data from 2004–2006 are suitable as estimates of next year’s betas. Estimate the expected excess returns for the seven stocks for next year.

### 17.9.1 Zero-beta Portfolios

A portfolio with beta = 0 is neutral to market risk and bounding the absolute weights of the portfolio reduces the portfolio’s unique risk. In the next problem, you will find a low-risk portfolio with a large alpha. The data in this section have been simulated and are only for illustration. Estimation of the alphas of stock is difficult, especially the prediction of future values of alphas.

**Problem 7** The file `AlphaBeta.csv` contains alphas and betas on 50 stocks. Use linear programming to find the portfolio containing these stocks that has the maximum possible alpha subject to the portfolio’s beta being equal to zero and weights satisfying  $-0.25 \leq w_i \leq 0.25$  for all  $i = 1, \dots, 50$ . What are the 50 weights of your portfolio? What is its alpha?

Hint: This is a linear programming problem. Use the function `solveLP()`. See Sect. 16.10.3.

**Problem 8** If you attempt to find a zero-beta portfolio with  $-0.25 \leq w_i \leq 0.25$  with a smaller number of stock, you will find that there is no solution. (If you like, try this with the first 20 stocks.) Discuss why is there no solution.

## 17.10 Exercises

- What is the beta of a portfolio if  $E(R_P) = 16\%$ ,  $\mu_f = 5.5\%$ , and  $E(R_M) = 11\%$ ?

2. Suppose that the risk-free rate of interest is 0.03 and the expected rate of return on the market portfolio is 0.14. The standard deviation of the market portfolio is 0.12.
- According to the CAPM, what is the efficient way to invest with an expected rate of return of 0.11?
  - What is the risk (standard deviation) of the portfolio in part (a)?
3. Suppose that the risk-free interest rate is 0.023, that the expected return on the market portfolio is  $\mu_M = 0.10$ , and that the volatility of the market portfolio is  $\sigma_M = 0.12$ .
- What is the expected return on an efficient portfolio with  $\sigma_R = 0.05$ ?
  - Stock A returns have a covariance of 0.004 with market returns. What is the beta of Stock A?
  - Stock B has beta equal to 1.5 and  $\sigma_\epsilon = 0.08$ . Stock C has beta equal to 1.8 and  $\sigma_\epsilon = 0.10$ .
    - What is the expected return of a portfolio that is one-half Stock B and one-half Stock C?
    - What is the volatility of a portfolio that is one-half Stock B and one-half Stock C? Assume that the  $\epsilon$ s of Stocks B and C are independent.
4. Show that equation (17.16) follows from equation (7.8).
5. True or false: The CAPM implies that investors demand a higher return to hold more volatile securities. Explain your answer.
6. Suppose that the riskless rate of return is 4% and the expected market return is 12%. The standard deviation of the market return is 11%. Suppose as well that the covariance of the return on Stock A with the market return is 165%<sup>2</sup>.<sup>9</sup>
- What is the beta of Stock A?
  - What is the expected return on Stock A?
  - If the variance of the return on Stock A is 220%<sup>2</sup>, what percentage of this variance is due to market risk?
7. Suppose there are three risky assets with the following betas and  $\sigma_{\epsilon_j}^2$ .

| $j$ | $\beta_j$ | $\sigma_{\epsilon_j}^2$ |
|-----|-----------|-------------------------|
| 1   | 0.9       | 0.010                   |
| 2   | 1.1       | 0.015                   |
| 3   | 0.6       | 0.011                   |

Suppose also that the variance of  $R_{Mt} - \mu_{ft}$  is 0.014.

- What is the beta of an equally weighted portfolio of these three assets?
- What is the variance of the excess return on the equally weighted portfolio?
- What proportion of the total risk of asset 1 is due to market risk?

---

<sup>9</sup> If returns are expressed in units of percent, then the units of variances and covariances are percent-squared. A variance of 165%<sup>2</sup> equals 165/10,000.

8. Suppose there are two risky assets, call them C and D. The tangency portfolio is 60% C and 40% D. The expected yearly returns are 4% and 6% for assets C and D. The standard deviations of the yearly returns are 10% and 18% for C and D and the correlation between the returns on C and D is 0.5. The risk-free yearly rate is 1.2%.
- What is the expected yearly return on the tangency portfolio?
  - What is the standard deviation of the yearly return on the tangency portfolio?
  - If you want an efficient portfolio with a standard deviation of the yearly return equal to 3%, what proportion of your equity should be in the risk-free asset? If there is more than one solution, use the portfolio with the higher expected yearly return.
  - If you want an efficient portfolio with an expected yearly return equal to 7%, what proportions of your equity should be in asset C, asset D, and the risk-free asset?
9. What is the beta of a portfolio if the expected return on the portfolio is  $E(R_P) = 15\%$ , the risk-free rate is  $\mu_f = 6\%$ , and the expected return on the market is  $E(R_M) = 12\%$ ? Make the usual CAPM assumptions including that the portfolio alpha is zero.
10. Suppose that the risk-free rate of interest is 0.07 and the expected rate of return on the market portfolio is 0.14. The standard deviation of the market portfolio is 0.12.
- According to the CAPM, what is the efficient way to invest with an expected rate of return of 0.11?
  - What is the risk (standard deviation) of the portfolio in part (a)?
11. Suppose there are three risky assets with the following betas and  $\sigma_{\epsilon_j}^2$  when regressed on the market portfolio.

| $j$ | $\beta_j$ | $\sigma_{\epsilon_j}^2$ |
|-----|-----------|-------------------------|
| 1   | 0.7       | 0.010                   |
| 2   | 0.8       | 0.025                   |
| 3   | 0.6       | 0.012                   |

Assume  $\epsilon_1$ ,  $\epsilon_2$ , and  $\epsilon_3$  are uncorrelated. Suppose also that the variance of  $R_M - \mu_f$  is 0.02.

- What is the beta of an equally weighted portfolio of these three assets?
  - What is the variance of the excess return on the equally weighted portfolio?
  - What proportion of the total risk of asset 1 is due to market risk?
12. As an analyst, you have constructed 2 possible portfolios. Both portfolios have the same beta and expected return, but portfolio 1 was constructed with only technology companies whereas portfolio 2 was constructed using technology, healthcare, energy, consumer products, and metals and mining companies. Should you be impartial to which portfolio you invest in? Explain why or why not.

## References

- Bodie, Z., and Merton, R. C. (2000) *Finance*, Prentice-Hall, Upper Saddle River, NJ.
- Bodie, Z., Kane, A., and Marcus, A. (1999) *Investments*, 4th ed., Irwin/McGraw-Hill, Boston.
- Campbell, J. Y., Lo, A. W., and MacKinlay, A. C. (1997) *The Econometrics of Financial Markets*, Princeton University Press, Princeton, NJ.
- Lintner, J. (1965a) The valuation of risky assets and the selection of risky investments in stock portfolios and capital budgets. *Review of Economics and Statistics*, **47**, 13–37.
- Lintner, J. (1965b) Security prices, risk, and maximal gains from diversification. *Journal of Finance*, **20**, 587–615.
- Mossin, J. (1966) Equilibrium in capital markets. *Econometrica*, **34**, 768–783.
- Sharpe, W. F. (1964) Capital asset prices: A theory of market equilibrium under conditions of risk. *Journal of Finance*, **19**, 425–442.
- Sharpe, W. F., Alexander, G. J., and Bailey, J. V. (1999) *Investments*, 6th ed., Prentice-Hall, Upper Saddle River, NJ.

## Factor Models and Principal Components

### 18.1 Dimension Reduction

High-dimensional data can be challenging to analyze. They are difficult to visualize, need extensive computer resources, and often require special statistical methodology. Fortunately, in many practical applications, high-dimensional data have most of their variation in a lower-dimensional space that can be found using *dimension reduction techniques*. There are many methods designed for dimension reduction, and in this chapter we will study two closely related techniques, *factor analysis* and *principal components analysis*, often called *PCA*.

PCA finds structure in the covariance or correlation matrix and uses this structure to locate low-dimensional subspaces containing most of the variation in the data.

Factor analysis explains returns with a smaller number of fundamental variables called *factors* or *risk factors*. Factor analysis models can be classified by the types of variables used as factors, macroeconomic or fundamental, and by the estimation technique, time series regression, cross-sectional regression, or statistical factor analysis.

### 18.2 Principal Components Analysis

PCA starts with a sample  $\mathbf{Y}_i = (Y_{i,1}, \dots, Y_{i,d})$ ,  $i = 1, \dots, n$ , of  $d$ -dimensional random vectors with mean vector  $\boldsymbol{\mu}$  and covariance matrix  $\boldsymbol{\Sigma}$ . One goal of PCA is finding “structure” in  $\boldsymbol{\Sigma}$ .

We will start with a simple example that illustrates the main idea. Suppose that  $\mathbf{Y}_i = \boldsymbol{\mu} + W_i \mathbf{o}$ , where  $W_1, \dots, W_n$  are i.i.d. mean-zero random variables and  $\mathbf{o}$  is some fixed vector, which can be taken to have norm 1. The  $\mathbf{Y}_i$  lie on

the line that passes through  $\boldsymbol{\mu}$  and is in the direction given by  $\mathbf{o}$ , so that all variation among the mean-centered vectors  $\mathbf{Y}_i - \boldsymbol{\mu}$  is in the one-dimensional space spanned by  $\mathbf{o}$ . Also, the covariance matrix of  $\mathbf{Y}_i$  is

$$\boldsymbol{\Sigma} = E\{W_i^2 \mathbf{o}\mathbf{o}^\top\} = \sigma_W^2 \mathbf{o}\mathbf{o}^\top.$$

The vector  $\mathbf{o}$  is called the first principal axis of  $\boldsymbol{\Sigma}$  and is the only eigenvector of  $\boldsymbol{\Sigma}$  with a nonzero eigenvalue, so  $\mathbf{o}$  can be estimated by an eigen-decomposition (Appendix A.20) of the estimated covariance matrix.

A slightly more realistic situation is where  $\mathbf{Y}_i = \boldsymbol{\mu} + W_i \mathbf{o} + \boldsymbol{\epsilon}_i$ , where  $\boldsymbol{\epsilon}_i$  is a random vector uncorrelated with  $W_i$  and having a “small” covariance matrix. Then most of the variation among the  $\mathbf{Y}_i - \boldsymbol{\mu}$  vectors is in the space spanned by  $\mathbf{o}$ , but there is small variation in other directions due to  $\boldsymbol{\epsilon}_i$ . Having looked at some simple special cases, we now turn to the general case.

PCA can be applied to either the sample covariance matrix or the correlation matrix. We will use  $\boldsymbol{\Sigma}$  to represent whichever matrix is chosen. The correlation matrix is, of course, the covariance matrix of the standardized variables, so the choice between the two matrices is really a decision whether or not to standardize the variables before PCA. This issue will be addressed later. Even if the data have not been standardized, to keep notation simple, we assume that the mean  $\bar{\mathbf{Y}}$  has been subtracted from each  $\mathbf{Y}_i$ . By (A.50),

$$\boldsymbol{\Sigma} = \mathbf{O} \operatorname{diag}(\lambda_1, \dots, \lambda_d) \mathbf{O}^\top, \quad (18.1)$$

where  $\mathbf{O}$  is an orthogonal matrix whose columns  $\mathbf{o}_1, \dots, \mathbf{o}_d$  are the eigenvectors of  $\boldsymbol{\Sigma}$  and  $\lambda_1 > \dots > \lambda_d$  are the corresponding eigenvalues. The columns of  $\mathbf{O}$  have been arranged so that the eigenvalues are ordered from largest to smallest. This is not essential, but it is convenient. We also assume no ties among the eigenvalues, which almost certainly will be true in actual applications.

A *normed linear combination* of  $\mathbf{Y}_i$  (either standardized or not) is of the form  $\boldsymbol{\alpha}^\top \mathbf{Y}_i = \sum_{j=1}^p \alpha_j Y_{i,j}$ , where  $\|\boldsymbol{\alpha}\| = \sqrt{\sum_{j=1}^p \alpha_j^2} = 1$ . The first principal component is the normed linear combination with the greatest variance. The variation in the direction  $\boldsymbol{\alpha}$ , where  $\boldsymbol{\alpha}$  is any fixed vector with norm 1, is

$$\operatorname{Var}(\boldsymbol{\alpha}^\top \mathbf{Y}_i) = \boldsymbol{\alpha}^\top \boldsymbol{\Sigma} \boldsymbol{\alpha}. \quad (18.2)$$

The first principal component maximizes (18.2) over  $\boldsymbol{\alpha}$ . The maximizer is  $\boldsymbol{\alpha} = \mathbf{o}_1$ , the eigenvector corresponding to the largest eigenvalue, and is called the first principal axis. The projections  $\mathbf{o}_1^\top \mathbf{Y}_i$ ,  $i = 1, \dots, n$ , onto this vector are called the first principal component or principal component scores. Requiring that the norm of  $\boldsymbol{\alpha}$  be fixed is essential, because otherwise (18.2) is unbounded and there is no maximizer.

After the first principal component has been found, one searches for the direction of maximum variation perpendicular to the first principal axis (eigenvector). This means maximizing (18.2) subject to  $\|\boldsymbol{\alpha}\| = 1$  and  $\boldsymbol{\alpha}^\top \mathbf{o}_1 = 0$ .

The maximizer, called the second principal axis, is  $\mathbf{o}_2$ , and the second principal component is the set of projections  $\mathbf{o}_2^T \mathbf{Y}_i$ ,  $i = 1, \dots, n$ , onto this axis. The reader can probably see where we are going. The third principal component maximizes (18.2) subject to  $\|\boldsymbol{\alpha}\| = 1$ ,  $\boldsymbol{\alpha}^T \mathbf{o}_1 = 0$ , and  $\boldsymbol{\alpha}^T \mathbf{o}_2 = 0$  and is  $\mathbf{o}_3^T \mathbf{Y}_i$ , and so forth, so that  $\mathbf{o}_1, \dots, \mathbf{o}_d$  are the principal axes and the set of projections  $\mathbf{o}_j^T \mathbf{Y}_i$ ,  $i = 1, \dots, n$ , onto the  $j$ th eigenvector is the  $j$ th principal component. Moreover,

$$\lambda_i = \mathbf{o}_i^T \boldsymbol{\Sigma} \mathbf{o}_i$$

is the variance of the  $i$ th principal component,  $\lambda_i / (\lambda_1 + \dots + \lambda_d)$  is the proportion of the variance due to this principal component, and  $(\lambda_1 + \dots + \lambda_i) / (\lambda_1 + \dots + \lambda_d)$  is the proportion of the variance due to the first  $i$  principal components. The principal components are mutually uncorrelated since for  $j \neq k$  we have

$$\text{Cov}(\mathbf{o}_j^T \mathbf{Y}_i, \mathbf{o}_k^T \mathbf{Y}_i) = \mathbf{o}_j^T \boldsymbol{\Sigma} \mathbf{o}_k = 0$$

by (A.52).

Let

$$\mathbf{Y} = \begin{pmatrix} \mathbf{Y}_1^T \\ \vdots \\ \mathbf{Y}_n^T \end{pmatrix}$$

be the original data and let

$$\mathbf{S} = \begin{pmatrix} \mathbf{o}_1^T \mathbf{Y}_1 & \cdots & \mathbf{o}_d^T \mathbf{Y}_1 \\ \vdots & \ddots & \vdots \\ \mathbf{o}_1^T \mathbf{Y}_n & \cdots & \mathbf{o}_d^T \mathbf{Y}_n \end{pmatrix}$$

be the matrix of principal components. Then

$$\mathbf{S} = \mathbf{Y} \mathbf{O}.$$

Postmultiplication of  $\mathbf{Y}$  by  $\mathbf{O}$  to obtain  $\mathbf{S}$  is an orthogonal rotation of the data. For this reason, the eigenvectors are sometimes called the *rotations*, e.g., in output from R's `pca()` function.

In many applications, the first few principal components, such as, the first three to five, account for almost all of the variation, and, for most purposes, one can work solely with these principal components and discard the rest. This can be a sizable reduction in dimension. See Example 18.2 for an illustration.

So far, we have left unanswered the question of how one should decide between working with the original or the standardized variables. If the components of  $\mathbf{Y}_i$  are comparable, e.g., are all daily returns on equities or all are yields on bonds, then working with the original variables should cause no problems. However, if the variables are not comparable, e.g., one is an unemployment rate and another is the GDP in dollars, then some variables may be many orders of magnitude larger than the others. In such cases, the large

variables could completely dominate the PCA, so that the first principal component is in the direction of the variable with the largest standard deviation. To eliminate this problem, one should standardize the variables.

*Example 18.1. PCA with unstandardized and standardized variables*

As a simple illustration of the difference between using standardized and unstandardized variables, suppose there are two variables ( $d = 2$ ) with a correlation of 0.9. Then the correlation matrix is

$$\begin{pmatrix} 1 & 0.9 \\ 0.9 & 1 \end{pmatrix}$$

with normalized eigenvectors  $(0.71, 0.71)$  and  $(-0.71, 0.71)$ <sup>1</sup> and eigenvalues 1.9 and 0.1. Most of the variation is in the direction  $(1, 1)$ , which is consistent with the high correlation between the two variables.

However, suppose that the first variable has variance 1,000,000 and the second has variance 1. The covariance matrix is

$$\begin{pmatrix} 1,000,000 & 900 \\ 900 & 1 \end{pmatrix},$$

which has eigenvectors, after rounding, equal to  $(1.0000, 0.0009)$  and  $(-0.0009, 1)$  and eigenvalues 1,000,000 and 0.19. The first variable dominates the principal components analysis based on the covariance matrix. This principal components analysis does correctly show that almost all of the variation is in the first variable, but this is true only with the original units. Suppose that variable 1 had been in dollars and is now converted to millions of dollars. Then its variance is equal to  $10^{-6}$ , so that the principal components analysis using the covariance matrix will now show most of the variation to be due to variable 2. In contrast, principal components analysis based on the correlation matrix does not change as the variables' units change.  $\square$

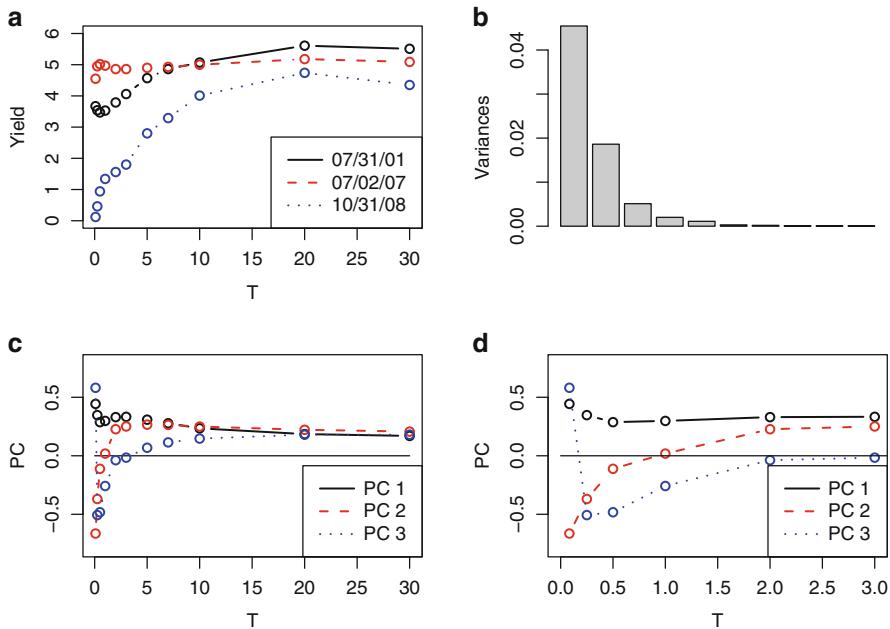
*Example 18.2. Principal components analysis of yield curves*

This example uses yields on Treasury bonds at 11 maturities,  $T = 1, 3$ , and 6 months and 1, 2, 3, 5, 7, 10, 20, and 30 years. Daily yields were taken from a U.S. Treasury website for the time period January 2, 1990, to October 31, 2008. A subset of these data was used in Example 15.1. The yield curves are shown in Fig. 18.1a for three different dates. Notice that the yield curves can have a variety of shapes. In this example, we will use PCA to study how the curves change from day to day.

To analyze daily changes in yields, all 11 time series were differenced. Daily yields were missing from some values of  $T$  because, for example to quote the

---

<sup>1</sup> The normalized eigenvalues are determined only up to sign so they could multiplied by  $-1$  to become  $(-0.71, -0.71)$  and  $(0.71, -0.71)$ .



**Fig. 18.1.** (a) Treasury yields on three dates. (b) Scree plot for the changes in Treasury yields. Note that the first three principal components have most of the variation, and the first five have virtually all of it. (c) The first three eigenvectors for changes in the Treasury yields. (d) The first three eigenvectors for changes in the Treasury yields in the range  $0 \leq T \leq 3$ .

website, “Treasury discontinued the 20-year constant maturity series at the end of calendar year 1986 and reinstated that series on October 1, 1993.” Differencing caused a few additional days to have missing values. In the analysis, all days with missing values of the differenced data were omitted. This left 819 days of data starting on July 31, 2001, when the one-month series started and ending on October 31, 2008, with the exclusion of the period February 19, 2002 to February 2, 2006 when the 30-year Treasury was discontinued. One could use much longer series by not including the one-month and 30-year series.

The covariance matrix, not the correlation matrix, was used, because in this example the variables are comparable and in the same units.

First, we will look at the 11 eigenvalues using R’s function `prcomp()`. The code is:

```
datNoOmit = read.table("treasury_yields.txt", header = TRUE)
diffdatNoOmit = diff(as.matrix(datNoOmit[, 2:12]))
dat = na.omit(datNoOmit)
diffdat = na.omit(diffdatNoOmit)
n = dim(diffdat)[1]
```

```
options(digits = 5)
pca = prcomp(diffdat)
summary(pca)
```

The results are:

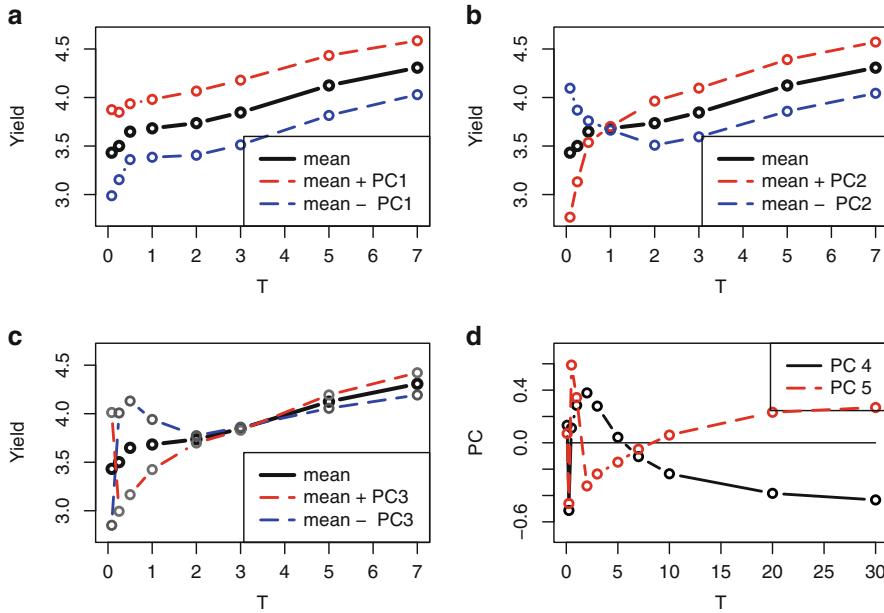
| Importance of components: |        |        |         |         |       |        |
|---------------------------|--------|--------|---------|---------|-------|--------|
|                           | PC1    | PC2    | PC3     | PC4     | PC5   | PC6    |
| Standard deviation        | 0.21   | 0.14   | 0.071   | 0.045   | 0.033 | 0.0173 |
| Proportion of Variance    | 0.62   | 0.25   | 0.070   | 0.028   | 0.015 | 0.0041 |
| Cumulative Proportion     | 0.62   | 0.88   | 0.946   | 0.974   | 0.989 | 0.9932 |
| PC7                       | PC8    | PC9    | PC10    | PC11    |       |        |
| 0.0140                    | 0.0108 | 0.0092 | 0.00789 | 0.00610 |       |        |
| 0.0027                    | 0.0016 | 0.0012 | 0.00085 | 0.00051 |       |        |
| 0.9959                    | 0.9975 | 0.9986 | 0.99949 | 1.00000 |       |        |

The first row gives the values of  $\sqrt{\lambda_i}$ , the second row the values of  $\lambda_i/(\lambda_1 + \dots + \lambda_d)$ , and the third row the values of  $(\lambda_1 + \dots + \lambda_i)/(\lambda_1 + \dots + \lambda_d)$  for  $i = 1, \dots, 11$ . One can see, for example, that the standard deviation of the first principal component is 0.21 and represents 62 % of the total variance. Also, the first three principal components have 94.6 % of the variation, and this increases to 97.4 % for the first four principal components and to 98.9 % for the first five. The variances (the squares of the first row) are plotted in Fig. 18.1b. This type of plot is called a “scree plot” since it looks like scree, fallen rocks that have accumulated at the base of a mountain.

We will concentrate on the first three principal components since approximately 95 % of the variation in the changes in yields is in the space they span. The eigenvectors, labeled “PC,” are plotted in Fig. 18.1c and d, the latter showing detail in the range  $T \leq 3$ . The eigenvectors have interesting interpretations. The first,  $\mathbf{o}_1$ , has all positive values.<sup>2</sup> A change in this direction either increases all yields or decreases all yields, and by roughly the same amounts. One could call such changes “parallel shifts” of the yield curve, though they are only approximately parallel. These shifts are shown in Fig. 18.2a, where the mean yield curve is shown as a solid black line, the mean plus  $\mathbf{o}_1$  is a dashed red line, and the mean minus  $\mathbf{o}_1$  is a dashed blue line. Only the range  $T \leq 7$  is shown, since the curves change less after this point. Since the standard deviation of the first principal component is only 0.21, a  $\pm 1$  shift in a single day is huge and is used only for better graphical presentation.

---

<sup>2</sup> As mentioned previously, the eigenvectors are determined only up to a sign reversal, since multiplication by  $-1$  would not change the spanned space or the norm. Thus, we could instead say the eigenvector has only negative values, but this would not change the interpretation.



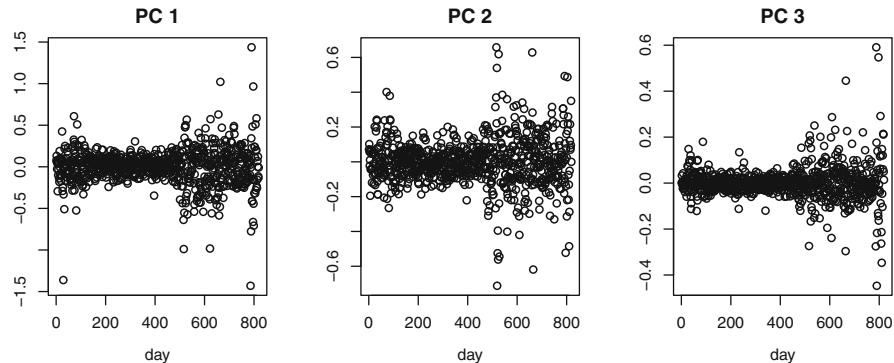
**Fig. 18.2.** (a) The mean yield curve plus and minus the first eigenvector. (b) The mean yield curve plus and minus the second eigenvector. (c) The mean yield curve plus and minus the third eigenvector. (d) The fourth and fifth eigenvectors for changes in the Treasury yields.

The graph of  $\mathbf{o}_2$  is everywhere decreasing<sup>3</sup> and changes in this direction either increase or decrease the slope of the yield curve. The result is that a graph of the mean plus or minus PC2 will cross the graph of the mean curve at approximately  $T = 1$ , where  $\mathbf{o}_2$  equals zero; see Fig. 18.2b.

The graph of  $\mathbf{o}_3$  is first decreasing and then increasing, and the changes in this direction either increase or decrease the convexity of the yield curve. The result is that a graph of the mean plus or minus PC3 will cross the graph of the mean curve twice; see Fig. 18.2c. It is worth repeating a point just made in connection with PC1, since it is even more important here. The standard deviations in the directions of PC2 and PC3 are only 0.14 and 0.071, respectively, so observed changes in these directions will be much smaller than those shown in Fig. 18.2b and c. Moreover, parallel shifts will be larger than changes in slope, which will be larger than changes in convexity.

Figure 18.2d plots the fourth and fifth eigenvectors. The patterns in their graphs are complex and do not have easy interpretations. Fortunately, the variation in the space they span is too small to be of much importance.

<sup>3</sup> The graph would, of course, be everywhere increasing if  $\mathbf{o}_2$  were multiplied by  $-1$ .



**Fig. 18.3.** Time series plots of the first three principal components of the Treasury yields. There are 819 days of data, but they are not consecutive because of missing data; see text.

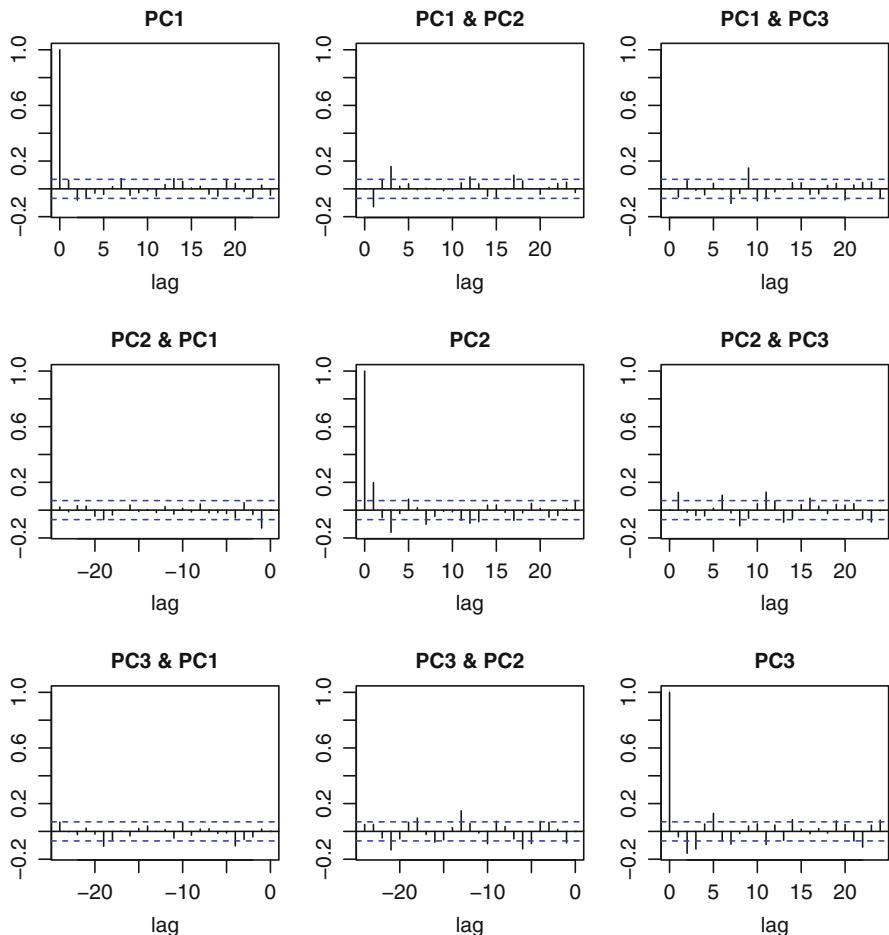
A bond portfolio manager would be interested in the behavior of the yield changes over time. Time series analysis based on the changes in the 11 yields could be useful, but a better approach would be to use the first three principal components. Their time series and auto- and cross-correlation plots are shown in Figs. 18.3 and 18.4, respectively. The latter shows moderate short-term auto-correlations which could be modeled with an ARMA process, though the correlation is small enough that it might be ignored. Notice that the lag-0 cross-correlations are zero; this is not a coincidence but rather is due to the way the principal components are defined. They are defined to be uncorrelated with each other, so their lag-0 correlations are exactly zero. Cross-correlations at nonzero lags are not zero, but in this example they are small. The practical implication is that parallel shifts, changes in slopes, and changes in convexity are nearly uncorrelated and could be analyzed separately. The time series plots show substantial volatility clustering which could be modeled using the GARCH models of Chap. 14. □

#### *Example 18.3. Principal components analysis of equity funds*

This example uses the data set `equityFunds.csv`. The variables are daily returns from January 1, 2002 to May 31, 2007 on eight equity funds: EASTEU, LATAM, CHINA, INDIA, ENERGY, MINING, GOLD, and WATER. The following code was run:

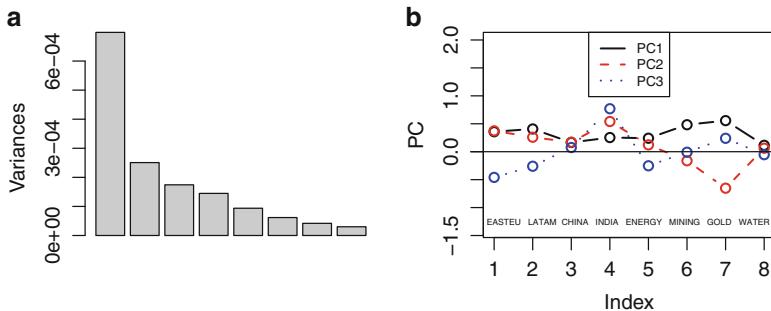
```
equityFunds = read.csv("equityFunds.csv")
pcaEq = prcomp(equityFunds[, 2:9])
summary(pcaEq)
```

The results in this example are below and are different than those for the changes in yields, because in this example the variation is less concentrated in the first few principal components. For example, the first three principal



**Fig. 18.4.** Sample auto- and cross-correlations of the first three principal components of the Treasury yields.

components have only 75 % of the variance, compared to 95 % for the yield changes. For the equity funds, one needs six principal components to get 95 %. A scree plot is shown in Fig. 18.5a.



**Fig. 18.5.** (a) Scree plot for the Equity Funds example. (b) The first three eigenvectors for the Equity Funds example.

Importance of components:

|                        | PC1    | PC2    | PC3    | PC4   | PC5    |
|------------------------|--------|--------|--------|-------|--------|
| Standard deviation     | 0.026  | 0.016  | 0.013  | 0.012 | 0.0097 |
| Proportion of Variance | 0.467  | 0.168  | 0.117  | 0.097 | 0.0627 |
| Cumulative Proportion  | 0.467  | 0.635  | 0.751  | 0.848 | 0.9107 |
|                        | PC6    | PC7    | PC8    |       |        |
|                        | 0.0079 | 0.0065 | 0.0055 |       |        |
|                        | 0.0413 | 0.0280 | 0.0201 |       |        |
|                        | 0.9520 | 0.9799 | 1.0000 |       |        |

The first three eigenvectors are plotted in Fig. 18.5b. The first eigenvector has only positive values, and returns in this direction are either positive for all of the funds or negative for all of them. The second eigenvector is negative for mining and gold (funds 6 and 7) and positive for the other funds. Variation along this eigenvector has mining and gold moving in the opposite direction of the other funds. Gold and mining stock moving counter to the rest of the stock market is a common occurrence and, in fact, these types of stock often have negative betas, so it is not surprising that the second principal component has 17% of the variation. The third principal component is less easy to interpret, but its loading on India (fund 4) is higher than on the other funds, which might indicate that there is something different about Indian equities. □

#### *Example 18.4. Principal components analysis of the Dow Jones 30*

As a further example, we will use returns on the 30 stocks on the Dow Jones average. The data are in the data set `DowJones30.csv` and cover the period from January 2, 1991 to January 2, 2002. The first five principal components have over 97% of the variation:

**Importance of components:**

|                        | PC1   | PC2    | PC3   | PC4    | PC5    |
|------------------------|-------|--------|-------|--------|--------|
| Standard deviation     | 88.53 | 24.967 | 13.44 | 10.602 | 8.2165 |
| Proportion of Variance | 0.87  | 0.069  | 0.02  | 0.012  | 0.0075 |
| Cumulative Proportion  | 0.87  | 0.934  | 0.95  | 0.967  | 0.9743 |

In contrast to the analysis of the equity funds where six principal components were needed to obtain 95 % of the variance, here the first three principal components have over 95 % of the variance. Why are the Dow Jones stocks behaving differently compared to the equity funds? The Dow Jones stocks are similar to each other since they are all large companies in the United States. Thus, we can expect that their returns will be highly correlated with each other and a few principal components will explain most of the variation.  $\square$

## 18.3 Factor Models

A factor model for excess equity returns is

$$R_{j,t} = \beta_{0,j} + \beta_{1,j}F_{1,t} + \cdots + \beta_{p,j}F_{p,t} + \epsilon_{j,t}, \quad (18.3)$$

where  $R_{j,t}$  is either the return or the excess return on the  $j$ th asset at time  $t$ ,  $F_{1,t}, \dots, F_{p,t}$  are variables, called *factors* or *risk factors*, that represent the “state of the financial markets and world economy” at time  $t$ , and  $\epsilon_{1,t}, \dots, \epsilon_{n,t}$  are uncorrelated, mean-zero random variables called the *unique risks* of the individual stocks. The assumption that unique risks are uncorrelated means that all cross-correlation between the returns is due to the factors. Notice that the factors do not depend on  $j$  since they are common to all returns. The parameter  $\beta_{i,j}$  is called a factor loading and specifies the sensitivity of the  $j$ th return to the  $i$ th factor. Depending on the type of factor model, either the loadings, the factors, or both the factors and the loadings are unknown and must be estimated.

The CAPM is a factor model where  $p = 1$  and  $F_{1,t}$  is the excess return on the market portfolio. In the CAPM, the market risk factor is the only source of risk besides the unique risk of each asset. Because the market risk factor is the only risk that any two assets share, it is the sole source of correlation between asset returns. Factor models generalize the CAPM by allowing more factors than simply the market risk and the unique risk of each asset. A *factor* can be any variable thought to affect asset returns. Examples of factors include:

1. returns on the market portfolio;
2. growth rate of the GDP;
3. interest rate on short term Treasury bills or changes in this rate;
4. inflation rate or changes in this rate;
5. interest rate spreads, for example, the difference between long-term Treasury bonds and long-term corporate bonds;

6. return on some portfolio of stocks, for example, all U.S. stocks or all stocks with a high ratio of book equity to market equity — this ratio is called BE/ME in Fama and French (1992, 1995, 1996);
7. the difference between the returns on two portfolios, for example, the difference between returns on stocks with high BE/ME values and stocks with low BE/ME values.

With enough factors, most, and perhaps all, commonalities between assets should be accounted for in the model. Then the  $\epsilon_{j,t}$  should represent factors truly unique to the individual assets and therefore should be uncorrelated across  $j$  (across assets), as is being assumed.

Factor models that use macroeconomic variables such as 1–5 as factors are called *macroeconomic factor models*. *Fundamental factor models* use observable asset characteristics (fundamentals) such as 6 and 7 as factors. Both types of factor models can be fit by time series regression, the topic of the next section. Fundamental factor models can also be fit by cross-sectional regression, as explained in Sect. 18.5.

## 18.4 Fitting Factor Models by Time Series Regression

Equation (18.3) is a regression model. If  $j$  is fixed, then it is a univariate multiple regression model, “univariate” because there is one response (the return on the  $j$ th asset) and “multiple” since there can be several predictor variables (the factors). If we combine these models across  $j$ , then we have a multivariate regression model, that is, a regression model with more than one response. Multivariate regression is used when fitting a set of returns to factors.

As discussed in Sect. 17.6, when fitting time series regression models, one should use data at the highest sampling frequency available, which is often daily or weekly, though only monthly data were available for the next example.

### *Example 18.5. A macroeconomic factor model*

The efficient market hypothesis implies that stock prices change because of new information. Although there is considerable debate about the extent to which markets are efficient, one still can expect that stock returns will be influenced by unpredictable changes in macroeconomic variables. Accordingly, the factors in a macroeconomic model are not the macroeconomic variables themselves, but rather the residuals when changes in the macroeconomic variables are predicted from past data by a time series model, such as, a multivariate AR model.

In this example, we look at a subset of a case study that has been presented by other authors; see the bibliographical notes in Sect. 18.7. The macroeconomic variables in this example are changes in the logs of CPI (Consumer

Price Index) and IP (Industrial Production). The changes in these series have been analyzed before in Examples 12.10, 12.11, and 13.10 and in that last example a bivariate AR model was fit. It was found that the AR(5) model minimized AIC, but the AR(1) had an AIC value nearly as small as the AR(5) model.

In this example, we will use the residuals from the AR(5) model as the factors. Monthly returns on nine stocks were taken from the `berndtInvest.csv` data set. The returns are from January 1978 to December 1987. The CPI and IP series from July 1977 to December 1987 were used, but the month of July 1977 was lost through differencing. This left enough data (the five months August 1977 to December 1977) for forecasting CPI and IP beginning January 1978 when the return series started.

$R^2$  and the slopes for the regressions of the stock returns on the CPI residuals and the IP residuals are plotted in Fig. 18.6 for each of the 9 stocks. Note that the  $R^2$ -values are very small, so the macroeconomic factors have little explanatory power. The problem of low explanatory power is common with macroeconomic factor models and has been noticed by other authors. For this reason, fundamental factor models are more widely used than macroeconomic models.  $\square$

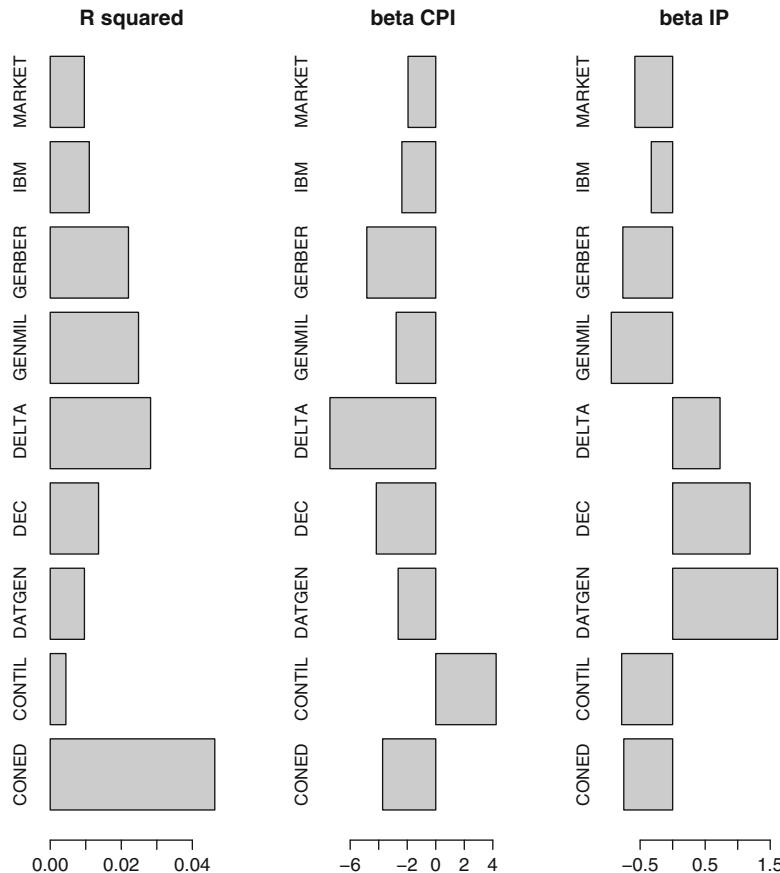
### 18.4.1 Fama and French Three-Factor Model

Fama and French (1995) have developed a fundamental factor model with three risk factors, the first being the excess return of the market portfolio, which is the sole factor in the CAPM. The second risk factor, which is called small minus big (SMB), is the difference in returns on a portfolio of small stocks and a portfolio of large stocks. Here “small” and “big” refer to the size of the *market value*, which is the share price times the number of shares outstanding. The third factor, HML (high minus low), is the difference in returns on a portfolio of high book-to-market value (BE/ME) stocks and a portfolio of low BE/ME stocks. *Book value* is the net worth of the firm according to its accounting balance sheet. Fama and French argue that most pricing anomalies that are inconsistent with the CAPM disappear in the three-factor model. Their model of the return on the  $j$ th asset for the  $t$ th holding period is

$$R_{j,t} - \mu_{f,t} = \beta_{0,j} + \beta_{1,j}(R_{M,t} - \mu_{f,t}) + \beta_{2,j}\text{SMB}_t + \beta_{3,j}\text{HML}_t + \epsilon_{j,t},$$

where  $\text{SMB}_t$  and  $\text{HML}_t$  are the values of SMB and HML and  $\mu_{f,t}$  is the risk-free rate for the  $t$ th holding period. Returns on portfolios have little autocorrelation, so the returns themselves, rather than residuals from a time series model, can be used.

Notice that this model does *not* use the size or the BE/ME ratio of the  $j$ th asset to explain returns. The coefficients  $\beta_{2,j}$  and  $\beta_{3,j}$  are the loading of the  $j$ th asset on SMB and HML. These loadings may, but need not, be



**Fig. 18.6.**  $R^2$  and slopes of regressions of stock returns on CPI residuals and IP residuals.

related to the size and to the BE/ME ratio of the  $j$ th asset. In any event, the loadings are estimated by regression, not by measuring the size or BE/ME of the  $j$ th asset. If the loading  $\beta_{2,j}$  of the  $j$ th asset on SMB is high, that might be because the  $j$ th asset is small or it might be because that asset is large but, in terms of returns, behaves similarly to small assets.

For emphasis, it is mentioned again that the factors  $SMB_t$  and  $HML_t$  do not depend on  $j$  since they are differences between returns on two fixed portfolios, not variables that are measured on the  $j$ th asset. This is true in general of the factors and loadings in model (18.3), not just the Fama–French model—only the loadings, that is, the parameters  $\beta_{k,j}$ , depend on the asset  $j$ . The factors are macroeconomic variables, linear combinations of returns on portfolios, or other variables that depend only on the financial markets and the economy as a whole.

There are many reasons why book and market values may differ. Book value is determined by accounting methods that do not necessarily reflect market values. Also, a stock might have a low book-to-market value because investors expect a high return on equity, which increases its market value relative to its book value. Conversely, a high book-to-market value could indicate a firm that is in trouble, which decreases its market value. A low market value relative to the book value is an indication of a stock's "cheapness," and stocks with a high market-to-book value are considered *growth stocks* for which investors are willing to pay a premium because of the promise of higher future earnings. Stocks with a low market-to-book value are called *value stocks* and investing in them is called *value investing*.

SMB and HML are the returns on portfolio that are long on one group of stocks and short on another. Such portfolios are called *hedge portfolios* since they are hedged, though perhaps not perfectly, against changes in the overall market.

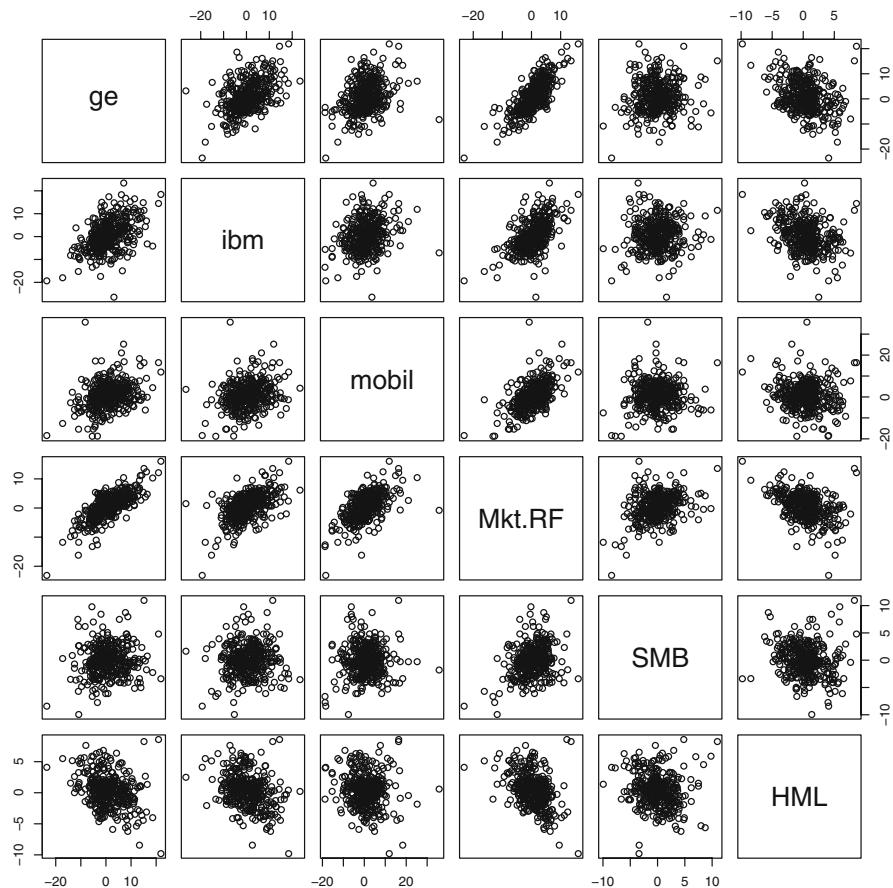
#### *Example 18.6. Fitting the Fama–French model to GE, IBM, and Mobil*

This example uses two data sets. The first is CRSPmon in R's Ecdat package. This is similar to the CRSPday data set used in previous examples except that the returns are now monthly rather than daily. There are returns on three equities, GE, IBM, and Mobil, as well as on the CRSP average, though we will not use the last one here. The returns are from January 1969 to December 1998. The second data set is the Fama–French factors and was taken from the website of Prof. Kenneth French.

Figure 18.7 is a scatterplot matrix of the GE, IBM, and Mobil excess returns and the factors. Focusing on GE, we see that, as would be expected, GE excess returns are highly correlated with the excess market returns. The GE returns are negatively related with the factor HML which would indicate that GE behaves as a growth stock, since it moves in the same direction as low BE/ME stocks and in the opposite direction of high BE/ME stocks. However, this is a false impression caused by the lack of adjustment for associations between GE excess returns and the other factors. Regression analysis will be used soon to address this problem. The two Fama–French factors are not quite hedge portfolios since SMB is positively and HML negatively related to the excess market return. However, these associations are far weaker than that between the excess returns on the stocks and the market excess returns. Moreover, SMB and HML have little association between each other, so multicollinearity is not a problem.

The three excess equity returns were regressed on the three factors using the lm() function in R. The code is:

```
FF_data = read.table("FamaFrench_mon_69_98.txt", header = TRUE)
attach(FF_data)
library("Ecdat")
```



**Fig. 18.7.** Scatterplot matrix of the excess returns on GE, IBM, and Mobil and the three factors in the Fama–French model. Mkt.RF is the return on the market portfolio minus the risk-free rate.

```
library("robust")
data(CRSPmon)
ge = 100*CRSPmon[,1] - RF
ibm = 100*CRSPmon[,2] - RF
mobil = 100*CRSPmon[,3] - RF
stocks = cbind(ge, ibm, mobil)
fit = lm(cbind(ge, ibm, mobil) ~ Mkt.RF + SMB + HML)
fit
```

and the estimated coefficients are

```
Call:
lm(formula = cbind(ge, ibm, mobil) ~ Mkt.RF + SMB + HML)
```

Coefficients:

|             | ge      | ibm     | mobil   |
|-------------|---------|---------|---------|
| (Intercept) | 0.3443  | 0.1460  | 0.1635  |
| Mkt.RF      | 1.1407  | 0.8114  | 0.9867  |
| SMB         | -0.3719 | -0.3125 | -0.3753 |
| HML         | 0.0095  | -0.2983 | 0.3725  |

The coefficients of HML indicate that GE and Mobil are value stocks and IBM is a growth stock. Notice that GE now has a positive relationship with HML, not the negative relationship seen in Fig. 18.7, although its coefficient is close to 0. GE seems to be somewhere in between being a growth stock and a value stock.

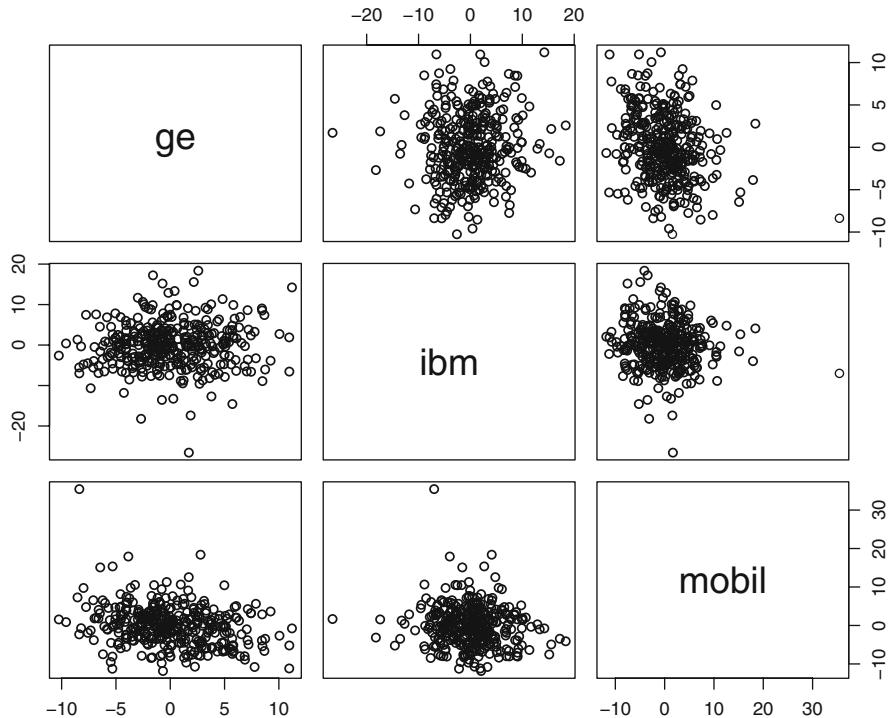
All three equity returns have negative relationships with SMB, so, not surprisingly, they behave like large stocks.

Recall that one important assumption of the factor model is that the  $\epsilon_{j,t}$  in (18.3) are uncorrelated. Violation of this assumption, that is, cross-correlations between  $\epsilon_{j,t}$  and  $\epsilon_{j',t}$ ,  $j \neq j'$ , will create biases when the factor model is used to estimate correlations between the equity returns, a topic explained in the next section. Lack of cross-correlation is not an assumption of the multivariate regression model and does not cause bias in the estimation of the regression coefficients or the variances of the  $\epsilon_{j,t}$ . The biases arise only when estimating covariances between the equity returns.

To check for cross-correlations, we will use the residuals from the multivariate regression. Their sample correlation matrix is

```
> cor(fit$residuals)
      ge    ibm mobil
ge   1.000  0.071 -0.25
ibm  0.071  1.000 -0.10
mobil -0.254 -0.102  1.00
```

The correlation between GE and Mobil is rather far from zero and is worth checking. A 95% confidence interval for the residual correlations between GE excess returns and Mobil excess returns does not include 0, so a test would reject the null hypotheses that the true correlation is 0. The other correlations are not significantly different from 0. Because of the large negative GE–Mobil correlation, we should be careful about using the Fama–French model for estimation of the covariance matrix of the equity returns. As always, it is good practice to look at scatterplot matrices as well as correlations, since scatterplots may be outliers or nonlinear relationships affecting the correlations. Figure 18.8 contains a scatterplot matrix of the residuals. One sees that there are few outliers, although none of the outliers is really extreme, it seems worthwhile to compute robust correlations estimates and to compare them with the ordinary sample correlation matrix. Robust estimates were found using the function `covRob()` in R's `robust` package. What was found is that the robust estimates are all closer to zero than the nonrobust estimates, but the robust correlation estimate for GE and Mobil is still a large negative value.



**Fig. 18.8.** Scatterplot matrix of the residuals for GE, IBM, and Mobil from the Fama–French model.

Call:

```
covRob(data = fit$residuals, corr = T)
```

Robust Estimate of Correlation:

|       | ge     | ibm     | mobil   |
|-------|--------|---------|---------|
| ge    | 1.000  | 0.0360  | -0.2479 |
| ibm   | 0.036  | 1.0000  | -0.0687 |
| mobil | -0.248 | -0.0687 | 1.0000  |

This example is atypical of real applications because, for illustration purposes, the number of returns has been kept low, only three, whereas in portfolio management the number of returns will be larger and might be in the hundreds.  $\square$

#### 18.4.2 Estimating Expectations and Covariances of Asset Returns

Section 17.7 discussed how the CAPM can simplify the estimation of expectations and covariances of asset returns. However, using the CAPM for this

purpose can be dangerous since the estimates depend on the validity of the CAPM. Fortunately, it is also possible to estimate return expectations and covariances using a more realistic factor model instead of the CAPM.

We start with two factors for simplicity. From (18.3), now with  $p = 2$ , we have

$$R_{j,t} = \beta_{0,j} + \beta_{1,j}F_{1,t} + \beta_{2,j}F_{2,t} + \epsilon_{j,t}. \quad (18.4)$$

It follows from (18.4) that

$$E(R_{j,t}) = \beta_{0,j} + \beta_{1,j}E(F_{1,t}) + \beta_{2,j}E(F_{2,t}) \quad (18.5)$$

and

$$\text{Var}(R_{j,t}) = \beta_{1,j}^2 \text{Var}(F_1) + \beta_{2,j}^2 \text{Var}(F_2) + 2\beta_{1,j}\beta_{2,j}\text{Cov}(F_1, F_2) + \sigma_{\epsilon,j}^2.$$

Also, because  $R_{j,t}$  and  $R_{j',t}$  are two linear combinations of the risk factors, it follows from (7.8) that for any  $j \neq j'$ ,

$$\begin{aligned} \text{Cov}(R_{j,t}, R_{j',t}) &= \beta_{1,j}\beta_{1,j'}\text{Var}(F_1) + \beta_{2,j}\beta_{2,j'}\text{Var}(F_2) \\ &\quad + (\beta_{1,j}\beta_{2,j'} + \beta_{1,j'}\beta_{2,j})\text{Cov}(F_1, F_2). \end{aligned} \quad (18.6)$$

More generally, let

$$\mathbf{F}_t^\top = (F_{1,t}, \dots, F_{p,t}) \quad (18.7)$$

be the vector of  $p$  factors at time  $t$  and suppose that  $\boldsymbol{\Sigma}_F$  is the  $p \times p$  covariance matrix of  $\mathbf{F}_t$ . Define the vector of intercepts

$$\boldsymbol{\beta}_0^\top = (\beta_{0,1}, \dots, \beta_{0,n})$$

and the matrix of loadings

$$\boldsymbol{\beta} = \begin{pmatrix} \beta_{1,1} & \cdots & \beta_{1,j} & \cdots & \beta_{1,n} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ \beta_{p,1} & \cdots & \beta_{p,j} & \cdots & \beta_{p,n} \end{pmatrix}.$$

Also, define

$$\boldsymbol{\epsilon}^\top = (\epsilon_{1,t}, \dots, \epsilon_{n,t}) \quad (18.8)$$

and let  $\boldsymbol{\Sigma}_\epsilon$  be the  $n \times n$  diagonal covariance matrix of  $\boldsymbol{\epsilon}$ :

$$\boldsymbol{\Sigma}_\epsilon = \begin{pmatrix} \sigma_{\epsilon,1}^2 & \cdots & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & \cdots & \sigma_{\epsilon,j}^2 & \cdots & 0 \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & \cdots & \sigma_{\epsilon,n}^2 \end{pmatrix}.$$

Finally, let

$$\mathbf{R}_t^T = (R_{1,t}, \dots, R_{n,t}) \quad (18.9)$$

be the vector of all returns at time  $t$ . Model (18.3) then can be reexpressed in matrix notation as

$$\mathbf{R}_t = \boldsymbol{\beta}_0 + \boldsymbol{\beta}^T \mathbf{F}_t + \boldsymbol{\epsilon}_t. \quad (18.10)$$

Therefore, the  $n \times n$  covariance matrix of  $\mathbf{R}_t$  is

$$\boldsymbol{\Sigma}_R = \boldsymbol{\beta}^T \boldsymbol{\Sigma}_F \boldsymbol{\beta} + \boldsymbol{\Sigma}_{\epsilon}. \quad (18.11)$$

In particular, if  $\boldsymbol{\beta}_j = (\beta_{1,j} \ \cdots \ \beta_{p,j})^T$  is the  $j$ th column of  $\boldsymbol{\beta}$ , then the variance of the  $j$ th return is

$$\text{Var}(R_j) = \boldsymbol{\beta}_j^T \boldsymbol{\Sigma}_F \boldsymbol{\beta}_j + \sigma_{\epsilon_j}^2, \quad (18.12)$$

and the covariance between the  $j$ th and  $j'$ th returns is

$$\text{Cov}(R_j, R_{j'}) = \boldsymbol{\beta}_j^T \boldsymbol{\Sigma}_F \boldsymbol{\beta}_{j'}. \quad (18.13)$$

To use (18.11), (18.12) or (18.13), one needs estimates of  $\boldsymbol{\beta}$ ,  $\boldsymbol{\Sigma}_F$ , and  $\boldsymbol{\Sigma}_{\epsilon}$ . The regression coefficients are used to estimate  $\boldsymbol{\beta}$ , the sample covariance of the factors can be used to estimate  $\boldsymbol{\Sigma}_F$ , and  $\hat{\boldsymbol{\Sigma}}_{\epsilon}$  can be the diagonal matrix of the mean residual sum of squared errors from the regressions; see equation (9.13).

Why estimate  $\boldsymbol{\Sigma}_R$  via a factor model instead of simply using the sample covariance matrix? One reason is estimation accuracy. This is another example of bias–variance tradeoff. The sample covariance matrix is unbiased, but it contains  $n(n+1)/2$  estimates, one for each covariance and each variance. Each of these parameters is estimated with error and when this many errors accumulate, the result can be a sizable loss of precision. In contrast, the factor model requires estimates of  $n \times p$  parameters in  $\boldsymbol{\beta}$ ,  $p(p+1)/2$  parameters in  $\boldsymbol{\Sigma}_F$ , and  $n$  parameters in the diagonal matrix  $\boldsymbol{\Sigma}_{\epsilon}$ , for a total of  $np + n + p(p+1)/2$  parameters. Typically,  $n$ , the number of returns, is large but  $p$ , the number of factors, is much smaller, so  $np + n + p(p+1)/2$  is much smaller than  $n(n+1)/2$ . For example, suppose there are 200 returns and 5 factors. Then  $n(n+1)/2 = 20,100$  but  $np + n + p(p+1)/2$  is only 1,215. The downside of the factor model is that there will be bias in the estimate of  $\boldsymbol{\Sigma}_R$  if the factor model is misspecified, especially if  $\boldsymbol{\Sigma}_{\epsilon}$  is not diagonal as the factor model assumes.

Another advantage of the factor model is expediency. Having fewer parameters to estimate is one convenience and another is ease of updating. Suppose a portfolio manager has implemented a factor model for  $n$  equities and now needs to add another equity. If the manager uses the sample covariance matrix, then the  $n$  sample covariances between the new return time series and the old ones must be computed. This requires that all  $n$  of the old time series be available. In comparison, with a factor model, the portfolio manager needs only to regress the new return time series on the factors. Only the  $p$  factor time series need to be available.

*Example 18.7. Estimating the covariance matrix of GE, IBM, and Mobil excess returns*

This example continues Example 18.6. Recall that the number of returns has been kept artificially low, since with more returns it would not have been possible to display the results. Therefore, this example merely illustrates the calculations and is not a typical application of factor modeling.

The estimate of  $\Sigma_F$  is the sample covariance matrix of the factors:

|        | Mkt.RF  | SMB     | HML     |
|--------|---------|---------|---------|
| Mkt.RF | 21.1507 | 4.2326  | -5.1045 |
| SMB    | 4.2326  | 8.1811  | -1.0760 |
| HML    | -5.1045 | -1.0760 | 7.1797  |

The estimate of  $\beta$  is the matrix of regression coefficients (without the intercepts):

|       | Mkt.RF  | SMB      | HML       |
|-------|---------|----------|-----------|
| ge    | 1.14071 | -0.37193 | 0.009503  |
| ibm   | 0.81145 | -0.31250 | -0.298302 |
| mobil | 0.98672 | -0.37530 | 0.372520  |

The estimate of  $\Sigma_\epsilon$  is the diagonal matrix of residual error MS values:

|      | [,1]   | [,2]   | [,3]   |
|------|--------|--------|--------|
| [1,] | 16.077 | 0.000  | 0.000  |
| [2,] | 0.000  | 31.263 | 0.000  |
| [3,] | 0.000  | 0.000  | 27.432 |

Therefore, the estimate of  $\beta^T \Sigma_F \beta$  is

|       | ge     | ibm    | mobil  |
|-------|--------|--------|--------|
| ge    | 24.960 | 19.303 | 19.544 |
| ibm   | 19.303 | 15.488 | 14.467 |
| mobil | 19.544 | 14.467 | 16.155 |

and the estimate of  $\beta^T \Sigma_F \beta + \Sigma_\epsilon$  is

|       | ge     | ibm    | mobil  |
|-------|--------|--------|--------|
| ge    | 41.036 | 19.303 | 19.544 |
| ibm   | 19.303 | 46.752 | 14.467 |
| mobil | 19.544 | 14.467 | 43.587 |

For comparison, the sample covariance matrix of the equity returns is

|       | ge     | ibm    | mobil  |
|-------|--------|--------|--------|
| ge    | 40.902 | 20.878 | 14.255 |
| ibm   | 20.878 | 46.491 | 11.518 |
| mobil | 14.255 | 11.518 | 43.357 |

The largest difference between the estimate of  $\beta^T \Sigma_F \beta + \Sigma_\epsilon$  and the sample covariance matrix is in the covariance between the excess returns on GE and Mobil. The reason for this large discrepancy is that the factor model assumes a zero residual correlation between these two variables, but, as we learned earlier, the data show a negative correlation of  $-0.25$ .

The code for the calculations in this example continues the code in Example 18.6. The addition code is:

```

sigF = as.matrix(var(cbind(Mkt.RF, SMB, HML)))
bbeta = as.matrix(fit$coef)
bbeta = t( bbeta[-1, ])
n = dim(CRSPmon)[1]
sigeps = (n - 1) / (n - 4) * as.matrix((var(as.matrix(fit$resid))))
sigeps = diag(as.matrix(sigeps))
sigeps = diag(sigeps, nrow = 3)
cov_equities = bbeta %*% sigF %*% t(bbeta) + sigeps
options(digits = 5)
sigF
bbeta
sigeps
bbeta %*% sigF %*% t(bbeta)
cov_equities
cov(stocks)

```

□

## 18.5 Cross-Sectional Factor Models

Models of the form (18.3) are *time series factor models*. They use time series data, one single asset at a time, to estimate the loadings.

As just discussed, time series factor models do not make use of variables such as dividend yields, book-to-market value, or other variables specific to the  $j$ th firm. An alternative is a *cross-sectional factor model*, which is a regression model using data from many assets but from only a single holding period. For example, suppose that  $R_j$ ,  $(B/M)_j$ , and  $D_j$  are the return, book-to-market value, and dividend yield for the  $j$ th asset for some fixed time  $t$ . Since  $t$  is fixed, it will not be made explicit in the notation. Then a possible cross-sectional factor model is

$$R_j = \beta_0 + \beta_1(B/M)_j + \beta_2 D_j + \epsilon_j.$$

The parameters  $\beta_1$  and  $\beta_2$  are unknown values at time  $t$  of a book-to-market value risk factor and a dividend yield risk factor. These values are estimated by regression.

There are two fundamental differences between time series factor models and cross-sectional factor models. The first is that with a time series factor model one estimates parameters, one asset at a time, using multiple holding

periods, while in a cross-sectional model one estimates parameters, one single holding period at a time, using multiple assets. The other major difference is that in a time series factor model, the factors are directly measured and the loadings are the unknown parameters to be estimated by regression. In a cross-sectional factor model the opposite is true; the loadings are directly measured and the factor values are estimated by regression.

*Example 18.8. An industry cross-sectional factor model*

This example uses the `berndtInvest.csv` used in Example 18.5. This data set has monthly returns on 15 stocks over 10 years, 1978 to 1987. The 15 stocks were classified into three industries, “Tech,” “Oil,” and “Other,” as follows:

|        | tech | oil | other |
|--------|------|-----|-------|
| CITCRP | 0    | 0   | 1     |
| CONED  | 0    | 0   | 1     |
| CONTIL | 0    | 1   | 0     |
| DATGEN | 1    | 0   | 0     |
| DEC    | 1    | 0   | 0     |
| DELTA  | 0    | 1   | 0     |
| GENMIL | 0    | 0   | 1     |
| GERBER | 0    | 0   | 1     |
| IBM    | 1    | 0   | 0     |
| MOBIL  | 0    | 1   | 0     |
| PANAM  | 0    | 1   | 0     |
| PSNH   | 0    | 0   | 1     |
| TANDY  | 1    | 0   | 0     |
| TEXACO | 0    | 1   | 0     |
| WEYER  | 0    | 0   | 1     |

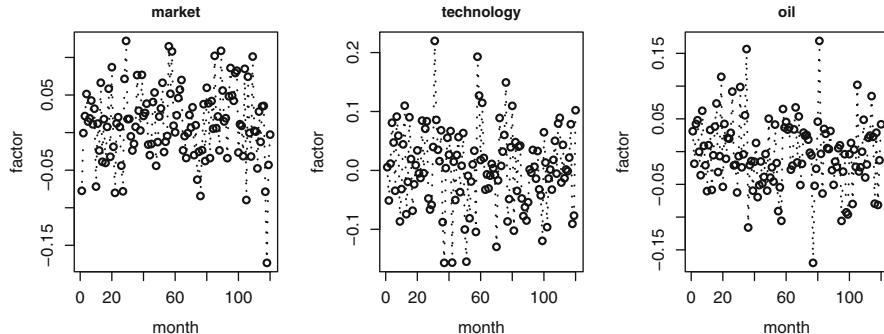
We used the indicator variables of “tech” and “oil” as loadings and fit the model

$$R_j = \beta_0 + \beta_1 \text{tech}_j + \beta_2 \text{oil}_j + \epsilon_j, \quad (18.14)$$

where  $R_j$  is the return on the  $j$ th stock,  $\text{tech}_j$  equals 1 if the  $j$ th stock is a technology stock and equals 0 otherwise, and  $\text{oil}_j$  is defined similarly. Model (18.14) was fit separately for each of the 120 months. The estimates  $\hat{\beta}_0$ ,  $\hat{\beta}_1$ , and  $\hat{\beta}_3$  for a month were the values of the three factors for that month. The loadings were the known values of  $\text{tech}_j$  and  $\text{oil}_j$ .

Factor 1, the values of  $\hat{\beta}_0$ , can be viewed as an overall market factor, since it affects all 15 returns. Factors 2 and 3 are the technology and oil factors. For example, if the value of factor 2 is positive in any given month, then Tech stocks have better-than-market returns that month. Figure 18.9 contains time series plots of the three factor series, and Fig. 18.10 shows their auto- and

cross-correlation functions. The largest cross-correlation is between the tech and oil factors at lag 0, which indicates that above- (below-) market returns for technology stocks are associated with above (below) market returns for oil stocks.



**Fig. 18.9.** Time series plots of the estimated values of the three factors in the cross-sectional factor model.

The standard deviations of the three factors are

| market | tech  | oil   |
|--------|-------|-------|
| 0.049  | 0.069 | 0.053 |

There are other ways of defining the factors. For example, Zivot and Wang (2006) use the model

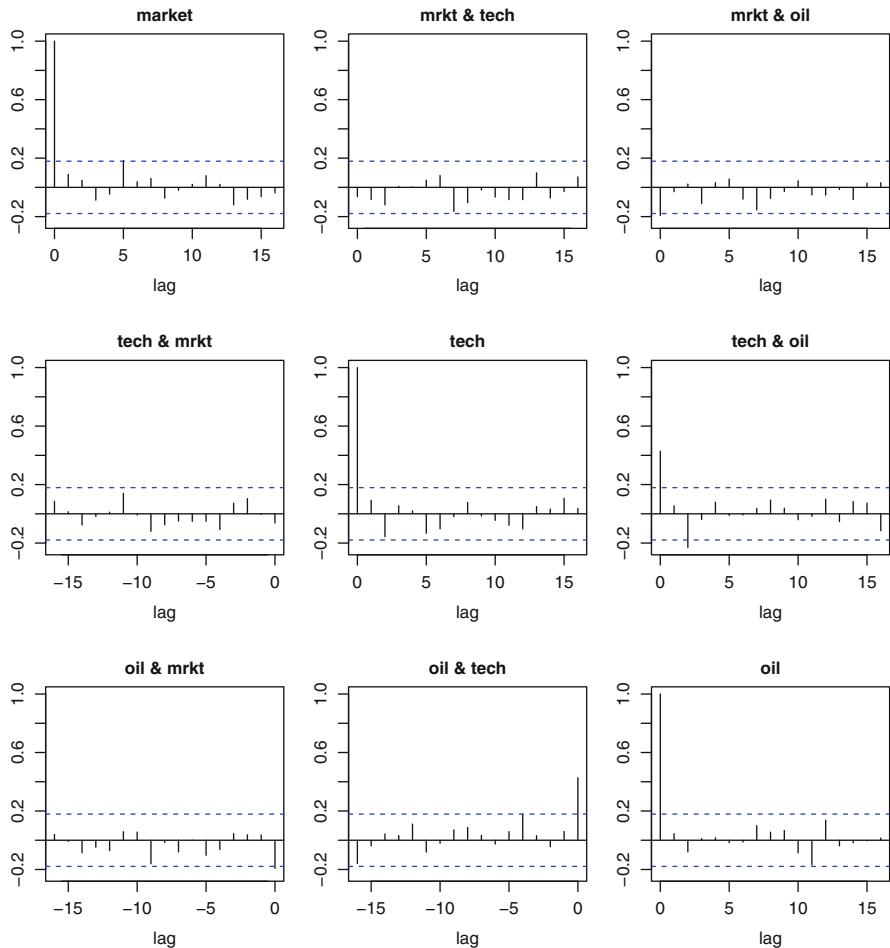
$$R_j = \beta_1 \text{tech}_j + \beta_2 \text{oil}_j + \beta_3 \text{other}_j + \epsilon_j, \quad (18.15)$$

with no intercept but with  $\text{other}_J$  as a third variable. With this model, there is no market factor but instead factors for all three industries. The model with an intercept but without  $\text{other}$  is equivalent to the model with  $\text{other}$  in place of the intercept, in the sense that the two models produce the same fitted values.  $\square$

Cross-sectional factor models are sometimes called BARRA models after BARRA, Inc., a company that has been developing cross-sectional factor models and marketing the output of their models to financial managers.

## 18.6 Statistical Factor Models

In a statistical factor model, neither the factor values nor the loadings are directly observable. All that is available is the sample  $\mathbf{Y}_1, \dots, \mathbf{Y}_n$  or, perhaps, only the sample covariance matrix. This is the same type of data available



**Fig. 18.10.** Auto- and cross-correlation plots of the estimated three factors in the cross-sectional factor model. Series 1–3 are the market, tech, and oil factors, respectively.

for PCA and we will see that statistical factor analysis and PCA have some common characteristics. As with PCA, one can work with either the standardized or unstandardized variables. R's `factanal()` function automatically standardizes the variables.

We start with the multifactor model in matrix notation (18.10) and the return covariance matrix (18.11) which for convenience will be repeated as

$$\mathbf{R}_t = \boldsymbol{\beta}_0 + \boldsymbol{\beta}^T \mathbf{F}_t + \boldsymbol{\epsilon}_t. \quad (18.16)$$

and

$$\boldsymbol{\Sigma}_R = \boldsymbol{\beta}^\top \boldsymbol{\Sigma}_F \boldsymbol{\beta} + \boldsymbol{\Sigma}_\epsilon. \quad (18.17)$$

Here  $\boldsymbol{\beta}^\top$  is  $d \times p$  where  $d$  is the dimension of  $R_t$  and  $p$  is the number of factors.

The only component of (18.17) that can be estimated directly from the data is  $\boldsymbol{\Sigma}_R$ . One can use this estimate to find estimates of  $\boldsymbol{\beta}$ ,  $\boldsymbol{\Sigma}_F$ , and  $\boldsymbol{\Sigma}_\epsilon$ . However, it is too much to ask that all three of these matrices be identified from  $\boldsymbol{\Sigma}_R$  alone. Here is the problem: Let  $\mathbf{A}$  be any  $p \times p$  invertible matrix. Then the returns vector  $\mathbf{R}_t$  in (18.16) is unchanged if  $\boldsymbol{\beta}^\top$  is replaced by  $\boldsymbol{\beta}^\top \mathbf{A}^{-1}$  and  $\mathbf{F}_t$  is replaced by  $\mathbf{A}\mathbf{F}_t$ . Therefore, the returns only determine  $\boldsymbol{\beta}$  and  $\mathbf{F}_t$  up to a nonsingular linear transformation, and consequently a set of constraints is needed to identify the parameters. The usual constraints are the factors are uncorrelated and standardized, so that

$$\boldsymbol{\Sigma}_F = \mathbf{I}, \quad (18.18)$$

where  $\mathbf{I}$  is the  $p \times p$  identity matrix. With these constraints, (18.17) simplifies to the statistical factor model

$$\boldsymbol{\Sigma}_R = \boldsymbol{\beta}^\top \boldsymbol{\beta} + \boldsymbol{\Sigma}_\epsilon. \quad (18.19)$$

However, even with this simplification,  $\boldsymbol{\beta}$  is only determined up to a rotation, that is, by multiplication by an orthogonal matrix. To appreciate why this is so, let  $\mathbf{P}$  be any orthogonal matrix, so that  $\mathbf{P}^\top = \mathbf{P}^{-1}$ . Then (18.19) is unchanged if  $\boldsymbol{\beta}$  is replaced by  $\mathbf{P}\boldsymbol{\beta}$  since

$$(\mathbf{P}\boldsymbol{\beta})^\top (\mathbf{P}\boldsymbol{\beta}) = \boldsymbol{\beta}^\top \mathbf{P}^\top \mathbf{P}\boldsymbol{\beta} = \boldsymbol{\beta}^\top \mathbf{P}^{-1} \mathbf{P}\boldsymbol{\beta} = \boldsymbol{\beta}^\top \boldsymbol{\beta}.$$

Therefore, to determine  $\boldsymbol{\beta}$  a further set of constraints is needed. One possible set of constraints is that  $\boldsymbol{\beta}\boldsymbol{\Sigma}_\epsilon^{-1}\boldsymbol{\beta}^\top$  is diagonal (Mardia et al., 1979, p. 258). Output from R's function `factanal()` satisfies this constraint when the argument `rotation` is set to "none". If  $\boldsymbol{\beta}$  is rotated as discussed in Sect. 18.6.1, then this constraint no longer holds.

If the main purpose of the statistical factor model is to estimate  $\boldsymbol{\Sigma}_R$  by (18.19), then the choice of constraint is irrelevant since all constraints lead to the same product  $\boldsymbol{\beta}^\top \boldsymbol{\beta}$ . In particular, rotation of  $\boldsymbol{\beta}$  does not change the estimate of  $\boldsymbol{\Sigma}_R$ .

It is helpful to compare three estimates of  $\boldsymbol{\Sigma}_R$ . The sample covariance matrix has full rank (rank =  $d$ ) provided that  $n > d$  as will be assumed here. Instead of the sample covariance matrix, one can perform PCA and estimate  $\boldsymbol{\Sigma}_R$  by the sample covariance matrix of the first  $p < d$  principal components. Then

$$\widehat{\boldsymbol{\Sigma}}_R = \mathbf{O}^\top \mathbf{O}.$$

where  $\mathbf{O}^\top$  is the  $d \times p$  matrix whose columns are the first  $d$  principal axes (eigenvectors) and the rank of  $\widehat{\boldsymbol{\Sigma}}_R$  is only  $p$  so less than full rank. In contrast, (18.19) provides a full-rank estimate of  $\boldsymbol{\Sigma}_R$  but with a simple structure, the sum of a rank  $p$  matrix and a diagonal matrix.

*Example 18.9. Factor analysis of equity funds*

This example continues the analysis of the equity funds data set that was used in Example 18.3 to illustrate PCA. The code for fitting a 4-factor model ( $p = 4$ ) using `factanal()` is:

```
equityFunds = read.csv("equityFunds.csv")
fa_none = factanal(equityFunds[, 2:9], 4, rotation = "none")
print(fa_none, cutoff = 0.1)
```

Here we specify no rotations. The output is:

```
> factanal(equityFunds[,2:9],4,rotation="none")
```

```
Call:
factanal(x = equityFunds[, 2:9], factors = 4,
          rotation = "none")
```

Uniquenesses:

|  | EASTEU | LATAM | CHINA | INDIA | ENERGY | MINING | GOLD  | WATER |
|--|--------|-------|-------|-------|--------|--------|-------|-------|
|  | 0.735  | 0.368 | 0.683 | 0.015 | 0.005  | 0.129  | 0.005 | 0.778 |

Loadings:

|        | Factor1 | Factor2 | Factor3 | Factor4 |
|--------|---------|---------|---------|---------|
| EASTEU | 0.387   | 0.169   | 0.293   |         |
| LATAM  | 0.511   | 0.167   | 0.579   |         |
| CHINA  | 0.310   | 0.298   | 0.362   |         |
| INDIA  | 0.281   | 0.951   |         |         |
| ENERGY | 0.784   |         |         | 0.614   |
| MINING | 0.786   |         | 0.425   | -0.258  |
| GOLD   | 0.798   |         |         | -0.596  |
| WATER  | 0.340   |         | 0.298   | 0.109   |

|                | Factor1 | Factor2 | Factor3 | Factor4 |
|----------------|---------|---------|---------|---------|
| SS loadings    | 2.57    | 1.07    | 0.82    | 0.82    |
| Proportion Var | 0.32    | 0.13    | 0.10    | 0.10    |
| Cumulative Var | 0.32    | 0.46    | 0.56    | 0.66    |

Test of the hypothesis that 4 factors are sufficient.  
The chi square statistic is 17 on 2 degrees of freedom.  
The p-value is 2e-04

The “loadings” are the estimates  $\hat{\beta}^T$ . Since there are eight funds and four factors, the loadings are in an  $8 \times 4$  matrix `fa_none$loadings`. The output above gives the sums of squares of the eight loadings for each factor. The `Proportion Var` row contains the `SS loadings` divided by 8, where 8 is the sum of the variances of the eight variables, since each variable has been standardized to have variance equal to 1.

By convention, any loading with an absolute value less than the parameter `cutoff` is not printed, and the default value of `cutoff` is 0.1.

Because all its loadings have the same sign, the first factor is an overall index of the eight funds. The second factor has large loadings on the four regional funds (EASTEU, LATAM, CHINA, INDIA) and small loadings on the four industry section funds (ENERGY, MINING, GOLD, WATER). The four regions are all emerging markets, so the second factor might be interpreted as an emerging markets factor. The fourth factor is a contrast of MINING and GOLD with ENERGY and WATER, and mimics a hedge portfolio that is long on ENERGY and WATER and short on GOLD and MINING. The third factor is less interpretable. The uniquenesses are the diagonal elements of the estimate  $\hat{\Sigma}_\epsilon$ .

The output gives a  $p$ -value for testing the null hypothesis that there are at most four factors. The  $p$ -value is small, indicating that the null hypothesis should be rejected. However, four is that maximum number of factors that can be used by `factanal()` when there are only eight returns. Should we be concerned that we are not using enough factors? Recall the important distinction between statistical and practical significance that has been emphasized elsewhere in this book. One way to assess practical significance is to see how well the factor model can reproduce the sample correlation matrix. Since `factanal()` standardizes the variables, the factor model estimate of the correlation matrix is the estimate of the covariance matrix, that, using (18.19), is

$$\hat{\beta}^\top \hat{\beta} + \hat{\Sigma}_\epsilon. \quad (18.20)$$

The code to calculate this estimate is

```
B_none = fa_none$loadings[, ]
BB_none = B_none %*% t(B_none)
D_none = diag(fa_none$unique)
Sigma_R_hat = BB_none + D_none
```

Here `B_none` is  $\hat{\beta}^\top$  with no rotation, `BB_none` equals  $\hat{\beta}^\top \hat{\beta}$  and `D_none` equals  $\hat{\Sigma}_\epsilon$ .

The difference between this estimate and the sample correlation matrix is a  $8 \times 8$  matrix. We would like all of its entries to be close to 0. Unfortunately, they are not as small as we would like. There are various ways to check if a matrix this size is “small.” The smallest entry is  $-0.063$  and the largest is  $0.03$ . These are reasonably large discrepancies between correlation matrices. Also, the eigenvalues of the difference are

```
-7.5e-02 -6.0e-03 -3.4e-15 -2.0e-15
-1.3e-15  3.0e-15  7.7e-03  7.3e-02
```

Another way to check for smallness of the difference between the two estimates is to look at the estimates of the variance of an equally weighted portfolio (of the standardized returns), which is

$$\mathbf{w}^\top \boldsymbol{\Sigma}_R \mathbf{w},$$

where  $\mathbf{w}^\top = (1/8, \dots, 1/8)$ . These estimates are 0.37 and 0.42 using the factor model and the sample correlation matrix, respectively. The absolute difference, 0.06, is relatively large compared to either of the estimates. It is unclear whether this difference is due to a more parsimonious and accurate fit by the factor model (good) or due to bias from a lack of fit by the factor model (not good).  $\square$

### 18.6.1 Varimax Rotation of the Factors

As discussed earlier, the estimate of the covariance matrix is unchanged if the loadings  $\beta$  are rotated by multiplication by an orthogonal matrix. Rotation might increase the interpretability of the loadings. In some applications, it is desirable for each loading to be either close to 0 or large, so that a variable will load only on a few factors, or even on only one factor. *Varimax* rotation attempts to make each loading either small or large by maximizing the sum of the variances of the squared loadings. Varimax rotation is the default with R's `factanal()` function, but this can be changed as in Example 18.9 where no rotation was used. In finance, having variables loading on only one or a few factors is not that important, and may even be undesirable, so varimax rotation may not be advantageous.

We repeat again for emphasis that the estimate of  $\boldsymbol{\Sigma}_R$  is not changed by rotation. The uniquenesses are also unchanged. Only the loadings change.

*Example 18.10. Factor analysis of equity funds: Varimax rotation*

The statistical factor analysis in Example 18.9 is repeated here but now with varimax rotation.

Call:

```
factanal(x = equityFunds[, 2:9], factors = 4,
          rotation = "varimax")
```

Uniquenesses:

|        |       |       |       |        |        |       |       |
|--------|-------|-------|-------|--------|--------|-------|-------|
| EASTEU | LATAM | CHINA | INDIA | ENERGY | MINING | GOLD  | WATER |
| 0.735  | 0.368 | 0.683 | 0.015 | 0.005  | 0.129  | 0.005 | 0.778 |

Loadings:

|        | Factor1 | Factor2 | Factor3 | Factor4 |
|--------|---------|---------|---------|---------|
| EASTEU | 0.436   | 0.175   | 0.148   | 0.148   |
| LATAM  | 0.748   | 0.174   |         | 0.180   |
| CHINA  | 0.494   |         | 0.247   |         |
| INDIA  | 0.243   |         | 0.959   |         |
| ENERGY | 0.327   | 0.118   |         | 0.934   |

|        |       |       |       |
|--------|-------|-------|-------|
| MINING | 0.655 | 0.637 | 0.168 |
| GOLD   | 0.202 | 0.971 |       |
| WATER  | 0.418 |       | 0.188 |

|                | Factor1 | Factor2 | Factor3 | Factor4 |
|----------------|---------|---------|---------|---------|
| SS loadings    | 1.80    | 1.45    | 1.03    | 1.00    |
| Proportion Var | 0.23    | 0.18    | 0.13    | 0.12    |
| Cumulative Var | 0.23    | 0.41    | 0.54    | 0.66    |

Test of the hypothesis that 4 factors are sufficient.  
The chi square statistic is 17 on 2 degrees of freedom.  
The p-value is 2e-04

The most notable change compared to the nonrotated loadings is that now all loadings with an absolute value above 0.1 are positive. Therefore, the factors all represent long positions, whereas before some were more like hedge portfolios. However, the rotated factors seem less interpretable compared to the unrotated factors, so a financial analyst might prefer the unrotated factors.  $\square$

## 18.7 Bibliographic Notes

The Fama–French three-factor model was introduced by Fama and French (1993) and discussed further in Fama and French (1995, 1996). Connor (1995) compares the three types of factor models and finds that macroeconomic factor models have less explanatory power than other factor models. Example 18.5 was adopted from Zivot and Wang (2006). Sharpe, Alexander, and Bailey (1999) has a brief description of the BARRA, Inc. factor model. The `yields.txt` data set is from the `Rsafd` package distributed by Professor René Carmona.

## 18.8 R Lab

### 18.8.1 PCA

In the first section of this lab, you will do a principal components analysis of daily yield data in the file `yields.txt`. R has functions, which we will use later, that automate PCA, but it is easy to do PCA “from scratch” and it is instructive to do this. First load the data and, to get a feel for what yield curves look like, plot the yield curves on days 1, 101, 201, 301, ..., 1101. There are 1352 yield curves in the data, so you will see a representative sample of them. The yield curves change slowly, which is why one should look at yield curves that are spaced rather far (100 days) apart.

```

yieldDat = read.table("yields.txt", header = T)
maturity = c((0:5), 5.5, 6.5, 7.5, 8.5, 9.5)
pairs(yieldDat)
par(mfrow = c(4,3))
for (i in 0:11)
{
  plot(maturity, yieldDat[100 * i + 1, ], type = "b")
}

```

Next compute the eigenvalues and eigenvectors of the sample covariance matrix, print the results, and plot the eigenvalues as a scree plot.

```

eig = eigen(cov(yieldDat))
eig$values
eig$vectors
par(mfrow = c(1, 1))
barplot(eig$values)

```

The following R code plots the first four eigenvectors.

```

par(mfrow=c(2, 2))
plot(eig$vector[ , 1], ylim = c(-0.7, 0.7), type = "b")
abline(h = 0)
plot(eig$vector[ , 2], ylim = c(-0.7, 0.7), type = "b")
abline(h = 0)
plot(eig$vector[ , 3], ylim = c(-0.7, 0.7), type = "b")
abline(h = 0)
plot(eig$vector[ , 4], ylim = c(-0.7, 0.7), type = "b")
abline(h = 0)

```

**Problem 1** *It is generally recommended that PCA be applied to time series that are stationary. Plot the first column of yieldDat. (You can look at other columns as well. You will see that they are fairly similar.) Does the plot appear stationary? Why or why not? Include your plot with your work.*

Another way to check for stationarity is to run the augmented Dickey–Fuller test. You can do that with the following code:

```

library("tseries")
adf.test(yieldDat[ , 1])

```

**Problem 2** *Based on the augmented Dickey–Fuller test, do you think the first column of yieldDat is stationary? Why or why not?*

Run the following code to compute changes in the yield curves. Notice the use of `[-1, ]` to delete the first row and similarly the use of `[-n, ]`.

```
n=dim(yieldDat)[1]
delta_yield = yieldDat[-1, ] - yieldDat[-n, ]
```

Plot the first column of `delta_yield` and run the augmented Dickey–Fuller test to check for stationarity.

**Problem 3** Do you think the first column of `delta_yield` is stationary? Why or why not?

Run the following code to perform a PCA using the function `princomp()`, which is similar, although not identical, to `prcomp()`. By default, `princomp()` does a PCA on the covariance matrix, though there is an option to use the correlation matrix instead. We will use the covariance matrix. The second line of the code will print the names of the components in the object that is returned by `princomp()`. As you can see, the `names` function can be useful for learning just what is being returned. You can also get this information by typing `?princomp`.

```
pca_del = princomp(delta_yield)
names(pca_del)
summary(pca_del)
par(mfrow = c(1, 1))
plot(pca_del)
```

**Problem 4 (a)** The output from `names` includes the following:

```
[1] "sdev"   "loadings" "center"  "scores"
```

Describe each of these components in mathematical terms. To answer this part of the question, you can print and plot the components to see what they contain and use R’s help for further information.

- (b) What are the first two eigenvalues of the covariance matrix?
- (c) What is the eigenvector corresponding to the largest eigenvalue?
- (d) Suppose you wish to “explain” at least 95 % of the variation in the changes in the yield curves. Then how many principal components should you use?

### 18.8.2 Fitting Factor Models by Time Series Regression

In this section, we will start with the one-factor CAPM model of Chap. 17 and then extend this model to the three-factor Fama–French model. We will use the data set `Stock_Bond_2004_to_2005.csv` on the book’s website, which contains stock prices and other financial time series for the years 2004 and 2005. Data on the Fama–French factors are available at Prof. Kenneth French’s website

[http://mba.tuck.dartmouth.edu/pages/faculty/ken.french/  
data\\_library.html#Research](http://mba.tuck.dartmouth.edu/pages/faculty/ken.french/data_library.html#Research)

where RF is the risk-free rate and Mkt.RF, SMB, and HML are the Fama–French factors.

Go to Prof. French’s website and get the daily values of RF, Mkt.RF, SMB, and HML for the years 2004–2005. It is assumed here that you’ve put the data in a text file `FamaFrenchDaily.txt`. Returns on this website are expressed as percentages.

Now fit the CAPM to the four stocks using the `lm` command. This code fits a linear regression model separately to the four responses. In each case, the independent variable is Mkt.RF.

```
# Uses daily data 2004–2005

stocks = read.csv("Stock_Bond_2004_to_2005.csv", header=T)
attach(stocks)
stocks_subset = as.data.frame(cbind(GM_AC, F_AC, UTX_AC, MRK_AC))
stocks_diff = as.data.frame(100 * apply(log(stocks_subset),
  2, diff) - FF_data$RF)
names(stocks_diff) = c("GM", "Ford", "UTX", "Merck")

FF_data = read.table("FamaFrenchDaily.txt", header = TRUE)
FF_data = FF_data[-1, ] # delete first row since stocks_diff
# lost a row due to differencing

fit1 = lm(as.matrix(stocks_diff) ~ FF_data$Mkt.RF)
summary(fit1)
```

**Problem 5** *The CAPM predicts that all four intercepts will be zero. For each stock, using  $\alpha = 0.025$ , can you accept the null hypothesis that its intercept is zero? Why or why not? Include the p-values with your work.*

**Problem 6** *The CAPM also predicts that the four sets of residuals will be uncorrelated. What is the correlation matrix of the residuals? Give a 95 % confidence interval for each of the six correlations. Can you accept the hypothesis that all six correlations are zero?*

**Problem 7** *Regardless of your answer to Problem 6, assume for now that the residuals are uncorrelated. Then use the CAPM to estimate the covariance matrix of the excess returns on the four stocks. Compare this estimate with the sample covariance matrix of the excess returns. Do you see any large discrepancies between the two estimates of the covariance matrix?*

Next, you will fit the Fama–French three-factor model. Run the following R code, which is much like the previous code except that the regression model has two additional predictor variables, `SMB` and `HML`.

```
fit2 = lm(as.matrix(stocks_diff) ~ FF_data$Mkt.RF +
  FF_data$$SMB + FF_data$$HML)
summary(fit2)
```

**Problem 8** *The CAPM predicts that for each stock, the slope (beta) for `SMB` and `HML` will be zero. Explain why the CAPM makes this prediction. Do you accept this null hypothesis? Why or why not?*

**Problem 9** *If the Fama–French model explains all covariances between the returns, then the correlation matrix of the residuals should be diagonal. What is the estimated correlations matrix? Would you accept the hypothesis that the correlations are all zero?*

**Problem 10** *Which model, CAPM or Fama–French, has the smaller value of AIC? Which has the smaller value of BIC? What do you conclude from this?*

**Problem 11** *What is the covariance matrix of the three Fama–French factors?*

**Problem 12** *In this problem, Stocks 1 and 2 are two stocks, not necessarily in the `Stock_FX_Bond_2004_to_2005.csv` data set. Suppose that Stock 1 has betas of 0.5, 0.4, and -0.1 with respect to the three factors in the Fama–French model and a residual variance of 23.0. Suppose also that Stock 2 has betas of 0.6, 0.15, and 0.7 with respect to the three factors and a residual variance of 37.0. Regardless of your answer to Problem 9, when doing this problem, assume that the three factors do account for all covariances.*

- (a) *Use the Fama–French model to estimate the variance of the excess return on Stock 1.*
- (b) *Use the Fama–French model to estimate the variance of the excess return on Stock 2.*
- (c) *Use the Fama–French model to estimate the covariance between the excess returns on Stock 1 and Stock 2.*

### 18.8.3 Statistical Factor Models

This section applies statistical factor analysis to the log returns of 10 stocks in the data set `Stock_FX_Bond.csv`. The data set contains adjusted closing

(AC) prices of the stocks, as well as daily volumes and other information that we will not use here.

The following R code will read the data, compute the log returns, and fit a two-factor model. Note that `factanal` works with the correlation matrix or, equivalently, with standardized variables.

```
dat = read.csv("Stock_FX_Bond.csv")
stocks_ac = dat[ , c(3, 5, 7, 9, 11, 13, 15, 17)]
n = length(stocks_ac[ , 1])
stocks_returns = log(stocks_ac[-1, ] / stocks_ac[-n, ])
fact = factanal(stocks_returns, factors = 2, rotation = "none")
print(fact)
```

Loadings less than the parameter `cutoff` are not printed. The default value of `cutoff` is 0.1, but you can change it as in “`print(fact, cutoff = 0.01)`” or “`print(fact, cutoff = 0)`”.

**Problem 13** *What are the factor loadings? What are the variances of the unique risks for Ford and General Motors?*

**Problem 14** *Does the likelihood ratio test suggest that two factors are enough? If not, what is the minimum number of factors that seems sufficient?*

The following code will extract the loadings and uniquenesses.

```
loadings = matrix(as.numeric(loadings(fact)), ncol = 2)
unique = as.numeric(fact$unique)
```

**Problem 15** *Regardless of your answer to Problem 14, use the two-factor model to estimate the correlation of the log returns for Ford and IBM.*

## 18.9 Exercises

1. The file `yields2009.csv` on this book’s website contains daily Treasury yields for 2009. Perform a principal components analysis on changes in the yields. Describe your findings. How many principal components are needed to capture 98 % of the variability?
2. Perform a statistical factor analysis of the returns in the data set `mid-capD.ts` on the book’s website. How many factors did you select? Use (18.20) to estimate the covariance matrix of the returns.
3. Verify equation (18.6).
4. Compute the eigenvectors in Example 18.3 and offer an interpretation of the first two eigenvectors.

## References

- Connor, G. (1995) The three types of factor models: a comparison of their explanatory power. *Financial Analysts Journal*, 42–46.
- Fama, E. F., and French, K. R. (1992) The cross-section of expected stock returns. *Journal of Finance*, 47, 427–465.
- Fama, E. F., and French, K. R. (1993) Common risk factors in the returns on stocks and bonds. *Journal of Financial Economics*, 33, 3–56.
- Fama, E. F., and French, K. R. (1995) Size and book-to-market factors in earnings and returns. *Journal of Finance*, 50, 131–155.
- Fama, E. F., and French, K. R. (1996) Multifactor explanations of asset pricing anomalies. *Journal of Finance*, 51, 55–84.
- Mardia, K. V., Kent, J. T., and Bibby, J. M. (1979) *Multivariate Analysis*, Academic Press, London.
- Sharpe, W. F., Alexander, G. J., and Bailey, J. V. (1999) *Investments*, 6th ed., Prentice-Hall, Upper Saddle River, NJ.
- Zivot, E., and Wang, J. (2006) *Modeling Financial Time Series with S-PLUS*, 2nd ed., Springer, New York.

## Risk Management

### 19.1 The Need for Risk Management

The financial world has always been risky, and financial innovations such as the development of derivatives markets and the packaging of mortgages have now made risk management more important than ever, but also more difficult.

There are many different types of risk. *Market risk* is due to changes in prices. *Credit risk* is the danger that a counterparty does not meet contractual obligations, for example, that interest or principal on a bond is not paid. *Liquidity risk* is the potential extra cost of liquidating a position because buyers are difficult to locate. *Operational risk* is due to fraud, mismanagement, human errors, and similar problems.

Early attempts to measure risk such as duration analysis, discussed in Sect. 3.8.1 and used to estimate the market risk of fixed income securities, were somewhat primitive and of only limited applicability. In contrast, value-at-risk (VaR) and expected shortfall (ES) are widely used because they can be applied to all types of risks and securities, including complex portfolios.

VaR uses two parameters, the time horizon and the confidence level, which are denoted by  $T$  and  $1 - \alpha$ , respectively. Given these, the VaR is a bound such that the loss over the horizon is less than this bound with probability equal to the confidence coefficient. For example, if the horizon is one week, the confidence coefficient is 99 % (so  $\alpha = 0.01$ ), and the VaR is \$5 million, then there is only a 1 % chance of a loss exceeding \$5 million over the next week. We sometimes use the notation  $\text{VaR}(\alpha)$  or  $\text{Var}(\alpha, T)$  to indicate the dependence of VaR on  $\alpha$  or on both  $\alpha$  and the horizon  $T$ . Usually,  $\text{VaR}(\alpha)$  is used with  $T$  being understood.

If  $\mathcal{L}$  is the loss over the holding period  $T$ , then  $\text{VaR}(\alpha)$  is the  $\alpha$ th upper quantile of  $\mathcal{L}$ . Equivalently, if  $\mathcal{R} = -\mathcal{L}$  is the revenue, then  $\text{VaR}(\alpha)$  is minus the  $\alpha$ th quantile of  $\mathcal{R}$ . For continuous loss distributions,  $\text{VaR}(\alpha)$  solves

$$P\{\mathcal{L} > \text{VaR}(\alpha)\} = P\{\mathcal{L} \geq \text{VaR}(\alpha)\} = \alpha, \quad (19.1)$$

and for any loss distribution, continuous or not,

$$\text{VaR}(\alpha) = \inf\{x : P(\mathcal{L} > x) \leq \alpha\}. \quad (19.2)$$

As will be discussed later, VaR has a serious deficiency—it discourages diversification—and for this reason it is being replaced by newer risk measures. One of these newer risk measures is the expected loss given that the loss exceeds VaR, which is called by a variety of names: *expected shortfall*, the *expected loss given a tail event*, *tail loss*, and *shortfall*. The name *expected shortfall* and the abbreviation ES will be used here.

For any loss distribution, continuous or not,

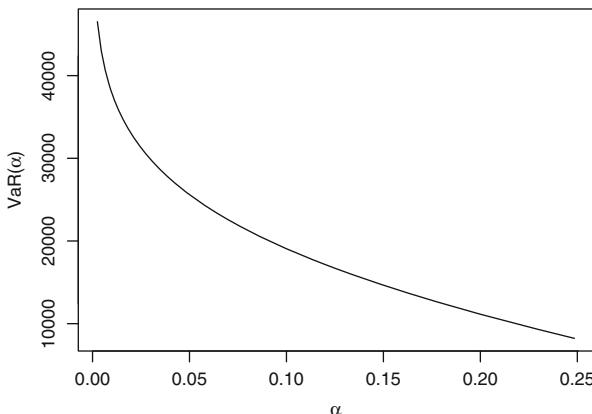
$$\text{ES}(\alpha) = \frac{\int_0^\alpha \text{VaR}(u) du}{\alpha}, \quad (19.3)$$

which is the average of  $\text{VaR}(u)$  over all  $u$  that are less than or equal to  $\alpha$ . If  $\mathcal{L}$  has a continuous distribution,

$$\text{ES}(\alpha) = E\{\mathcal{L} \mid \mathcal{L} > \text{VaR}(\alpha)\} = E\{\mathcal{L} \mid \mathcal{L} \geq \text{VaR}(\alpha)\}. \quad (19.4)$$

*Example 19.1. VaR with a normally distributed loss*

Suppose that the yearly return on a stock is normally distributed with mean 0.04 and standard deviation 0.18. If one purchases \$100,000 worth of this stock, what is the VaR with  $T$  equal to one year?



**Fig. 19.1.**  $\text{VaR}(\alpha)$  for  $0.025 < \alpha < 0.25$  when the loss distribution is normally distributed with mean  $-4000$  and standard deviation  $18,000$ .

To answer this question, we use the fact that the loss distribution is normal with mean  $-4000$  and standard deviation  $18,000$ , with all units in dollars. Therefore, VaR is

$$-4000 + 18,000z_\alpha,$$

where  $z_\alpha$  is the  $\alpha$ -upper quantile of the standard normal distribution.  $\text{VaR}(\alpha)$  is plotted as a function of  $\alpha$  in Fig. 19.1. VaR depends heavily on  $\alpha$  and in this figure ranges from 46,527 when  $\alpha$  is 0.025 to 8,226 when  $\alpha$  is 0.25.  $\square$

In applications, risk measures will rarely, if ever, be known exactly as in these simple examples. Instead, risk measures are estimated, and estimation error is another source of uncertainty. This uncertainty can be quantified using a confidence interval for the risk measure. We turn next to these topics.

## 19.2 Estimating VaR and ES with One Asset

To illustrate the techniques for estimating VaR and ES, we begin with the simple case of a single asset. In this section, these risk measures are estimated using historic data to estimate the distribution of returns. We make the assumption that returns are stationary, at least over the historic period we use. This is usually a reasonable assumption. We will also assume that the returns are independent. Independence is a much less reasonable assumption because of volatility clustering, and later we will remove this assumption by using GARCH models.

Two cases are considered, first without and then with the assumption of a parametric model for the return distribution.

### 19.2.1 Nonparametric Estimation of VaR and ES

We start with *nonparametric* estimates of VaR and ES, meaning that the loss distribution is not assumed to be in a parametric family such as the normal or  $t$ -distributions.

Suppose that we want a confidence coefficient of  $1 - \alpha$  for the risk measures. Therefore, we estimate the  $\alpha$ -quantile of the return distribution, which is the  $\alpha$ -upper quantile of the loss distribution. In the nonparametric method, this quantile is estimated as the  $\alpha$ -quantile of a sample of historic returns, which we will call  $\hat{q}(\alpha)$ . If  $S$  is the size of the current position, then the nonparametric estimate of VaR is

$$\widehat{\text{VaR}}^{\text{np}}(\alpha) = -S \times \hat{q}(\alpha),$$

with the minus sign converting revenue (return times initial investment) to a loss. In this chapter, superscripts and subscripts will sometimes be placed on VaR and ES to provide information. Here, the superscript “np” means “nonparametrically estimated.”

To estimate ES, let  $R_1, \dots, R_n$  be the historic returns and define  $\mathcal{L}_i = -S \times R_i$ . Then

$$\widehat{\text{ES}}^{\text{np}}(\alpha) = \frac{\sum_{i=1}^n \mathcal{L}_i I\{\mathcal{L}_i > \widehat{\text{VaR}}(\alpha)\}}{\sum_{i=1}^n I\{\mathcal{L}_i > \widehat{\text{VaR}}(\alpha)\}} = -S \times \frac{\sum_{i=1}^n R_i I\{R_i < \widehat{q}(\alpha)\}}{\sum_{i=1}^n I\{R_i < \widehat{q}(\alpha)\}}, \quad (19.5)$$

which is the average of all  $\mathcal{L}_i$  exceeding  $\widehat{\text{VaR}}^{\text{np}}(\alpha)$ . Here  $I\{\mathcal{L}_i > \widehat{\text{VaR}}^{\text{np}}(\alpha)\}$  is the indicator that  $\mathcal{L}_i$  exceeds  $\widehat{\text{VaR}}^{\text{np}}(\alpha)$ , and similarly for  $I\{R_i < \widehat{q}(\alpha)\}$ .

*Example 19.2. Nonparametric VaR and ES for a position in an S&P 500 index fund*

As a simple example, suppose that you hold a \$20,000 position in an S&P 500 index fund, so your returns are those of this index, and that you want a 24-h VaR. We estimate this VaR using the 1,000 daily returns on the S&P 500 for the period ending in April 1991. These log returns are a subset of the data set `SP500` in R's `Ecdat` package. The full time series is plotted in Fig. 4.1. Black Monday, with a log return of  $-0.23$ , occurs near the beginning of the shortened time series used in this example.

Suppose you want 95 % confidence. The 0.05 quantile of the returns computed by R's `quantile()` function is  $-0.0169$ . In other words, a daily return of  $-0.0169$  or less occurred only 5 % of the time in the historic data, so we estimate that there is a 5 % chance of a return of that size occurring during the next 24 h. A return of  $-0.0169$  on a \$20,000 investment yields a revenue of  $-\$337.5$ , and therefore the estimated  $\text{VaR}(0.05, 24 \text{ h})$  is \$337.43.

$\text{ES}(0.05)$  is obtained by averaging all returns below  $-0.0169$  and multiplying this average by  $-20,000$ . The result is  $\widehat{\text{ES}}^{\text{np}}(0.05) = \$619.3$ . The code for this example is below.

```

1 data(SP500, package="Ecdat")
2 n = 2783
3 SPreturn = SP500$r500[(n - 999):n]
4 year = 1981 + (1:n) * (1991.25 - 1981) / n
5 year = year[(n - 999):n]
6 alpha = 0.05
7 q = as.numeric(quantile(SPreturn, alpha))
8 VaR_nonp = -20000 * q
9 IEVaR = (SPreturn < q)
10 sum(IEVaR)
11 ES_nonp = -20000 * sum(SPreturn * IEVaR) / sum(IEVaR)
12 options(digits = 5)
13 VaR_nonp
14 ES_nonp

```

□

### 19.2.2 Parametric Estimation of VaR and ES

Parametric estimation of VaR and ES has a number of advantages. For example, parametric estimation allows the use of GARCH models to adapt the risk measures to the current estimate of volatility. Also, risk measures can be easily computed for a portfolio of stocks if we assume that their returns have a joint parametric distribution, such as a multivariate  $t$ -distribution. Nonparametric estimation using sample quantiles works best when the sample size and  $\alpha$  are reasonably large. With smaller sample sizes or smaller values of  $\alpha$ , it is preferable to use parametric estimation. In this section, we look at parametric estimation of VaR and ES when there is a single asset.

Let  $F(y|\boldsymbol{\theta})$  be a parametric family of distributions used to model the return distribution and suppose that  $\hat{\boldsymbol{\theta}}$  is an estimate of  $\boldsymbol{\theta}$ , such as, the MLE computed from historic returns. Then  $F^{-1}(\alpha|\hat{\boldsymbol{\theta}})$  is an estimate of the  $\alpha$ -quantile of the return distribution and

$$\widehat{\text{VaR}}^{\text{par}}(\alpha) = -S \times F^{-1}(\alpha|\hat{\boldsymbol{\theta}}) \quad (19.6)$$

is a parametric estimate of  $\text{VaR}(\alpha)$ . As before,  $S$  is the size of the current position.

Let  $f(y|\boldsymbol{\theta})$  be the density of  $F(y|\boldsymbol{\theta})$ . Then the estimate of expected shortfall is

$$\widehat{\text{ES}}^{\text{par}}(\alpha) = -\frac{S}{\alpha} \times \int_{-\infty}^{F^{-1}(\alpha|\hat{\boldsymbol{\theta}})} xf(x|\hat{\boldsymbol{\theta}}) dx. \quad (19.7)$$

The superscript “par” denotes “parametrically estimated.” Computing this integral is not always easy, but in the important cases of normal and  $t$ -distributions there are convenient formulas.

Suppose the return has a  $t$ -distribution with mean equal to  $\mu$ , scale parameter equal to  $\lambda$ , and tail index<sup>1</sup>  $\nu$ . Let  $f_\nu$  and  $F_\nu$  be, respectively, the  $t$ -density and  $t$ -distribution function with  $\nu$  degrees of freedom. The expected shortfall is

$$\widehat{\text{ES}}^t(\alpha) = S \times \left\{ -\mu + \lambda \left( \frac{f_\nu\{F_\nu^{-1}(\alpha)\}}{\alpha} \left[ \frac{\nu + \{F_\nu^{-1}(\alpha)\}^2}{\nu - 1} \right] \right) \right\}. \quad (19.8)$$

The formula for normal loss distributions is obtained by a direct calculation or letting  $\nu \rightarrow \infty$  in (19.8). The result is

$$\text{ES}^{\text{norm}}(\alpha) = S \times \left\{ -\mu + \sigma \left( \frac{\phi\{\Phi^{-1}(\alpha)\}}{\alpha} \right) \right\}, \quad (19.9)$$

where  $\mu$  and  $\sigma$  are the mean and standard deviation of the returns and  $\phi$  and  $\Phi$  are the standard normal density and CDF. The superscripts “t” and

---

<sup>1</sup> The tail index parameter for the  $t$ -distribution is also commonly referred to as the degrees-of-freedom parameter by its association with the theory of linear regression, and some R functions use the abbreviations df or nu.

“norm” denote estimates assuming a  $t$ -distributed return and normal return, respectively.

Parametric estimation with one asset is illustrated in the next example.

*Example 19.3. Parametric VaR and ES for a position in an S&P 500 index fund*

This example uses the same data set as in Example 19.2 so that parametric and nonparametric estimates can be compared. We will assume that the returns are i.i.d. with a  $t$ -distribution. Under this assumption, VaR is

$$\widehat{\text{VaR}}^t(\alpha) = -S \times \{\widehat{\mu} + q_{\alpha,t}(\widehat{\nu})\widehat{\lambda}\}, \quad (19.10)$$

where  $\widehat{\mu}$ ,  $\widehat{\lambda}$ , and  $\widehat{\nu}$  are the estimated mean, scale parameter, and tail index of a sample of returns. Also,  $q_{\alpha,t}(\widehat{\nu})$  is the  $\alpha$ -quantile of the  $t$ -distribution with tail index  $\widehat{\nu}$ , so that  $\{\widehat{\mu} + q_{\alpha,t}(\widehat{\nu})\widehat{\lambda}\}$  is the  $\alpha$ th quantile of the fitted distribution.

The  $t$ -distribution was fit using R’s `fitdistr()` function and the estimates were  $\widehat{\mu} = 0.000689$ ,  $\widehat{\lambda} = 0.007164$ , and  $\widehat{\nu} = 2.984$ . For later reference, the estimated standard deviation is  $\widehat{\sigma} = \widehat{\lambda}\sqrt{\widehat{\nu}/(\widehat{\nu}-2)} = 0.01248$ .

The 0.05-quantile of the  $t$ -distribution with tail index 2.984 is  $-2.3586$ . Therefore, by (19.6),

$$\widehat{\text{VaR}}^t(0.05) = -20000 \times \{0.000689 - (2.3586)(0.007164)\} = \$324.17.$$

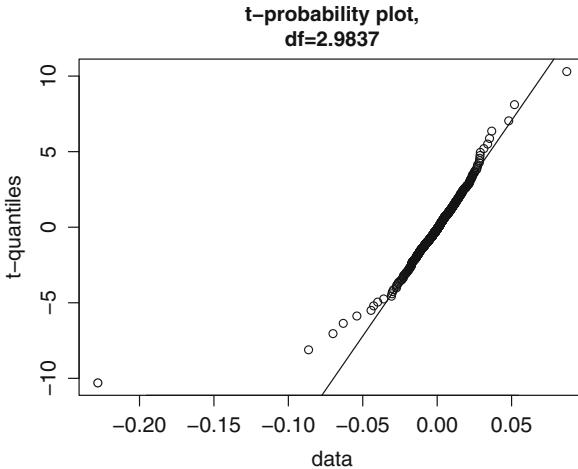
Notice that the nonparametric estimate,  $\widehat{\text{VaR}}^{\text{np}}(0.05) = \$337.55$ , is similar to, but somewhat larger than the parametric estimate,  $\$324.17$ .

The parametric estimate of  $\text{ES}^t(0.05)$  is  $\$543.81$  and is found by substituting  $S = 20,000$ ,  $\alpha = 0.05$ ,  $\widehat{\mu} = 0.000689$ ,  $\widehat{\lambda} = 0.007164$ , and  $\widehat{\nu} = 2.984$  into (19.8). The parametric estimate of  $\text{ES}^t(0.05)$  is noticeably shorter than the nonparametric. The reason the two estimates differ is that the extreme left tail of the returns, roughly the smallest 10 of 1,000 returns, is heavier than the tail of a  $t$ -distribution with 2.984 degrees of freedom; see the  $t$ -plot in Fig. 19.2. The code for this example is below.

```

15 data(SP500, package="Ecdat")
16 n = 2783
17 SPreturn = SP500$r500[(n - 999):n]
18 year = 1981 + (1:n) * (1991.25 - 1981) / n
19 year = year[(n - 999):n]
20 alpha = 0.05
21 library(MASS)
22 fitt = fitdistr(SPreturn, "t")
23 param = as.numeric(fitt$estimate)
24 mean = param[1]
25 df = param[3]
26 sd = param[2] * sqrt((df) / (df - 2))

```



**Fig. 19.2.** *t-plot of the S&P 500 returns used in Examples 19.2 and 19.3. The deviations from linearity in the tails, especially the left tail, indicate that the t-distribution does not fit the data in the extreme tails. The reference line goes through the first and third quartiles. The t-quantiles use 2.98 degrees of freedom, the MLE. The deviation in the left tail of the data from the t-distribution explains why the parametric estimate of ES is smaller than the nonparametric estimate.*

```

27 lambda = param[2]
28 qalpha = qt(alpha, df = df)
29 VaR_par = -20000 * (mean + lambda * qalpha)
30 es1 = dt(qalpha, df = df) / (alpha)
31 es2 = (df + qalpha^2) / (df - 1)
32 es3 = -mean + lambda * es1 * es2
33 ES_par = 20000*es3
34 VaR_par
35 ES_par

```

□

### 19.3 Bootstrap Confidence Intervals for VaR and ES

The estimates of VaR and ES are precisely that, just estimates. If we had used a different sample of historic data, then we would have gotten different estimates of these risk measures. We just calculated VaR and ES values to five significant digits, but do we really have that much precision? The reader has probably guessed (correctly) that we do not, but how much precision do we have? How can we learn the true precision of the estimates? Fortunately, a confidence interval for VaR or ES is rather easily obtained by bootstrapping. Any of the confidence interval procedures in Sect. 6.3 can be used. We will

see that even with 1,000 returns to estimate VaR and ES, these risk measures are estimated with considerable uncertainty.

For now, we will assume an i.i.d. sample of historic returns and use model-free resampling. In Sect. 19.4 we will allow for dependencies, for instance, GARCH effects, in the data and we will use model-based resampling.

Suppose we have a large number,  $B$ , of resamples of the returns data. Then a  $\text{VaR}(\alpha)$  or  $\text{ES}(\alpha)$  estimate is computed from each resample and for the original sample. The confidence interval can be based upon either a parametric or nonparametric estimator of  $\text{VaR}(\alpha)$  or  $\text{ES}(\alpha)$ . Suppose that we want the confidence coefficient of the interval to be  $1 - \gamma$ . The interval's confidence coefficient should not be confused with the confidence coefficient of VaR, which we denote by  $1 - \alpha$ . The  $\gamma/2$ -lower and -upper quantiles of the bootstrap estimates of  $\text{VaR}(\alpha)$  and  $\text{ES}(\alpha)$  are the limits of the basic percentile method confidence intervals.

It is worthwhile to restate the meanings of  $\alpha$  and  $\gamma$ , since it is easy to confuse these two confidence coefficients, but they need to be distinguished since they have rather different interpretations.  $\text{VaR}(\alpha)$  is defined so that the probability of a loss being greater than  $\text{VaR}(\alpha)$  is  $\alpha$ . On the other hand,  $\gamma$  is the confidence coefficient for the confidence interval for  $\text{VaR}(\alpha)$  and  $\text{ES}(\alpha)$ . If many confidence intervals are constructed, then approximately  $\gamma$  of them do not contain the true risk measure. Thus,  $\alpha$  is about the loss from the investment while  $\gamma$  is about the confidence interval being correct. An alternative way to view the difference between  $\alpha$  and  $\gamma$  is that  $\text{VaR}(\alpha)$  and  $\text{ES}(\alpha)$  are measuring risk due to uncertainty about future losses, assuming perfect knowledge of the loss distribution, while the confidence intervals tell us the uncertainty of these risk measures due to imperfect knowledge of the loss distribution.

*Example 19.4. Bootstrap confidence intervals for VaR and ES for a position in an S&P 500 index fund*

In this example, we continue Examples 19.2 and 19.3 and find an approximate confidence interval for  $\text{VaR}(\alpha)$  and  $\text{ES}(\alpha)$ . We use  $\alpha = 0.05$  as before and  $\gamma = 0.1$ .  $B = 5,000$  resamples were taken.

The basic percentile confidence intervals for  $\text{VaR}(0.05)$  were (297, 352) and (301, 346) using nonparametric and parametric estimators of  $\text{VaR}(0.05)$ , respectively. For  $\text{ES}(0.05)$ , the corresponding basic percentile confidence intervals were (487, 803) and (433, 605). We see that there is considerable uncertainty in the risk measures, especially for  $\text{ES}(0.05)$  and especially using nonparametric estimation.

When the first edition was written, the bootstrap computation took 33.3 minutes using an R program and a 2.13 GHz Pentium<sup>TM</sup> processor running under Windows<sup>TM</sup>. The computations took this long because the optimization step to find the MLE for parametric estimation is moderately expensive in computational time, at least if it is repeated 5,000 times. However, the same

computation took only 5.23 minutes on a 2.9 GHz MacBook Pro when the second edition was being written in 2014. Nonetheless, more computationally expensive estimators could easily take one-half hour or more to bootstrap even on a fast computer.

Waiting over a half an hour for the confidence interval may not be an attractive proposition. However, a reasonable measure of precision can be obtained with far fewer bootstrap repetitions. One might use only 50 repetitions, which would take less than a minute. This is not enough resamples to use basic percentile bootstrap confidence intervals, but instead one can use the normal approximation bootstrap confidence interval, (6.4). As an example, the normal approximation interval for the nonparametric estimate of  $\widehat{\text{VaR}}(0.05)$  is (301, 361) using only the first 50 bootstrap resamples. This interval gives the same general impression of accuracy as the above basic percentile method interval, (297, 352), that uses all 5,000 resamples.

The normal approximation interval assumes that  $\widehat{\text{VaR}}(0.05)$  is approximately normally distributed. This assumption is justified by the central limit theorem for sample quantiles (Sect. 4.3.1) and the fact that  $\widehat{\text{VaR}}(0.05)$  is a multiple of a sample quantile. The normal approximation does *not* require that the returns are normally distributed. In fact, we are modeling them as  $t$ -distributed when computing the parametric estimates.  $\square$

## 19.4 Estimating VaR and ES Using ARMA+GARCH Models

As we have seen in Chaps. 12 and 14, daily equity returns typically have a small amount of autocorrelation and a greater amount of volatility clustering. When calculating risk measures, the autocorrelation can be ignored if it is small enough, but the volatility clustering is less ignorable. In this section, we use ARMA+GARCH models so that  $\text{VaR}(\alpha)$  and  $\text{ES}(\alpha)$  can adjust to periods of high or low volatility.

Assume that we have  $n$  returns,  $R_1, \dots, R_n$  and we need to estimate VaR and ES for the next return  $R_{n+1}$ . Let  $\widehat{\mu}_{n+1|n}$  and  $\widehat{\sigma}_{n+1|n}$  be the estimated conditional mean and variance of tomorrow's return  $R_{n+1}$ , conditional on the current information set, which in this context is simply  $\{R_1, \dots, R_n\}$ . We will also assume that  $R_{n+1}$  has a conditional  $t$ -distribution with tail index  $\nu$ . After fitting an ARMA+GARCH model, we have estimates of  $\widehat{\nu}$ ,  $\widehat{\mu}_{n+1|n}$ , and  $\widehat{\sigma}_{n+1|n}$ . The estimated conditional scale parameter is

$$\widehat{\lambda}_{n+1|n} = \sqrt{(\widehat{\nu} - 2)/\widehat{\nu}} \widehat{\sigma}_{n+1|n}. \quad (19.11)$$

VaR and ES are estimated as in Sect. 19.2.2 but with  $\widehat{\mu}$  and  $\widehat{\lambda}$  replaced by  $\widehat{\mu}_{n+1|n}$  and  $\widehat{\lambda}_{n+1|n}$ .

*Example 19.5. VaR and ES for a position in an S&P 500 index fund using a GARCH(1,1) model*

An AR(1)+GARCH(1,1) model was fit to the log returns on the S&P 500. The AR(1) coefficient was small and not significantly different from 0, so a GARCH(1,1) was used for estimation of VaR and ES. The GARCH(1,1) fit is

```

46 library(rugarch)
47 garch.t = ugarchspec(mean.model=list(armaOrder=c(0,0)),
48                      variance.model=list(garchOrder=c(1,1)),
49                      distribution.model="std")
50 sp.garch.t = ugarchfit(data=SPreturn, spec=garch.t)
51 show(sp.garch.t)

Optimal Parameters
-----
      Estimate Std. Error   t value Pr(>|t|)
mu     0.000714  0.000264  2.70872 0.006754
omega  0.000003  0.000004  0.79083 0.429046
alpha1 0.032459  0.019439  1.66979 0.094961
beta1  0.939176  0.009296 101.02598 0.000000
shape   4.417464  0.560553  7.88054 0.000000

52 pred = ugarchforecast(sp.garch.t, data=SPreturn, n.ahead=1) ; pred

      Series      Sigma
T+1 0.0007144 0.009478

53 alpha = 0.05
54 nu = as.numeric(coef(sp.garch.t)[5])
55 q = qstd(alpha, mean=fitted(pred), sd=sigma(pred), nu=nu)
56 VaR = -20000*q ; VaR

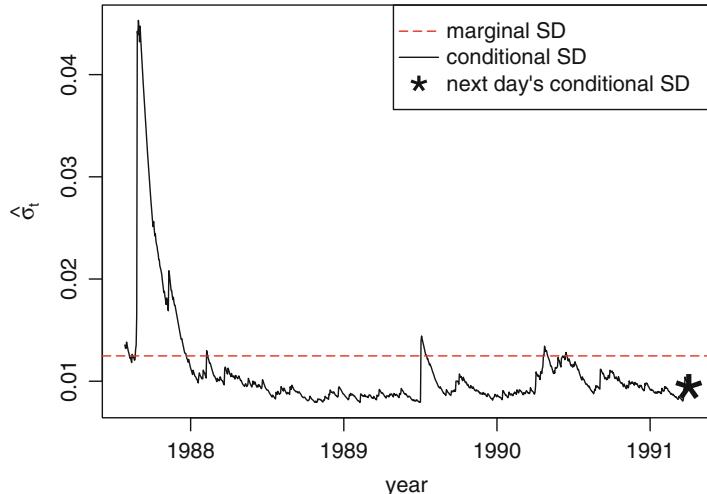
T+1          276.7298

57 lambda = sigma(pred)/sqrt( (nu)/(nu-2) )
58 qalpha = qt(alpha, df=nu)
59 es1 = dt(qalpha, df=nu)/(alpha)
60 es2 = (nu + qalpha^2) / (nu - 1)
61 es3 = -mean + lambda*es1*es2
62 ES_par = 20000*es3 ; ES_par

T+1          413.6518

```

The conditional mean and standard deviation of the next return were estimated to be 0.00071 and 0.00950. For the estimation of VaR and ES, the next return was assumed to have a  $t$ -distribution with these values for the mean and standard deviation and tail index 4.417. The estimate of VaR was \$276.73 and the estimate of ES was \$413.65. The VaR and ES estimates using the GARCH model are considerably smaller than the parametric estimates



**Fig. 19.3.** Estimated conditional standard deviation of the daily S&P 500 index returns based on a GARCH(1,1) model. The asterisk is a forecast of the next day's conditional standard deviation from the end of the return series, and the height of the dashed horizontal line is an estimate of the marginal (unconditional) standard deviation.

in Example 19.2 (\$323.42 and \$543.81), because the conditional standard deviation used here (0.00950) is smaller than the marginal standard deviation (0.01248) used in Example 19.2; see Fig. 19.3, where the dashed horizontal line's height is the marginal standard deviation and the conditional standard deviation of the next day's return is indicated by a large asterisk. The marginal standard deviation is inflated by periods of higher volatility such as in October 1987 (near Black Monday) on the left-hand side of Fig. 19.3. □

## 19.5 Estimating VaR and ES for a Portfolio of Assets

When VaR is estimated for a portfolio of assets rather than a single asset, parametric estimation based on the assumption of multivariate normal or  $t$ -distributed returns is very convenient, because the portfolio's return will have a univariate normal or  $t$ -distributed return. The portfolio theory and factor models developed in Chaps. 16 and 18 can be used to estimate the mean and variance of the portfolio's return.

Estimating VaR becomes complex when the portfolio contains stocks, bonds, options, foreign exchange positions, and other assets. However, when a portfolio contains only stocks, then VaR is relatively straightforward to estimate, and we will restrict attention to this case—see Sect. 19.10 for discussion of the literature covering more complex cases.

With a portfolio of stocks, means, variances, and covariances of returns could be estimated directly from a sample of returns as discussed in Chap. 16 or using a factor model as discussed in Sect. 18.4.2. They can also be estimated using the multivariate time series models discussed in Chaps. 13 and 14. Once these estimates are available, they can be plugged into Eqs. (16.6) and (16.7) to obtain estimates of the expected value and variance of the return on the portfolio, which are denoted by  $\hat{\mu}_P$  and  $\hat{\sigma}_P^2$ . Then, analogous to (19.10), VaR can be estimated, assuming normally distributed returns on the portfolio (denoted with a subscript “P”), by

$$\widehat{\text{VaR}}_P^{\text{norm}}(\alpha) = -S \times \{\hat{\mu}_P + \Phi^{-1}(\alpha)\hat{\sigma}_P\}, \quad (19.12)$$

where  $S$  is the initial value of the portfolio. Moreover, using (19.9), the estimated expected shortfall is

$$\widehat{\text{ES}}_P^{\text{norm}}(\alpha) = S \times \left\{ -\hat{\mu}_P + \hat{\sigma}_P \left( \frac{\phi\{\Phi^{-1}(\alpha)\}}{\alpha} \right) \right\}. \quad (19.13)$$

If the stock returns have a joint  $t$ -distribution, then the returns on the portfolio have a univariate  $t$ -distribution with the same tail index, and VaR and ES for the portfolio can be calculated using formulas in Sect. 19.2.2. If the returns on the portfolio have a  $t$ -distribution with mean  $\mu_P$ , scale parameter  $\lambda_P$ , and tail index  $\nu$ , then the estimated VaR is

$$\widehat{\text{VaR}}_P^t(\alpha) = -S \times \{\hat{\mu}_P + F_{\nu}^{-1}(\alpha)\hat{\lambda}_P\}, \quad (19.14)$$

and the estimated expected shortfall is

$$\widehat{\text{ES}}_P^t(\alpha) = S \times \left\{ -\hat{\mu}_P + \hat{\lambda}_P \left( \frac{f_{\hat{\nu}}\{F_{\hat{\nu}}^{-1}(\alpha)\}}{\alpha} \left[ \frac{\hat{\nu} + \{F_{\hat{\nu}}^{-1}(\alpha)\}^2}{\hat{\nu} - 1} \right] \right) \right\}. \quad (19.15)$$

*Example 19.6.* VaR and ES for portfolios of the three stocks in the CRSPday data set

This example uses the data set `CRSPday` used earlier in Examples 7.1 and 7.4. There are four variables—returns on GE, IBM, Mobil, and the CRSP index and we found in Example 7.4 that their returns can be modeled as having a multivariate  $t$ -distribution with tail index 5.94. In this example, we will only consider the returns on the three stocks. The  $t$ -distribution parameters were reestimated without the CRSP index and  $\hat{\nu}$  changed slightly to 5.81.

The estimated mean was

$$\hat{\mu} = (0.000858 \quad 0.000325 \quad 0.000616)^T$$

and the estimated covariance matrix was

$$\widehat{\Sigma} = \begin{pmatrix} 1.27e-04 & 5.04e-05 & 3.57e-05 \\ 5.04e-05 & 1.81e-04 & 2.40e-05 \\ 3.57e-05 & 2.40e-05 & 1.15e-04 \end{pmatrix}.$$

For an equally weighted portfolio with  $w = (1/3 \ 1/3 \ 1/3)^\top$ , the mean return for the portfolio is estimated to be

$$\widehat{\mu}_P = \widehat{\mu}^\top w = 0.0006$$

and the standard deviation of the portfolio's return is estimated as

$$\widehat{\sigma}_P = \sqrt{w^\top \widehat{\Sigma} w} = 0.00846.$$

The return on the portfolio has a  $t$ -distribution with this mean and standard deviation, and the same tail index as the multivariate  $t$ -distribution of the three stock returns. The scale parameter, using  $\widehat{\nu} = 5.81$ , is

$$\widehat{\lambda}_P = \sqrt{(\widehat{\nu} - 2)/\widehat{\nu}} \times 0.00846 = 0.00685.$$

Therefore,

$$\widehat{\text{VaR}}^t(0.05) = -S \times \{\widehat{\mu}_P + \widehat{\lambda}_P \widehat{q}_{0.05,t}(\widehat{\nu})\} = S \times 0.0128,$$

so, for example, with  $S = \$20,000$ ,  $\widehat{\text{VaR}}^t(0.05) = \$256$ .

The estimated ES using (19.8) and  $S = \$20,000$  is

$$\widehat{\text{ES}}^t(0.05) = S \times \left\{ -\widehat{\mu}_P + \widehat{\lambda}_P \left( \frac{f_{\widehat{\nu}}\{\widehat{q}_{0.05,t}(\widehat{\nu})\}}{\alpha} \left[ \frac{\widehat{\nu} + \{\widehat{q}_{0.05,t}(\widehat{\nu})\}^2}{\widehat{\nu} - 1} \right] \right) \right\} = \$363.$$

□

## 19.6 Estimation of VaR Assuming Polynomial Tails

There is an interesting compromise between using a totally nonparametric estimator of VaR as in Sect. 19.2.1 and a parametric estimator as in Sect. 19.2.2. The nonparametric estimator is feasible for large  $\alpha$ , but not for small  $\alpha$ . For example, if the sample had 1,000 returns, then reasonably accurate estimation of the 0.05-quantile is feasible, but not estimation of the 0.0005-quantile. Parametric estimation can estimate VaR for any value of  $\alpha$ , but is sensitive to misspecification of the tail when  $\alpha$  is small. Therefore, a methodology intermediary between totally nonparametric and parametric estimation is attractive.

The approach used in this section assumes that the return density has a polynomial left tail, or equivalently that the loss density has a polynomial right tail. Under this assumption, it is possible to use a nonparametric estimate of

$\text{VaR}(\alpha_0)$  for a *large* value of  $\alpha_0$  to obtain estimates of  $\text{VaR}(\alpha_1)$  for *small* values of  $\alpha_1$ . It is assumed here that  $\text{VaR}(\alpha_1)$  and  $\text{VaR}(\alpha_0)$  have the same horizon  $T$ .

Because the return density is assumed to have a polynomial left tail, the return density  $f$  satisfies

$$f(y) \sim Ay^{-(a+1)}, \text{ as } y \rightarrow -\infty, \quad (19.16)$$

where  $A > 0$  is a constant,  $a > 0$  is the tail index, and “ $\sim$ ” means that the ratio of the left-hand to right-hand sides converges to 1. Therefore,

$$P(R \leq y) \sim \int_{-\infty}^y f(u) du = \frac{A}{a} y^{-a}, \text{ as } y \rightarrow -\infty, \quad (19.17)$$

and if  $y_0 > 0$  and  $y_1 > 0$ , then

$$\frac{P(R < -y_0)}{P(R < -y_1)} \approx \left( \frac{y_0}{y_1} \right)^{-a}. \quad (19.18)$$

Now suppose that  $y_0 = \text{VaR}(\alpha_1)$  and  $y_1 = \text{VaR}(\alpha_0)$ , where  $0 < \alpha_1 < \alpha_0$  and, for simplicity and without loss of generality, we use  $S = 1$  in the following calculation. Then (19.18) becomes

$$\frac{\alpha_1}{\alpha_0} = \frac{P\{R < -\text{VaR}(\alpha_1)\}}{P\{R < -\text{VaR}(\alpha_0)\}} \approx \left( \frac{\text{VaR}(\alpha_1)}{\text{VaR}(\alpha_0)} \right)^{-a} \quad (19.19)$$

or

$$\frac{\text{VaR}(\alpha_1)}{\text{VaR}(\alpha_0)} \approx \left( \frac{\alpha_0}{\alpha_1} \right)^{1/a},$$

so, now dropping the subscript “1” of  $\alpha_1$  and writing the approximate equality as exact, we have

$$\text{VaR}(\alpha) = \text{VaR}(\alpha_0) \left( \frac{\alpha_0}{\alpha} \right)^{1/a}. \quad (19.20)$$

Equation (19.20) becomes an estimate of  $\text{VaR}(\alpha)$  when  $\text{VaR}(\alpha_0)$  is replaced by a nonparametric estimate and the tail index  $a$  is replaced by one of the estimates discussed soon in Sect. 19.6.1. Notice another advantage of (19.20), that it provides an estimate of  $\text{VaR}(\alpha)$  not just for a single value of  $\alpha$  but for all values. This is useful if one wants to compute and compare  $\text{VaR}(\alpha)$  for a variety of values of  $\alpha$ , as is illustrated in Example 19.7 ahead. The value of  $\alpha_0$  must be large enough that  $\text{VaR}(\alpha_0)$  can be accurately estimated, but  $\alpha$  can be any value less than  $\alpha_0$ .

A model combining parametric and nonparametric components is called *semiparametric*, so estimator (19.20) is semiparametric because the tail index is specified by a parameter, but otherwise the distribution is unspecified.

To find a formula for ES, we will assume further that for some  $c < 0$ , the returns density satisfies

$$f(y) = A|y|^{-(a+1)}, \quad y \leq c, \quad (19.21)$$

so that we have equality in (19.16) for  $y \leq c$ . Then, for any  $d \leq c$ ,

$$P(R \leq d) = \int_{-\infty}^d A|y|^{-(a+1)} dy = \frac{A}{a}|d|^{-a}, \quad (19.22)$$

and the conditional density of  $R$  given that  $R \leq d$  is

$$f(y|R \leq d) = \frac{Ay^{-(a+1)}}{P(R \leq d)} = a|d|^a|y|^{-(a+1)}. \quad (19.23)$$

It follows from (19.23) that for  $a > 1$ ,

$$E(|R| | R \leq d) = a|d|^a \int_{-\infty}^d |y|^{-a} dy = \frac{a}{a-1}|d|. \quad (19.24)$$

(For  $a \leq 1$ , this expectation is  $+\infty$ .) If we let  $d = -\text{VaR}(\alpha)$ , then we see that

$$\text{ES}(\alpha) = \frac{a}{a-1}\text{VaR}(\alpha) = \frac{1}{1-a^{-1}}\text{VaR}(\alpha), \text{ if } a > 1. \quad (19.25)$$

Formula (19.25) enables one to estimate  $\text{ES}(\alpha)$  using an estimate of  $\text{VaR}(\alpha)$  and an estimate of  $a$ .

### 19.6.1 Estimating the Tail Index

In this section, we estimate the tail index assuming a polynomial left tail. Two estimators will be introduced, the regression estimator and the Hill estimator.

#### Regression Estimator of the Tail Index

It follows from (19.17) that

$$\log\{P(R \leq -y)\} = \log(L) - a \log(y), \quad (19.26)$$

where  $L = A/a$ .

If  $R_{(1)}, \dots, R_{(n)}$  are the order statistics of the returns, then the number of observed returns less than or equal to  $R_{(k)}$  is  $k$ , so we estimate  $\log\{P(R \leq R_{(k)})\}$  to be  $\log(k/n)$ . Then, from (19.26), we have

$$\log(k/n) \approx \log(L) - a \log(-R_{(k)}) \quad (19.27)$$

or, rearranging (19.27),

$$\log(-R_{(k)}) \approx (1/a) \log(L) - (1/a) \log(k/n). \quad (19.28)$$

The approximation (19.28) is expected to be accurate only if  $-R_{(k)}$  is large, which means  $k$  is small, perhaps only 5%, 10%, or 20% of the sample size  $n$ . If we plot the points  $\{\log(k/n), \log(-R_{(k)})\} : k = 1, \dots, m\}$  for  $m$  equal to a small percentage of  $n$ , say 10%, then we should see these points fall on roughly a straight line. Moreover, if we fit the straight-line model (19.28) to these points by least squares, then the estimated slope, call it  $\hat{\beta}_1$ , estimates  $-1/a$ . Therefore, we will call  $-1/\hat{\beta}_1$  the *regression estimator of the tail index*.

## Hill Estimator

The Hill estimator of the left tail index  $a$  of the return density  $f$  uses all data less than a constant  $c$ , where  $c$  is sufficiently small such that

$$f(y) = A|y|^{-(a+1)} \quad (19.29)$$

is assumed to be true for  $y < c$ . The choice of  $c$  is crucial and will be discussed below. Let  $Y_{(1)}, \dots, Y_{(n)}$  be order statistics of the returns and  $n(c)$  be the number of  $Y_i$  less than or equal to  $c$ . By (19.23), the conditional density of  $Y_i$  given that  $Y_i \leq c$  is

$$a|c|^a|y|^{-(a+1)}. \quad (19.30)$$

Therefore, the likelihood for  $Y_{(1)}, \dots, Y_{(n(c))}$  is

$$L(a) = \left( \frac{a|c|^a}{|Y_1|^{a+1}} \right) \left( \frac{a|c|^a}{|Y_2|^{a+1}} \right) \cdots \left( \frac{a|c|^a}{|Y_{n(c)}|^{a+1}} \right),$$

and the log-likelihood is

$$\log\{L(a)\} = \sum_{i=1}^{n(c)} \{\log(a) + a \log(|c|) - (a+1) \log(|Y_{(i)}|)\}. \quad (19.31)$$

Differentiating the right-hand side of (19.31) with respect to  $a$  and setting the derivative equal to 0 gives the equation

$$\frac{n(c)}{a} = \sum_{i=1}^{n(c)} \log(Y_{(i)}/c).$$

Therefore, the MLE of  $a$ , which is called the *Hill estimator*, is

$$\hat{a}^{\text{Hill}}(c) = \frac{n(c)}{\sum_{i=1}^{n(c)} \log(Y_{(i)}/c)}. \quad (19.32)$$

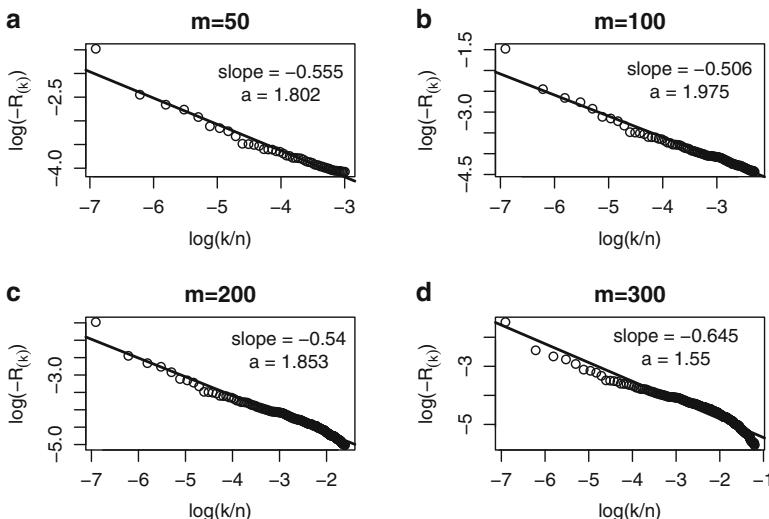
Note that  $Y_{(i)} \leq c < 0$ , so that  $Y_{(i)}/c$  is positive.

How should  $c$  be chosen? Usually  $c$  is equal to one of  $Y_1, \dots, Y_n$  so that  $c = Y_{(n(c))}$ , and therefore choosing  $c$  means choosing  $n(c)$ . The choice involves a bias-variance tradeoff. If  $n(c)$  is too large, then  $f(y) = A|y|^{-(a+1)}$  will not hold for all values of  $y \leq c$ , causing bias. If  $n(c)$  is too small, then there will be too few  $Y_i$  below  $c$  and  $\hat{a}^{\text{Hill}}(c)$  will be highly variable and unstable because it uses too few data. However, we can hope that there is a range of values of  $n(c)$  where  $\hat{a}^{\text{Hill}}(c)$  is reasonably constant because it is neither too biased nor too variable.

A *Hill plot* is a plot of  $\hat{a}^{\text{Hill}}(c)$  versus  $n(c)$  and is used to find this range of values of  $n(c)$ . In a Hill plot, one looks for a range of  $n(c)$  where the estimator is nearly constant and then chooses  $n(c)$  in this range.

*Example 19.7. Estimating the left tail index of the daily S&P 500 index returns*

This example uses the 1,000 daily S&P 500 index returns used in Examples 19.2 and 19.3. First, the regression estimator of the tail index was calculated. The values  $\{\log(k/n), \log(-R_{(k)}) : k = 1, \dots, m\}$  were plotted for  $m = 50, 100, 200$ , and 300 to find the largest value of  $m$  giving a roughly linear plot, of which  $m = 100$  was selected. The plotted points and the least-squares lines can be seen in Fig. 19.4. The slope of the line with  $m = 100$  was  $-0.506$ , so  $a$  was estimated to be  $1/0.506 = 1.975$ .

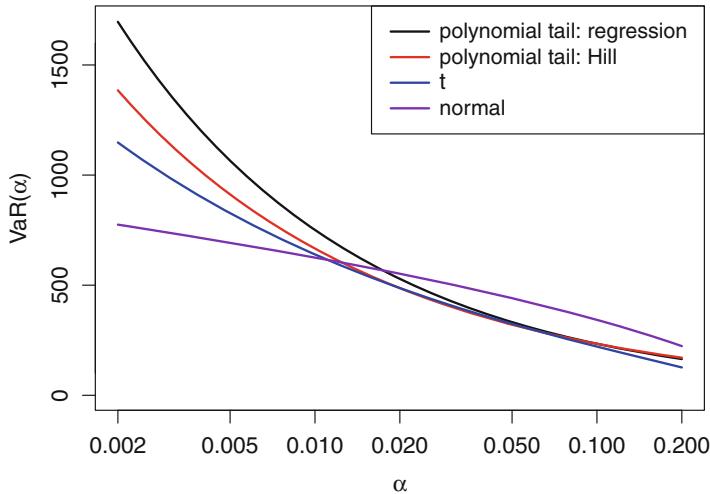


**Fig. 19.4.** Plots for estimating the left tail index of the S&P 500 returns by regression. The “slope” is the least-squares slope estimate and “ $a$ ” is  $-1/\text{slope}$ .

Suppose we have invested \$20,000 in an S&P 500 index fund. We will use  $\alpha_0 = 0.1$ .  $\text{VaR}(0.1, 24 \text{ h})$  is estimated to be  $-\$20,000$  times the 0.1-quantile of the 1,000 returns. The sample quantile is  $-0.0117$ , so  $\widehat{\text{VaR}}^{\text{np}}(0.1, 24 \text{ h}) = \$234$ . Using (19.20) and  $a = 1.975$  (i.e.,  $1/a = 0.506$ ), we have

$$\widehat{\text{VaR}}(\alpha) = 234 \left( \frac{0.1}{\alpha} \right)^{0.506}. \quad (19.33)$$

The black curve in Fig. 19.5 is a plot of  $\widehat{\text{VaR}}(\alpha)$  for  $0.0025 \leq \alpha \leq 0.25$  using (19.33) and the regression estimator of  $a$ . The red curve is the same plot but with the Hill estimator of  $a$ , which is 2.2—see below. The blue curve is  $\text{VaR}(\alpha)$  estimated assuming  $t$ -distributed returns as discussed in Sect. 19.2.2, and the purple curve is estimated assuming normally distributed returns. The



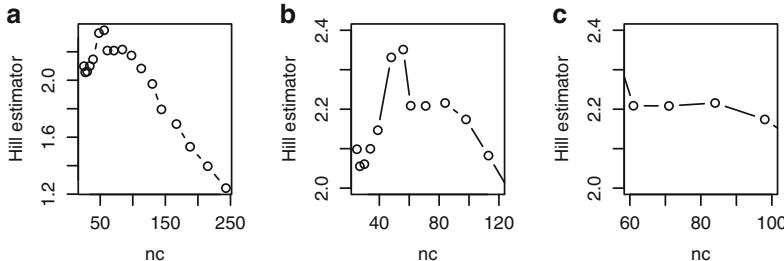
**Fig. 19.5.** Estimation of  $\text{VaR}(\alpha)$  using formula (19.33) and the regression estimator of the tail index (black line), using formula (19.33) and the Hill estimator of the tail index (red line), assuming  $t$ -distributed returns (blue line), and assuming normally distributed returns (purple line). Note the log-scale on the  $x$ -axis.

return distribution has much heavier tails than a normal distribution, and the latter curve is included only to show the effect of model misspecification. The parametric estimates based on the  $t$ -distribution are similar to the estimates assuming a polynomial tail except when  $\alpha$  is very small. The difference between the two estimates for small  $\alpha$  ( $\alpha < 0.01$ ) is to be expected because the polynomial tail with tail index 1.975 or 2.2 is heavier than the tail of the  $t$ -distribution with  $\nu = a = 2.984$ . The estimate based on the  $t$ -distribution is somewhat biased since it assumes a symmetric density and uses data in the right, as well as the left, tails to estimate the left tail; the problem with this is that the right tail is lighter than the left tail. If  $\alpha$  is in the range 0.01 to 0.2, then  $\widehat{\text{VaR}}(\alpha)$  is relatively insensitive to the choice of model, except for the poorly fitting normal model. This is a good reason for preferring  $\alpha \geq 0.01$ .

It follows from (19.25) using the regression estimate  $\hat{a} = 1.975$  that

$$\widehat{\text{ES}}(\alpha) = \frac{1.975}{0.975} \widehat{\text{VaR}}(\alpha) = 2.026 \widehat{\text{VaR}}(\alpha). \quad (19.34)$$

The Hill estimator of  $a$  was also implemented. Figure 19.6 contains Hill plots, that is, plots of the Hill estimate  $\hat{a}_{\text{Hill}}(c)$  versus  $n(c)$ . In panel (a),  $n(c)$  ranges from 25 to 250. There seems to be a region of stability when  $n(c)$  is between 25 and 120, which is shown in panel (b). In panel (b), we see a region of even greater stability when  $n(c)$  is between 60 and 100. Panel (c) zooms in on this region. We see in panel (c) that the Hill estimator is close to 2.2 when  $n(c)$  is between 60 and 100, and we will take 2.2 as the Hill estimate. Thus, the Hill estimate is similar to the regression estimate (1.975) of the tail index.



**Fig. 19.6.** Estimation of the tail index by applying a Hill plot to the daily returns on the S&P 500 index for 1,000 consecutive trading days ending on March 4, 2003. (a) Full range of  $n_c$ . (b) Zoom in to  $n_c$  between 25 and 120. (c) Zoom in further to  $n_c$  between 60 and 100.

The advantage of the regression estimate is that one can use the linearity of the plots of  $\{[\log(k/n), \log(-R_{(k)})] : k = 1, \dots, m\}$  for different  $m$  to guide the choice of  $m$ , which is analogous to  $n_c$ . A linear plot indicates a polynomial tail. In contrast, the Hill plot checks for the stability of the estimator and does not give a direct assessment whether or not the tail is polynomial.  $\square$

## 19.7 Pareto Distributions

The Pareto distribution with location parameter  $c > 0$  and shape parameter  $a > 0$  has density

$$f(y|a, c) = \begin{cases} ac^a y^{-(a+1)}, & y > c, \\ 0, & \text{otherwise.} \end{cases} \quad (19.35)$$

The expectation is  $ac/(a-1)$  if  $a > 1$ , and  $+\infty$  otherwise. The Pareto distribution has a polynomial tail and, in fact, a polynomial tail is often called a Pareto tail.

Equation (19.30) states that the loss, conditional on being above  $|c|$ , has a Pareto distribution. A property of the Pareto distribution that was exploited before [see (19.23)] is that if  $Y$  has a Pareto distribution with parameters  $a$  and  $c$  and if  $d > c$ , then the conditional distribution of  $Y$ , given that  $Y > d$ , is Pareto with parameters  $a$  and  $d$ .

## 19.8 Choosing the Horizon and Confidence Level

The choice of horizon and confidence coefficient are somewhat interdependent, and also depend on the eventual use of the VaR estimate. For shorter horizons such as one day, a large  $\alpha$  (small confidence coefficient =  $1 - \alpha$ ) would result in frequent losses exceeding VaR. For example,  $\alpha = 0.05$  would result in a loss

exceeding VaR approximately once per month since there are slightly more than 20 trading days in a month. Therefore, we might wish to use smaller values of  $\alpha$  with a shorter horizon.

One should be wary, however, of using extremely small values of  $\alpha$ , such as, values less than 0.01. When  $\alpha$  is very small, then VaR and, especially, ES are impossible to estimate accurately and are very sensitive to assumptions about the left tail of the return distribution. As we have seen, it is useful to create bootstrap confidence intervals to indicate the amount of precision in the VaR and ES estimates. It is also important to compare estimates based on different tail assumptions as in Fig. 19.5, for example, where the three estimates of VaR are increasingly dissimilar as  $\alpha$  decreases below 0.01.

There is, of course, no need to restrict attention to only one horizon or confidence coefficient. When VaR is estimated parametrically and i.i.d. normally distributed returns are assumed, then it is easy to reestimate VaR with different horizons. Suppose that  $\hat{\mu}_P^{1\text{day}}$  and  $\hat{\sigma}_P^{1\text{day}}$  are the estimated mean and standard deviation of the return for one day. Assuming only that returns are i.i.d., the mean and standard deviation for  $M$  days are

$$\hat{\mu}_P^{M\text{ days}} = M\hat{\mu}_P^{1\text{ day}} \quad (19.36)$$

and

$$\hat{\sigma}_P^{M\text{ days}} = \sqrt{M}\hat{\sigma}_P^{1\text{ day}}. \quad (19.37)$$

Therefore, if one assumes further that the returns are normally distributed, then the VaR for  $M$  days is

$$\text{VaR}_P^{M\text{ days}} = -S \times \left\{ M\hat{\mu}_P^{1\text{ day}} + \sqrt{M}\Phi^{-1}(\alpha)\hat{\sigma}_P^{1\text{ day}} \right\}, \quad (19.38)$$

where  $S$  is the size of the initial investment. The power of Eq. (19.38) is, for example, that it allows one to change from a daily to a weekly horizon without reestimating the mean and standard deviation with weekly instead of daily returns. Instead, one simply uses (19.38) with  $M = 5$ . The danger in using (19.38) is that it assumes normally distributed returns and no autocorrelation or GARCH effects (volatility clustering) of the daily returns. If there is positive autocorrelation, then (19.38) underestimates the  $M$ -day VaR. If there are GARCH effects, then (19.38) gives VaR based on the marginal distribution, but one should be using VaR based on the conditional distribution given the current information set.

If the returns are not normally distributed, then there is no simple analog to (19.38). For example, if the daily returns are i.i.d. but  $t$ -distributed then one cannot simply replace the normal quantile  $\Phi^{-1}(\alpha)$  in (19.38) by a  $t$ -quantile. The problem is that the sum of i.i.d.  $t$ -distributed random variables is not itself  $t$ -distributed. Therefore, if the daily returns are  $t$ -distributed then the sum of  $M$  daily returns is not  $t$ -distributed. However, for large values of  $M$  and i.i.d. returns, the sum of  $M$  independent returns will be close to normally distributed by the central limit theorem, so (19.38) could be used for large  $M$  even if the returns are not normally distributed.

## 19.9 VaR and Diversification

A serious problem with VaR is that it may *discourage* diversification. This problem was studied by Artzner, Delbaen, Eber, and Heath (1997, 1999), who ask the question, what properties can reasonably be required of a risk measure? They list four properties that any risk measure should have, and they call a risk measure *coherent* if it has all of them.

One property among the four that is very desirable is *subadditivity*. Let  $\mathfrak{R}(P)$  be a risk measure of a portfolio  $P$ , for example, VaR or ES. Then  $\mathfrak{R}$  is said to be subadditive, if for any two portfolios  $P_1$  and  $P_2$ ,  $\mathfrak{R}(P_1 + P_2) \leq \mathfrak{R}(P_1) + \mathfrak{R}(P_2)$ . Subadditivity says that the risk for the combination of two portfolios is at most the sum of their individual risks, which implies that diversification reduces risk or at least does not increase risk. For example, if a bank has two traders, then the risk of them combined is less than or equal to the sum of their individual risks if a subadditive risk measure is used. Subadditivity extends to more than two portfolios, so if  $\mathfrak{R}$  is subadditive, then for  $m$  portfolios,  $P_1, \dots, P_m$ ,

$$\mathfrak{R}(P_1 + \dots + P_m) \leq \mathfrak{R}(P_1) + \dots + \mathfrak{R}(P_m).$$

Suppose a firm has 100 traders and monitors the risk of each trader's portfolio. If the firm uses a subadditive risk measure, then it can be sure that the total risk of the 100 traders is at most the sum of the 100 individual risks. Whenever this sum is acceptable, there is no need to compute the risk measure for the entire firm. If the risk measure used by the firm is not subadditive, then there is no such guarantee.

Unfortunately, as the following example shows, VaR is *not* subadditive and therefore is incoherent. ES is subadditive, which is a strong reason for preferring ES to VaR.

*Example 19.8. An example where VaR is not subadditive*

This simple example has been designed to illustrate that VaR is not subadditive and can discourage diversification. A company is selling par \$1,000 bonds with a maturity of one year that pay a simple interest of 5% so that the bond pays \$50 at the end of one year if the company does not default. If the bank defaults, then the entire \$1,000 is lost. The probability of no default is 0.96. To make the loss distribution continuous, we will assume that the loss is  $N(-50, 1)$  with probability 0.96 and  $N(1000, 1)$  with probability 0.04. The main purpose of making the loss distribution continuous is to simplify calculations. However, the loss would be continuous, for example, if the portfolio contained both the bond and some stocks. Suppose that there is a second company selling bonds with exactly the same loss distribution and that the two companies are independent.

Consider two portfolios. Portfolio 1 buys two bonds from the first company and portfolio 2 buys one bond from each of the two companies. Both portfolios

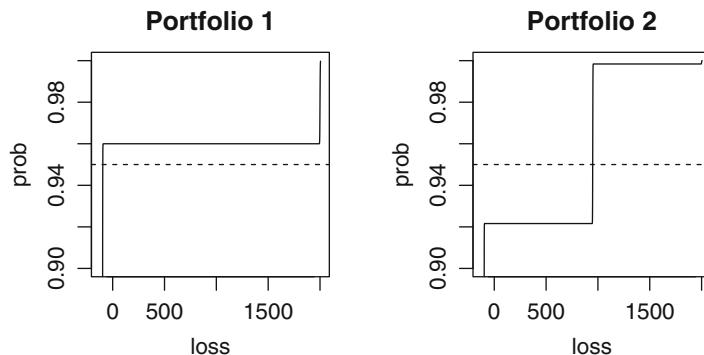
have the same expected loss, but the second is more diversified. Let  $\Phi(x|\mu, \sigma^2)$  be the normal CDF with mean  $\mu$  and variance  $\sigma^2$ . For portfolio 1, the loss CDF is

$$0.04\Phi(x|2000, 4) + 0.96\Phi(x|-100, 4),$$

while for portfolio 2, by independence of the two companies, the loss distribution CDF is

$$0.04^2\Phi(x|2000, 2) + 2(0.96)(0.04)\Phi(x|950, 2) + 0.96^2\Phi(x|-100, 2).$$

We should expect the second portfolio to seem less risky, but VaR(0.05) indicates the opposite. Specifically, VaR(0.05) is  $-95.38$  and  $949.53$  for portfolios 1 and 2, respectively. Notice that a negative VaR means a negative loss (positive revenue). Therefore, portfolio 1 is much less risky than portfolio 2, as measured by VaR(0.05). For each portfolio, VaR(0.05) is shown in Fig. 19.7 as the loss at which the CDF crosses the horizontal dashed line at 0.95.



**Fig. 19.7.** Example where VaR discourages diversification. Plots of the CDF of the loss distribution. VaR(0.05) is the loss at which the CDF crosses the horizontal dashed line at 0.95.

Notice as well that which portfolio has the highest value of VaR( $\alpha$ ) depends heavily on the values of  $\alpha$ . When  $\alpha$  is below the default probability, 0.04, portfolio 1 is more risky than portfolio 2.  $\square$

Although VaR is often considered the industry standard for risk management, Artzner, Delbaen, Eber, and Heath (1997) make an interesting observation. They note that when setting margin requirements, an exchange should use a subadditive risk measure so that the aggregate risk due to all customers is guaranteed to be smaller than the sum of the individual risks. Apparently, no organized exchanges use quantiles of loss distributions to set margin requirements. Thus, exchanges may be aware of the shortcomings of VaR, and VaR is not the standard for measuring risk within exchanges.

## 19.10 Bibliographic Notes

Risk management is an enormous subject and we have only touched upon a few aspects, focusing on statistical methods for estimating risk. We have not considered portfolios with bonds, foreign exchange positions, interest rate derivatives, or credit derivatives. We also have not considered risks other than market risk or how VaR and ES can be used for risk management. To cover risk management thoroughly requires at least a book-length treatment of that subject. Fortunately, excellent books exist, for example, Dowd (1998), Crouhy, Galai, and Mark (2001), Jorion (2001), and McNeil, Frey, and Embrechts (2005). The last has a strong emphasis on statistical techniques, and is recommended for further reading along the lines of this chapter. Generalized Pareto distributions were not covered here but are discussed in McNeil, Frey, and Embrechts.

Alexander (2001), Hull (2003), and Gourieroux and Jasiak (2001) have chapters on VaR and risk management. The semiparametric method of estimation based on the assumption of a polynomial tail and Eq. (19.20) are from Gourieroux and Jasiak (2001). Drees, de Haan, and Resnick (2000) and Resnick (2001) are good introductions to Hill plots.

## 19.11 R Lab

### 19.11.1 Univariate VaR and ES

In this section we will compare VaR and ES parametric (unconditional) estimates with those from using ARMA+GARCH (conditional) models. Consider the daily returns for Coca-Cola stock from January 2007 to November 2012.

```

1 CokePepsi = read.table("CokePepsi.csv", header=T)
2 price = CokePepsi[,1]
3 returns = diff(price)/lag(price)[-1]
4 ts.plot(returns)
```

First, assume that the returns are iid and follow a  $t$ -distribution. Run the following commands to get parameter estimates in R.

```

5 S = 4000
6 alpha = 0.05
7 library(MASS)
8 res = fitdistr(returns,'t')
9 mu = res$estimate['m']
10 lambda = res$estimate['s']
11 nu = res$estimate['df']
12 qt(alpha, df=nu)
13 dt(qt(alpha, df=nu), df=nu)
```

**Problem 1** What quantities are being computed in the last two lines above?

**Problem 2** For an investment of \$4,000, what are estimates of  $VaR^t(0.05)$  and  $ES^t(0.05)$ ?

Now, fit a ARMA(0,0)+GARCH(1,1) model to the returns and calculate one step forecasts.

```

14 library(fGarch) # for qstd() function
15 library(rugarch)
16 garch.t = ugarchspec(mean.model=list(armaOrder=c(0,0)),
17                       variance.model=list(garchOrder=c(1,1)),
18                       distribution.model = "std")
19 K0.garch.t = ugarchfit(data=returns, spec=garch.t)
20 show(K0.garch.t)
21 plot(K0.garch.t, which = 2)
22 pred = ugarchforecast(K0.garch.t, data=returns, n.ahead=1) ; pred
23 fitted(pred) ; sigma(pred)
24 nu = as.numeric(coef(K0.garch.t)[5])
25 q = qstd(alpha, mean = fitted(pred), sd = sigma(pred), nu = nu) ; q
26 sigma(pred)/sqrt( (nu)/(nu-2) )
27 qt(alpha, df=nu)
28 dt(qt(alpha, df=nu), df=nu)

```

**Problem 3** Carefully express the fitted ARMA(0,0)+GARCH(1,1) model in mathematical notation.

**Problem 4** What are the one-step ahead predictions of the conditional mean and conditional standard deviation?

**Problem 5** Again, for an investment of \$4,000, what are estimates of  $VaR^t(0.05)$  and  $ES^t(0.05)$  for the next day based on the fitted ARMA+GARCH model?

### 19.11.2 VaR Using a Multivariate-t Model

Run the following code to create a data set of returns on two stocks, DATGEN and DEC.

```

1 library(mnormt)
2 berndtInvest = read.csv("berndtInvest.csv")
3 Berndt = berndtInvest[,5:6]
4 names(Berndt)

```

**Problem 6** Fit a multivariate-t model to the returns in Berndt; see Sect. 7.13.3 for an example of fitting such a model. What are the estimates of the mean vector, tail index, and scale matrix? Include your R code and output with your work.

**Problem 7**

- (a) What is the distribution of the return on a \$100,000 portfolio that is 30 % invested in DATGEN and 70 % invested in DEC? Include your R code and output with your work.
- (b) Find  $VaR^t(0.05)$  and  $ES^t(0.05)$  for this portfolio.

**Problem 8** Use the model-free bootstrap to find a basic percentile bootstrap confidence interval for  $VaR(0.05)$  for the portfolio in Problem 7. Use a 90 % confidence coefficient for the confidence interval. Use 250 bootstrap resamples. This amount of resampling is not enough for a highly accurate confidence interval, but will give a reasonably good indication of the uncertainty in the estimate of  $VaR(0.05)$ , which is all that is really needed.

Also, plot kernel density estimates of the bootstrap distribution of the tail index and  $VaR^t(0.05)$ . Do the densities appear Gaussian or skewed? Use a normality test to check if they are Gaussian.

**Problem 9** This problem uses the variable DEC. Estimate the left tail index using the Hill estimator. Use a Hill plot to select  $n_c$ . What is your choice of  $n_c$ ?

## 19.12 Exercises

1. This exercise uses daily BMW returns in the `bmwRet` data set on the book's website. For this exercise, assume that the returns are i.i.d., even though there may be some autocorrelation and volatility clustering is likely. Suppose a portfolio holds \$1,000 in BMW stock (and nothing else).
  - (a) Compute nonparametric estimates of  $VaR(0.01, 24 \text{ h})$  and  $ES(0.01, 24 \text{ h})$ .
  - (b) Compute parametric estimates of  $VaR(0.01, 24 \text{ h})$  and  $ES(0.01, 24 \text{ h})$  assuming that the returns are normally distributed.
  - (c) Compute parametric estimates of  $VaR(0.01, 24 \text{ h})$  and  $ES(0.01, 24 \text{ h})$  assuming that the returns are  $t$ -distributed.
  - (d) Compare the estimates in (a), (b), and (c). Which do you feel are most realistic?
2. Assume that the loss distribution has a polynomial tail and an estimate of  $a$  is 3.1. If  $VaR(0.05) = \$252$ , what is  $VaR(0.005)$ ?
3. Find a source of stock price data on the Internet and obtain daily prices for a stock of your choice over the last 1,000 days.
  - (a) Assuming that the loss distribution is  $t$ , find the parametric estimate of  $VaR(0.025, 24 \text{ h})$ .
  - (b) Find the nonparametric estimate of  $VaR(0.025, 24 \text{ h})$ .
  - (c) Use a  $t$ -plot to decide if the normality assumption is reasonable.

- (d) Estimate the tail index assuming a polynomial tail and then use the estimate of  $\text{VaR}(0.025, 24 \text{ h})$  from part (a) to estimate  $\text{VaR}(0.0025, 24 \text{ h})$ .
4. This exercise uses daily Microsoft price data in the `msft.dat` data set on the book's website. Use the closing prices to compute daily returns. Assume that the returns are i.i.d., even though there may be some autocorrelation and volatility clustering is likely. Suppose a portfolio holds \$1,000 in Microsoft stock (and nothing else). Use the model-free bootstrap to find 95 % confidence intervals for parametric estimates of  $\text{VaR}(0.005, 24 \text{ h})$  and  $\text{ES}(0.005, 24 \text{ h})$  assuming that the returns are  $t$ -distributed.
5. Suppose the risk measure  $\mathfrak{R}$  is  $\text{VaR}(\alpha)$  for some  $\alpha$ . Let  $P_1$  and  $P_2$  be two portfolios whose returns have a joint normal distribution with means  $\mu_1$  and  $\mu_2$ , standard deviations  $\sigma_1$  and  $\sigma_2$ , and correlation  $\rho$ . Suppose the initial investments are  $S_1$  and  $S_2$ . Show that  $\mathfrak{R}(P_1 + P_2) \leq \mathfrak{R}(P_1) + \mathfrak{R}(P_2)$  under joint normality.<sup>2</sup>
6. This problem uses daily stock price data in the file `Stock_Bond.csv` on the book's website. In this exercise, use only the first 500 prices on each stock. The following R code reads the data and extracts the first 500 prices for five stocks. “AC” in the variables’ names means “adjusted closing” price.

```
dat = read.csv("Stock_Bond.csv", header = T)
prices = as.matrix(dat[1:500, c(3, 5, 7, 9, 11)])
```

- (a) What are the sample mean vector and sample covariance matrix of the 499 returns on these stocks?
- (b) How many shares of each stock should one buy to invest \$50 million in an equally weighted portfolio? Use the prices at the end of the series, e.g., `prices[,500]`.
- (c) What is the one-day  $\text{VaR}(0.1)$  for this equally weighted portfolio? Use a parametric VaR assuming normality.
- (d) What is the five-day  $\text{Var}(0.1)$  for this portfolio? Use a parametric VaR assuming normality. You can assume that the daily returns are uncorrelated.

## References

- Alexander, C. (2001) *Market Models: A Guide to Financial Data Analysis*, Wiley, Chichester.
- Artzner, P., Delbaen, F., Eber, J.-M., and Heath, D. (1997) Thinking coherently. *RISK*, **10**, 68–71.

---

<sup>2</sup> This result shows that VaR is subadditive on a set of portfolios whose returns have a joint normal distribution, as might be true for portfolios containing only stocks. However, portfolios containing derivatives or bonds with nonzero probabilities of default generally do not have normally distributed returns.

- Artzner, P., Delbaen, F., Eber, J.-M., and Heath, D. (1999) Coherent measures of risk. *Mathematical Finance*, **9**, 203–238.
- Crouhy, M., Galai, D., and Mark, R. (2001) *Risk Management*, McGraw-Hill, New York.
- Drees, H., de Haan, L., and Resnick, S. (2000) How to make a Hill plot, *Annals of Statistics*, **28**, 254–274.
- Dowd, K. (1998) *Beyond Value At Risk*, Wiley, Chichester.
- Gourieroux, C., and Jasiak, J. (2001) *Financial Econometrics*, Princeton University Press, Princeton, NJ.
- Hull, J. C. (2003) *Options, Futures, and Other Derivatives*, 5th ed., Prentice-Hall, Upper Saddle River, NJ.
- Jorion, P. (2001) *Value At Risk*, McGraw-Hill, New York.
- McNeil, A. J., Frey, R., and Embrechts, P. (2005) *Quantitative Risk Management*, Princeton University Press, Princeton, NJ.
- Resnick, S. I. (2001) *Modeling Data Networks*, School of Operations Research and Industrial Engineering, Cornell University, Technical Report #1345.

## Bayesian Data Analysis and MCMC

### 20.1 Introduction

Bayesian statistics is based up a philosophy different from that of other methods of statistical inference. In Bayesian statistics all unknowns, and in particular unknown parameters, are considered to be random variables and their probability distributions specify our beliefs about their likely values. Estimation, model selection, and uncertainty analysis are implemented by using Bayes's theorem to update our beliefs as new data are observed.

Non-Bayesians distinguish between two types of unknowns, parameters and latent variables. To a non-Bayesian, parameters are fixed quantities without probability distributions while latent variables are random unknowns with probability distributions. For example, to a non-Bayesian, the mean  $\mu$ , the moving average coefficients  $\theta_1, \dots, \theta_q$ , and the white noise variance  $\sigma_\epsilon^2$  of an MA( $q$ ) process are fixed parameters while the unobserved white noise process itself consists of latent variables. In contrast, to a Bayesian, the parameters and the white noise process are both unknown random quantities. Since this chapter takes a Bayesian perspective, there is no need to distinguish between the parameters and latent variables, since they can now be treated in the same way. Instead, we will let  $\boldsymbol{\theta}$  denote the vector of all unknowns and call it the “parameter vector.” In the context of time series forecasting, for example,  $\boldsymbol{\theta}$  could include both the unobserved white noise and the future values of the series being forecast.

A hallmark of Bayesian statistics is that one *must* start by specifying prior beliefs about the values of the parameters. Many statisticians have been reluctant to use Bayesian analysis since the need to start with prior beliefs seems too subjective. Consequently, there have been heated debates between Bayesian and non-Bayesian statisticians over the philosophical basis of statistics. However, much of mainstream statistical thought now supports the more pragmatic notion that we should use whatever works satisfactorily.

If one has little prior knowledge about a parameter, this lack of knowledge can be accommodated by using a so-called noninformative prior that provides very little information about the parameter relative to the information supplied by the data. In practice, Bayesian and non-Bayesian analyses of data usually arrive at similar conclusions when the Bayesian analysis uses only weak prior information so that knowledge of the parameters comes predominately from the data.

Moreover, in finance and many other areas of application, analysts often have substantial prior information and are willing to use it. In business and finance, there is no imperative to strive for objectivity as there is in scientific study. The need to specify a prior can be viewed as a strength, not a weakness, of the Bayesian view of statistics, since it forces the analyst to think carefully about how much and what kind of prior knowledge is available.

There has been a tremendous increase in the use of Bayesian statistics over the past few decades, because the Bayesian philosophy is becoming more widely accepted and because Bayesian estimators have become much easier to compute. In fact, Bayesian techniques often are the most satisfactory way to compute estimates for complex models. We have heard one researcher say “I am not a Bayesian but I use Bayesian methods” and undoubtedly others would agree.

For an overview of this chapter, assume we are interested in a parameter vector  $\boldsymbol{\theta}$ . A Bayesian analysis starts with a *prior* probability distribution for  $\boldsymbol{\theta}$  that summarizes all prior knowledge about  $\boldsymbol{\theta}$ ; “prior” means before the data are observed. The likelihood is defined in the same way in a non-Bayesian analysis, but in Bayesian statistics the likelihood has a different interpretation—the likelihood is the conditional distribution of the data given  $\boldsymbol{\theta}$ . The key step in Bayesian inference is the use of Bayes’s theorem to combine the prior knowledge about  $\boldsymbol{\theta}$  with the information in the data. This is done by computing the conditional distribution of  $\boldsymbol{\theta}$  given the data. This distribution is called the *posterior distribution*. In many, if not most, practical problems, it is impossible to compute the posterior analytically and numerical methods are used instead. A very successful class of numerical Bayesian methods is Markov chain Monte Carlo (MCMC), which simulates a Markov chain in such a way that the stationary distribution of the chain is the posterior distribution of the parameters. The simulated data from the chain are used to compute Bayes estimates and perform uncertainty analysis.

## 20.2 Bayes’s Theorem

Bayes’s theorem applies to both discrete events and to continuously distributed random variables. We will start with the case of discrete events. The continuous case is covered in Sect. 20.3.

Suppose that  $B_1, \dots, B_K$  is a partition of the sample space  $\mathcal{S}$  (the set of all possible outcomes). By “partition” is meant that  $B_i \cap B_j = \emptyset$  if  $i \neq j$  and  $B_1 \cup B_2 \cup \dots \cup B_K = \mathcal{S}$ . For any set  $A$ , we have that

$$A = (A \cap B_1) \cup \dots \cup (A \cap B_K),$$

and therefore, since  $B_1, \dots, B_K$  are disjoint,

$$P(A) = P(A \cap B_1) + \dots + P(A \cap B_K). \quad (20.1)$$

It follows from (20.1) and the definition of conditional probability that

$$P(B_j|A) = \frac{P(A|B_j)P(B_j)}{P(A)} = \frac{P(A|B_j)P(B_j)}{P(A|B_1)P(B_1) + \dots + P(A|B_K)P(B_K)}. \quad (20.2)$$

Equation (20.2) is called *Bayes's theorem*, and is also known as Bayes's rule or Bayes's law. Bayes's theorem is a simple, almost trivial, mathematical result, but its implications are profound. The importance of Bayes's theorem comes from its use for updating probabilities. Here is an example, one that is far too simple to be realistic but that illustrates how Bayes's theorem can be applied.

### *Example 20.1. Bayes's theorem in a discrete case*

Suppose that our prior knowledge about a stock indicates that the probability  $\theta$  that the price will rise on any given day is either 0.4 or 0.6. Based upon past data, say from similar stocks, we believe that  $\theta$  is equally likely to be 0.4 or 0.6. Thus, we have the *prior* probabilities

$$P(\theta = 0.4) = 0.5 \text{ and } P(\theta = 0.6) = 0.5.$$

We observe the stock for five consecutive days and its price rises on all five days. Assume that the price changes are independent across days, so that the probability that the price rises on each of five consecutive days is  $\theta^5$ . Given this information, we may suspect that  $\theta$  is 0.6, not 0.4. Therefore, the probability that  $\theta$  is 0.6, given five consecutive price increases, should be greater than the prior probability of 0.5, but how much greater? As notation, let  $A$  be the event that the prices rises on five consecutive days. Then, using Bayes's theorem, we have

$$\begin{aligned} P(\theta = 0.6|A) &= \frac{P(A|\theta = 0.6)P(\theta = 0.6)}{P(A|\theta = 0.6)P(\theta = 0.6) + P(A|\theta = 0.4)P(\theta = 0.4)} \\ &= \frac{(0.6)^5(0.5)}{(0.6)^5(0.5) + (0.4)^5(0.5)} \\ &= \frac{(0.6)^5}{(0.6)^5 + (0.4)^5} = \frac{0.07776}{0.07776 + 0.01024} = 0.8836. \end{aligned}$$

Thus, our probability that  $\theta$  is 0.6 was 0.5 before we observed five consecutive price increases but is 0.8836 after observing this event. Probabilities before observing data are called the *prior probabilities* and the probabilities conditional on observed data are called the *posterior probabilities*, so the prior probability that  $\theta$  equals 0.6 is 0.5 and the posterior probability is 0.8836.  $\square$

Bayes's theorem is extremely important because it tells us exactly how to update our beliefs in light of new information. Revising beliefs after receiving additional information is something that humans do poorly without the help of mathematics.<sup>1</sup> There is a human tendency to put either too little or too much emphasis on new information, but this problem can be mitigated by using Bayes's theorem for guidance.

### 20.3 Prior and Posterior Distributions

We now assume that  $\boldsymbol{\theta}$  is a continuously distributed parameter vector. The *prior distribution* with density  $\pi(\boldsymbol{\theta})$  expresses our beliefs about  $\boldsymbol{\theta}$  prior to observing data. The likelihood function is interpreted as the conditional density of the data  $\mathbf{Y}$  given  $\boldsymbol{\theta}$  and written as  $f(\mathbf{y}|\boldsymbol{\theta})$ . Using Eq. (A.19), the joint density of  $\boldsymbol{\theta}$  and  $\mathbf{Y}$  is the product of the prior and the likelihood; that is,

$$f(\mathbf{y}, \boldsymbol{\theta}) = \pi(\boldsymbol{\theta})f(\mathbf{y}|\boldsymbol{\theta}). \quad (20.3)$$

The marginal density of  $\mathbf{Y}$  is found by integrating  $\boldsymbol{\theta}$  out of the joint density so that

$$f(\mathbf{y}) = \int \pi(\boldsymbol{\theta})f(\mathbf{y}|\boldsymbol{\theta})d\boldsymbol{\theta}, \quad (20.4)$$

and the conditional density of  $\boldsymbol{\theta}$  given  $\mathbf{Y}$  is

$$\pi(\boldsymbol{\theta}|\mathbf{Y}) = \frac{\pi(\boldsymbol{\theta})f(\mathbf{Y}|\boldsymbol{\theta})}{f(\mathbf{Y})} = \frac{\pi(\boldsymbol{\theta})f(\mathbf{Y}|\boldsymbol{\theta})}{\int \pi(\boldsymbol{\theta})f(\mathbf{Y}|\boldsymbol{\theta})d\boldsymbol{\theta}}. \quad (20.5)$$

Equation (20.5) is another form of Bayes's theorem. The density on the left-hand side of (20.5) is called the *posterior density* and gives the probability distribution of  $\boldsymbol{\theta}$  after observing the data  $\mathbf{Y}$ .

Notice our use of  $\pi$  to denote densities of  $\boldsymbol{\theta}$ , so that  $\pi(\boldsymbol{\theta})$  is the prior density and  $\pi(\boldsymbol{\theta}|\mathbf{Y})$  is the posterior density. In contrast,  $f$  is used to denote densities of the data, so that  $f(\mathbf{y})$  is the marginal density of the data and  $f(\mathbf{y}|\boldsymbol{\theta})$  is the conditional density given  $\boldsymbol{\theta}$ .

Bayesian estimation and uncertainty analysis are based upon the posterior. The most common Bayes estimators are the mode and the mean of the

---

<sup>1</sup> See Edwards (1982).

posterior density. The mode is called the *maximum a posteriori estimator*, or *MAP estimator*. The mean of the posterior is

$$E(\boldsymbol{\theta}|\mathbf{Y}) = \int \boldsymbol{\theta} \pi(\boldsymbol{\theta}|\mathbf{Y}) d\boldsymbol{\theta} = \frac{\int \boldsymbol{\theta} \pi(\boldsymbol{\theta}) f(\mathbf{Y}|\boldsymbol{\theta}) d\boldsymbol{\theta}}{\int \pi(\boldsymbol{\theta}) f(\mathbf{Y}|\boldsymbol{\theta}) d\boldsymbol{\theta}} \quad (20.6)$$

and is also called the posterior expectation.

*Example 20.2. Updating the prior beliefs about the probability that a stock price will increase*

We continue Example 20.1 but change the simple, but unrealistic, prior that said that  $\theta$  was either 0.4 or 0.6 to a more plausible prior where  $\theta$  could be any value in the interval [0, 1], but with values near 1/2 more likely. Specifically, we use a Beta(2,2) prior so that

$$\pi(\theta) = 6\theta(1-\theta), \quad 0 < \theta < 1.$$

Let  $Y$  be the number of times the stock price increases on five consecutive days. Then  $Y$  is Binomial( $n, \theta$ ) and the density of  $Y$  is

$$f(y|\theta) = \binom{5}{y} \theta^y (1-\theta)^{5-y}, \quad y = 0, 1, \dots, 5.$$

Since we observed that  $Y = 5$ ,  $f(Y|\theta) = f(5|\theta) = \theta^5$  and the posterior density is

$$\pi(\theta|5) = \frac{6\theta(1-\theta)\theta^5}{\int 6\theta(1-\theta)\theta^5 d\theta} = 56\theta^6(1-\theta),$$

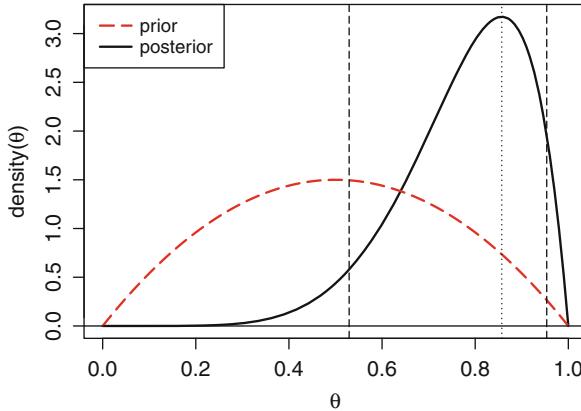
which is a Beta(7,2) density.

The prior and posterior densities are shown in Fig. 20.1. The posterior density is shifted towards the right compared to the prior because five consecutive days saw increased prices. The 0.05 lower and upper quantiles of the posterior distribution are 0.529 and 0.953, respectively, and are shown on the plot. Thus, there is 90% posterior probability that  $\theta$  is between 0.529 and 0.953. For this reason, the interval [0.529, 0.953] is called a *90% posterior interval* and provides us with the set of likely values of  $\theta$ . Posterior intervals are Bayesian analogs of confidence intervals and are discussed further in Sect. 20.6. Posterior intervals are also called *credible intervals*.

The posterior expectation is

$$\int_0^1 \theta \pi(\theta|5) d\theta = \int_0^1 56\theta^7(1-\theta) d\theta = \frac{56}{72} = 0.778. \quad (20.7)$$

The MAP estimate is  $6/7 = 0.857$  and its location is shown by a dotted vertical line in Fig. 20.1.



**Fig. 20.1.** Prior and posterior densities in Example 20.2. The dashed vertical lines are at the lower and upper 0.05-quantiles of the posterior, so they mark off a 90 % equal-tailed posterior interval. The dotted vertical line shows the location of the posterior mode at  $\theta = 6/7 = 0.857$ .

The posterior CDF is

$$\int_0^\theta \pi(x|5)dx = \int_0^\theta 56x^6(1-x)dx = 56 \left( \frac{\theta^7}{7} - \frac{\theta^8}{8} \right), \quad 0 \leq \theta \leq 1.$$

□

## 20.4 Conjugate Priors

In Example 20.2, the prior and the posterior were both beta distributions. This is an example of a family of conjugate priors. A family of distributions is called a *conjugate prior family* for a statistical model (or, equivalently, for the likelihood) if the posterior is in this family whenever the prior is in the family. Conjugate families are convenient because they make calculation of the posterior straightforward. All one needs to do is to update the parameters in the prior. To see how this is done, we will generalize Example 20.2.

*Example 20.3. Computing the posterior density of the probability that a stock price will increase—general case of a conjugate prior*

Suppose now that the prior for  $\theta$  is Beta( $\alpha, \beta$ ) so that the prior density is

$$\pi(\theta) = K_1 \theta^{\alpha-1} (1-\theta)^{\beta-1}, \quad (20.8)$$

where  $K_1$  is a constant. As we will see, knowing the exact value of  $K_1$  is not important, but from (A.14) we know that  $K_1 = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}$ . The parameters in

a prior density must be known, so here  $\alpha$  and  $\beta$  are chosen by the data analyst in accordance with the prior knowledge about the value of  $\theta$ . The choice of these parameters will be discussed later.

Suppose that the stock price is observed on  $n$  days and increases on  $Y$  days (and does not increase on  $n - Y$  days). Then the likelihood is

$$f(Y|\theta) = K_2 \theta^Y (1-\theta)^{n-Y}, \quad (20.9)$$

where  $K_2 = \binom{n}{Y}$  is another constant. The joint density of  $\theta$  and  $Y$  is

$$\pi(\theta)f(Y|\theta) = K_3 \theta^{\alpha+Y-1} (1-\theta)^{\beta+n-Y-1}, \quad (20.10)$$

where  $K_3 = K_1 K_2$ . Then, the posterior density is

$$\pi(\theta|Y) = \frac{\pi(\theta)f(Y|\theta)}{\int_0^1 \pi(\theta)f(Y|\theta)d\theta} = K_4 \theta^{\alpha+Y-1} (1-\theta)^{\beta+n-Y-1}. \quad (20.11)$$

where

$$K_4 = \frac{1}{\int_0^1 \theta^{\alpha+Y-1} (1-\theta)^{\beta+n-Y-1} d\theta}. \quad (20.12)$$

The posterior distribution is  $\text{Beta}(\alpha + Y, \beta + n - Y)$ .

We did not need to keep track of the values of  $K_1, \dots, K_4$ . Since (20.11) is proportional to a  $\text{Beta}(\alpha + Y, \beta + n - Y)$  density and since all densities integrate to 1, we can deduce that the constant of proportionality is 1 and the posterior is  $\text{Beta}(\alpha + Y, \beta + n - Y)$ . It follows from (A.14) that

$$K_4 = \frac{\Gamma(\alpha + \beta + n)}{\Gamma(\alpha + Y)\Gamma(\beta + n - Y)}.$$

It is worth noticing how easily the posterior can be found. One simply updates the prior parameters  $\alpha$  and  $\beta$  to  $\alpha + Y$  and  $\beta + n - Y$ , respectively.

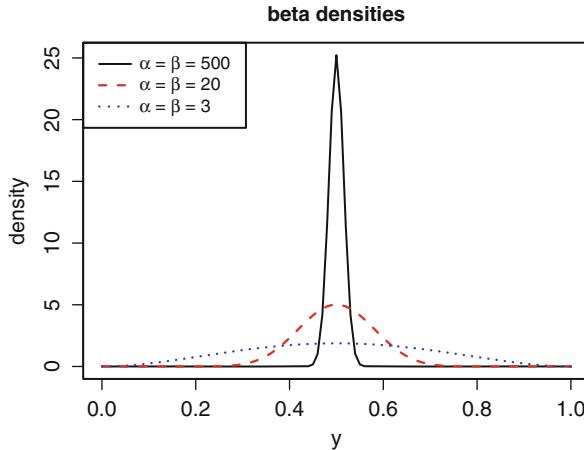
Using the results in Appendix A.9.7 about the mean and variance of beta distributions, the mean of the posterior is

$$E(\theta|Y) = \frac{\alpha + Y}{\alpha + \beta + n} \quad (20.13)$$

and the posterior variance is

$$\begin{aligned} \text{var}(\theta|Y) &= \frac{(\alpha + Y)(\beta + n - Y)}{(\alpha + \beta + n)^2(\alpha + \beta + n + 1)} \\ &= \frac{E(\theta|Y)\{1 - E(\theta|Y)\}}{(\alpha + \beta + n + 1)}. \end{aligned} \quad (20.14)$$

For values of  $\alpha$  and  $\beta$  that are small relative to  $Y$  and  $n$ ,  $E(\theta|Y)$  is approximately equal to the MLE, which is  $Y/n$ . If we had little prior knowledge



**Fig. 20.2.** Examples of beta probability densities with  $\alpha = \beta$ .

of  $\theta$ , we might take both  $\alpha$  and  $\beta$  close to 0. However, since  $\theta$  is the probability of a positive daily return on a stock, we might be reasonably certain that  $\theta$  is close to 1/2. In that case, choosing  $\alpha = \beta$  and both fairly large (so that the prior precision is large) makes sense. One could plot several beta densities with  $\alpha = \beta$  and decide which seem reasonable choices of the prior. For example, Fig. 20.2 contains plots of beta densities with  $\alpha = \beta = 3$ , 20, and 500. When 500 is the common value of  $\alpha$  and  $\beta$ , then the prior is quite concentrated about 1/2. This prior could be used by someone who is rather sure that  $\theta$  is close to 1/2. Someone with less certainty might instead prefer to use  $\alpha = \beta = 20$ , which has almost all of the prior probability between 0.3 and 0.6. The choice  $\alpha = \beta = 3$  leads to a very diffuse prior and would be chosen if one had very little prior knowledge of  $\theta$  and wanted to “let the data speak for themselves.”

The posterior mean in (20.13) has an interesting interpretation. Suppose that we had prior information from a previous sample of size  $\alpha + \beta$  and in that sample the stock price increased  $\alpha$  times. If we combined the two samples, then the total sample size would be  $\alpha + \beta + n$ , the number of days with a price increase would be  $\alpha + Y$ , and the MLE of  $\theta$  would be  $(\alpha + Y)/(\alpha + \beta + n)$ , the posterior mean given by (20.13). We can think of the prior as having as much information as would be given by a prior sample of size  $\alpha + \beta$  and  $\alpha/(\alpha + \beta)$  can be interpreted as the MLE of  $\theta$  from that sample. Therefore, the three priors in Fig. 20.2 can be viewed as having as much information as samples of sizes 6, 40, and 1000. For a fixed value of  $E(\theta|Y)$ , we see from (20.14) that the posterior variance of  $\theta$  becomes smaller as  $\alpha$ ,  $\beta$ , or  $n$  increases; this makes sense since  $n$  is the sample size and  $\alpha + \beta$  quantifies the amount of information in the prior.

Since it is not necessary to keep track of constants, we could have omitted them from the previous calculations and, for example, written (20.8) as

$$\pi(\theta) \propto \theta^{\alpha-1} (1-\theta)^{\beta-1}. \quad (20.15)$$

In the following examples, we will omit constants in this manner.  $\square$

*Example 20.4. Posterior distribution when estimating the mean of a normal population with known variance*

Suppose  $Y_1, \dots, Y_n$  are i.i.d.  $N(\mu, \sigma^2)$  and  $\sigma^2$  is known. The unrealistic assumption that  $\sigma^2$  is known is made so that we can start simple and will be removed later.

The conjugate prior for  $\mu$  is the family of normal distributions. To show this, assume that the prior on  $\mu$  is  $N(\mu_0, \sigma_0^2)$  for known values of  $\mu_0$  and  $\sigma_0^2$ . We learned in Example 20.3 that it is not necessary to keep track of quantities that do not depend on the unknown parameters (but could depend on the data or known parameters), so we will keep track only of terms that depend on  $\mu$ .

Simple algebra shows that the likelihood is

$$\begin{aligned} f(Y_1, \dots, Y_n | \mu) &= \prod_{i=1}^n \left[ \frac{1}{\sqrt{2\pi}\sigma} \exp \left\{ -\frac{1}{2\sigma^2} (Y_i - \mu)^2 \right\} \right] \\ &\propto \exp \left\{ -\frac{1}{2\sigma^2} (-2n\bar{Y}\mu + n\mu^2) \right\}. \end{aligned} \quad (20.16)$$

The prior density is

$$\pi(\mu) = \frac{1}{\sqrt{2\pi}\sigma_0} \exp \left\{ -\frac{1}{2\sigma_0^2} (\mu - \mu_0)^2 \right\} \propto \exp \left\{ -\frac{1}{2\sigma_0^2} (-2\mu\mu_0 + \mu^2) \right\}. \quad (20.17)$$

A *precision* is the reciprocal of a variance, and we let  $\tau = 1/\sigma^2$  denote the population precision. Multiplying (20.16) and (20.17), we can see that the posterior density is

$$\begin{aligned} \pi(\mu | Y_1, \dots, Y_n) &\propto \exp \left\{ \left( \frac{n\bar{Y}}{\sigma^2} + \frac{\mu_0}{\sigma_0^2} \right) \mu - \left( \frac{n}{2\sigma^2} + \frac{1}{2\sigma_0^2} \right) \mu^2 \right\} \\ &= \exp \left\{ (\tau_{\bar{Y}}\bar{Y} + \tau_0\mu_0)\mu - \frac{1}{2}(\tau_{\bar{Y}} + \tau_0)\mu^2 \right\}, \end{aligned} \quad (20.18)$$

where  $\tau_{\bar{Y}} = n\tau = n/\sigma^2$  and  $\tau_0 = 1/\sigma_0^2$ , so that  $\tau_{\bar{Y}}$  is the precision of  $\bar{Y}$  and  $\tau_0$  is the precision of the prior distribution.

One can see that  $\log\{\pi(\mu | Y_1, \dots, Y_n)\}$  is a quadratic function of  $\mu$ , so  $\pi(\mu | Y_1, \dots, Y_n)$  is a normal density. Therefore, to find the posterior distribution we need only compute the posterior mean and variance. The posterior mean is the value of  $\mu$  that maximizes the posterior density, that is, the posterior mode, so to calculate the posterior mean, we solve

$$0 = \frac{\partial}{\partial \mu} \log\{\pi(\mu|Y_1, \dots, Y_n)\} \quad (20.19)$$

and find that the mean is

$$E(\mu|Y_1, \dots, Y_n) = \frac{\tau_{\bar{Y}}\bar{Y} + \tau_0\mu_0}{\tau_{\bar{Y}} + \tau_0} = \frac{\frac{n\bar{Y}}{\sigma^2} + \frac{\mu_0}{\sigma_0^2}}{\frac{n}{\sigma^2} + \frac{1}{\sigma_0^2}}. \quad (20.20)$$

We can see from (A.10) that the precision of a normal density  $f(y)$  is  $-2$  times the coefficient of  $y^2$  in  $\log\{f(y)\}$ . Therefore, the posterior precision is  $-2$  times the coefficient of  $\mu^2$  in (20.18). Consequently, the posterior precision is  $\tau_{\bar{Y}} + \tau_0 = n/\sigma^2 + 1/\sigma_0^2$ , and the posterior variance is

$$\text{Var}(\mu|Y_1, \dots, Y_n) = \frac{1}{\frac{n}{\sigma^2} + \frac{1}{\sigma_0^2}}. \quad (20.21)$$

In summary, the posterior distribution is

$$N\left(\frac{\frac{n\bar{Y}}{\sigma^2} + \frac{\mu_0}{\sigma_0^2}}{\frac{n}{\sigma^2} + \frac{1}{\sigma_0^2}}, \frac{1}{\frac{n}{\sigma^2} + \frac{1}{\sigma_0^2}}\right) = N\left(\frac{\tau_{\bar{Y}}\bar{Y} + \tau_0\mu_0}{\tau_{\bar{Y}} + \tau_0}, \frac{1}{\tau_{\bar{Y}} + \tau_0}\right). \quad (20.22)$$

We can see that the posterior precision  $(\tau_{\bar{Y}} + \tau_0)$  is the sum of the precision of  $\bar{Y}$  and the precision of the prior; this makes sense since the posterior combines the information in the data with the information in the prior.

Notice that as  $n \rightarrow \infty$ , the posterior precision  $\tau_{\bar{Y}}$  converges to  $\infty$  and the posterior distribution is approximately

$$N(\bar{Y}, \sigma^2/n). \quad (20.23)$$

What this result tells us is that as the amount of data increases, the effect of the prior becomes negligible. The posterior density also converges to (20.23) as  $\sigma_0 \rightarrow \infty$  with  $n$  fixed, that is, as the prior becomes negligible because the prior precision decreases to zero.

A common Bayes estimator is the posterior mean given by the right-hand side of (20.20). Many statisticians are neither committed Bayesians nor committed non-Bayesians and like to look at estimators from both perspectives. A non-Bayesian would analyze the posterior mean by examining its bias, variance, and mean-squared error. We will see that, in general, the Bayes estimator is biased but is less variable than  $\bar{Y}$ , and the tradeoff between bias and variance is controlled by the choice of the prior.

To simplify notation, let  $\hat{\mu}$  denote the posterior mean. Then

$$\hat{\mu} = \delta\bar{Y} + (1 - \delta)\mu_0, \quad (20.24)$$

where  $\delta = \tau_{\bar{Y}}/(\tau_{\bar{Y}} + \tau_0)$ , and  $E(\hat{\mu}|\mu) = \delta\mu + (1 - \delta)\mu_0$ , so the bias of  $\hat{\mu}$  is  $\{E(\hat{\mu}|\mu) - \mu\} = (\delta - 1)(\mu - \mu_0)$  and  $\hat{\mu}$  is biased unless  $\delta = 1$  or  $\mu_0 = \mu$ .

We will have  $\delta = 1$  only in the limit as the prior precision  $\tau_0$  converges to 0 and  $\mu_0 = \mu$  means that the prior mean is exactly equal to the true parameter, but of course this beneficial situation cannot be arranged since  $\mu$  is not known.

The variance of  $\hat{\mu}$  is

$$\text{Var}(\hat{\mu}|\mu) = \frac{\delta^2\sigma^2}{n},$$

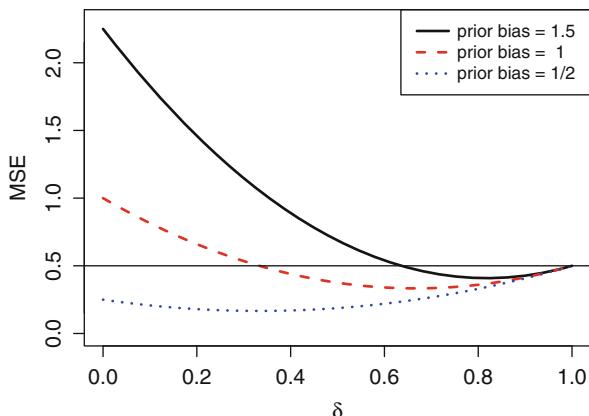
which is less than  $\text{Var}(\bar{Y}) = \sigma^2/n$ , except in the extreme case where  $\delta = 1$ . We see that smaller values of  $\delta$  lead to more bias but smaller variance. The best bias-variance tradeoff minimizes the mean square error of  $\hat{\mu}$ , which is

$$\text{MSE}(\hat{\mu}) = \text{BIAS}^2(\hat{\mu}) + \text{Var}(\hat{\mu}) = (\delta - 1)^2(\mu - \mu_0)^2 + \frac{\delta^2\sigma^2}{n}. \quad (20.25)$$

It is best, of course, to have  $\mu_0 = \mu$ , but this is not possible since  $\mu$  is unknown. What is known is  $\delta = \tau_{\bar{Y}}/(\tau_{\bar{Y}} + \tau_0)$  and  $\delta$  can be controlled by the choice of  $\tau_0$ .

Figure 20.3 shows the MSE as a function of  $\delta \in (0, 1)$  for three values of  $\mu - \mu_0$ , which is called the “prior bias” since it is the difference between the true value of the parameter and the prior mean. In this figure  $\sigma^2/n = 1/2$ . For each of the two larger values of the prior bias, there is a range of values of  $\delta$  where the Bayes estimator has a smaller MSE than  $\bar{Y}$ , but if  $\delta$  is below this range, then the Bayes estimator has a larger MSE than  $\bar{Y}$  and the range of “good”  $\delta$ -values decreases as the prior bias increases. If the prior bias is large and  $\delta$  is too small, then the MSE of the Bayes estimator can be quite large since it converges to the squared prior bias as  $\delta \rightarrow 0$ ; see (20.25) or Fig. 20.3. This result shows the need either to have a good prior guess of  $\mu$  or to keep the prior precision small so that  $\delta$  is large. However, when  $\delta$  is large, then the Bayes estimator cannot improve much over  $\bar{Y}$  and, in fact, converges to  $\bar{Y}$  as  $\delta \rightarrow 1$ .

In summary, it can be challenging to choose a prior that offers a substantial improvement over  $\bar{Y}$ . One way to do this is to combine several related



**Fig. 20.3.** *MSE versus  $\delta$  for three values of “prior bias”  $= \mu - \mu_0$  when  $\sigma^2/n = 1/2$ . The horizontal line represents the MSE of the maximum likelihood estimator ( $\bar{Y}$ ).*

estimation problems using a hierarchical prior; see Sect. 20.8. When it is not possible to combine related problems and there is no other way to get information about  $\mu$ , then the prudent data analyst will forgo the attempt to improve upon the MLE and instead will choose a small value for the prior precision  $\tau_0$ .  $\square$

*Example 20.5. Posterior distribution when estimating a normal precision*

Now suppose that  $Y_1, \dots, Y_n$  are i.i.d. with a known mean  $\mu$  and an unknown variance  $\sigma^2$  and precision  $\tau = 1/\sigma^2$ . We will show that the conjugate priors for  $\tau$  are the gamma distributions and we will find the posterior distribution of  $\tau$ . Define  $s^2 = n^{-1} \sum_{i=1}^n (Y_i - \mu)^2$ , which is the MLE of  $\sigma^2$ .

Simple algebra shows that the likelihood is

$$f(Y_1, \dots, Y_n | \tau) \propto \exp\left(-\frac{1}{2}n\tau s^2\right) \tau^{n/2}. \quad (20.26)$$

Let the prior distribution be the gamma distribution with shape parameter  $\alpha$  and scale parameter  $b$  which has density

$$\pi(\tau) = \frac{\tau^{\alpha-1}}{\Gamma(\alpha)b^\alpha} \exp(-\tau/b) \propto \tau^{\alpha-1} \exp(-\tau/b). \quad (20.27)$$

Multiplying (20.26) and (20.27), we see that the posterior density for  $\tau$  is

$$\pi(\tau | Y_1, \dots, Y_n) \propto \tau^{n/2+\alpha-1} \exp\{-ns^2/2 + b^{-1}\tau\}, \quad (20.28)$$

which shows that the posterior distribution is gamma with shape parameter  $n/2 + \alpha$  and scale parameter  $(ns^2/2 + b^{-1})^{-1}$ ; that is,

$$\pi(\tau | Y_1, \dots, Y_n) = \text{Gamma}\left\{n/2 + \alpha, (ns^2/2 + b^{-1})^{-1}\right\}. \quad (20.29)$$

We have shown that gamma distributions are conjugate for a normal precision parameter.

The expected value of a gamma distribution is the product of the shape and scale parameters, so the posterior mean of  $\tau$  is

$$E(\tau | Y_1, \dots, Y_n) = \frac{\frac{n}{2} + \alpha}{\frac{ns^2}{2} + b^{-1}}.$$

Notice that  $E(\tau | Y_1, \dots, Y_n)$  converges to  $s^{-2}$  as  $n \rightarrow \infty$ , which is not surprising since the MLE of  $\sigma^2$  is  $s^2$ , so that the MLE of  $\tau$  is  $s^{-2}$ .  $\square$

## 20.5 Central Limit Theorem for the Posterior

For large sample sizes, the posterior distribution obeys a central limit theorem that can be roughly stated as follows:

**Result 20.6.** *Under suitable assumptions and for large enough sample sizes, the posterior distribution of  $\theta$  is approximately normal with mean equal to the true value of  $\theta$  and with variance equal to the inverse of the Fisher information matrix.*

This result is also known as the *Bernstein–von Mises Theorem*. See Sect. 20.14 for references to a precise statement of the theorem.

This theorem is an important result for several reasons. First, a comparison with Result 5.1 shows that the Bayes estimator and the MLE have the same large-sample distributions. In particular, we see that for large sample sizes, the effect of the prior becomes negligible, because the asymptotic distribution does not depend on the prior. Moreover, the theorem shows a connection between confidence and posterior intervals that is discussed in the next section.

One of the assumptions of this theorem is that the prior remains fixed as the sample size increases, so that eventually nearly all of the information comes from the data. The more informative the prior, the larger the sample size needed for the posterior distribution to approach its asymptotic limit.

## 20.6 Posterior Intervals

Bayesian posterior intervals were mentioned in Example 20.2 and will now be discussed in more depth.

Posterior intervals have a different probabilistic interpretation than confidence intervals. The theory of confidence intervals views the parameter as fixed and the interval as random because it is based on a random sample. Thus, when we say “the probability that the confidence interval will include the true parameter is . . .,” it is the probability distribution of the interval, not the parameter, that is being considered. Moreover, the probability expresses the likelihood *before* the data are collected about what will happen after the data are collected. For example, if we use 95 % confidence, then the probability is 0.95 that we will obtain a sample whose interval covers the parameter. After the data have been collected and the interval is known, a non-Bayesian will say that either the interval covers the parameter or it does not, so the probability that the interval covers the parameter is either 1 or 0, though, of course, we do not know which value is the actual probability.

In the Bayesian theory of posterior intervals, the opposite is true. The sample is considered fixed since we use posterior probabilities, that is, probabilities conditional on the data. Therefore, the posterior interval is considered a fixed quantity. But in Bayesian statistics, parameters are treated as random. Therefore, when a Bayesian says “the probability that the posterior interval will include the true parameter is . . .,” the probability distribution being considered is the posterior distribution of the parameter. The random quantity is the parameter, the interval is fixed, and the probability is after the data have been collected.

Despite these substantial philosophical differences between confidence and posterior intervals, in many examples where both a confidence interval and a posterior interval have been constructed, one finds that they are nearly equal. This is especially common when the prior is relatively noninformative compared to the data, for example, in Example 20.3 if  $\alpha + \beta$  is much smaller than  $n$ .

There are solid theoretical reasons based on central limit theorems why confidence and posterior intervals are nearly equal for large sample sizes. By Result 20.6 (the central limit theorem for the posterior), a large-sample posterior interval for the  $i$ th component of  $\boldsymbol{\theta}$  is

$$E(\theta_i | \mathbf{Y}) \pm z_{\alpha/2} \sqrt{\text{var}(\theta_i | \mathbf{Y})}. \quad (20.30)$$

By Results 5.1 and 7.6 (the univariate and multivariate central limit theorems for the MLE), the large-sample confidence interval (5.20) based on the MLE and the large-sample posterior interval (20.30) will approach each other as the sample size increases. Therefore, practically-minded non-Bayesian data analysts are often happy to use a posterior interval and interpret it as a large-sample approximation to a confidence interval. Except in simple problems, all confidence intervals are based on large-sample approximations. This is true for confidence intervals that use profile likelihood, the central limit theorem for the MLE and Fisher information, or the bootstrap, in other words, for all of the major methods for constructing confidence intervals.

There are two major types of posterior intervals, highest probability and equal-tails. Let  $\psi = \psi(\boldsymbol{\theta})$  be a scalar function of the parameter vector  $\boldsymbol{\theta}$  and let  $\pi(\psi | \mathbf{Y})$  be the posterior density of  $\psi$ . A highest-probability interval is of the form  $\{\psi : \pi(\psi | \mathbf{Y}) > k\}$  for some constant  $k$ . As  $k$  increases from 0 to  $\infty$ , the posterior probability of this interval decreases from 1 to 0, and  $k$  is chosen so that the probability is  $1 - \alpha$ . If  $\pi(\psi | \mathbf{Y})$  has multiple modes, then the set  $\{\psi : \pi(\psi | \mathbf{Y}) > k\}$  might not be an interval and in that case it should be called a posterior set or posterior region rather than a posterior interval. In any case, this region has the interpretation of being the smallest set with  $1 - \alpha$  posterior probability. When the highest-posterior region is an interval, it can be found by computing all intervals that range from the  $\alpha_1$ -lower quantile of  $\pi(\psi | \mathbf{Y})$  to the  $\alpha_2$ -upper quantile of  $\pi(\psi | \mathbf{Y})$ , where  $\alpha_1 + \alpha_2 = \alpha$ , and the shortest of these intervals.

The equal-tails posterior interval has lower and upper limits equal to the lower and upper  $\alpha/2$ -quantiles of  $\pi(\psi | \mathbf{Y})$ . The two types of intervals coincide when  $\pi(\psi | \mathbf{Y})$  is symmetric and unimodal, which will be at least approximately true for large samples by the central limit theorem for the posterior.

Posterior intervals of either type are easy to compute when using Monte Carlo methods; see Sect. 20.7.3.

*Example 20.7. Posterior interval for a normal mean when the variance is known*

This example continues Example 20.4. By (20.20) and (20.21), a  $(1 - \alpha)$  100% posterior interval for  $\mu$  is

$$\frac{\tau_{\bar{Y}}\bar{Y} + \tau_0\mu_0}{\tau_{\bar{Y}} + \tau_0} \pm z_{\alpha/2} \sqrt{\frac{1}{\frac{n}{\sigma^2} + \frac{1}{\sigma_0^2}}}, \quad (20.31)$$

where  $z_{\alpha/2}$  is the  $\alpha/2$ -upper quantile of the standard normal distribution.

If either  $n \rightarrow \infty$  or  $\sigma_0 \rightarrow \infty$ , then the information in the prior becomes negligible relative to the information in the data because  $\tau_{\bar{Y}}/\tau_0 \rightarrow \infty$ , and the posterior interval converges to

$$\bar{Y} \pm z_{\alpha/2} \frac{\sigma}{\sqrt{n}},$$

which is the usual non-Bayesian confidence interval.  $\square$

## 20.7 Markov Chain Monte Carlo

Although the Bayesian calculations in the simple examples of the last few sections were straightforward, this is generally not true for problems of practical interest. Frequently, the integral in the denominator of posterior density (20.5) is impossible to calculate analytically. The same is true of the integral in the numerator of the posterior mean given by (20.6). Because of computational difficulties, until approximately 1990 Bayesian data analysis was much less widely used than now. Fortunately, Monte Carlo simulation methods for approximating posterior densities and expectations have been developed. They have been a tremendous advance and not only have they made Bayesian methods practical, but also they have led to the solution of applied problems that heretofore could not be tackled.

The most widely applicable Monte Carlo method for Bayesian analysis simulates a Markov chain whose stationary distribution is the posterior. The sample from this chain is used for Bayesian inference. This technique is called *Markov chain Monte Carlo*, or *MCMC*. The **BUGS** language implements MCMC. There are three widely used versions of **BUGS**, **OpenBUGS**, **WinBUGS**, and **JAGS**. Most **BUGS** programs will run on all three versions, but there are exceptions. **JAGS** is in one way the most versatile of the three versions since it is the only one that will run under MacOS.<sup>2</sup>

This section is an introduction to MCMC and **BUGS**. First, we discuss Gibbs sampling, the simplest type of MCMC. Gibbs sampling works well when it is applicable, but it is applicable only to limited set of problems. Next, the Metropolis–Hastings algorithm is discussed. Metropolis–Hastings is applicable to nearly every type of Bayesian analysis. **BUGS** is a sophisticated program that is able to select an MCMC algorithm that is suitable for a particular model.

---

<sup>2</sup> **OpenBUGS** will run on a Mac under WINE.

### 20.7.1 Gibbs Sampling

Gibbs sampling is the simplest MCMC method. Suppose that the parameter vector  $\boldsymbol{\theta}$  can be partitioned into  $M$  subvectors so that

$$\boldsymbol{\theta} = \begin{pmatrix} \boldsymbol{\theta}_1 \\ \vdots \\ \boldsymbol{\theta}_M \end{pmatrix}.$$

Let  $[\boldsymbol{\theta}_j | \mathbf{Y}, \boldsymbol{\theta}_k, k \neq j]$  be the conditional distribution of  $\boldsymbol{\theta}_j$  given the data  $\mathbf{Y}$  and the values of the other subvectors;  $[\boldsymbol{\theta}_j | \mathbf{Y}, \boldsymbol{\theta}_k, k \neq j]$  is called the *full conditional distribution* of  $\boldsymbol{\theta}_j$ . Gibbs sampling is feasible if one can sample from each of the full conditionals.

Gibbs sampling creates a Markov chain that repeatedly samples the subvectors  $\boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_M$  in the following manner. The chain starts with an arbitrary starting value  $\boldsymbol{\theta}^{(0)}$  for the parameter vector  $\boldsymbol{\theta}$ . Then the subvector  $\boldsymbol{\theta}_1^{(1)}$  is sampled from the full conditional  $[\boldsymbol{\theta}_1 | \mathbf{Y}, \boldsymbol{\theta}_k, k \neq 1]$  with each of the remaining subvectors  $\boldsymbol{\theta}_k, k \neq 1$ , set at its current value which is  $\boldsymbol{\theta}_k^{(0)}$ . Next  $\boldsymbol{\theta}_2^{(1)}$  is sampled from  $[\boldsymbol{\theta}_2 | \mathbf{Y}, \boldsymbol{\theta}_k, k \neq 2]$  with  $\boldsymbol{\theta}_k, k \neq 2$ , set at its current value, which is  $\boldsymbol{\theta}_k^{(1)}$  for  $k = 1$  and  $\boldsymbol{\theta}_k^{(0)}$  for  $k \geq 2$ . One continues it this way until each of  $\boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_M$  has been updated and one has  $\boldsymbol{\theta}^{(1)} = (\boldsymbol{\theta}_1^{(1)}, \dots, \boldsymbol{\theta}_M^{(1)})^\top$ .

Then  $\boldsymbol{\theta}^{(2)}$  is found starting at  $\boldsymbol{\theta}^{(1)}$  in the same way that  $\boldsymbol{\theta}^{(1)}$  was obtained starting at  $\boldsymbol{\theta}^{(0)}$ . Continuing in this way, we obtain the sequence  $\boldsymbol{\theta}^{(1)}, \dots, \boldsymbol{\theta}^{(N)}$  that is a Markov chain with the remarkable property that its stationary distribution is the posterior distribution of  $\boldsymbol{\theta}$ . Moreover, regardless of the starting value  $\boldsymbol{\theta}^{(0)}$ , the chain will converge to the stationary distribution. After convergence to the stationary distribution, the Markov chain samples the posterior distribution and the MCMC sample is used to compute posterior expectations, quantiles, and other characteristics of the posterior distribution.

Since the Gibbs sample does not start in the stationary distribution, the first  $N_0$  iterations are discarded as a burn-in period for an appropriately chosen value of  $N_0$ . We will assume that this has been done and  $\boldsymbol{\theta}^{(1)}, \dots, \boldsymbol{\theta}^{(N)}$  is the sample from the chain after the burn-in period. In Sect. 20.7.5, methods for choosing  $N_0$  are discussed.

#### *Example 20.8. Gibbs sampling for a normal mean and precision*

In Example 20.7, we found the posterior for a normal mean when the precision is known, and in Example 20.5, we found the posterior for a normal precision when the mean is known. These two results specify the two full conditionals and allow one to apply Gibbs sampling to the problem of estimating a normal mean and precision when both are unknown. The idea is simple. A starting value  $\tau^{(0)}$  for  $\tau$  is selected. The starting value might be the MLE, for example. However, there are advantages to using multiple chains with random starting values that are *overdispersed*, meaning that their probability distribution is more scattered than that posterior distribution; see Sect. 20.7.5.

Then, treating  $\tau$  as known and equal to  $\tau^{(0)}$ ,  $\mu^{(1)}$  is drawn randomly from its Gaussian full conditional posterior distribution given in (20.22). Note: the starting value  $\tau^{(0)}$  for the population precision  $\tau$  should not be confused with the precision  $\tau_0$  in the prior for  $\mu$ ;  $\tau^{(0)}$  is used only once, to start the Gibbs sampling algorithm; after burn-in, the Gibbs sample will not depend on the actual value of  $\tau^{(0)}$ . In contrast,  $\tau_0$  is fixed and is part of the posterior so the Gibbs sample should and will depend on  $\tau_0$ .

After  $\mu^{(1)}$  has been sampled,  $\mu$  is treated as known and equal to  $\mu^{(1)}$  and  $\tau^{(1)}$  is drawn from the full conditional (20.29). Gibbs sampling continues in this way, alternatively between sampling  $\mu$  and  $\tau$  from their full conditionals.  $\square$

### 20.7.2 Other Markov Chain Monte Carlo Samplers

It is often difficult or impossible to sample directly from the full conditionals of the posterior and then Gibbs sampling is infeasible. Fortunately, there is a large variety of other sampling algorithms that can be used when Gibbs sampling cannot be used. Programming Monte Carlo algorithms “from scratch” is beyond the scope of this book but is explained in the references in Sect. 20.14. The BUGS language discussed in Sect. 20.7.4 allows analysts to use MCMC without the time-consuming and error-prone process of programming the details.

### 20.7.3 Analysis of MCMC Output

The analysis of MCMC output typically examines scalar-valued functions of the parameter vector  $\boldsymbol{\theta}$ . The analysis should be performed on each scalar quantity of interest. Let  $\psi = \psi(\boldsymbol{\theta})$  be one such function. Suppose  $\boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_N$  is an MCMC sample from the posterior distribution of  $\boldsymbol{\theta}$ , either from a single Markov chain or from combining multiple chains, and define  $\psi_i = \psi(\boldsymbol{\theta}_i)$ . We will assume that the burn-in period and the chain lengths are sufficient so that  $\psi_1, \dots, \psi_N$  is a representative sample from the posterior distribution of  $\psi$ . Methods for diagnosing convergence and adequacy of the Monte Carlo sample size are explained in Sect. 20.7.5.

The MCMC sample mean  $\bar{\psi} = N^{-1} \sum_{i=1}^N \psi_i$  estimates the posterior expectation  $E(\psi|\mathbf{Y})$ , which is the most common Bayes estimator. The MCMC sample standard deviation  $s_\psi = \left\{ (N-1)^{-1} \sum_{i=1}^N (\psi_i - \bar{\psi})^2 \right\}^{1/2}$  estimates the posterior standard deviation of  $\psi$  and will be called the *Bayesian standard error*. If the sample size of the data is sufficiently large, then the posterior distribution will be approximately normal by Result 20.6 and an approximate  $(1 - \alpha)$  posterior interval for  $\psi$  is

$$\bar{\psi} \pm z_{\alpha/2} s_\psi. \quad (20.32)$$

Interval (20.32) is an MCMC approximation to (20.30).

However, one need not use this normal approximation to find posterior intervals. If  $L(\alpha_1)$  is the  $\alpha_1$ -lower sample quantile and  $U(\alpha_2)$  is the  $\alpha_2$ -upper sample quantile of  $\psi_1, \dots, \psi_N$ , then  $[L(\alpha_1), U(\alpha_2)]$  is a  $1 - (\alpha_1 + \alpha_2)$  posterior interval. For an equal-tailed posterior interval, one uses  $\alpha_1 = \alpha_2 = \alpha/2$ . For a highest-posterior density interval, one chooses  $\alpha_1$  and  $\alpha_2$  on a fine grid such that  $\alpha_1 + \alpha_2 = \alpha$  and  $U(\alpha_2) - L(\alpha_1)$  is minimized. One should check that the posterior density of  $\psi$  is unimodal using a kernel density estimate. If there are several modes and sufficiently deep troughs between them, then highest-posterior density posterior region could be a union of intervals, not a single interval. However, even in this somewhat unusual case,  $[L(\alpha_1), U(\alpha_2)]$  might still be used as the shortest  $1 - \alpha$  posterior *interval*.

Kernel density estimates can be used to visualize the shapes of the posterior densities. As an example, see Fig. 20.4 discussed in Example 20.9 ahead. Most automatic bandwidth selectors for kernel density estimation are based on the assumption of an independent sample. When applied to MCMC output, they might undersmooth. If the `density()` function in R is used, one might correct this undersmoothing by using a value of the `adjust` parameter greater than the default value of 1. However, Fig. 20.4 uses the default value and the amount of smoothing seems adequate; this could be due to the large Monte Carlo sample size,  $N = 10,000$ .

#### 20.7.4 JAGS

JAGS is a implementation of the BUGS (Bayesian analysis Using Gibbs Sampling) program that can be run from Windows, Mac OS, or Linux. JAGS can be used as a standalone program or it can be called from within R using the `rjags` package.

*Example 20.9. Using JAGS to fit a t-distribution to returns*

In this example, a  $t$ -distribution will be fit to S&P 500 returns using JAGS called from R. The BUGS program below is in the file `univt.bug`. The program will run under any of OpenBUGS, WinBUGS, or JAGS. In this example, JAGS will be used.

```

1 model{
2 for(i in 1:N)
3 {
4   r[i] ~ dt(mu, tau, k)
5 }
6 mu ~ dnorm(0.0, 1.0E-6)
7 tau ~ dgamma(0.1, 0.01)
8 k ~ dunif(2, 50)
9 sigma2 <- (k / (k - 2)) / tau
10 sigma <- sqrt(sigma2)
11 }
```

In BUGS, `dnorm(mu, tau)` is the normal distribution with mean equal to `mu` and precision equal to `tau`. Also, `dt(mu, tau, k)` is the  $t$ -distribution with mean equal to `mu`, degrees of freedom equal to `k`, and inverse scale parameter equal to the square root of `tau` (so `tau` is proportional to, rather than equal to, the precision and the constant of proportionality is  $\sqrt{(k-2)/k}$ ). In the BUGS program, the “for loop” (lines 3–5) specifies the likelihood and lines 6–8 specify the priors for `mu`, `tau`, and `k`. Line 9 computes the variance from `tau` and line 10 computes the standard deviation.

The R program is:

```

1 library(rjags)
2 library("Ecdat")
3 data(SP500)
4 r = SP500$r500
5 N = length(r)
6 data = list(r = r, N = N)
7 inits = function(){list(mu = rnorm(1, mean = mean(r),
8     sd = 2 * sd(r)), tau = runif(1, 0.2/var(r), 2/var(r)),
9     k = runif(1, 2.5, 10))}
10 t1 = proc.time()
11 univt.mcmc <- jags.model("univt.bug", data = data, inits = inits,
12     n.chains = 3, n.adapt = 1000, quiet = FALSE)
13 nthin = 20
14 univt.coda = coda.samples(univt.mcmc, c("mu", "k", "sigma"),
15     100*nthin, thin = nthin)
16 summary(univt.coda, digits = 2)
17 t2 = proc.time()
18 (t2 - t1) / 60
19 pdf("basic_plot.pdf", width = 4, height = 7) ## Figure 20.4
20 par(mfrow = c(4, 2))
21 plot(univt.coda, auto.layout = F) ## Figure 20.4
22 graphics.off()
23 gelman.diag(univt.coda)
24 effectiveSize(univt.coda)
25 pdf("gelman_plot.pdf", width = 6, height = 6) ## Figure 20.6
26 gelman.plot(univt.coda)
27 graphics.off()
28 dic.samples(univt.mcmc, 100*nthin, thin = nthin, type = "pD:")

```

Line 6 creates a data list that is given to JAGS and lines 7–9 creates a function `inits()` that generates starting values for each chain. The function `jags.model()` at lines 11–12 creates an object `univt.mcmc` of class `jags` containing a graphical model description of the model specified in the file `univt.bug`. This object is one of the arguments of `coda.samples()` at lines 14–15. The function `coda.samples()` produces an object `univt.coda` of class `mcmc.list` containing MCMC output. Objects of this class can be used as input to functions in the `coda` package such as `gelman.diag()`, `effectiveSize()`, `gelman.plot()`, and `summary()`. Line 18 prints the

computation time in minutes. The computation time for this example was about 6 minutes, but this number is, of course, hardware dependent.

The output from line 16 is:

```
> summary(univt.coda, digits = 2)
```

```
Iterations = 3020:5000
Thinning interval = 20
Number of chains = 3
Sample size per chain = 100
```

1. Empirical mean and standard deviation for each variable,  
plus standard error of the mean:

|       | Mean      | SD        | Naive SE  | Time-series SE |
|-------|-----------|-----------|-----------|----------------|
| k     | 6.0451630 | 0.5443919 | 3.143e-02 | 3.137e-02      |
| mu    | 0.0005129 | 0.0001850 | 1.068e-05 | 1.071e-05      |
| sigma | 0.0103017 | 0.0002078 | 1.200e-05 | 1.146e-05      |

2. Quantiles for each variable:

|       | 2.5%      | 25%       | 50%       | 75%       | 97.5%     |
|-------|-----------|-----------|-----------|-----------|-----------|
| k     | 5.1407126 | 5.6780998 | 6.0380664 | 6.4039149 | 7.1849194 |
| mu    | 0.0001289 | 0.0003821 | 0.0005046 | 0.0006191 | 0.0008763 |
| sigma | 0.0099304 | 0.0101560 | 0.0102910 | 0.0104427 | 0.0107435 |

Figure 20.4 produced at line 21 contains trace plots and kernel density estimates for `mu`, `k`, and `sigma` arranged alphabetically. The trace plots are simply time series plots of the three chains. The interpretation of trace plots is discussed in Sect. 20.7.5. The Gelman plot produced by line 26 is shown later as Fig. 20.6.

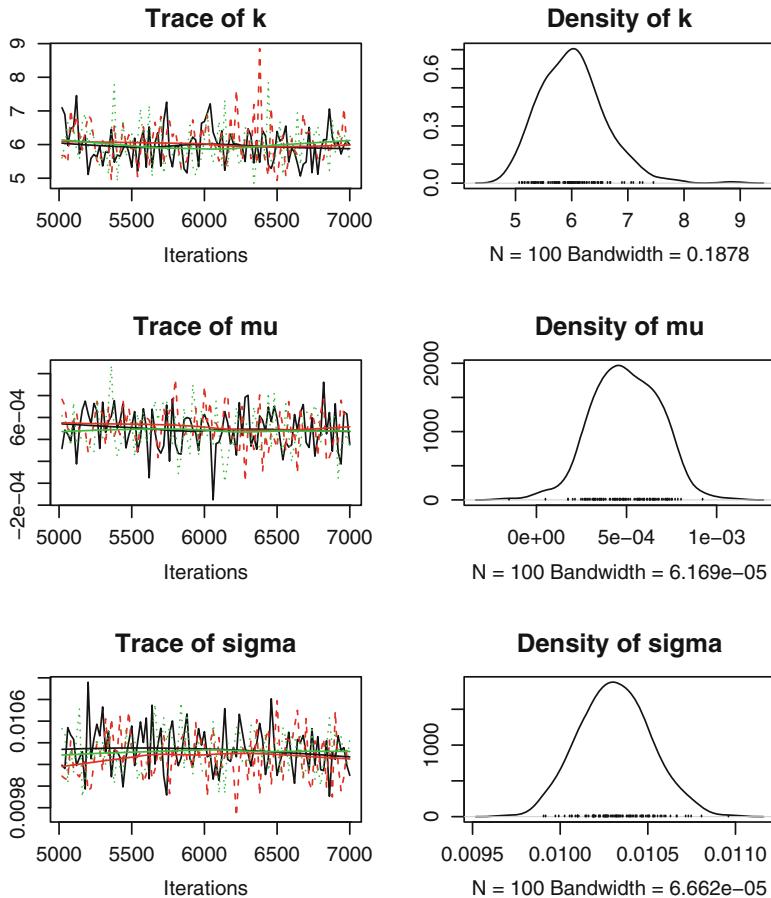
The diagnostics from lines 23–28 will discuss briefly here and described in more detail later. The Gelman diagnostics produced by line 23 are:

```
> gelman.diag(univt.coda)
Potential scale reduction factors:

      Point est. Upper C.I.
k            1.00     1.01
mu          1.00     1.01
sigma        1.02     1.06

Multivariate psrf
1.02
```

The effective sample sizes calculated at line 24 are:



**Fig. 20.4.** Trace plots and kernel density estimates in Example 20.9.

```
> effectiveSize(univt.coda)
      k       mu     sigma
366.8874 300.0000 300.0000
```

Gelman diagnostics and effective sample sizes are discussed soon in Sect. 20.7.5. DIC and  $p_D$ , which are discussed in Sect. 20.7.6 and produced at line 28, are DIC = -18,062 and  $p_D$  = 2.664:

```
> dic.samples(univt.mcmc, 100*nthin, thin = nthin, type = "pD")
|*****| 100%
Mean deviance: -18065
penalty 2.664
Penalized deviance: -18062
```

□

### 20.7.5 Monitoring MCMC Convergence and Mixing

The length  $N_0$  of the burn-in period must be sufficiently large that the Markov chain has converged to the stationary distribution by the end of burn-in. The length  $N$  of the chain after burn-in must be large enough that moments, quantiles, and other quantities computed from the MCMC sample are accurate estimates of the corresponding characteristics of posterior. Markov chains are dependent sequences and the chains used in MCMC typically have positive autocorrelation. Because of the autocorrelation, to achieve accurate estimates Markov chain samples must be larger, often far larger, than would be necessary with independent sampling. A chain that moves about the posterior slowly is said to mix poorly. The slower the mixing of the chain, the larger the sample size needed for accurate estimation.

In principle, one long Markov chain is all that is needed to sample the posterior. However, if several chains are generated, then one can compare them to decide if the burn-in period  $N_0$  and chain length  $N$  are sufficiently large. If the amount of between-chain variation in the chain means is large relative to the within-chain variation, then the chains are mixing poorly. Consequently, diagnostics for convergence and mixing can be based on between- and within-chain variation.

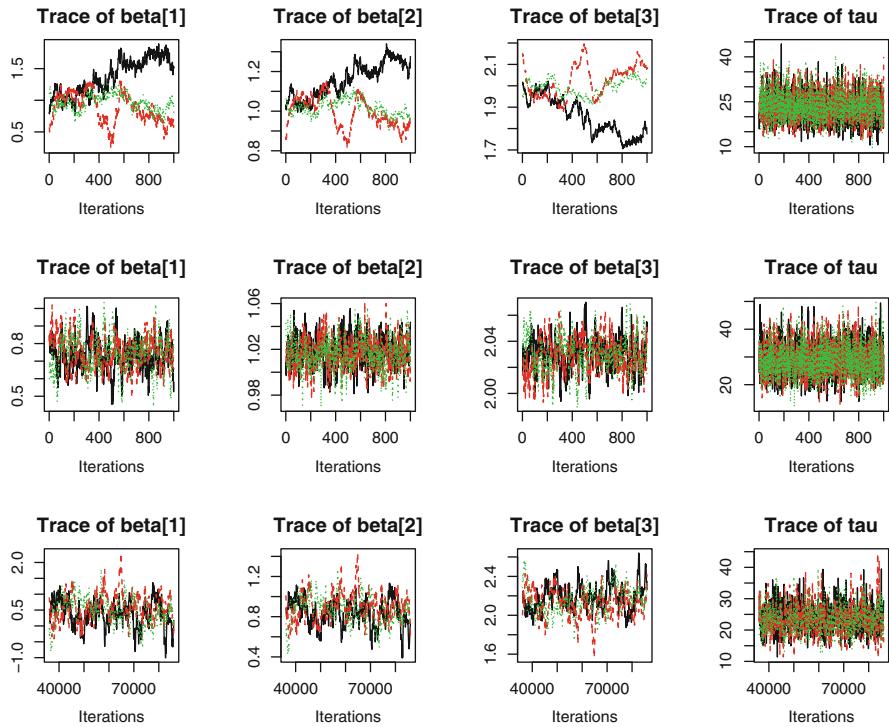
Between-chain variability will be artificially low if the chains have similar starting values. For this reason, it is recommended that the starting values be randomly sampled from a distribution with greater dispersion than the posterior. For example, one might use a Gaussian or  $t$ -distribution with mean equal to the MLE and covariance matrix equal to  $k$  times the inverse Fisher information for some  $k > 1$ , e.g.,  $k = 1.5$  or  $2$ .

*Example 20.10. Good mixing and poor mixing*

Excellent and poor mixing are contrasted in Fig. 20.5. The model is linear regression with two predictor variables and i.i.d. Gaussian noise. There are two simulated data sets. In the first data set the predictors are highly correlated (sample correlation = 0.996). Trace plots for this data set are in the top row. In the second data set the predictors are independent and the trace plots are in the middle row. Except for this difference in the amount of collinearity, the two data sets have the same distributions. In both of these cases, there are three chains and for each chain there is a burn-in period of  $N_0 = 100$  iterations and then 1000 iterations that are retained. In each row, trace plots, are shown for the three regression coefficients (intercept and two slopes) and for the residual precision (inverse variance).

In each case the three chains were started at randomly chosen initial values. The probability distribution was centered at the least-squares and “overdispersed” relative to the posterior distribution. Specifically, the regression coefficients have a Gaussian starting value distribution centered at the least-squares estimate and with covariance matrix 1.5 times the covariance matrix of the least-squares estimator. The noise variance had a starting distribution that

was uniformly distributed between 0.25 and 4 times the least-squares estimate [e.g.,  $\hat{\sigma}_\epsilon^2$  in (9.16)] of the noise variance. By using overdispersed starting values, one can discover how quickly the chains move from their starting values to the stationary distribution. The chains for the regression coefficients move very quickly in the middle row but slowly in the top row. The residual precision is unaffected by collinearity and moves quickly even in the high collinearity case.



**Fig. 20.5.** MCMC analysis of a linear regression model with two predictor variables. Simulated data. Trace plots of the regression coefficients and residual precision for three chains. The burn-in period was 100 and the chain lengths are 1000. The trace plots contain the MCMC output after the burn-in period. The intercept is  $\beta_1$  and the slopes are  $\beta_2$  and  $\beta_3$ . **Top row:** The two predictors are highly correlated and the strong collinearity is causing poor mixing of the regression coefficients. Notice that the chains have not converged to the stationary distribution by the start of the sampling period and that the between-chain variation is large. **Middle row:** The burn-in period was 100 and the chain lengths are 1000 as in the top row. The two predictors are independent and there is very good mixing because there is no collinearity. Notice that the chains have converged to the stationary distribution by the start of the sampling period and there is little between-chain variation. **Bottom row:** Same data set as the top row but with a burn-in period of 5000 and chain lengths of 30,000. The chains have been thinned so that only every 10th iteration is retained.

One solution to poor mixing is to increase the burn-in period and the chain lengths. The bottom row uses the same data set as in the top row but with a longer burn-in (5000 iterations) and longer chains (30,000 iterations). The chains have been thinned so that only every 10th iteration is retained. Thinning can speed the analysis of the MCMC output by reducing the Monte Carlo sample size and can improve the appearance of trace plots—a trace plot of 3 chains of 30,000 iterations each would be almost completely filled in. The chains appear to have converged to the stationary distribution by the end of the burn-in and to mix reasonably well over 30,000 iterations (3000 after thinning).

The BUGS code for this model is:

```

1 model{
2 for(i in 1:N){
3   y[i] ~ dnorm(mu[i],tau)
4   mu[i] <- x[i,1]*beta[1] + x[i,2]*beta[2] + x[i,3]*beta[3]
5 }
6 for(i in 1:3)beta[i] ~ dnorm(0,.00001)
7 tau ~ dgamma(0.01,0.01)
8 }
```

The R code is:

```

1 library(rjags)
2 library(coda)
3 library(mvtnorm)
4 set.seed(90201)
5 N = 50
6 beta1 = 1
7 beta2 = 2
8 alpha = 1
9 x1 = rnorm(N, mean = 3, sd = 2)
10 x2 = x1 + rnorm(N, mean = 3, sd = 0.2)
11 x = cbind(rep(1, N), x1, x2)
12 y = alpha + beta1 * x1 + beta2 * x2 + rnorm(N, mean = 0, sd = 0.2)
13 data = list(y = y, x = x, N = N)
14 summ = summary(lm(y ~ x1 + x2))
15 betahat = as.numeric(summ$coeff)[1:3]
16 covbetahat = summ$sigma^2 * solve(t(x) %*% x)
17 inits = function(){list(beta = as.numeric(rmvnrm(n = 1,
18   mean = betahat, sigma=1.5 * covbetahat)),
19   tau=runif(1, 1/(4 * summ$sigma^2), 4 / summ$sigma^2))}
20 regr <- jags.model("lin_reg_vect.bug", data = data, inits = inits,
21   n.chains = 3, n.adapt = 1000, quiet = FALSE)
22 regr.coda = coda.samples(regr, c("beta", "tau"), 1000, thin = 1)
23 regr.coda.largeN = coda.samples(regr, c("beta", "tau"),
24   50000, thin = 100)
25 ##### no collinearity #####
26 set.seed(90201)
```

```

27 x1 = rnorm(N, mean = 3, sd = 2)
28 x2 = rnorm(N, mean = 3, sd = 2) + rnorm(N, mean = 3, sd = 0.2)
29 x = cbind(rep(1, N), x1, x2)
30 y = alpha + beta1 * x1 + beta2 * x2 + rnorm(N, mean = 0, sd = 0.2)
31 data = list(y = y, x = x, N = N)
32 summ = summary(lm(y ~ x1 + x2))
33 betahat = as.numeric(summ$coeff)[1:3]
34 covbetahat = summ$sigma^2 * solve(t(x) %*% x)
35 inits=function(){list(beta = as.numeric(rmvnorm(n = 1,
36     mean = betahat, sigma = 1.5 * covbetahat)) ,
37     tau=runif(1, 1 / (4 * summ$sigma^2), 4 / summ$sigma^2))}
38 regr.noco <- jags.model("lin_reg_vect.bug", data = data,
39     inits = inits, n.chains = 3, n.adapt = 1000, quiet = FALSE)
40 regr.coda.noco = coda.samples(regr.noco, c("beta", "tau"),
41     1000, thin = 1)
42 pdf("linRegMCMC.pdf", width = 7, height = 6)
43 par(mfrow=c(3,4))
44 traceplot(regr.coda)
45 traceplot(regr.coda.noco)
46 traceplot(regr.coda.largeN)
47 graphics.off()

```

□

We now introduce two widely used diagnostics, `Rhat` and `n.eff`. Suppose one samples  $M$  chains, each of length  $N$  after burn-in. Let  $\boldsymbol{\theta}_{i,j}$  be the  $i$ th iterate from the  $j$ th chain and let  $\psi_{i,j} = \psi(\boldsymbol{\theta}_{i,j})$  for some scalar-valued function  $\psi$ . For example, to extract the  $k$ th parameter, one would use  $\psi(\mathbf{x}) = x_k$ , or  $\psi$  might compute the standard deviation or the variance from the precision. We also use  $\psi$  to denote the estimand  $\psi(\boldsymbol{\theta})$ .

Let

$$\bar{\psi}_{\cdot,j} = N^{-1} \sum_{i=1}^N \psi_{i,j} \quad (20.33)$$

be the mean of the  $j$ th chain and let

$$\bar{\psi}_{\cdot,\cdot} = M^{-1} \sum_{j=1}^M \bar{\psi}_{\cdot,j}. \quad (20.34)$$

$\bar{\psi}_{\cdot,\cdot}$  is the average of the chain means and is the Monte Carlo approximation to  $E(\psi|\mathbf{Y})$ . Then define

$$B = \frac{N}{M-1} \sum_{j=1}^M (\bar{\psi}_{\cdot,j} - \bar{\psi}_{\cdot,\cdot})^2. \quad (20.35)$$

$B/N$  is the sample variance of the chain means. Define

$$s_j^2 = (N - 1)^{-1} \sum_{i=1}^N (\psi_{i,j} - \bar{\psi}_{\cdot,j})^2, \quad (20.36)$$

the variance of the  $j$ th chain, and define

$$W = M^{-1} \sum_{j=1}^M s_j^2. \quad (20.37)$$

$W$  is the pooled within-chain variance. The two variances,  $B$  and  $W$ , are combined into

$$\widehat{\text{var}}^+(\psi|\mathbf{Y}) = \frac{N-1}{N}W + \frac{1}{N}B, \quad (20.38)$$

where, as before,  $\mathbf{Y}$  is the data.

To assess convergence, one can use

$$\hat{R} = \sqrt{\frac{\widehat{\text{var}}^+(\psi|\mathbf{Y})}{W}}. \quad (20.39)$$

$\hat{R}$  is called the “potential scale reduction factor” in output produced by the function `gelman.diag()`; see the output in Example 20.9.  $\hat{R}$  is also called the “shrink factor” or sometime the “Gelman shrink factor.”

When the chains have not yet reached the stationary distribution, the numerator  $\widehat{\text{var}}^+(\psi|\mathbf{Y})$  inside the radical is an upward-biased estimate of  $\text{var}(\psi|\mathbf{Y})$  and the denominator  $W$  is a downward-biased estimator of this quantity. Both biases converge to 0 as the burn-in period and Monte Carlo sample size increase. Therefore, larger values of  $\hat{R}$  indicate nonconvergence. If  $\hat{R}$  is approximately equal to 1, say at most 1.1, then the chains are considered to have converged to the stationary distribution and  $\widehat{\text{var}}^+(\psi|\mathbf{Y})$  can be used as an estimate of  $\text{var}(\psi|\mathbf{Y})$ . A larger value of  $\hat{R}$  is an indication that a longer burn-in period is needed. A small value of  $\hat{R}$  is evidence that the burn-in period is adequate, but we need another diagnostic, the effective sample size, to know if the sampling period was long enough.

The *effective sample size* of the chain is

$$N_{\text{eff}} = MN \frac{\widehat{\text{var}}^+(\psi|\mathbf{Y})}{B}. \quad (20.40)$$

The interpretation of  $N_{\text{eff}}$  is that the Markov chain can estimate the posterior expectation of  $\psi$  with approximately the same precision as would be obtained from an independent sample from the posterior of size  $N_{\text{eff}}$ . (Of course, it is usually impossible to actually obtain an independent sample, which is why MCMC is used.)

$N_{\text{eff}}$  is derived by comparing the variance of  $\bar{\psi}_{\cdot,\cdot}$  from Markov chain sampling with the variance of  $\bar{\psi}_{\cdot,\cdot}$  under hypothetical independent sampling. Since

$\bar{\psi}_{\cdot,\cdot}$  is the average of the means of  $M$  independent chains and since  $B/N$  is the sample variance of these  $M$  chain means,

$$M^{-1} \frac{B}{N} \quad (20.41)$$

estimates the Monte Carlo variance of  $\bar{\psi}_{\cdot,\cdot}$ . Suppose instead of sampling  $M$  chains, each of length  $N$ , one could take an independent sample of size  $N^*$  from the posterior. The Monte Carlo variance of the mean of this sample would be

$$\frac{\text{var}(\psi|\mathbf{Y})}{N^*},$$

which can be estimated by

$$\frac{\widehat{\text{var}}^+(\psi|\mathbf{Y})}{N^*}. \quad (20.42)$$

By definition  $N_{\text{eff}}$  is the value of  $N^*$  that makes (20.41) equal to (20.42) and therefore  $N^*$  is given by (20.40). Because  $B/N$  is the sample variance of  $M$  chains and because  $M$  is typically quite small, often between 2 and 5,  $B$  has considerable Monte Carlo variability. Therefore,  $N_{\text{eff}}$  is at best a crude estimate of the effective sample size.

$\hat{R}$  and  $N_{\text{eff}}$  are computed by the functions `gelman.diag()` and `effectiveSize()` in the `coda` package. The function `gelman.plot()` in the `coda` package plots the  $\hat{R}$  evaluated at various times along the simulation. The documentation for this function notes that “A potential problem with `gelman.diag` is that it may mis-diagnose convergence if the shrink factor happens to be close to 1 by chance. By calculating the shrink factor at several points in time, `gelman.plot` shows if the shrink factor has really converged, or whether it is still fluctuating.” Figure 20.6 shows the Gelman plot from Example 20.9. The dashed red line is an upper 97.5 % confidence limit. Although  $\hat{R}$  varies during the simulations, it appears to have converged to values close to 1 by the end of the simulations.

How large should  $N_{\text{eff}}$  be? Of course, larger means better Monte Carlo accuracy, but larger values of  $N_{\text{eff}}$  require more or longer chains, so we do not want  $N_{\text{eff}}$  to be unnecessarily large. The effect of  $N_{\text{eff}}$  on estimation error can be seen by decomposing the estimation error  $\psi - \bar{\psi}_{\cdot,\cdot}$  into two parts, which will be called  $E_1$  and  $E_2$ :

$$\psi - \bar{\psi}_{\cdot,\cdot} = \{\psi - E(\psi|\mathbf{Y})\} + \{E(\psi|\mathbf{Y}) - \bar{\psi}_{\cdot,\cdot}\} = E_1 + E_2. \quad (20.43)$$

If  $E\{\psi|\mathbf{Y}\}$  could be computed exactly so that it, not  $\bar{\psi}_{\cdot,\cdot}$ , would be the estimator of  $\psi$ , then  $E_1$  would be the only error.  $E_2$  is the error due to the Monte Carlo approximation of  $E\{\psi|\mathbf{Y}\}$  by  $\bar{\psi}_{\cdot,\cdot}$ . The two errors  $E_1$  and  $E_2$  are uncorrelated, so

$$\text{var}\{(\psi - \bar{\psi}_{\cdot,\cdot})|\mathbf{Y}\} = \text{var}(E_1|\mathbf{Y}) + \text{var}(E_2|\mathbf{Y})$$

$$\begin{aligned}
&= \text{var}(\psi|\mathbf{Y}) + \frac{\text{var}(\psi|\mathbf{Y})}{N_{\text{eff}}} \\
&= \text{var}(\psi|\mathbf{Y}) \left( 1 + \frac{1}{N_{\text{eff}}} \right)
\end{aligned}$$

by the definitions of  $\text{var}(\psi|\mathbf{Y})$  and  $N_{\text{eff}}$  and using the approximation  $\widehat{\text{var}}^+(\psi|\mathbf{Y}) \approx \text{var}(\psi|\mathbf{Y})$ . Using the Taylor series approximation  $\sqrt{1+\delta} \approx 1 + \delta/2$  for small values of  $\delta$ , we see that

$$\sqrt{\text{var}\{(\psi - \bar{\psi}_{\cdot,\cdot})|\mathbf{Y}\}} \approx \sqrt{\text{var}(\psi|\mathbf{Y})} \left( 1 + \frac{1}{2N_{\text{eff}}} \right). \quad (20.44)$$

Recall that  $\sqrt{\text{var}\{(\psi - \bar{\psi}_{\cdot,\cdot})|\mathbf{Y}\}}$  is the “Bayesian standard error.” If  $N_{\text{eff}} \geq 50$ , then we see from (20.44) that the standard error is inflated by Monte Carlo error by at most 1%. Thus, one might use the rule-of-thumb that  $N_{\text{eff}}$  should be at least 50. Remember, however, that  $N_{\text{eff}}$  is estimated only crudely because the number of chains is small. Thus, we might want to have  $N_{\text{eff}}$  at least 100 to provide some leeway for error in the estimation of  $N_{\text{eff}}$ .

The value of  $N_{\text{eff}}$  can vary between different choices of  $\psi$ . Recall that the values of  $N_{\text{eff}}$  from Example 20.9 were:

```
> effectiveSize(univt.coda)
      k       mu     sigma
366.8874 300.0000 300.0000
```

In this example,  $N_{\text{eff}}$  is 300 for `mu` and `sigma` and only slightly larger for `k`. (In more complex models, much greater variation in  $N_{\text{eff}}$  is common.) In this example, 300 is the Monte Carlo sample size since there are three chains, each of length 100 after burn-in and thinning.<sup>3</sup> Therefore, in this simple example, MCMC sampling is as effective as independent sampling; this is not a typical case.

For convenience,  $\hat{R}$  values in Example 20.9 are listed again:

```
> gelman.diag(univt.coda)
Potential scale reduction factors:
```

|                    | Point est. | Upper C.I. |
|--------------------|------------|------------|
| <code>k</code>     | 1.00       | 1.01       |
| <code>mu</code>    | 1.00       | 1.01       |
| <code>sigma</code> | 1.02       | 1.06       |

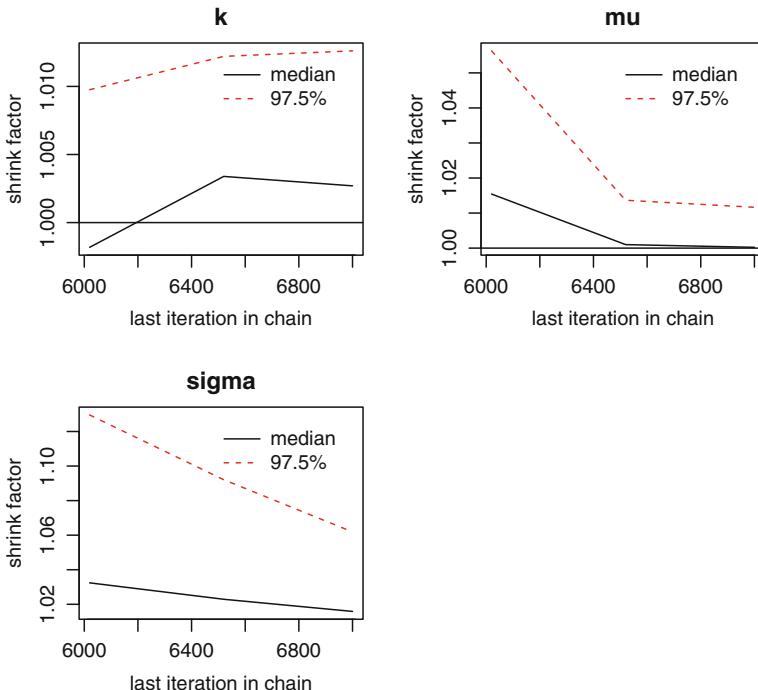
Multivariate psrf

1.02

---

<sup>3</sup> The effective sample size can be larger than the actual sample size if there is negative correlation, but negative correlation is unlikely. It is more likely that some of the effective sample sizes exceed the actual sample sizes due to random variation, i.e., estimation error.

One can see that  $\widehat{R}$  is at most 1.02 for all of the parameters that were monitored, which is another indication that the amount of MCMC sampling was sufficient. Even the 95 % upper confidence limits are satisfactory, at most 1.06.



**Fig. 20.6.** Gelman plot in Example 20.9.

### 20.7.6 DIC and $p_D$ for Model Comparisons

In this section, we introduce two widely used statistics, DIC and  $p_D$ . DIC is used to compare several models for the same data set and is a Bayesian analog of AIC.  $p_D$  is a Bayesian analog to the number of parameters in the model.

Recall from Sect. 5.12 that the deviance, denoted now by  $D(\mathbf{Y}, \boldsymbol{\theta})$ , is minus twice the log-likelihood, and AIC defined by (5.29) is

$$\text{AIC} = D(\mathbf{Y}, \boldsymbol{\theta}_{\text{ML}}) + 2p, \quad (20.45)$$

where  $\widehat{\boldsymbol{\theta}}_{\text{ML}}$  is the MLE and  $p$  is the dimension of  $\boldsymbol{\theta}$ . A Bayesian analog of the MLE is the posterior mean, the usual Bayes estimator, which can be estimated by MCMC.

We need a Bayesian analog of  $p$ , the number of parameters. It may seem strange at first that we do not simply use  $p$  itself as in a non-Bayesian analysis. After all, the number of parameters has not changed just because we now have a prior and are using Bayesian estimation. However, the prior information used in a Bayesian analysis somewhat constrains the estimated parameters, which makes the *effective* number of parameters less than  $p$ . To appreciate why this is true, consider an example where there are  $d$  returns on equities that are believed to be similar. Assume the returns have a multivariate normal distribution. Let's focus on the  $d$  expected returns, call them  $\mu_1, \dots, \mu_d$ . To a non-Bayesian, there are two ways to model  $\mu_1, \dots, \mu_d$ . The first is to assume that they are all equal, say to  $\mu$ , and then there is only one parameter to model the means. The other possibility is to assume that the expected returns are not equal so that there are  $d$  parameters.

A Bayesian can achieve a compromise between these two extreme by specifying a prior such that  $\mu_1, \dots, \mu_d$  are similar but not identical. For example, we could assume that they are i.i.d.  $N(\mu, \sigma_\mu^2)$ , and  $\sigma_\mu^2$  would specify the degree of similarity. It is important to appreciate that  $\sigma_\mu^2$  can be estimated from the data, that is, there is no need to specify in advance the degree of similarity between the means. The result of using such prior information is that the *effective* number of parameters to specify  $\mu_1, \dots, \mu_d$  is greater than 1 but less than  $d$ .

The effective number of parameters is defined as

$$p_D = \widehat{D}_{\text{avg}} - D(\mathbf{Y}, \bar{\boldsymbol{\theta}}), \quad (20.46)$$

where

$$\bar{\boldsymbol{\theta}} = (NM)^{-1} \sum_{j=1}^M \sum_{i=1}^N \boldsymbol{\theta}_{i,j}$$

is the average of the MCMC sample of  $\boldsymbol{\theta}_{i,j}$  and estimates the posterior expectation of  $\boldsymbol{\theta}$ , and

$$\widehat{D}_{\text{avg}} = (NM)^{-1} \sum_{j=1}^M \sum_{i=1}^N D(\mathbf{Y}, \boldsymbol{\theta}_{i,j})$$

is an MCMC estimate of

$$D_{\text{avg}} = E\{D(\mathbf{Y}, \boldsymbol{\theta}) | \mathbf{Y}\}. \quad (20.47)$$

In analogy with (20.45), DIC is defined as

$$\text{DIC} = D(\mathbf{Y}, \bar{\boldsymbol{\theta}}) + 2p_D.$$

By (20.46), we have as well that

$$\text{DIC} = \widehat{D}_{\text{avg}} + p_D.$$

The function `dic.samples()` in the `rjags` package reports  $\hat{D}_{\text{avg}}$  as the “mean deviance,”  $p_D$  as the “penalty,” and DIC as the “penalized deviance.” See the output in Example 20.9.

As the following example illustrates,  $p_D$  is primarily a measure of the posterior variability of  $\boldsymbol{\theta}$ , which increases as  $p$  increases or the amount of prior information about  $\boldsymbol{\theta}$  decreases relative to the information in the sample.

*Example 20.11.  $p_D$  when estimating a normal mean with known precision*

Suppose that  $\mathbf{Y} = (Y_1, \dots, Y_n)$  are i.i.d.  $N(\mu, 1)$ , so  $\boldsymbol{\theta} = \mu$  in this example. Then the log-likelihood is

$$\begin{aligned}\log\{L(\mu)\} &= -\frac{1}{2} \sum_{i=1}^n (Y_i - \mu)^2 - \frac{n}{2} \log(2\pi) \\ &= -\frac{1}{2} \left\{ \sum_{i=1}^n (Y_i - \bar{Y})^2 + n(\bar{Y} - \mu)^2 \right\} - \frac{n}{2} \log(2\pi),\end{aligned}$$

and so

$$D(\mathbf{Y}, \mu) = \sum_{i=1}^n (Y_i - \bar{Y})^2 + n(\bar{Y} - \mu)^2 + n \log(2\pi). \quad (20.48)$$

When  $p_D$  is computed, quantities not depending on  $\mu$  cancel with the subtraction in (20.46). Therefore, for the purpose of computing  $p_D$ , we can use

$$D(\mathbf{Y}, \mu) = n(\bar{Y} - \mu)^2. \quad (20.49)$$

Then

$$D\{\mathbf{Y}, E(\mu|\mathbf{Y})\} = \{\bar{Y} - E(\mu|\mathbf{Y})\}^2, \quad (20.50)$$

and

$$\begin{aligned}D_{\text{avg}} &= n E\{(\bar{Y} - \mu)^2 | \mathbf{Y}\} \\ &= n \left( \{\bar{Y} - E(\mu|\mathbf{Y})\}^2 + E[\{E(\mu|\mathbf{Y}) - \mu\}^2 | \mathbf{Y}] \right) \\ &= n \left[ \{\bar{Y} - E(\mu|\mathbf{Y})\}^2 + \text{Var}(\mu|\mathbf{Y}) \right] \\ &= D\{\mathbf{Y}, E(\mu|\mathbf{Y})\} + n \text{Var}(\mu|\mathbf{Y}),\end{aligned} \quad (20.51)$$

because  $\{\bar{Y} - E(\mu|\mathbf{Y})\}$  and  $\{E(\mu|\mathbf{Y}) - \mu\}$  are conditionally uncorrelated given  $\mathbf{Y}$ . Therefore,

$$\begin{aligned}p_D &= \hat{D}_{\text{avg}} - D\{\mathbf{Y}, E(\mu|\mathbf{Y})\} \\ &\approx D_{\text{avg}} - D\{\mathbf{Y}, E(\mu|\mathbf{Y})\} = n \text{Var}(\mu|\mathbf{Y}) = \frac{n}{n + \tau_0},\end{aligned} \quad (20.52)$$

where the last equality uses (20.21) and  $\tau_0$  is the prior precision for  $\mu$ . The approximation (“ $\approx$ ”) in (20.52) becomes equality as the Monte Carlo sample size  $N$  increases to  $\infty$ .

As  $\tau_0 \rightarrow 0$ , the amount of prior information becomes negligible and the right-hand side of (20.52) converges to  $p = 1$ . Conversely, as  $\tau_0 \rightarrow \infty$ , the amount of prior information increases without bound and the right-hand side of (20.52) converges to 0. This is an example of a general phenomenon—more prior information means less effective parameters.  $\square$

Generally,  $p_D \approx p$  when  $p$  is small and there is little prior information. In other cases, such as when  $d$  means are modeled as coming from a common normal distribution,  $p_D$  could be considerably less than  $p$ —see Example 20.12.

When comparing models using DIC, smaller is better, though, like AIC and BIC, DIC should never be used blindly. Often subject-matter considerations or model simplicity will lead an analyst to select a model other than the one minimizing DIC.

The function `dic.sample()` returns both DIC and  $p_D$ , as can be seen in the output from Example 20.9 which was:

```
> dic.samples(univt.mcmc, 100*nthin, thin = nthin, type = "pD")
|*****| 100%
Mean deviance: -18065
penalty 2.664
Penalized deviance: -18062
```

## 20.8 Hierarchical Priors

A common situation is having a number of parameters that are believed to have similar, but not identical, values. For example, the expected returns on several equities might be thought similar. In such cases, it can be useful to pool information about the parameters to improve the specification of the prior, because the use of good prior information will improve the accuracy of the estimation. A effective method for pooling information is a Bayesian analysis with a so-called “hierarchical prior” that allows one to shrink the estimates toward each other or toward some other target. An example of the latter would be shrinking the sample covariance matrix of returns toward an estimate from the CAPM or another factor model. This type of shrinkage would achieve a tradeoff between the high variability of the sample covariance matrix and the potential bias of the covariance matrix estimator from a factor model when the factor model does not fit perfectly.

As before, let the likelihood be  $f(\mathbf{y}|\boldsymbol{\theta})$ . The likelihood is the first layer (or stage) in the hierarchy. So far in this chapter, the prior density of  $\boldsymbol{\theta}$ , which is the second layer, has been  $\pi(\boldsymbol{\theta}|\boldsymbol{\gamma})$ , where the parameter vector  $\boldsymbol{\gamma}$  in the prior has a known value, say  $\boldsymbol{\gamma}_0$ . For example, in Example 20.3 the prior had a beta distribution with both parameters fixed.

In a *hierarchical* or multistage prior,  $\gamma$  is unknown and has its own prior  $\pi(\gamma|\delta)$  (the third layer). Typically,  $\delta$  has a known value, though one can add further layers to the hierarchy by making  $\delta$  unknown with its own prior, and so forth.

It is probably easiest to understand hierarchical priors using examples.

*Example 20.12. Estimating expected returns on midcap stocks*

This example uses the `midcapD.ts` dataset. This data set contains 500 daily returns on 20 midcap stocks and on the market and was used in Example 5.2.

The data set will be divided into the “training data,” which contains the first 100 days of returns and the “test” data containing the last 400 days of returns. Only the training data will be used for estimation. The test data will be used to compare the estimates from the training data. The test data sample size was chosen intentionally to be relatively large so that we can consider the mean returns from the test data to be the “true” expected returns on the 20 stocks, though, of course, this is only an approximation. The “true” expected returns will be estimated using the training data.

We will compare three possible estimators of the true expected returns.

- (a) sample means (the 20 mean returns on the midcap stocks for the first 100 days);
- (b) pooled estimation (total shrinkage where every expected return has the same estimate);
- (c) Bayes estimation with a hierarchical prior (shrinkage).

Method (a) is the “usual” non-Bayesian estimator where each expected return is estimated by the sample mean of that stock. In method (b), every expected return has the same estimate, which is the “mean of means,” that is, the average of the 20 means from (a). Bayes shrinkage, which will be explained in this example, shrinks the 20 individual means toward the mean of means using a hierarchical prior. Bayesian shrinkage is a compromise between (a) and (b). Shrinkage was also used in Example 16.10 though in that example the amount of shrinkage was chosen arbitrarily because Bayesian methods had not yet been introduced.

Let  $R_{i,t}$  be the  $t$ th daily return on  $i$  stock expressed as a percentage. For Bayesian shrinkage, the first layer will be the simple model

$$R_{i,t} = \mu_i + \epsilon_{i,t},$$

where  $\epsilon_{i,t}$  are i.i.d.  $N(0, \sigma_\epsilon^2)$ . This model has several unrealistic aspects: (a) the assumption that the standard deviation of  $\epsilon_{i,t}$  does not depend on  $i$ ; (b) the assumption that  $\epsilon_{i,t}$  and  $\epsilon_{i',t}$  are independent (we know that there will be cross-sectional correlations); (c) the assumption that there are no GARCH effects; (d) the assumption that the  $\epsilon_{i,t}$  are normally distributed rather than

having heavy tails. Nonetheless, for the purpose of estimating expected returns, this model should be adequate. Remember, “all models are wrong but some models are useful,” and, of course, what is “useful” depends on the objectives of the analysis.

The hierarchy prior has second layer

$$\mu_i \sim \text{i.i.d. } N(\alpha, \sigma_\mu^2).$$

The assumption here is that the expected returns for the 20 midcap stocks have been sampled from a large population of expected returns, perhaps of all midcap stocks or even a larger population. The mean of that population is  $\alpha$  and the standard deviation is  $\sigma_\mu$ .

If we used a non-hierarchical prior, then we would need to specify values of  $\alpha$  and  $\sigma_\mu$ . This is exactly what was done in Example 20.4, except in that example  $\sigma_\epsilon^2$  also was known. We probably have a rough idea of the values of  $\alpha$  and  $\sigma_\mu$ , but it is unlikely that we have precise information about them, and we saw in Example 20.4 that a rather accurate specification of the prior is needed for the Bayes estimator to improve upon the sample means. In fact, the Bayes estimator can easily be inferior to the sample means if the prior is poorly chosen.

The third layer will be a prior on  $\alpha$  and  $\sigma_\mu$  and will let us use the data to estimate these parameters. It is important to appreciate why we can estimate  $\alpha$  and  $\sigma_\mu$  in this example, but they could not be estimated in Example 20.4. The reason is that we now have 20 expected returns (the  $\mu_i$ ) that are distributed with the same mean  $\alpha$  and standard deviation  $\sigma_\mu$ . In contrast, in Example 20.4 there is only a single  $\mu$  and so it not possible to estimate the mean and variance of the population from which this  $\mu$  was sampled.

Because there is now a substantial amount of information in the data about  $\alpha$ ,  $\sigma_\epsilon^2$ , and  $\sigma_\mu^2$ , we could use fairly noninformative priors for them to “let the data speak for themselves.”

The BUGS program for this example is:

```

1 model{
2   for (i in 1:n)
3   {
4     for (j in 1:m)
5     {
6       x1[i,j] ~ dnorm(mu[j], tau_eps)
7     }
8   }
9   for (j in 1:m)
10  {
11     mu[j] ~ dnorm(alpha, tau_mu)
12  }
13 alpha ~ dnorm(0.0, 1.0E-3)
14 tau_eps ~ dgamma(0.1, 0.01)
15 tau_mu ~ dgamma(0.1, 0.01)
```

```

16 sigma_eps <- 1 / sqrt(tau_eps)
17 sigma_mu <- 1 / sqrt(tau_mu)
18 }

```

The R code for this example is below:

```

1 library(rjags)
2 dat = read.csv("midcapD.ts.csv")
3 market = 100 * as.matrix(dat[, 22])
4 x = 100 * as.matrix(dat[, -c(1, 22)])
5 m = 20
6 k = 100
7 x1 = x[1:k, ]
8 x2 = x[(k+1):500, ]
9 mu1 = apply(x1, 2, mean)
10 mu2 = apply(x2, 2, mean)
11 means = apply(x1, 2, mean)
12 sd2 = apply(x1, 2, sd)
13 tau_mu = 1 / mean(sd2^2)
14 tau_eps = 1 / sd(means)^2
15 n = k
16 data = list(x1 = x1, n = n, m = m)
17 inits.midCap = function(){list(alpha = 0.001, mu = means,
18   tau_eps = tau_eps, tau_mu = tau_mu)}
19 midCap <- jags.model("midCap.bug", data = data, inits = inits.midCap,
20   n.chains = 3, n.adapt = 1000, quiet = FALSE)
21 nthin = 20
22 midCap.coda = coda.samples(midCap, c("mu", "tau_mu", "tau_eps",
23   "alpha", "sigma_mu", "sigma_eps"), 500 * nthin, thin = nthin)
24 summ.midCap = summary(midCap.coda)
25 summ.midCap
26 post.means = summ.midCap[[1]][2:21, 1]
27 pdf("midcap.pdf", width = 6, height = 3.75)
28 par(mfrow = c(1, 2))
29 plot(c(rep(1, m), rep(2, m)), c(mu1, mu2),
30   xlab = "estimate", target", ylab = "mean",
31   main = "sample means",
32   ylim = c(-0.3, 0.7), axes = FALSE)
33 axis(2)
34 axis(1, labels = FALSE, tick = TRUE, lwd.tick = 0)
35 for (i in 1:m){lines(1:2, c(mu1[i], mu2[i]), col = i)}
36 plot(c(rep(1, m), rep(2, m)), c(post.means, mu2),
37   xlab = "estimate", target", ylab = "mean",
38   main = "Bayes",
39   ylim=c(-0.3, 0.7), axes = FALSE)
40 axis(2)
41 axis(1, labels = FALSE, tick = TRUE, lwd.tick = 0)
42 for (i in 1:m){lines(1:2, c(post.means[i], mu2[i]) ,col=i)}
43 graphics.off()
44 options(digits = 2)

```

```

45 sum((mu1 - mu2)^2 )
46 sum((post.means - mu2)^2)
47 sum((mean(mu1) - mu2)^2)

```

The output is below.

```
> summ.midCap
```

```
Iterations = 20:10000
```

```
Thinning interval = 20
```

```
Number of chains = 3
```

```
Sample size per chain = 500
```

1. Empirical mean and standard deviation for each variable,  
plus standard error of the mean:

|       | Mean    | SD       | Naive SE  | Time-series SE |
|-------|---------|----------|-----------|----------------|
| alpha | 0.08730 | 0.102433 | 2.645e-03 | 3.101e-03      |
| mu[1] | 0.11121 | 0.171456 | 4.427e-03 | 4.546e-03      |
| mu[2] | 0.12128 | 0.169230 | 4.369e-03 | 4.892e-03      |
| mu[3] | 0.07871 | 0.170849 | 4.411e-03 | 4.702e-03      |

(edited to save space)

|           |          |           |           |           |
|-----------|----------|-----------|-----------|-----------|
| mu[19]    | 0.05082  | 0.175466  | 4.531e-03 | 4.422e-03 |
| mu[20]    | 0.03997  | 0.184614  | 4.767e-03 | 4.873e-03 |
| sigma_eps | 4.30691  | 0.067810  | 1.751e-03 | 1.629e-03 |
| sigma_mu  | 0.14970  | 0.067054  | 1.731e-03 | 1.671e-03 |
| tau_eps   | 0.05395  | 0.001699  | 4.386e-05 | 4.074e-05 |
| tau_mu    | 75.70128 | 68.715669 | 1.774e+00 | 1.738e+00 |

2. Quantiles for each variable:

|       | 2.5%     | 25%       | 50%     | 75%     | 97.5%   |
|-------|----------|-----------|---------|---------|---------|
| alpha | -0.11465 | 0.017800  | 0.08600 | 0.15560 | 0.29111 |
| mu[1] | -0.21768 | -0.001040 | 0.10910 | 0.21864 | 0.43408 |
| mu[2] | -0.21107 | 0.018025  | 0.11704 | 0.22382 | 0.47727 |
| mu[3] | -0.27262 | -0.032547 | 0.08392 | 0.19634 | 0.40900 |

(edited to save space)

|           |          |           |          |          |           |
|-----------|----------|-----------|----------|----------|-----------|
| mu[19]    | -0.30441 | -0.059642 | 0.05612  | 0.16521  | 0.38525   |
| mu[20]    | -0.34436 | -0.069959 | 0.04462  | 0.16479  | 0.37048   |
| sigma_eps | 4.17808  | 4.261352  | 4.30714  | 4.35226  | 4.43733   |
| sigma_mu  | 0.06155  | 0.100600  | 0.13853  | 0.18225  | 0.31293   |
| tau_eps   | 0.05079  | 0.052792  | 0.05390  | 0.05507  | 0.05729   |
| tau_mu    | 10.21311 | 30.107752 | 52.10738 | 98.81013 | 263.94725 |

The posterior means of  $\sigma_\mu$  and  $\sigma_\epsilon$  are 0.150 % and 4.31 %, respectively (the returns are as percentages). If we look at precisions instead of standard

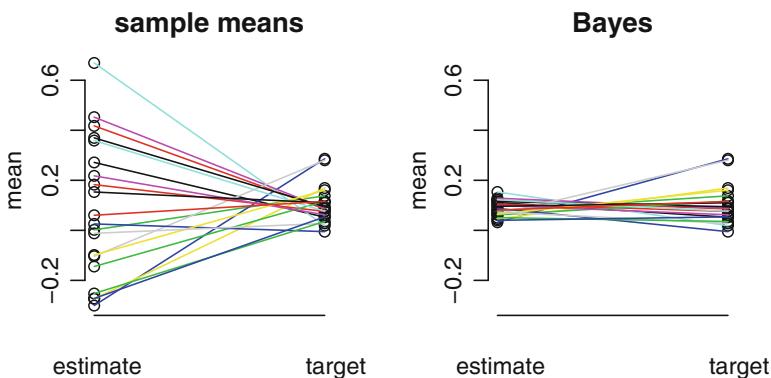
deviations, then we find that the posterior means of  $\tau_\mu$  and  $\tau_\epsilon$  are 75.7 and 0.0540. Using the notation of (20.24), in the present example  $\tau_{\bar{Y}}$  is  $100\tau_\epsilon = 5.4$  and  $\tau_0 = \tau_\mu = 75.7$ . Therefore,  $\delta$  in (20.24) is  $5.4/(5.4 + 75.7) = 0.067$ . Recall that  $\delta$  close to 0 (far from 1) results in substantial shrinkage, so  $\delta$  equal to 0.064 causes a great amount of shrinkage of the sample means toward the mean of means, as can be seen in Fig. 20.7.

To compare the estimators, we use the sum of squared errors (SSE) defined as

$$\text{SSE} = \sum_{i=1}^{20} (\hat{\mu}_i - \mu_i)^2,$$

where  $\mu_i$  is the  $i$ th “true” mean from the test data and  $\hat{\mu}_i$  is an estimate from the training data. The values of the SSE are found in Table 20.1. The SSE for the sample means is about 11 (1.9/0.18) times larger than for the Bayes estimate. Clearly, shrinkage is very successful in this example.

Interestingly, complete shrinkage to the pooled mean is even better than Bayesian shrinkage. Bayesian shrinkage attempts to estimate the optimal amount of shrinkage, but, of course, it cannot do this perfectly. Although complete shrinkage is better than Bayesian shrinkage in this example, complete shrinkage is, in general, dangerous since it will have a large SSE in examples where the true means differ more than in this case. If one has a strong prior belief that the true means are very similar, one should use this



**Fig. 20.7.** Estimation of the average returns for 20 midcap stocks. “Target” is the quantity being estimated, specifically the average return over 400 days of test data. “Estimate” is an estimate based on the 100 previous days of training data. On the left, the estimates are the 20 individual sample means. On the right, the estimates are the sample means shrunk toward their mean. In each panel, the estimate and target for each stock are connected by a line. On the left, the sample means of the training data are so variable that the stocks with smaller (larger) means in the training data often have larger (smaller) means in the test data. The Bayes estimates on the right are much closer to the targets.

belief when specifying a prior for  $\sigma_\mu$ . Instead of using a noninformative prior as in this example, one would use a prior more concentrated near 0.  $\square$

**Table 20.1.** Sum of squared errors (SSE) for three estimators of the expected returns of 20 midcap stocks.

| Estimate         | SSE  |
|------------------|------|
| (a) sample means | 1.9  |
| (b) pooled mean  | 0.12 |
| (c) Bayes        | 0.18 |

## 20.9 Bayesian Estimation of a Covariance Matrix

In this section, we assume that  $\mathbf{Y}_1, \dots, \mathbf{Y}_n$  is an i.i.d. sample from a  $d$ -dimensional  $N(\boldsymbol{\mu}, \boldsymbol{\Sigma})$  distribution or a  $d$ -dimensional  $t_\nu(\boldsymbol{\mu}, \boldsymbol{\Lambda})$  distribution. We will focus on estimation of the covariance matrix  $\boldsymbol{\Sigma}$  of a multivariate normal distribution or the scale matrix  $\boldsymbol{\Lambda}$  of a multivariate  $t$ -distribution. The *precision matrix* is defined as  $\boldsymbol{\Sigma}^{-1}$  or  $\boldsymbol{\Lambda}^{-1}$  for the Gaussian and  $t$ -distributions, respectively. This definition is analogous to the univariate case where the precision is defined as the reciprocal of the variance or squared scale parameter.

We will start with Gaussian distributions.

### 20.9.1 Estimating a Multivariate Gaussian Covariance Matrix

In the multivariate Gaussian case, the conjugate prior for the precision matrix  $\boldsymbol{\Sigma}^{-1}$  is the Wishart distribution. The Wishart distribution, denoted by  $\text{Wishart}(\nu, \mathbf{A})$ , has a univariate parameter  $\nu$  called the degrees of freedom and a matrix parameter  $\mathbf{A}$  that can be any nonsingular covariance matrix. There is a simple definition of the  $\text{Wishart}(\nu, \mathbf{A})$  distribution when  $\nu$  is an integer. Let  $\mathbf{Z}_1, \dots, \mathbf{Z}_n$  be i.i.d.  $N(\boldsymbol{\mu}, \mathbf{A})$ . In this case, the distribution of

$$\sum_{i=1}^n (\mathbf{Z}_i - \boldsymbol{\mu})(\mathbf{Z}_i - \boldsymbol{\mu})^\top$$

is  $\text{Wishart}(n, \mathbf{A})$ . Also, the distribution of

$$\sum_{i=1}^n (\mathbf{Z}_i - \bar{\mathbf{Z}})(\mathbf{Z}_i - \bar{\mathbf{Z}})^\top \tag{20.53}$$

is  $\text{Wishart}(n-1, \mathbf{A})$ . Because the sum in (20.53) is  $n-1$  times the sample covariance matrix, the Wishart distribution is important for inference about the covariance matrix of a Gaussian distribution.

The density of a  $\text{Wishart}(\nu, \mathbf{A})$  distribution for any positive value of  $\nu$  is

$$f(\mathbf{W}) = C(\nu, d) |\mathbf{A}|^{-\nu/2} |\mathbf{W}|^{(\nu-d-1)/2} \exp \left\{ -\frac{1}{2} \text{tr}(\mathbf{A}^{-1} \mathbf{W}) \right\} \quad (20.54)$$

with normalizing constant

$$C(\nu, d) = \left\{ 2^{\nu d/2} \pi^{d(d-1)/4} \prod_{i=1}^d \Gamma \left( \frac{\nu + 1 - i}{2} \right) \right\}^{-1}.$$

The argument  $\mathbf{W}$  is a nonsingular covariance matrix. The expected value is  $E(\mathbf{W}) = \nu \mathbf{A}$ . In the univariate case ( $d = 1$ ), the Wishart distribution is a gamma distribution.

If  $\mathbf{W}$  is  $\text{Wishart}(\nu, \mathbf{A})$  distributed, then the distribution of  $\mathbf{W}^{-1}$  is called the inverse Wishart distribution with parameters  $\nu$  and  $\mathbf{A}^{-1}$  and denoted  $\text{Inv-Wishart}(\nu, \mathbf{A}^{-1})$ .

Let  $\mathbf{Y} = (\mathbf{Y}_1, \dots, \mathbf{Y}_n)$  denote the data. To derive the full conditional for the precision matrix  $\boldsymbol{\Sigma}^{-1}$ , assume that  $\boldsymbol{\mu}$  is known. We know from (7.15) that the likelihood is

$$f(\mathbf{Y} | \boldsymbol{\Sigma}^{-1}) = \prod_{i=1}^n \left[ \frac{1}{(2\pi)^{d/2} |\boldsymbol{\Sigma}|^{1/2}} \exp \left\{ -\frac{1}{2} (\mathbf{Y}_i - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1} (\mathbf{Y}_i - \boldsymbol{\mu}) \right\} \right].$$

After some simplification,

$$f(\mathbf{Y} | \boldsymbol{\Sigma}^{-1}) \propto |\boldsymbol{\Sigma}^{-1}|^{n/2} \exp \left\{ -\frac{1}{2} \sum_{i=1}^n (\mathbf{Y}_i - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1} (\mathbf{Y}_i - \boldsymbol{\mu}) \right\}.$$

Define

$$\mathbf{S} = \sum_{i=1}^n (\mathbf{Y}_i - \boldsymbol{\mu})(\mathbf{Y}_i - \boldsymbol{\mu})^\top.$$

Next

$$\sum_{i=1}^n (\mathbf{Y}_i - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1} (\mathbf{Y}_i - \boldsymbol{\mu}) = \text{tr} \left\{ \sum_{i=1}^n (\mathbf{Y}_i - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1} (\mathbf{Y}_i - \boldsymbol{\mu}) \right\} = \text{tr}(\boldsymbol{\Sigma}^{-1} \mathbf{S}). \quad (20.55)$$

The first equality in (20.55) is the trivial result that a scalar is also a  $1 \times 1$  matrix and equal to its trace. The second equality uses the result that  $\text{tr}(\mathbf{AB}) = \text{tr}(\mathbf{BA})$  for any matrices  $\mathbf{B}$  and  $\mathbf{A}$  such that the products  $\mathbf{BA}$  and  $\mathbf{AB}$  are defined. It follows that

$$f(\mathbf{Y} | \boldsymbol{\Sigma}^{-1}) \propto |\boldsymbol{\Sigma}^{-1}|^{n/2} \exp \left\{ -\frac{1}{2} \text{tr}(\boldsymbol{\Sigma}^{-1} \mathbf{S}) \right\}. \quad (20.56)$$

Suppose that the prior on the precision matrix  $\boldsymbol{\Sigma}^{-1}$  is  $\text{Wishart}(\nu_0, \boldsymbol{\Sigma}_0^{-1})$ . Then the prior density is

$$\pi(\boldsymbol{\Sigma}^{-1}) \propto |\boldsymbol{\Sigma}^{-1}|^{(\nu_0 - d - 1)/2} \exp\left\{-\frac{1}{2}\text{tr}(\boldsymbol{\Sigma}^{-1}\boldsymbol{\Sigma}_0)\right\}. \quad (20.57)$$

Since the posterior density is proportional to the product of the prior density and the likelihood, it follows from (20.56) and (20.57) that the posterior density is

$$\pi(\boldsymbol{\Sigma}^{-1}|\mathbf{Y}) \propto |\boldsymbol{\Sigma}^{-1}|^{(n+\nu_0-d-1)/2} \exp\left[-\frac{1}{2}\text{tr}\{\boldsymbol{\Sigma}^{-1}(\mathbf{S} + \boldsymbol{\Sigma}_0)\}\right]. \quad (20.58)$$

Therefore, the posterior distribution of  $\boldsymbol{\Sigma}^{-1}$  is Wishart $\{n + \nu_0, (\mathbf{S} + \boldsymbol{\Sigma}_0)^{-1}\}$ . The posterior expectation is

$$E(\boldsymbol{\Sigma}^{-1}|\mathbf{Y}) = (n + \nu_0) \{(\mathbf{S} + \boldsymbol{\Sigma}_0)^{-1}\}. \quad (20.59)$$

If  $\nu_0$  and  $\boldsymbol{\Sigma}_0$  are both small, then

$$E(\boldsymbol{\Sigma}^{-1}|\mathbf{Y}) \approx n\mathbf{S}^{-1} \quad (20.60)$$

The MLE of  $\boldsymbol{\Sigma}$  is  $n^{-1}\mathbf{S}$ , so the MLE of  $\boldsymbol{\Sigma}^{-1}$  is  $n\mathbf{S}^{-1}$ . Therefore, for small values of  $\nu_0$  and  $\boldsymbol{\Sigma}_0$ , the Bayesian estimator of  $\boldsymbol{\Sigma}^{-1}$  is close to the MLE.

The full conditional for  $\boldsymbol{\Sigma}^{-1}$  can be combined with a model for  $\boldsymbol{\mu}$  to estimate both parameters. For application to asset returns, a hierarchical prior for  $\boldsymbol{\mu}$  such as in Example 20.12 might be used. In either case, an MCMC analysis would be straightforward.

### 20.9.2 Estimating a Multivariate-*t* Scale Matrix

The Wishart distribution is not a conjugate prior for the scale matrix of a multivariate *t*-distribution, but it can be used as the prior nonetheless, since MCMC does not require the use of conjugate priors.

*Example 20.13. Estimating the correlation matrix of the CRSPday data*

In Example 7.4, the correlation matrix of the CRSPday returns data was estimated by maximum likelihood. In this example, the MLE will be compared to a Bayes estimate and the two estimates will be found to be very similar. The BUGS program used in this example is

```
model{
  for(i in 1:N)
  {
    y[i,1:m] ~ dmt(mu[], tau[,], df_likelihood)
  }
  mu[1:m] ~ dmt(mu0[], Prec_mu[,], df_prior)
  tau[1:m,1:m] ~ dwish(Prec_tau[,], df_wishart)
  lambda[1:m,1:m] <- inverse(tau[,])
}
```

In the BUGS program, `mu` is the mean vector, `tau` is the precision matrix, `lambda` is the scale matrix of the returns. Also, `dmt` is the multivariate-*t* distribution, and `dwish` is the Wishart distribution.

At the time of this writing (Dec 2014), JAGS does not have a sampler that will sample the posterior from this model. Therefore, WinBUGS will be used and will be called using the `bugs()` function in the R2WinBUGS package. The R code is below.

```

1 library(R2WinBUGS)
2 library(MASS) # need to mvrnorm
3 library(MCMCpack) # need for rwish
4 library(mnormt)
5 data(CRSPday, package = "Ecdat")
6 y = CRSPday[,4:7]
7 N = dim(y)[1]
8 m = dim(y)[2]
9 mu0 = rep(0,m)
10 Prec_mu = diag(rep(1, m)) / 10000
11 Prec_tau = diag(rep(1, m)) / 10000
12 df_wishart = 6
13 df_likelihood = 6
14 df_prior = 6
15 data = list(y = y, N = N, Prec_mu = Prec_mu,
16   Prec_tau = Prec_tau,
17   mu0 = mu0, m = m, df_likelihood = df_likelihood,
18   df_prior = df_prior, df_wishart = df_wishart)
19 inits_t_CRSP = function(){list(mu = mvrnorm(1, mu0,
20   diag(rep(1, m) / 100)),
21   tau = rwish(6, diag(rep(1, m)) / 100))}
22 library(R2WinBUGS)
23 multi_t.sim = bugs(data, inits_t_CRSP ,
24   model.file = "multi_t_CRSP.bug",
25   parameters = c("mu", "tau"), n.chains = 3,
26   n.iter = 2200, n.burnin = 200, n.thin = 2,
27   program = "WinBUGS", bugs.seed = 13, codaPkg = FALSE)
28 print(multi_t.sim, digits = 2)
29 tauhat = multi_t.sim$mean$tau
30 lambdahat = solve(tauhat)
31 sdinv = diag(1/sqrt(diag(lambdahat)))
32 cor = sdinv %*% lambdahat %*% sdinv
33 print(cor,digits=4)
```

The data list that is an input to the BUGS program contain `y` which is the matrix of returns, `df_likelihood` which is the degrees of freedom of the *t*-distribution in the likelihood, `mu0` which is the prior mean for `mu`, `df_prior` which is the degrees of freedom in the *t* prior on `mu`, and `df_wishart` which is the degrees of freedom of the Wishart prior on `tau`.

Ideally, `df_likelihood` should be an unknown parameter, but WinBUGS does not allow this parameter to be estimated. Instead, we fix it at the MLE

(rounded to 6) computed in Example 7.4. The need to fix this parameter at the MLE is due to limitations of WinBUGS and could, with considerably more effort, be circumvented by programming the MCMC in R or another language rather than using WinBUGS.

Note that `codaPkg = FALSE` was specified in the call to `bugs()`; this was not necessary since it is the default. When `codaPkg = FALSE` then `bugs()` returns an object of class `bugs` which cannot be used directly by functions in the `coda` package since these functions take objects of class `mcmc.list`. However, the function `as.mcmc.list()` will convert a `bugs` object to an `mcmc.list` object.

There were three chains, each of length 2000 after a burn-in of 200 and thinned to every second iteration. Thus, the total sample size was 3000 after thinning. The convergence to the stationary distribution and mixing were both quite rapid.  $N_{\text{eff}}$  was at least 1500 and  $R$  essentially 1 for all parameters, which indicate adequate burn-in and chain lengths.

```
> print(multi_t.sim, digits = 2)
Inference for Bugs model at "multi_t_CRSR.bug", fit using WinBUGS,
 3 chains, each with 2200 iterations (first 200 discarded), n.thin = 2
n.sims = 3000 iterations saved
      mean     sd    2.5%   25%   50%   75% 97.5% Rhat n.eff
mu[1]    0     0     0     0     0     0     0     0     1   3000
mu[2]    0     0     0     0     0     0     0     0     1   3000
mu[3]    0     0     0     0     0     0     0     0     1   3000
mu[4]    0     0     0     0     0     0     0     0     1   1800
tau[1,1] 14706  473  13780 14390 14690 15020 15630    1   3000

(edited to save space)

tau[4,4] 65197 2102 61180 63738 65190 66600 69360    1   3000
deviance -69858   61 -69980 -69900 -69860 -69820 -69730    1   3000

For each parameter, n.eff is a crude measure of effective sample size,
and Rhat is the potential scale reduction factor (at convergence, Rhat=1).

DIC info (using the rule, pD = Dbar-Dhat)
pD = 13.8 and DIC = -69843.9
DIC is an estimate of expected predictive error (lower deviance is better).
```

Since  $\mu$  is close to zero, `multi_t.sim` needed to be printed again, this time with more digits than 2:

|       | mean    | sd      | 2.5%     | 25%     | 50%     | 75%     | 97.5%   | Rhat | n.eff |
|-------|---------|---------|----------|---------|---------|---------|---------|------|-------|
| mu[1] | 9.4e-04 | 2.4e-04 | 4.6e-04  | 7.7e-04 | 9.4e-04 | 1.1e-03 | 1.4e-03 | 1    | 3000  |
| mu[2] | 4.4e-04 | 2.9e-04 | -1.3e-04 | 2.4e-04 | 4.6e-04 | 6.4e-04 | 1.0e-03 | 1    | 3000  |
| mu[3] | 6.9e-04 | 2.3e-04 | 2.3e-04  | 5.3e-04 | 6.9e-04 | 8.4e-04 | 1.1e-03 | 1    | 3000  |
| mu[4] | 7.7e-04 | 1.3e-04 | 5.1e-04  | 6.8e-04 | 7.7e-04 | 8.6e-04 | 1.0e-03 | 1    | 1800  |

The Bayes estimate of the precision matrix was converted to a correlation matrix at lines 29–33 of the R program. The estimated correlation matrix is below.

```
[,1]   [,2]   [,3]   [,4]
[1,] 1.0000 0.3191 0.2841 0.6756
[2,] 0.3191 1.0000 0.1586 0.4696
[3,] 0.2841 0.1586 1.0000 0.4300
[4,] 0.6756 0.4696 0.4300 1.0000
```

In Example 7.4, the MLE of the correlation matrix was found to be

```
$cor
 [,1]   [,2]   [,3]   [,4]
[1,] 1.0000 0.3192 0.2845 0.6765
[2,] 0.3192 1.0000 0.1584 0.4698
[3,] 0.2845 0.1584 1.0000 0.4301
[4,] 0.6765 0.4698 0.4301 1.0000
```

Notice the similarity between the Bayes estimate and the MLE.  $\square$

### 20.9.3 Non-Wishart Priors for the Covariate Matrix

We saw in Example 20.13 that a Wishart prior with noninformative choices of the prior parameters more or less replicates maximum likelihood estimation. Often, however, one wishes to shrink the covariance matrix toward some target, perhaps a estimate from a factor model. See Example 20.16.

## 20.10 Stochastic Volatility Models

Stochastic volatility models are an alternative to GARCH models for modeling conditional heteroscedasticity. In the ARIMA/GARCH models of Chap. 14, there was a single white noise process that drove both the conditional mean and the conditional variance. In contrast, stochastic volatility models use one white noise process to drive the conditional expectation and another to drive the conditional variance. Therefore, stochastic volatility models are more challenging to fit because the unobserved volatility process is driven by its own white noise process, which, of course, is also unobserved; see (20.63) below. Bayesian analysis is particularly good at dealing with unobserved variables and seems the best way to meet the challenge of fitting stochastic volatility models.

We will illustrate stochastic volatility models with the model

$$Y_t = \mu + \sum_{j=1}^k \beta_j X_{j,t} + a_t, \quad (20.61)$$

where  $Y_t$  is an observed process, e.g., the returns on an asset, and  $X_{j,t}$ ,  $j = 1, \dots, k$ , is the  $j$ th covariate at time  $t$  and could be a lagged value of  $Y_t$ . Also,  $a_t$  is a weak white noise process with conditional heteroscedasticity. Specifically,

$$a_t = \sqrt{h_t} \epsilon_t \quad (20.62)$$

where  $\log(h_t)$  follows the ARMA(p,q) process

$$\log(h_t) = \beta_0 + \sum_{j=1}^p \phi_j \log(h_{t-j}) + \sum_{j=1}^q \theta_j v_{t-j} + v_t, \quad (20.63)$$

$\epsilon_t$  and  $v_t$  are mutually independent iid white noises, and  $\text{Var}(\epsilon_t) = 1$ . Notice from (20.62) that  $\sqrt{h_t}$  is the conditional standard deviation of  $Y_t$ . As mentioned above, none of the variables in (20.63) are observable.

*Example 20.14. Fitting an ARMA(1,1) stochastic volatility model to the S&P 500 stock returns*

As an illustration, model (20.61)–(20.63) will be fit to daily S&P 500 log returns from January 2011 through October 2014. Model (20.61) will be used with no covariates so that  $Y_t = \mu + a_t$ . An ARMA(1,1) stochastic volatility model will be used so that  $\log(h_t) = \beta_0 + \phi \log(h_{t-1}) + \theta v_{t-1} + v_t$ . In (20.62)  $\epsilon_t$  has a  $t$ -distribution.

A BUGS program to fit the stochastic volatility model is below.

```

1 model
2 {
3   for (i in 1:N)
4   {
5     y[i] ~ dt(mu, tau[i], nu)
6   }
7   logh[1] ~ dnorm(0, 1.0E-6)
8   for(i in 2:N)
9   {
10    logh[i] ~ dnorm(beta0 + phi * logh[i-1] + theta * v[i-1], tau_v)
11    v[i] <- logh[i] - beta0 + phi * logh[i-1] + theta * v[i-1]
12  }
13  for (i in 1:N)
14  {
15    tau[i] <- exp(-logh[i])
16    h[i] <- 1/tau[i]
17  }
18  mu ~ dnorm(0.0, 1)
19  beta0 ~ dnorm(0, 0.0001)
20  phi ~ dnorm(0.4, 0.0001)
21  theta ~ dnorm(0, 0.0001)
22  tau_v ~ dgamma(0.01, 0.01)
23  v[1] ~ dnorm(0, 0.001)
24  nu ~ dunif(1,30)
25  sigma_v <- 1 / sqrt(tau_v)
26 }
```

Line 5 specifies the likelihood conditional on  $h_t$ . Line 7 gives a prior for the  $h_1$ . Lines 8–12 specify model (20.63) starting at  $t = 2$  with  $p = q = 1$ ; line 23 gives  $v_1$  a noninformative prior to start this recursion. Lines 18–25 specify diffuse priors on  $\mu$ ,  $\beta_0$ ,  $\phi$ ,  $\theta$ , and  $\sigma_v$ .

S&P 500 prices from Jan 3, 2011 to Oct 31, 2014 are in the file `S&P500_new.csv`. There are 964 log returns starting on Jan 4, 2011. The following R code computes the log returns and fit the stochastic volatility model to the log returns. The MCMC took about 10 minutes.

```

1 library(rjags)
2 dat = read.csv("S&P500_new.csv")
3 prices = dat$Adj.Close
4 y = diff(log(prices))
5 ##### get initial estimates #####
6 N = length(y)
7 logy2 = log(y^2)
8 fitar = lm(logy2[2:N] ~ logy2[1:(N - 1)])
9 beta0Init = as.numeric(fitar$coef[1])
10 phiInit = as.numeric(fitar$coef[2])
11 sfitar = summary(fitar)
12 tauInit = 1/sfitar$sigma^2
13 ##### Set up for MCMC #####
14 N = length(y)
15 data = list(y = y, N = N)
16 inits_stochVol_ARMA11 = function(){list(mu = rnorm(1, mean = mean(y),
17 sd = sd(y) / sqrt(N)), logh = log(y^2),
18 beta0 = runif(1, beta0Init * 1.5, beta0Init/1.5),
19 phi = runif(1,phiInit / 1.5, phiInit * 1.5),
20 tau_v = runif(1, tauInit / 1.5, tauInit * 1.5),
21 theta = runif(1, -0.5, 0.5))}
22 stochVol_ARMA11 <- jags.model("stochVol_ARMA11.bug", data = data,
23 inits = inits_stochVol_ARMA11,
24 n.chains = 3, n.adapt = 1000, quiet = FALSE)
25 nthin = 20
26 stochVol_ARMA.coda = coda.samples(stochVol_ARMA11, c("mu", "beta0",
27 "phi", "theta", "tau_v", "nu", "tau"), 100 * nthin, thin = nthin)
28 summ_stochVol_ARMA11 = summary(stochVol_ARMA.coda)
29 head(summ_stochVol_ARMA11[[1]], 8)
30 tail(summ_stochVol_ARMA11[[1]], 8)
31 dic.stochVol_ARMA11 = dic.samples(stochVol_ARMA11, 100 * nthin,
32 thin = nthin, type = "pD")
33 dic.stochVol_ARMA11

```

Lines 8–12 compute rough initial values for  $\beta_0$ ,  $\phi$ , and  $\sigma_v$  by using  $Y_t^2$  a proxy for  $h_t$  and regressing  $\log(Y_t^2)$  on  $\log(Y_{t-1}^2)$ .

Since nearly 1000 variables are monitored, we do not want to look at the output for all of them. Instead, the MCMC output is summarized for  $\beta_0$ ,  $\mu$ ,  $\phi$ ,  $\theta$ ,  $\tau_v$ , and the first and last few  $h_i$ .

```

> head(summ_stochVol_ARMA11[[1]], 8)
      Mean          SD   Naive SE Time-series SE
beta0 -1.327118e+01 6.986557e+00 4.033691e-01 2.956974e+00
mu     8.977085e-04 2.290611e-04 1.322485e-05 1.197846e-05
nu     2.116160e+01 5.785812e+00 3.340440e-01 3.884508e-01
phi    3.651321e-05 2.254858e-02 1.301843e-03 8.364831e-03
tau[1] 2.379568e+05 4.180067e+05 2.413363e+04 2.413332e+04
tau[2] 6.815322e+04 5.764209e+04 3.327968e+03 3.933972e+03
tau[3] 6.472217e+04 5.089951e+04 2.938685e+03 3.247533e+03

```

```

tau[4] 6.030660e+04 4.190914e+04 2.419625e+03 2.815956e+03
> tail(summ_stochVol_ARMA11[[1]], 8)
      Mean          SD   Naive SE Time-series SE
tau[959] 1.328635e+04 7.236487e+03 4.177988e+02 4.432135e+02
tau[960] 1.468882e+04 8.124815e+03 4.690864e+02 5.062925e+02
tau[961] 1.318162e+04 7.091475e+03 4.094265e+02 3.959164e+02
tau[962] 1.514622e+04 8.918848e+03 5.149299e+02 5.163099e+02
tau[963] 1.574039e+04 1.065051e+04 6.149072e+02 6.165450e+02
tau[964] 1.403511e+04 8.598674e+03 4.964447e+02 5.021540e+02
tau_v    5.161879e+00 1.409690e+00 8.138847e-02 1.765265e-01
theta    4.819691e-01 1.219201e-02 7.039058e-04 3.262414e-03

> dic.stochVol_ARMA11
Mean deviance: -6653
penalty 119.6
Penalized deviance: -6533

```

DIC and  $pD$  are called the “penalized deviance” and the “penalty” in the output of `dic.samples()` and in this example are  $-6536$  and  $127.3$ , respectively.  $\square$

## 20.11 Fitting GARCH Models with MCMC

Like stochastic volatility models, GARCH models are easy to fit by MCMC. One reason for fitting a GARCH model with BUGS is that this model can then be compared using DIC with a stochastic volatility model that is also fit using BUGS.

The following BUGS program fits a GARCH(1,1) model with  $t$ -distributed noise. This program runs under JAGS but crashes with no useful error messages under OpenBUGS and WinBUGS. The model is

$$\begin{aligned}
y_t &= \mu + a_t \\
a_t &= \sqrt{h_t} \epsilon_t \\
h_t &= \alpha_0 + \alpha_1 a_{t-1}^2 + \beta_1 h_{t-1} \\
\epsilon_t &\sim t_\nu(0, 1).
\end{aligned}$$

```

1 model{
2 for (t in 1:N)
3 {
4   y[t] ~ dt(mu, tau[t], nu)
5   a[t] <- y[t] - mu
6   tau[t] <- 1/h[t]
7 }
8 for (t in 2:N)
9 {
10   h[t] <- alpha0 + alpha1 * pow(a[t-1], 2) + beta1 * h[t-1]
}

```

```

11  }
12 mu ~ dnorm(0, 0.001)
13 h[1] ~ dunif(0, 0.0012)
14 alpha0 ~ dunif(0, 0.2)
15 alpha1 ~ dunif(0.00001, 0.8)
16 beta0 ~ dunif(0.00001, 0.8)
17 nu ~ dunif(1,30)
18 }
```

*Example 20.15. Fitting a GARCH(1,1) model to the S&P 500 stock returns*

The following R code fits a GARCH(1,1) model to the data in Example 20.14.

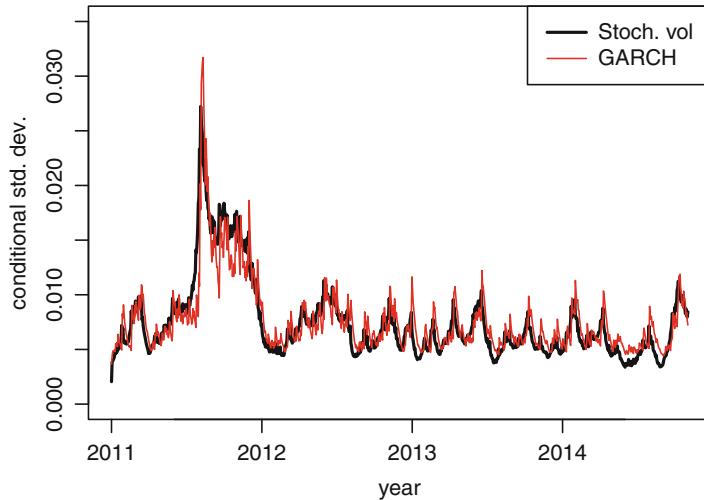
```

1 library(rjags)
2 dat = read.csv("S&P500_new.csv")
3 prices = dat$Adj.Close
4 y = diff(log(prices))
5 N = length(y)
6 data = list(y = y, N = N)
7 inits_garch11 = function(){list(alpha0 = runif(1, 0.001, 0.25),
8     beta1 = runif(1, 0.001, 0.25), mu = runif(1, 0.001, 0.25),
9     alpha1 = runif(1, 0.001, 0.25), nu = runif(1, 2, 10))}
10 garch11 <- jags.model("garch11.bug", data=data,
11     inits = inits_garch11,
12     n.chains = 3, n.adapt = 1000, quiet = FALSE)
13 nthin = 20
14 garch11.coda = coda.samples(garch11,c("mu", "beta1", "alpha0",
15     "alpha1", "nu", "tau"), 100*nthin, thin = nthin)
16 dic.garch11 = dic.samples(garch11, 100*nthin, thin = nthin)
17 dic.garch11
18 diffdic(dic.garch11, dic.stochVol_ARMA11)
19 summ_garch11 = summary(garch11.coda)
20 head(summ_garch11[[1]])
21 tail(summ_garch11[[1]])
```

The output is below.

```

> dic.garch11
Mean deviance: -6508
penalty 5.737
Penalized deviance: -6502
> diffdic(dic.garch11, dic.stochVol_ARMA11)
Difference: 30.90573
Sample standard error: 15.12843
> head(summ_garch11[[1]])
      Mean          SD    Naive SE Time-series SE
alpha0 3.875413e-06 9.405668e-07 5.430365e-08 6.060383e-08
alpha1 1.287614e-01 2.170856e-02 1.253344e-03 1.465383e-03
beta1 7.677071e-01 2.621658e-02 1.513615e-03 1.870831e-03
mu     8.663882e-04 2.290809e-04 1.322599e-05 1.247927e-05
```



**Fig. 20.8.** The conditional standard deviations of the log returns estimated by the stochastic volatility model in Example 20.14 and the GARCH(1,1) model.

```

nu      7.570399e+00 1.990408e+00 1.149163e-01   1.077795e-01
tau[1] 7.432585e+04 1.242142e+05 7.171511e+03   7.376238e+03
> tail(summ_garch11[[1]])
      Mean       SD Naive SE Time-series SE
tau[959] 10765.65 1050.613 60.65714      56.26545
tau[960] 12495.45 1207.416 69.71019      64.84224
tau[961] 15145.85 1500.491 86.63091      81.57595
tau[962] 14266.32 1339.490 77.33549      72.44932
tau[963] 17144.03 1653.541 95.46724      90.31518
tau[964] 19112.62 1869.545 107.93821     107.54300

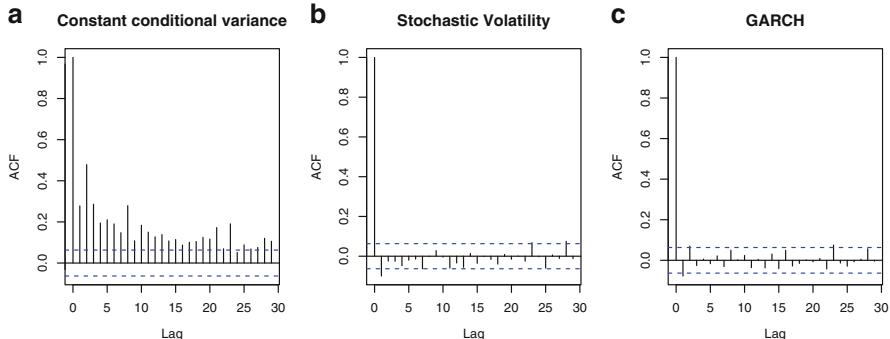
```

Line 18 of the R code uses the function `diffdic()` in the `rjags` package to compare the stochastic volatility and GARCH models by DIC. In the output from `diffdic()`, we see that DIC for the GARCH model is larger than for the stochastic volatility model; the difference between the two DIC values is 30.9 but with a standard error of 15.1. Figure 20.8 compares the estimates of the conditional standard deviations of the log returns by the two models. Figure 20.9 compares the ACF's of the squared standardized residuals<sup>4</sup> from the two models and also assuming that the log returns are i.i.d. Despite the large difference in DIC, the two models are about equally successful in modeling the conditional heteroscedasticity.

□

---

<sup>4</sup> A residual is standardized by dividing it by its conditional standard deviation.



**Fig. 20.9.** ACF's of the squared standardized residuals. (a) Assuming a constant conditional standard deviation. The model is  $R_t = \mu + \epsilon_t$  where  $\epsilon_t$  is independent white noise. (b) Assuming an ARMA(1,1) stochastic volatility model. (c) Assuming a GARCH(1,1) model.

## 20.12 Fitting a Factor Model

Factor models can be fit easily using JAGS. An advantage of a Bayesian analysis of a factor model is that a hierarchical model can be used to shrink the betas towards each other.

*Example 20.16. Fitting a one factor model to stock returns*

In this example, the factor will be the returns on the S&P 500 so this is a Bayesian version of the CAPM. The model that will be used is

$$R_{j,t} = \beta_j R_{M,t} + \epsilon_{j,t}.$$

where, as in Chap. 17,  $R_{j,t}$  is the return on the  $j$  stock at time  $t$ ,  $j = 1, \dots, 10$ , and  $R_{M,t}$  is the return on the S&P 500 at time  $t$ . It is assumed that for  $j = 1, \dots, 10$ ,  $\{\epsilon_{j,t}, t = 1, \dots\}$  are mutually independent i.i.d. white noise processes with  $\text{var}(\epsilon_{j,t}) = \sigma_{\epsilon,j}^2$ . For simplicity we have assumed that all the alphas are zero.

We will put a hierarchical on  $\beta_1, \dots, \beta_{10}$ , specifically

$$\beta_j \sim N(\mu_\beta, \sigma_\beta^2),$$

with non-informative priors on  $\mu_\beta$  and  $\sigma_\beta^2$ .

The BUGS program is below. At line 11,  $\mu_\beta$  is called `meanbeta` and  $\sigma_\beta^{-2}$  is called `taubeta`. Also, `tauepsilon` on line 6 is  $\sigma_\epsilon^{-2}$ .

```

1 model{
2   for (t in 1:N)
3   {
4     for (j in 1:m)
```

```

5   {
6     R[t,j] ~ dnorm(beta[j]*mkt[t], tauepsilon[j])
7   }
8 }
9 for (j in 1:m)
10 {
11   beta[j] ~ dnorm(meansbeta, taubeta)
12   tauy[j] ~ dgamma(0.1, 0.001)
13 }
14 meansbeta ~ dnorm(1, 0.000001)
15 taubeta ~ dunif(1, 100)
16 }

```

This example is a continuation of Example 16.11 in that it uses the stock price data in the file `Stock_Bond.csv`. Since there are nearly 5000 days of returns, one can create 20 blocks of returns, each block with 250 days except that the last block would be somewhat short of 250. Each block can be used for training (parameter estimation) and the next block for testing.

The R program below illustrates this strategy using only the first two blocks, the first as training data and the second as test data. During training, the optimal allocation vector  $w$  is estimated. Here “optimal” means maximizing the expected utility of the returns using the utility function

$$U(R; \lambda) = 1 - \exp\{-\lambda(1 + R)\}. \quad (20.64)$$

The value of  $\lambda$  is set equal to 3 at line 38 of the R code. A value of  $\lambda$  in the range 2 to 8 is reasonable since these are daily returns and typically in the range  $\pm 0.05$ . Also, in (20.64) we are implicitly assuming that the initial wealth is equal to 1, since the initial wealth is not explicitly included there.

The first part of the R program fits the factor model:

```

1 library(rjags)
2 dat = read.csv("Stock_Bond.csv")
3 y = dat[, c(3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 22)]
4 n = dim(y)[1]
5 m = dim(y)[2] - 1
6 r = y[-1, ] / y[-n, ] - 1
7 k1 = 250
8 k2 = k1 + 250
9 rtrain = r[1:k1, 1:m]
10 mkt_train = r[1:k1, 11]
11 rtest = r[(k1+1):k2, 1:m]
12 data = list(R = rtrain, N = k1, mkt = mkt_train, m = m)
13 inits.Capm = function(){list(beta = rep(1,m))}
14 Capm.jags <- jags.model("BayesCapm.bug.R", data = data,
15   inits = inits.Capm, n.chains = 1, n.adapt = 1000, quiet = FALSE)
16 nthin = 10
17 N = 500
18 Capm.coda = coda.samples(Capm.jags,

```

```

19   c("beta", "tauepsilon", "taubeta"),
20   N * nthin, thin = nthin)
21 MCMC_out = Capm.coda[[1]]
22 summ = as.matrix(summary(Capm.coda)[[1]][,1])
23 beta = summ[1:10]
24 taubeta = summ[11]
25 tauy = summ[12:21]
26 sigmaepsilon = tauepsilon^(-.5)

```

The next section of R code defines the utility function and finds the optimal allocation vector  $w$  by using quadratic programming as in Example 16.11. The vector  $w$  is found twice, once using the sample mean vector and covariance matrix of the training sample returns to estimate  $\mu$  and  $\Sigma$  in Eq. (16.20) (model-free) and once using the factor model to estimate  $\mu$  and  $\Sigma$  (CAPM).

```

27 ExUtil = function(w)
28 {
29   -1 + exp(-lambda * (1 + t(w) %*% mu) +
30   lambda^2 * t(w) %*% Omega %*% w / 2 )
31 }
32
33 mu_model_free = colMeans(rtrain)
34 Omega_model_free = cov(rtrain)
35 mu_Capm = beta * mean(mkt)
36 Omega_Capm = beta %o% beta * var(mkt_train) + diag(sigmaepsilon^2)
37
38 lambda = 3
39 library(quadprog)
40 mu = mu_model_free
41 Omega = Omega_model_free
42 opt1 = solve.QP(Dmat = as.matrix(lambda^2 * Omega),
43   dvec = lambda * mu, Amat = as.matrix(rep(1,10)),
44   bvec = 1, meq = 1)
45 w_model_free = opt1$solution
46
47 mu = mu_Capm
48 Omega = Omega_Capm
49 opt2 = solve.QP(Dmat = as.matrix(lambda^2 * Omega),
50   dvec = lambda * mu, Amat = as.matrix(rep(1,10)),
51   bvec = 1, meq = 1)
52 w_Capm = opt2$solution

```

Next, the utility of the portfolio's returns is averaged over the test data using the model-free and CAPM based estimates of the optimal portfolio. Also, the mean and standard deviations of the portfolio's returns on the test data are computed.

```

53 return_model_free = as.matrix(rtest) %*% w_model_free
54 ExUt_model_free = mean(1 - exp(-lambda * return_model_free))
55

```

```

56 return_Capm = as.matrix(rtest) %*% w_Capm
57 ExUt_Capm = mean(1 - exp(-lambda * return_Capm))
58
59 print(c(ExUt_model_free, ExUt_Capm), digits = 2)
60 print(c(mean(return_model_free), mean(return_Capm)), digits = 2)
61 print(c(sd(return_model_free), sd(return_Capm)), digits = 2)

```

The output is below. We see that the portfolio selected using the CAPM estimates outperformed the portfolio based on the sample mean vector and covariance matrix. Interestingly, the CAPM selected portfolio not only has a higher average utility, but it also has both a higher mean and a lower standard deviation of the returns compared to the model-free estimates. Usually a higher expected return comes with higher risk (larger standard deviation), but a better estimate can achieve a higher return without higher risk.

```

62 > print(c(ExUt_model_free, ExUt_Capm), digits = 2)
63 [1] -0.0179  0.0023
64 > print(c(mean(return_model_free), mean(return_Capm)), digits = 2)
65 [1] 0.00060  0.00099
66 > print(c(sd(return_model_free), sd(return_Capm)), digits = 2)
67 [1] 0.067  0.012

```

These results are based on only one block of training data and one block of test data, and by themselves they are not a convincing demonstration of the superiority of CAPM estimates. The analysis could be continued using all twenty blocks of data; see Exercise 4.

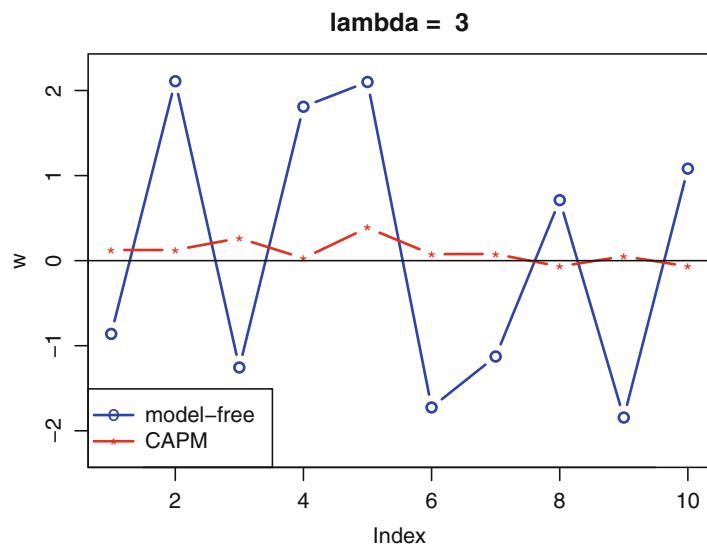
Figure 20.10 plots the allocation vectors,  $\mathbf{w}$ , using model-free and CAPM-based estimators. The model-free  $\mathbf{w}$  oscillates widely and has substantial short-selling. In contrast, the CAPM-based  $\mathbf{w}$  is much closer to assigning equal weights to the 10 stocks and has minimal short selling.

Another issue is how hierarchical Bayes estimation of the CAPM compares with ordinary least-squares estimation. Figure 20.11 plots the least-squares estimates of  $\beta$  versus the Bayes estimates. With the exception of the largest beta, the Bayes estimates are only slightly shrunk together and are similar to the least-squares estimates. This suggest that the Bayes estimates will, at best, be only a moderate improvement over least-squares in this example. Most of the improvement is due to using the CAPM to estimate the expected returns.  $\square$

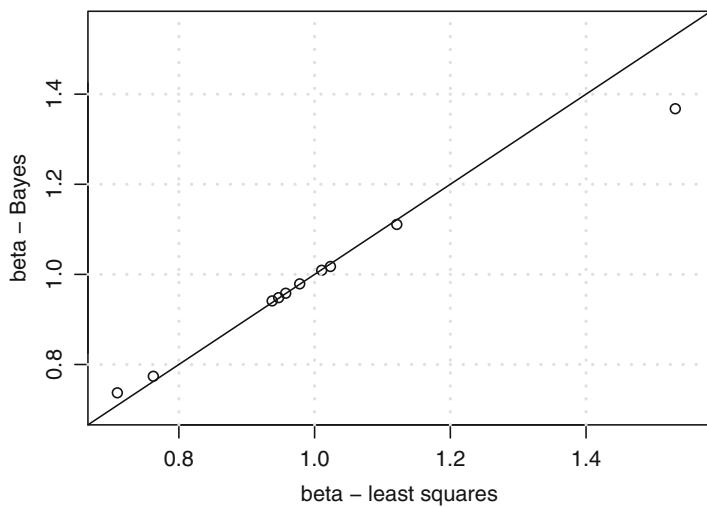
## 20.13 Sampling a Stationary Process

This section provides the theory behind the statistics  $B$ ,  $W$ , and  $\widehat{\text{var}}^+(\psi|\mathbf{Y})$  used in Sect. 20.7.5 to monitor MCMC convergence and mixing.

Suppose that  $Y_1, Y_2, \dots, Y_n$  is a sample from a stationary process with mean  $\mu$  and autocovariance function  $\gamma(h)$ . Let  $\bar{Y} = n^{-1} \sum_{i=1}^n Y_i$  be the sample mean. Then



**Fig. 20.10.** Allocation (or weight) vectors using the sample mean vector and covariance matrix (model-free) and CAPM-based estimates.



**Fig. 20.11.** Plot of the Bayes estimator of  $\beta$  versus the least-squares estimator.

$$\begin{aligned}
\text{var}(\bar{Y}) &= n^{-2} \sum_{i=1}^n \sum_{j=1}^n \text{Cov}(Y_i, Y_j) \\
&= n^{-2} \sum_{i=1}^n \sum_{j=1}^n \gamma(i-j) \\
&= n^{-2} \left\{ n\gamma(0) + 2 \sum_{h=1}^{n-1} \gamma(h)(n-h) \right\} \\
&= \frac{\gamma(0)}{n} R_n,
\end{aligned} \tag{20.65}$$

where  $R_n = \left\{ 1 + 2 \sum_{h=1}^{n-1} \rho(h) \left( 1 - \frac{h}{n} \right) \right\}$ . If  $Y_1, Y_2, \dots, Y_n$  is an uncorrelated process (white noise), then  $R_n = 1$  and (20.65) agrees with (7.13).

Most stationary processes generated by MCMC have  $\rho(h) \geq 0$  for all  $h$  so that  $R_n$  is inflated by the autocorrelation. The inflation can be severe. Consider the case of a stationary AR(1) process,  $Y_n = \phi Y_{n-1} + \epsilon_i$ . AR(1) processes often are reasonably good approximations to MCMC processes. For an AR(1) process we can approximate  $R_n$ :

$$R_n \approx \left\{ 1 + 2 \sum_{h=1}^{\infty} \rho(h) \right\} = \left\{ 2 \sum_{h=0}^{\infty} \phi^h - 1 \right\} = \left( \frac{2}{1-\phi} - 1 \right) = \frac{1+\phi}{1-\phi}, \tag{20.66}$$

where we have used summation formula for geometric series (3.4) with  $T = \infty$ . Notice that the right-hand side of (20.66) increases without bound as  $\phi \rightarrow 1$ .

From the identity

$$\sum_{i=1}^n (Y_i - \mu)^2 = \sum_{i=1}^n (Y_i - \bar{Y})^2 + n(\bar{Y} - \mu)^2,$$

we obtain

$$E \left\{ \sum_{i=1}^n (Y_i - \bar{Y})^2 \right\} = \gamma(0)(n - R_n) \tag{20.67}$$

since  $\gamma(0) = E \{(Y_i - \mu)^2\}$  and  $\gamma(0)R_n = E \{n(\bar{Y} - \mu)^2\}$  by definitions. Therefore, an unbiased estimate of the process variance  $\gamma(0)$  is

$$\hat{\gamma}(0) = \frac{\sum_{i=1}^n (Y_i - \bar{Y})^2}{n - R_n}. \tag{20.68}$$

When the process is uncorrelated so that  $R_n = 1$ , the right-hand side of (20.68) is the sample variance (A.7). For positively autocorrelated processes,  $R_n > 1$  and the sample variance (which uses 1 in place of  $R_n$ ) is biased downward.

To obtain an unbiased estimate of  $\gamma(0)$ , one can use

$$\frac{\sum_{i=1}^n (Y_i - \bar{Y})^2 + \widehat{\gamma(0)R_n}}{n}, \tag{20.69}$$

where  $\widehat{\gamma(0)R_n}$  is an unbiased estimator of  $\gamma(0)R_n$ . There are several methods for estimating  $\gamma(0)R_n$ . The simplest uses several independent realizations of the process. Let  $\bar{Y}_1, \dots, \bar{Y}_M$  be the means of  $M$  independent realizations of the process and let  $\bar{Y} = M^{-1} \sum_{j=1}^M \bar{Y}_j$ . Then

$$\widehat{\gamma(0)R_n} = \frac{\sum_{j=1}^M (\bar{Y}_j - \bar{Y})^2}{M - 1} \quad (20.70)$$

is an unbiased estimator of  $\gamma(0)R_n$ . The statistic  $\widehat{\text{var}}^+(\psi|\mathbf{Y})$  used in Sect. 20.7.5 for MCMC monitoring is a special case of (20.68) and (20.70).

## 20.14 Bibliographic Notes

There are many excellent books on Bayesian statistics. Gelman, Carlin, Stern, Dunson, Vehtari, and Rubin (2013) and Carlin and Louis (2008) are introductions to Bayesian statistics written at about the same mathematical level as this book. Box and Tiao (1973) is a classic work on Bayesian statistics with a wealth of examples and still worth reading despite its age. Berger (1985) is a standard reference on Bayesian analysis and decision theory. Bernardo and Smith (1994) and Robert (2007) are more recent books on Bayesian theory. Rachev, Hsu, Bagasheva, and Fabozzi (2008) covers many applications of Bayesian statistics to finance.

Albert (2007) is an excellent introduction to Bayesian computations in R. Chib and Greenberg (1995) explain how the Metropolis–Hastings algorithm works and why its stationary distribution is the posterior. Congdon (2001, 2003) covers the more recent developments in Bayesian computing with an emphasis on OpenBUGS software. There are other Bayesian Monte Carlo samplers besides MCMC, for example, importance sampling. Robert and Casella (2005) discuss these as well as MCMC. Gelman et al., (2013) have examples of Bayesian computations in R and OpenBUGS in an appendix. Lunn, Thomas, Best, and Spiegelhalter (2000) describe the design of OpenBUGS. Lunn, Jackson, Best, and Spiegelhalter (2013) is a comprehensive introduction to BUGS.

The diagnostics  $\hat{R}$  and  $N_{\text{eff}}$  are due to Gelman and Rubin (1992) though Sect. 20.7.5 uses the somewhat different notation of Gelman et al., (2013). Spiegelhalter, Best, Carlin, and van der Linde (2002) proposed DIC and  $p_D$ . The Gelman plot was introduced by Brooks and Gelman (1998). Kass et al. (1998) discuss their practical experiences with MCMC.

Bayesian modeling of yield curves models is discussed by Chib and Ergashev (2009). Bayesian time series are discussed by Albert and Chib (1993), Chib and Greenberg (1994), and Kim, Shephard, and Chib (1998); the first two papers cover ARMA process and the last covers ARCH and stochastic volatility models. There is a vast literature on the important and difficult problem of Bayesian estimation of covariance matrices with nonconjugate priors. Daniels and Kass (1999) review some of the literature in addition to providing their own suggestions.

We have not discussed empirical Bayes inference, but Carlin and Louis (2000) can be consulted for an introduction to that literature. Empirical Bayes inference uses a hierarchical prior but estimates the parameters in the lower level in a non-Bayesian manner and then, treating those parameter as known and fixed, performs a Bayesian analysis. The result is shrinkage estimation much like that achieved by a Bayesian analysis. The advantage of an empirical Bayes analysis is that it can be somewhat simpler than a fully Bayesian analysis. The disadvantage is that it underestimates uncertainty because estimated parameters in the prior are treated as if they were known. There are shrinkage estimators that are not exactly Bayesian or even empirical Bayes procedures. Ledoit and Wolf (2003) propose a shrinkage estimator for the covariance matrix of stock returns. Their shrinkage target is an estimate from a factor model, for example, the CAPM. Shrinkage estimation goes back at least to Stein (1956) and is often called Stein estimation.

The central limit theorem for the posterior is discussed by Gelman et al. (2013), Lehmann (1983), and van der Vaart (1998), in increasing order of technical level.

See Greyserman, Jones, and Strawderman (2006) for more information on portfolio selection by Bayesian methods.

## 20.15 R Lab

### 20.15.1 Fitting a *t*-Distribution by MCMC

In this section of the lab, you will fit the *t*-distribution to monthly returns on IBM using JAGS to estimate the posterior distribution by MCMC sampling.

Run the following R code to load the `rjags` package, input the data, and prepare the data for use by JAGS.

```
library(rjags)
data(CRSPmon, package = "Ecdat")
ibm = CRSPmon[, 2]
r = ibm
N = length(r)
ibm_data = list(r = r, N = N)
```

Next, put the following BUGS code in a text file. I will assume that you name this file `univt.bug`, though you can use another name provided you make appropriate changes in the R code that follows. BUGS code is somewhat similar to, but not the same as, R code. For example, in R “`dt`” is the *t*-density, but in BUGS it is the *t*-distribution.

```
model{
for (t in 1:N)
{
```

```

r[t] ~ dt(mu, tau, k)
}
mu ~ dnorm(0.0, 1.0E-6)
tau ~ dgamma(0.1, 0.01)
nu ~ dunif(2, 50)
sigma2 <- (k / (k - 2)) / tau
sigma <- sqrt(sigma2)
}

```

BUGS programs are difficult to debug, so be careful to enter the code exactly as it appears here. It has been tested and runs as written, but any error will cause problems. Our experience is that JAGS is better at providing error messages and easier to debug than WinBUGS and OpenBUGS.

The BUGS code above provides a description of the statistical model and specifies the prior distributions. The model states that the data are i.i.d. from a  $t$ -distribution. The  $\sim$  symbol assigns a distribution to a random variable so  $y[i] \sim dt(mu, tau, k)$  gives the likelihood of the data. Here `mu`, `tau`, and `k` are the mean, precision, and degrees of freedom, respectively, of the  $t$ -distribution. For a  $t$ -distribution, the precision is  $\tau = 1/\lambda^2$  where  $\lambda$  is the scale parameter. Also, `mu ~ dnorm(0.0, 1.0E-6)` specifies the prior for the mean `mu` to be normal with mean 0 and precision 1.0E-6. The precision of a normal distribution is the reciprocal of its variance, so here the prior variance of `mu` is 1.0E6.

The symbol `<-` is used to assign a value (rather than a distribution) to a variable. Thus, `sigma <- 1/sqrt(tau)` makes `sigma` the scale parameter of the  $t$ -distribution of the data. In R, “`=`” can be used in place of “`<-`” for assigning a value to a variable, but this is not true in BUGS. The parameter `sigma` is not needed, but, by defining this variable in the BUGS program, we generate a sample from its posterior distribution.

Next, run the following R code that defines a function `inits()`. This function is used to generate random starting values for the chains.

```

inits = function(){list(mu = rnorm(1, 0, 0.3),
                      tau = runif(1, 1, 10), k = runif(1, 1, 30))}

```

The next code uses the `jags.model()` and `coda.samples()` functions in the `rjags` package. Notice that the arguments specify the data, the function to create initial values of the chains, the file containing the BUGS program, the parameters to be monitored and returned, the number of chains, the number of iterations per chain, the number of iterations to discard as burn-in, the amount of thinning.

```

univ_t <- jags.model("univt.bug", data = ibm_data,
                      inits = inits, n.chains = 3, n.adapt = 1000, quiet = FALSE)
nthin = 2
univ_t.coda = coda.samples(univ_t, c("mu", "tau", "k",
   "sigma"), n.iter = 500 * nthin, thin = nthin)

```

Next, print and plot the results.

```
summary(univ_t.coda)
effectiveSize(univ_t.coda)
gelman.diag(univ_t.coda)
```

### Problem 1

- (a) Which parameter mixes best according to  $N_{\text{eff}}$  in the output?
- (b) Which parameter mixes worst according to  $N_{\text{eff}}$  in the output?
- (c) Give a 95 % posterior interval for the degrees-of-freedom parameter.

Next, plot the results to check for convergence to the stationary distribution (posterior distribution) using Gelman plots and trace plots.

```
gelman.plot(univ_t.coda)
par(mfrow = c(2, 2))
traceplot(univ_t.coda)
```

Plotting the ACFs gives much insight into how well the chains are mixing. The less autocorrelation, the better. The function `autocorr.plot()` plots ACFs separately for each chain.

```
par(mfrow = c(2, 2))
autocorr.plot(univ_t.coda, auto.layout = FALSE)
```

### Problem 2

- (a) Which parameter mixes best and which mixes worse according to the ACF plots? Explain your answers.
- (b) Find the posterior skewness and kurtosis of the degrees of freedom parameter.

The function `densityplot()` gives kernel density estimate from each chain.

```
library(lattice)
densityplot(univ_t.coda)
```

### Problem 3 Which posterior densities are most skewed?

The kurtosis of a  $t$ -distribution is  $3(\nu - 2)/(\nu - 4)$  if  $\nu > 4$  and is  $+\infty$  if  $\nu \leq 4$ . Variables in R can have infinite values: `Inf` is  $+\infty$  and `-Inf` is  $-\infty$ , so R can handle infinite values of kurtosis if they occur.

**Problem 4** Write R code to compute 1500 MCMC values of the kurtosis. ( $1500 = 3 \times 2000/2$ ; there are 3 chains of length 2000 after burn-in and they are thinned to every 2nd iteration.)

- (a) Find the 0.01, 0.05, 0.25, 0.5, 0.75, 0.95, and 0.99 quantiles of the posterior distribution of the kurtosis of IBM returns.
- (b) Estimate the posterior probability that the kurtosis of the distribution of IBM returns is finite.
- (c) Compute the 0.01, 0.05, 0.25, 0.5, 0.75, 0.95, and 0.99 quantiles of the bootstrap distribution of the sample kurtosis of IBM. Take 1000 resamples using both a model-free and a model-based bootstrap. Compare the two sets of bootstrap quantiles with the posterior quantiles in (a).
- (d) Compare 90% bootstrap basic percentile confidence intervals for the kurtosis with the 90% posterior interval. Which interval is shortest? Why might it be shortest?

### 20.15.2 AR Models

In this section of the lab, you will fit an AR(1) model to the changes in the log of the GDP. First, run the following code to process the data. Notice that the log-GDP time series is differenced before fitting.

```

1 library(rjags)
2 data(Tbrate, package = "Ecdat")
3 # r = the 91-day treasury bill rate
4 # y = the log of real GDP
5 # pi = the inflation rate
6 del_dat = diff(Tbrate)
7 y = del_dat[,2]
8 N = length(y)
9 GDP_data=list(y = y, N = N)

```

Next create a file called `ar1.bug` containing the following WinBUGS code.

```

1 model{
2 for(i in 2:N){
3   y[i] ~ dnorm(mu + phi * (y[i-1] - mu), tau)
4 }
5 mu ~ dnorm(0, 0.00001)
6 phi ~ dnorm(0, 0.00001)
7 tau ~ dgamma(0.1, 0.0001)
8 sigma <- 1/sqrt(tau)
9 }

```

Finally, run the following code to fit an AR(1) model using JAGS and also using R's `arima()` function to compute the MLE, which will be compared with the Bayes estimator.

```

1 inits = function(){list(mu = rnorm(1, 0, 2 * sd(y) / sqrt(N)),
2   phi = rnorm(1, 0, 0.3), tau = runif(1, 1, 10))}
3 ar1 <- jags.model("ar1.bug", data = GDP_data, inits = inits,
4   n.chains = 3, n.adapt = 1000, quiet = FALSE)
5 nthin = 20
6 ar1.coda = coda.samples(ar1, c("mu", "phi", "sigma"),
7   n.iter = 500 * nthin, thin = nthin)
8 summary(ar1.coda, digits = 3)
9 arima(y, order = c(1, 0, 0))

```

**Problem 5** Construct time series and ACF plots of the parameters  $\phi$  and  $\sigma$ .

- (a) Do you believe that the MCMC sample size is adequate? Why or why not? Is the burn-in iterations adequate? Why or why not? If you feel that either the number of iterations or the length of the burn-in period is inadequate, then rerun with a larger burn-in period and/or MCMC sample size.
- (b) Compute the MLEs for this model using `arima()`. How closely do the Bayes estimates and MLEs agree? Could you explain any possible disagreement?
- (c) The model in the BUGS program does not assume that the time series is in its stationary distribution. In fact, the model does not even assume that there is a stationary distribution. Explain why.
- (d) Modify the BUGS program to utilize the marginal distribution of  $y_1$ , assuming that the process starts in its stationary distribution.

### 20.15.3 MA Models

Next you will fit an MA(1) to simulated data. The function `arima.sim()` is used to create the data.

```

1 library(rjags)
2 set.seed(5640)
3 N = 600
4 y = arima.sim(n = N, list(ma = -0.5), sd = 0.4)
5 y = as.numeric(y) + 3
6 q = 5
7 ma.sim_data = list(y = y, N = N, q = q)
8 inits.ma = function(){list(mu = rnorm(1, mean(y), 2*sd(y)/sqrt(N)),
9   theta = rnorm(1, -0.05, 0.1), tau = runif(1, 5, 8))}
```

Put the following BUGS program in the file `ma1.bug`. This program not only fits the MA(1) model but also predicts  $q$  steps ahead;  $q$  is an input parameter chosen by the user and, from the viewpoint of BUGS,  $q$  is part of the data and is set equal to 5 in the code above. The predicted values will be included in the output and called `ypred`.

```

1 model{
2   for (i in 2:N)
3   {
4     w[i] <- y[i] - mu - theta * w[i-1]
5   }
6   w[1] ~ dnorm(0, 0.01)
7   for (i in 2:N)
8   {
9     y[i] ~ dnorm(mu + theta * w[i-1], tau)
10  }
11  mu ~ dnorm(0, 0.0001)
12  theta ~ dnorm(0, 0.0001)
13  tau ~ dgamma(0.01, 0.0001)
14  sigma <- 1/sqrt(tau)
15  for (i in 1:q)
16  {
17    ypred[i] ~ dnorm(theta * w[N + i - 1], tau)
18    w[i + N] <- ypred[i] - theta * w[N + i - 1]
19  }
20 }
```

Now run this R code.

```

1 ma1 <- jags.model("ma1.bug", data = ma.sim_data, inits = inits.ma,
2   n.chains = 3, n.adapt = 1000, quiet = FALSE)
3 nthin = 5
4 ma1.coda = coda.samples(ma1, c("mu", "theta", "sigma", "ypred"),
5   n.iter = 500 * nthin, thin = nthin)
6 summary(ma1.coda)
```

### Problem 6

- (a) Do you believe that the MCMC sample size is adequate? Why or why not?  
If you feel it is inadequate, than rerun JAGS with a larger MCMC sample size. Is the length of the burn-in periods adequate?
- (b) Construct time series and ACF plots of the parameters `theta`, `sigma`, `ypred[1]`, and `ypred[2]`. What do the plots tell us about MCMC mixing and convergence?
- (c) Find a 90 % posterior interval for the next observation after the observed data.

#### 20.15.4 ARMA Models

Create a simulated sample from an ARMA(1,1) process with the following R code.

```

set.seed(5640)
N = 600
```

```
y = arima.sim(n = N, list(ar = 0.9, ma = -0.5), sd = 0.4)
y = as.numeric(y)
```

**Problem 7** Create BUGS and R code to fit the ARMA(1,1) model to the simulated data. Monitor the result to make certain that the MCMC sample size is large enough.

- (a) Discuss how well the chains mix and whether the Monte Carlo sample size is adequate.
- (b) Find 99 % posterior intervals for the AR and MA parameters.

## 20.16 Exercises

1. Show in Example 20.2 that the MAP estimator is 6/7.
2. Verify (20.26).
3. In the derivation of (20.51), it was stated that “ $\{\bar{Y} - E(\mu|\mathbf{Y})\}$  and  $\{E(\mu|\mathbf{Y}) - \mu\}$  are conditionally uncorrelated given  $\mathbf{Y}$ .” Verify this statement.
4. Continue the analysis in Example 20.16. Divide the data into 20 blocks of 250 days each, except that the last block will have only 212 days. Use each of the first 19 blocks as training data with the subsequent block as test data. How does portfolio selection based on the CAPM compare with model-free estimation when averaged over the 19 pairs of training and test data sets?
5. One of the strength of fitting models by MCMC using BUGS is that a very wide range of models can be fit. As an example, in this exercise a regression model with MA(1) errors will be used. For data, use the first 1500 returns<sup>5</sup> on GM and on the S&P 500 index in the data set `Stock.Bond.csv`. Fit the model

$$R_t = \beta R_{M,t} + \epsilon_t + \theta \epsilon_{t-1}.$$

where  $R_t$  is the  $t$ th return on GM,  $R_{M,t}$  is the  $t$ th return on the S&P 500, and  $\epsilon_1, \dots, \epsilon_{1500}$  are i.i.d.  $N(0, \sigma_\epsilon^2)$ . Use non-informative priors on  $\beta$ ,  $\theta$ , and  $\sigma_\epsilon^2$ .

6. Expand the model in Exercise 5 so that  $\epsilon_1, \dots$  is a GARCH(1,1) process. Revise the BUGS and R code of Exercise 5 to fit this expanded model.
7. So far we have treated the sample mean vector and covariance matrix as fixed when considering the risk of a portfolio. Stated differently, estimation risk has been ignored. A methodology for taking risk due to estimation error into account was proposed by Greyserman, Jones, and Strawderman (2006). Assume that the vector of returns  $\mathbf{R}$  is  $N(\boldsymbol{\mu}, \boldsymbol{\Sigma})$  distributed. Let  $(\boldsymbol{\mu}^{(k)}, \boldsymbol{\Sigma}^{(k)})$ ,  $k = 1, \dots, K$ , be an MCMC sample from the posterior distribution of  $(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ . For each  $k$ , let  $R^{(k)}$  be  $N(\boldsymbol{\mu}^{(k)}, \boldsymbol{\Sigma}^{(k)})$  distributed. Then

---

<sup>5</sup> JAGS had trouble when the full data set was used, probably because there are nearly 5000 latent variables. This problem is likely hardware dependent.

$\mathbf{R}^{(1)}, \dots, \mathbf{R}^{(K)}$  is a sample from the posterior predictive distribution of  $\mathbf{R}$  and take uncertainty about  $\boldsymbol{\mu}$  and  $\boldsymbol{\Sigma}$  into account and

$$K^{-1} \sum_{k=1}^K U\{X_0(1 + \mathbf{w}^\top \mathbf{R}^{(k)})\} \quad (20.71)$$

estimates the expected utility if the allocation vector is  $\mathbf{w}$ . Here, as in Sect. 16.8,  $X_0$  is the initial wealth and  $U$  is the utility function. Continue the analysis in Example 20.16 using the CAPM model and maximize (20.71). Maximizing (20.71) is a nonlinear optimization problem, so a good starting value is essential. As a starting value, use the  $\mathbf{w}$  found in Example 20.16 that ignores estimation error.

## References

- Albert, J. (2007) *Bayesian Computation with R*, Springer, New York.
- Albert, J. H. and Chib, S. (1993) Bayes inference via Gibbs sampling of autoregressive time series subject to Markov mean and variance shifts, *Journal of Business & Economic Statistics*, 11, 1–15.
- Berger, J. O. (1985) *Statistical Decision Theory and Bayesian Analysis* 2nd ed., Springer-Verlag, Berlin.
- Bernardo, J. M., and Smith, A. F. M. (1994) *Bayesian Theory*, Wiley, Chichester.
- Box, G. E. P., and Tiao, G. C. (1973) *Bayesian Inference in Statistical Analysis*, Addison-Wesley, Reading, MA.
- Brooks, S. P. and Gelman, A. (1998) General Methods for Monitoring Convergence of Iterative Simulations. *Journal of Computational and Graphical Statistics*, 7, 434–455.
- Carlin, B. P., and Louis, T. A. (2000) Empirical Bayes: Past, present and future. *Journal of the American Statistical Association*, 95, 1286–1289.
- Carlin, B. , and Louis, T. A. (2008) *Bayesian Methods for Data Analysis*, 3rd ed., Chapman & Hall, New York.
- Chib, S., and Ergashev, B. (2009) Analysis of multifactor affine yield curve models. *Journal of the American Statistical Association*, 104, 1324–1337.
- Chib, S., and Greenberg, E. (1994) Bayes inference in regression models with ARMA( $p, q$ ) errors. *Journal of Econometrics*, 64, 183–206.
- Chib, S., and Greenberg, E. (1995) Understanding the Metropolis–Hastings algorithm. *American Statistician*, 49, 327–335.
- Congdon, P. (2001) *Bayesian Statistical Modelling*, Wiley, Chichester.
- Congdon, P. (2003) *Applied Bayesian Modelling*, Wiley, Chichester.
- Daniels, M. J., and Kass, R. E. (1999) Nonconjugate Bayesian estimation of covariance matrices and its use in hierarchical models. *Journal of the American Statistical Association*, 94, 1254–1263.

- Edwards, W. (1982) Conservatism in human information processing. In *Judgement Under Uncertainty: Heuristics and Biases*, D. Kahneman, P. Slovic, and A. Tversky, ed., Cambridge University Press, New York.
- Gelman, A., and Rubin, D. B. (1992) Inference from iterative simulation using multiple sequence (with discussion). *Statistical Science*, **7**, 457–511.
- Gelman, A., Carlin, J. B., Stern, H. S., Dunson, D. B., Vehtari, A., and Rubin, D. B. (2013) *Bayesian Data Analysis*, 3rd ed., Chapman & Hall, London.
- Greyserman, A., Jones, D. H., and Strawderman, W. E. (2006) Portfolio selection using hierarchical Bayesian analysis and MCMC methods, *Journal of Banking and Finance*, **30**, 669–678.
- Kass, R. E., Carlin, B. P., Gelman, A., and Neal, R. (1998) Markov chain Monte Carlo in practice: A roundtable discussion. *American Statistician*, **52**, 93–100.
- Kim, S., Shephard, N., and Chib, S. (1998) Stochastic volatility: likelihood inference and comparison with ARCH models. *Review of Economic Studies*, **65**, 361–393.
- Ledoit, O., and Wolf, M. (2003) Improved estimation of the covariance matrix of stock returns with an application to portfolio selection. *Journal of Empirical Finance*, **10**, 603–621.
- Lehmann, E. L. (1983) *Theory of Point Estimation*, Wiley, New York.
- Lunn, D., Jackson, C., Best, N., Thomas, A., and Spiegelhalter, D. (2013) *The BUGS Book*, Chapman & Hall.
- Lunn, D. J., Thomas, A., Best, N., and Spiegelhalter, D. (2000) OpenBUGS—A Bayesian modelling framework: Concepts, structure, and extensibility. *Statistics and Computing*, **10**, 325–337.
- Rachev, S. T., Hsu, J. S. J., Bagasheva, B. S., and Fabozzi, F. J. (2008) *Bayesian Methods in Finance*, Wiley, Hoboken, NJ.
- Robert, C. P. (2007) *The Bayesian Choice: From Decision-Theoretic Foundations to Computational Implementation*, 2nd ed., Springer, New York.
- Robert, C. P., and Casella, G. (2005) *Monte Carlo Statistical Methods*, 2nd ed., Springer, New York.
- Spiegelhalter, D. J., Best, N. G., Carlin, B. P., and van der Linde, A. (2002) Bayesian measures of model complexity and fit. *Journal of the Royal Statistical Society, Series B, Methodological*, **64**, 583–616.
- Stein, C. (1956) Inadmissibility of the usual estimator for the mean of a multivariate normal distribution. In *Proceedings of the Third Berkeley Symposium on Mathematical and Statistical Probability*, J. Neyman, ed., University of California, Berkeley, pp. 197–206, Volume 1.
- van der Vaart, A. W. (1998) *Asymptotic Statistics*, Cambridge University Press, Cambridge.

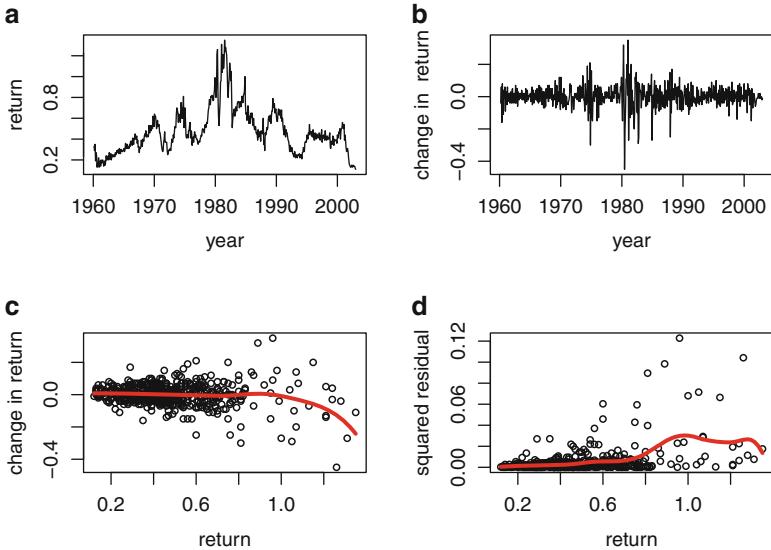
## Nonparametric Regression and Splines

### 21.1 Introduction

As discussed in Chap. 9, regression analysis estimates the conditional expectation of a response given predictor variables. The conditional expectation is called the regression function and is the best predictor of the response based upon the predictor variables, because it minimizes the expected squared prediction error.

There are three types of regression, linear, nonlinear parametric, and nonparametric. *Linear regression* assumes that the regression function is a linear function of the parameters and estimates the intercept and slopes (regression coefficients). *Nonlinear parametric regression*, which was discussed in Sect. 11.2, does not assume linearity but does assume that the regression function is of a *known* parametric form, for example, the Nelson-Siegel model. In this chapter, we study *nonparametric regression*, where the form of the regression function is also nonlinear but, unlike nonlinear parametric regression, not specified by a model but rather determined from the data. Nonparametric regression is used when we know, or suspect, that the regression function is curved, but we do not have a model for the curve.

There are many techniques for nonparametric regression, but local polynomial regression and splines are the most widely used, and only these will be discussed here. Local polynomial regression and splines generally work well and, since they usually give similar estimates, it is difficult to recommend one over the other. Local polynomial estimation might be somewhat simpler to understand. Splines are used in many areas of mathematics, such as, for interpolation, and so it is worthwhile to be familiar with them. Also, splines are useful as components in complex models. The R lab at the end of this chapter gives an example.



**Fig. 21.1.** Risk-free monthly returns. The returns are 1/12th the yearly rate. (a) Time series plot of the returns. (b) Time series plot of the changes in the returns. (c) Plot of changes in returns against lagged returns with a local linear estimate of the drift. (d) Plot of squared residuals against lagged returns with a local linear estimate of the squared diffusion coefficient.

Models for the evolution of short-term interest rates are important in finance, for example, because they are needed for the pricing of interest rate derivatives. Figure 21.1 contains plots of the monthly risk-free returns<sup>1</sup> in the `Capm` data set in R's `Ecdat` package. This data set has been used for various purposes in several previous chapters. Here we will use it to illustrate nonparametric regression. Panels (a) and (b) are time series plots of the returns and the changes in the returns.

A common model for changes in short-term interest rates is

$$\Delta r_t = \mu(r_{t-1}) + \sigma(r_{t-1})\epsilon_t, \quad (21.1)$$

where  $r_t$  is the rate at time  $t$ ,  $\Delta r_t = r_t - r_{t-1}$ ,  $\mu(\cdot)$  is the drift function,  $\sigma(\cdot)$  is the volatility function, also called the diffusion function, and  $\epsilon_t$  is  $N(0, 1)$  noise. Many different parametric models have been proposed for  $\mu(\cdot)$  and  $\sigma(\cdot)$ , for example, by Merton (1973), Vasicek (1977), Cox, Ingersoll, and Ross (1985), Yau and Kohn (2003), and Chan et al. (1992). The simplest model, due to Merton (1973), is that  $\mu(\cdot)$  and  $\sigma(\cdot)$  are constant. Chan et al. (1992) assume that  $\mu(r) = \beta(r - \alpha)$  and  $\sigma(r) = \theta r^\gamma$ , where  $\alpha > 0$ ,  $\beta < 0$ ,  $\theta > 0$ , and  $\gamma$  are unknown parameters—this process reverts to a mean equal to  $\alpha$ . Chan et al.'s model was used as an example of nonlinear regression in Sect. 11.12.1.

<sup>1</sup> The risk-free rate is called the risk-free return in the `Capm` package.

The approach of Yau and Kohn (2003) that is used here is to model both  $\mu(\cdot)$  and  $\sigma(\cdot)$  nonparametrically. Doing this allows one to check which parametric models, if any, fit the data and to have a nonparametric alternative if none of the parametric models fits well.

The solid red curves in Fig. 21.1c and d are estimates of  $\mu(\cdot)$  and  $\sigma^2(\cdot)$  by a nonparametric regression method *local linear regression*, a special case of *local polynomial regression*. By (21.1),  $E(\Delta r_t) = \mu(r_{t-1})$  and  $\text{Var}(\Delta r_t) = \sigma^2(r_{t-1})$ , so  $\hat{\mu}(\cdot)$  is obtained by regressing  $\Delta r_t$  on  $r_{t-1}$  and  $\hat{\sigma}^2(\cdot)$  by regressing  $\{\Delta r_t - \hat{\mu}(r_{t-1})\}^2$  on  $r_{t-1}$ . The latter is an example of estimating a conditional variance; see Sect. 14.2.

The code to produce Fig. 21.1 is below. The local linear regression estimates were produced by the function `locpoly()` in the `KernSmooth` package. This function computes the fitted function on a grid of 401 equally spaced points; this is sufficient for plotting the fitted function as in lines 21 and 24 but not for computing the residuals. Instead, to compute squared residuals at line 11, the `spline()` function is used at line 10 to interpolate the fit from the 401-point grid to the values of the explanatory variable.

```

1 library(Ecdat)
2 library(KernSmooth)
3 data(Capm)
4 attach(Capm)
5 n = length(rf)
6 year = seq(1960.125, 2003, length = n)
7 diffrrf = diff(Capm$rf)
8 rf_lag = rf[1:(n-1)]
9 ll_mu <- locpoly(rf_lag, diffrrf, bandwidth = dpill(rf_lag, diffrrf))
10 muhat = spline(ll_mu$x, ll_mu$y, xout = rf_lag)$y
11 epsilon_sqr = (diffrrf-muhat)^2
12 ll_sig <- locpoly(rf_lag, epsilon_sqr,
13   bandwidth = dpill(rf_lag, epsilon_sqr) )
14 pdf("riskfree01.pdf", width = 6, height = 5)
15 par(mfrow=c(2, 2))
16 plot(year, rf, ylab = "return", main = "(a)", type = "l" )
17 plot(year[2:n], diffrrf, ylab = "change in return", main = "(b)",
18   type = "l", xlab = "year")
19 plot(rf_lag, diffrrf, ylab = "change in return", xlab = "return",
20   main=""(c)",type="p",cex=.7)
21 lines(ll_mu$x, ll_mu$y, lwd = 4, col = "red")
22 plot(rf_lag, (diffrrf - muhat)^2, xlab = "return",
23   ylab = "squared residual", main = "(d)", cex = 0.7)
24 lines(ll_sig$x, ll_sig$y, lwd = 4, col = "red")
25 graphics.off()

```

## 21.2 Local Polynomial Regression

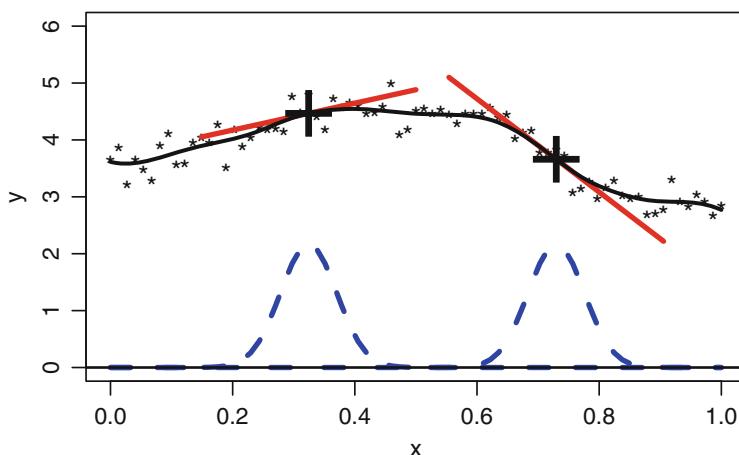
Local polynomial regression is based on the principle that a smooth function can be approximated locally by a low-degree polynomial. Suppose we have a sample  $(X_i, Y_i)$ ,  $i = 1, \dots, n$ , and  $E(Y|X = x) = \mu(x)$  for a smooth function  $\mu$ . The function  $\mu$  will be estimated on a grid of  $x$ -values,  $x_1, \dots, x_M$ . These could, but need not, be the same values  $X_1, \dots, X_n$ , as where we observe  $Y$ .

The estimation is done one point at a time on the grid  $x_1, \dots, x_M$ . To estimate  $\mu$  at  $x_\ell$ , one fits a  $p$ th-degree polynomial using only  $(X_i, Y_i)$  with  $X_i$  near  $x_\ell$ . This is done using weights determined by a kernel function  $K$ .  $K$  is a probability density function symmetric about 0 and such that  $K(x)$  decreases as  $|x|$  increases, for instance, a normal density with mean 0. We have seen kernels used for density estimation in Sect. 4.2.

The regression function at  $x_\ell$  is estimated by kernel-weighted least squares, which minimizes

$$\sum_{i=1}^n \left[ Y_i - \{ \beta_0 + \beta_1(X_i - x_\ell) + \dots + \beta_p(X_i - x_\ell)^p \} \right]^2 K\{(X_i - x_\ell)/h\} \quad (21.2)$$

and then  $\hat{\mu}(x_\ell) = \hat{\beta}_0$  since the regression model  $\beta_0 + \beta_1(x - x_\ell) + \dots + \beta_p(x - x_\ell)^p$  equals  $\beta_0$  at  $x = x_\ell$ . The weights  $K\{(X_i - x_\ell)/h\}$  decrease as  $|X_i - x_\ell|$  increases, so only the data near  $x_\ell$  are used. The parameter  $h$  is called the bandwidth and determines how much data are used for estimation; the larger the value of  $h$ , the more data used.



**Fig. 21.2.** Local linear fit (solid curve) to 75 data points (asterisks) with bandwidth chosen by the direct plug-in method. The regression function  $\mu$  is estimated at each of the 75 points and the estimates are connected to create the solid curve. Estimation at  $x_{25} = 0.32$  and  $x_{55} = 0.72$  is illustrated by the kernels (dashed curves), the linear fits (solid lines), and the fitted points (large +).

Local linear estimation, where  $p = 1$ , is illustrated in Fig. 21.2. The kernel functions are shown as blue dashed curves at two points,  $x_{25} = 0.32$  and  $x_{75} = 0.72$ . Above each kernel, the local linear fit is shown as a red line and the large “+” is placed at  $\{x, \hat{\mu}(x)\}$ . The black curve  $\hat{\mu}$  is obtained by finding local fits on a grid of 75  $x_\ell$ -values and plotting  $\{x_\ell, \hat{\mu}(x_\ell)\}$  for all  $x_\ell$  on this grid. For example, the curve in Fig. 21.2 used the R function `locpoly()` in R’s `KernSmooth` package and has a grid of 401 equally spaced  $x$ -values (the default). Often the grid is simply the observed  $X$ -values,  $X_1, \dots, X_n$ .

The bandwidth  $h$  is called a “smoothing parameter” because it determines the smoothness of  $\hat{\mu}$ . A larger value of  $h$  gives a smoother curve. The choice of  $h$  is important. If  $h$  is too large, then the polynomial approximation may be poor and the estimate of  $\mu(x)$  will be badly biased. Conversely, if  $h$  is too small, then too few data are used and the estimate of  $\mu$  will be too variable. A good choice of the bandwidth minimizes the mean squared error of the estimator, which is the variance plus the squared bias. Both the squared bias and variance of the estimator are unknown and must be estimated, or at least their sum must be estimated. Automatic bandwidth selection, which either directly or indirectly estimates and attempts to minimize the mean-squared error, has been an area of intense research and a number of data-based bandwidth selectors are available. The curve in Fig. 21.2 used the bandwidth chosen by the popular direct plug-in (dpi) bandwidth selector of Ruppert, Sheather and Wand (1995). The dpi selector estimates the mean integrated squared error (MISE) of  $\hat{\mu}$ , which is

$$E \left[ \int_{\min(X_i)}^{\max(X_i)} \{\mu(x) - \hat{\mu}(x)\}^2 dx \right], \quad (21.3)$$

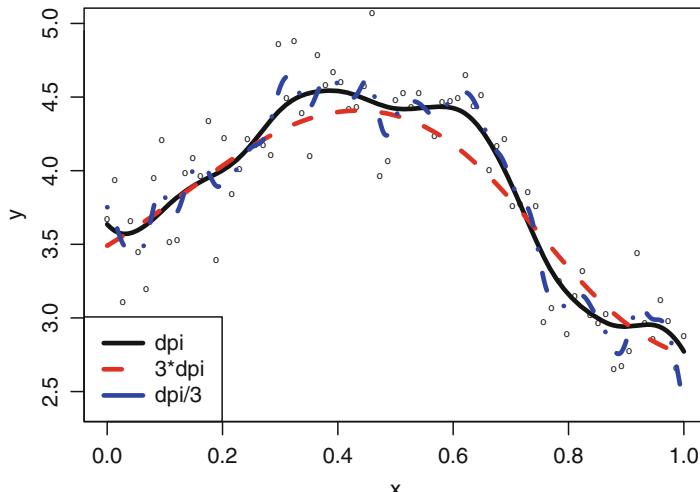
and finds the bandwidth that minimizes the estimated MISE.

Nonparametric regression estimators are also called *smoothers* because they smooth out the noise in the data. Using a bandwidth that is too small causes *overfitting*, which is *undersmoothing*. Conversely, a bandwidth that is too large will result in *underfitting*, which is *oversmoothing*—see Sect. 4.2 for further discussion of under- and oversmoothing in the context of kernel density estimation.

Figure 21.3 illustrates the effect of varying the bandwidth. The solid black curve uses the dpi bandwidth, the dashed red curve uses three times the dpi bandwidth, and the dotted-and-dashed blue curve uses one-third the dpi bandwidth. The dashed red curve is too smooth to follow the data closely, that is, it underfits, while the dotted-and-dashed blue curve is wiggly because it is tracking random noise in the data, that is, it overfits. In this example, the data were simulated, so the true regression function,  $\mu(x) = 3.6 + 0.1x + \sin(5x^{1.5})$ , is known and it is possible to calculate the average squared error,  $\sum_{i=1}^n \{\hat{\mu}(X_i) - \mu(X_i)\}^2$ , for each bandwidth. The average squared errors are 1.34 and 2.27 times larger using  $3*\text{dpi}$  and  $\text{dpi}/3$ , respectively, compared to using dpi.

Besides the dpi bandwidth selector, the bandwidth can also be chosen by minimizing either the AIC or GCV (generalized cross-validation) criterion. The definition of AIC for a parametric model uses the number of parameters in the model, but local polynomial estimation is not parametric, so one cannot count parameters. Nonetheless, it is possible to define the “effective number of parameters” and this is done in Sect. 21.3.1. GCV is defined in Sect. 21.3.2.

*Example 21.1. Local polynomial estimation of forward rates.*



**Fig. 21.3.** Local linear estimators with three bandwidths: dpi (direct plug-in), which gives an appropriate amount of smoothing; three times the dpi, which oversmooths (underfits); and one-third the dpi, which undersmooths (overfits). Simulated data.

The R code in this section computes two estimates of the forward rate function, a parametric estimate based on the Nelson-Siegel model and a non-parametric estimates using local polynomial smoothing. The program is split into several chunks of code with discussion between the chunks.

Line 1 reads in prices of STRIPS on Dec 31, 1995. This data set was used in Sect. 11.3. There are 29 1/4 years of quarterly maturities  $T$ , for a total of 117 (= 29.25 \* 4) prices.

The price of a zero-coupon bond as a percentage of par is  $\text{price} = 100 \exp \left\{ - \int_0^T f(s) ds \right\}$  so that  $-\log(\text{price}) = \int_0^T f(s) ds - \log(100)$ . Thus, the integrated forward rate, called `Int_F`, is defined on line 7 to be  $-\log(\text{price}) + \log(100)$ . Lines 4–7 use the `order()` function to order the maturities  $T$  from smallest to largest and to order the prices accordingly. Ordering the data by  $T$  is needed when the plotted points are connected by lines. If they were not ordered by  $T$ , then the plot could look like a spider web.

```

1 dat = read.table("strips_dec95.txt", header = T)
2 T = dat$T
3 n = length(T)
4 ord = order(T)
5 T = T[ord]
6 price = dat$price[ord]
7 Int_F = - log(price) + log(100)

```

In lines 8–9, the function `locfit()` in the `locfit` package estimates the first derivative of `Int_F` by local cubic fitting to produce an estimate of the forward rate (since `Int_F` estimates the integrated forward rate). Note that `deriv = 1` specifies estimation of the first derivative and `deg = 3` specifies using cubic polynomial fitting. The function `locfit()` is similar to `locpoly()` used in Sect. 21.1 to create the curves in Fig. 21.1. We could have used `locpoly()` again here but wanted to illustrate both local polynomial regression functions.

```

8 library(locfit)
9 fit_loc_Int_F = locfit(Int_F ~ T, deriv = 1, deg = 3)

```

The function `Nelson-Siegel()` in lines 10–18 returns a list containing the forward rate, called `f`, and the integrated forward rate, called `int_f`. Lines 19–24 estimate the parameters of the Nelson-Siegel model by fitting the Nelson-Siegel integrated forward rate to `Int_F`. On line 21, `Yhat` is the integrated forward rate because “[2]” is the second element of the list that is output by `NelsonSiegel()`.

```

10 NelsonSiegel = function(theta){
11   ##### f = forward rate and int_f = intergrated forward rate #####
12   f = theta[1] + (theta[2] + theta[3] * T) * exp(-theta[4] * T)
13   int_f = theta[1] * T - theta[2] / theta[4]
14   * (exp(-theta[4] * T) - 1) -
15   theta[3] * (T * exp(-theta[4] * T) / theta[4] +
16   (exp(-theta[4] * T) - 1) / theta[4]^2)
17   list("f" = f, "inf_t" = int_f)
18 }
19 fit_NS = optim(c(0.05, 0.001, 0.001, 0.08),
20 fn = function(theta){
21   Yhat = NelsonSiegel(theta)[[2]]
22   sum((Int_F - Yhat)^2)},
23   control = list(maxit=30000, reltol = 1e-10))
24 NS_yhat = NelsonSiegel(fit_NS$par)[[1]]

```

Figure 21.4 is produced by lines 25–36. Notice that the Nelson-Siegel fit (in blue) tends to overestimate the forward rate when  $5 < T < 15$  and  $T > 25$  and to underestimate when  $T < 3$  and  $15 < T < 25$ . In comparison, the local polynomial estimates show no bias.

```

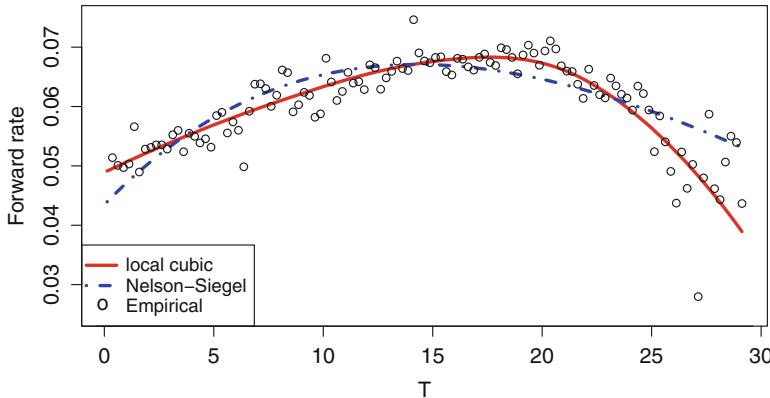
25 pdf("strips02.pdf", width=6, height=5)
26 par(mfrow=c(1,1))

```

```

27 plot(fit_loc_Int_F, ylim = c(.025,.075), ylab = "Forward rate",
28       col = "red", lwd = 3)
29 lines(fit_loc_Int_F, col = "red", lwd=3) # to widen to linewidth = 3
30 lines(T, NS_yhat, col = "blue", lwd = 3, lty = 4)
31 points(T[2:n], diff(Int_F) / diff(T))
32 legend("bottomleft", c("local cubic",
33   "Nelson-Siegel", "Empirical"),
34   lty=c(1, 4,NA),pch=c(NA, NA, "o"),
35   col=c("red", "blue", "black"), lwd = c(3, 3, NA))
36 graphics.off()

```



**Fig. 21.4.** Local cubic (nonparametric), Nelson-Siegel (parametric), and empirical estimates of the forward rate curve. Notice that the nonparametric estimates are much closer than the parametric estimates to the empirical estimates.

□

### 21.2.1 Lowess and Loess

Loess and its earlier version lowess are local polynomial smoothers with spatially varying bandwidths controlled by a parameter called *span*. Span is the fraction of the data used for estimation at each point. The bandwidth, call it  $h(x, \text{span})$ , for estimation at a point  $x$  is adjusted so that whenever  $\text{span} \leq 1$  then  $K\{(X_i - x)/h(x, \text{span})\}$  is nonzero for  $\text{span} \times 100\%$  of the  $X_i$ .

If  $\text{span} = 1$ , then all of the data are used for estimation at each point, but the data farthest from  $X_i$  get small weights. Because of these small weights, for small data sets, a lowess (or loess) smooth with a span of 1 might not be smooth enough. To solve this problem, span is defined for values greater than 1 by

$$h(x, \text{span}) = \text{span} \times h(x, 1).$$

As span increases beyond 1, the weights  $K\{(X_i - x)/h(x, \text{span})\}$  become more and more equal. As  $\text{span} \rightarrow \infty$ , the weights converge to a constant,  $K(0)$ , and the lowess (or loess) fit converges to a polynomial regression fit.

## 21.3 Linear Smoothers

Local polynomial regression as well as penalized spline regression—to be covered soon—are examples of linear smoothers. A linear smoother has an  $n \times n$  smoother matrix  $\mathbf{H}$ , which does not depend on  $\mathbf{Y}$ , such that

$$\hat{\mathbf{Y}} = \mathbf{H}\mathbf{Y}, \quad (21.4)$$

where  $\mathbf{Y} = (Y_1, \dots, Y_n)^\top$  is the vector of responses and  $\hat{\mathbf{Y}} = (\hat{Y}_1, \dots, \hat{Y}_n)^\top$  is the vector of fitted values. Equation (21.4) can be written as

$$\hat{Y}_i = \sum_{j=1}^n H_{ij} Y_j, \quad i = 1, \dots, n. \quad (21.5)$$

The smoother matrix will depend on a smoothing parameter, which for local polynomial regression is the bandwidth. We will let  $\lambda$  denote the smoothing parameter and denote the smoother matrix by  $\mathbf{H}(\lambda)$ . The smoother matrix is an analog of the hat matrix of linear regression and is, itself, often called a hat matrix.

### 21.3.1 The Smoother Matrix and the Effective Degrees of Freedom

In a parametric model, the number of parameters quantifies the ability of the model to fit the data. In nonparametric estimation, the potential to fit (and overfit) can be quantified by the *effective number of parameters* or the *effective degrees of freedom of the fit*. Conceptually, the effective number of parameters is similar to the Bayesian  $p_D$  in Sect. 20.7.6.

By (21.5), the hat diagonal  $H(\lambda)_{ii}$  gives the *leverage* or *self-influence* of the  $Y_i$  since it is the weight given to  $Y_i$  when calculating  $\hat{Y}_i$ . A large value of  $H(\lambda)_{ii}$  means a high potential for overfitting. The effective number of parameters is the sum of the leverages:

$$p_{\text{eff}} = \sum_{i=1}^n H(\lambda)_{ii} = \text{tr}\{\mathbf{H}(\lambda)\}. \quad (21.6)$$

If  $p_{\text{eff}}$  is too small (too large), then the data are underfit (overfit).

The residual mean sum of squares is

$$\sum_{i=1}^n (Y_i - \hat{Y}_i)^2 = \|\mathbf{Y} - \hat{\mathbf{Y}}\|^2 = \|\{\mathbf{I} - \mathbf{H}(\lambda)\}\mathbf{Y}\|^2, \quad (21.7)$$

where  $\mathbf{I}$  is the  $n \times n$  identity matrix. The noise variance is estimated by

$$\hat{\sigma}(\lambda)^2 = \frac{\|\{\mathbf{I} - \mathbf{H}(\lambda)\}\mathbf{Y}\|^2}{n - p_{\text{eff}}}, \quad (21.8)$$

which is a direct analog of (9.16).

### 21.3.2 AIC, CV, and GCV

For linear regression models, AIC is

$$\text{AIC} = n \log(\hat{\sigma}^2) + 2(1 + p),$$

where  $1 + p$  is the number of parameters (intercept plus  $p$  slopes). For a linear smoother, AIC uses  $p_{\text{eff}}$  in place of  $p + 1$ , so that

$$\text{AIC}(\lambda) = n \log\{\hat{\sigma}^2(\lambda)\} + 2p_{\text{eff}}.$$

We can then select  $\lambda$  by minimizing AIC.

The cross-validation or CV statistic is

$$\text{CV}(\lambda) = \sum_{i=1}^n \{Y_i - \hat{Y}_{-i}(\lambda)\}^2$$

where, to prevent overfitting,  $\hat{Y}_{-i}(\lambda)$  is the  $i$ th fitted value computed with the  $i$ th observation deleted. One, of course, should choose a  $\lambda$  with a small value of  $\text{CV}(\lambda)$ .

The generalized cross-validation statistic (GCV) is

$$\text{GCV}(\lambda) = \frac{\|\mathbf{Y} - \hat{\mathbf{Y}}(\lambda)\|^2}{(n - p_{\text{eff}})^2}. \quad (21.9)$$

$\text{GCV}(\lambda)$  can be computed more quickly than  $\text{CV}(\lambda)$  and  $\text{GCV}(\lambda)$  is a good approximation to  $\text{CV}(\lambda)/n^2$  so minimizing GCV is another way to choose  $\lambda$ .

AIC and GCV can both be computed very quickly and usually give essentially the same amount of smoothing. In fact, it has been shown theoretically that both criteria should give similar estimates. Therefore, it does not matter much which is used, but GCV is more commonly used than AIC in nonparametric regression.

## 21.4 Polynomial Splines

The use of polynomial splines in nonparametric regression, as well as many other areas of mathematics, is based on the same principle as local polynomial regression—a smooth function can be accurately approximated locally by a low-degree polynomial. A  $p$ th-degree polynomial spline is constructed by piecing together  $p$ th-degree polynomials, so that they join together at specified locations called *knots*. The polynomials are spliced together, so that the spline has  $p - 1$  continuous derivatives. The  $p$ th derivative of the spline is constant between knots and can jump at the knots.

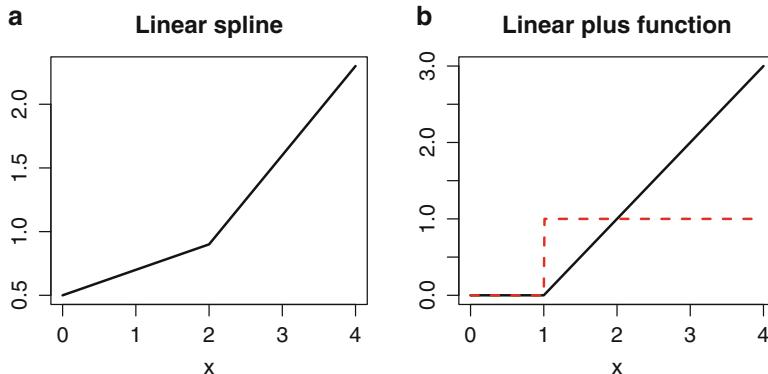
### 21.4.1 Linear Splines with One Knot

We start simple, a linear spline with one knot. Figure 21.5a illustrates such a spline. This spline is defined as

$$f(x) = \begin{cases} 0.5 + 0.2x, & x < 2, \\ -0.5 + 0.7x, & x \geq 2. \end{cases}$$

Because  $0.5 + 0.2x = 0.9 = -0.5 + 0.7x$  when  $x = 2$ , the two linear components are equal at the point  $x = 2$ , so that they join together there.

The point  $x = 2$  where the spline switches from one linear function to the other is called a *knot*. A linear spline with a knot at the point  $t$  can be constructed as follows. The spline is defined to be  $s(x) = a + bx$  for  $x < t$  and  $s(x) = c + dx$  for  $x > t$ . The parameters  $a$ ,  $b$ ,  $c$ , and  $d$  can be chosen arbitrarily except that they must satisfy the equality constraint



**Fig. 21.5.** (a) Example of a linear spline with a knot at 2. (b) The linear plus function  $(x - 1)_+$  with a knot at 1 (black) and its first derivative (red).

$$a + bt = c + dt, \quad (21.10)$$

which assures us that the two lines join together at  $x = t$ . Solving for  $c$  in (21.10), we get  $c = a + (b - d)t$ . Substituting this expression for  $c$  into the definition of  $s(x)$  and doing some rearranging, we have

$$s(x) = \begin{cases} a + bx, & x < t, \\ a + bx + (d - b)(x - t), & x \geq t. \end{cases} \quad (21.11)$$

Recall the definition that for any number  $y$ ,

$$(y)_+ = \begin{cases} 0, & y < 0, \\ y, & y \geq 0. \end{cases}$$

By this definition,

$$(x - t)_+ = \begin{cases} 0, & x < t, \\ x - t, & x \geq t. \end{cases}$$

We call  $(x - t)_+$  a linear *plus function* with a knot at  $t$ . It is also called a truncated line, though we will stick with “plus function.” The spline  $s(x)$  in (21.11) can be written using this plus function:

$$s(x) = a + bx + (d - b)(x - t)_+.$$

The plus function simplifies the problem of keeping the spline continuous at  $t$ . Figure 21.5b illustrates a linear plus function with a knot at 1 and its first derivative. Notice that

$$\frac{d}{dx}(x - t)_+ = \begin{cases} 0, & x < t, \\ 1, & x > t. \end{cases}$$

### 21.4.2 Linear Splines with Many Knots

Plus functions are very convenient when defining linear splines with more than one knot, because plus functions automatically join the component linear functions together so that the spline is continuous. For example, suppose we want a linear spline to have  $K$  knots,  $t_1 < \dots < t_K$ , for the spline to equal  $s(x) = \beta_0 + \beta_1 x$  for  $x < t_1$ , and for the first derivative of the spline to jump by the amount  $b_k$  at knot  $t_k$ , for  $k = 1, \dots, K$ . Then the spline can be constructed from linear plus functions, one for each knot:

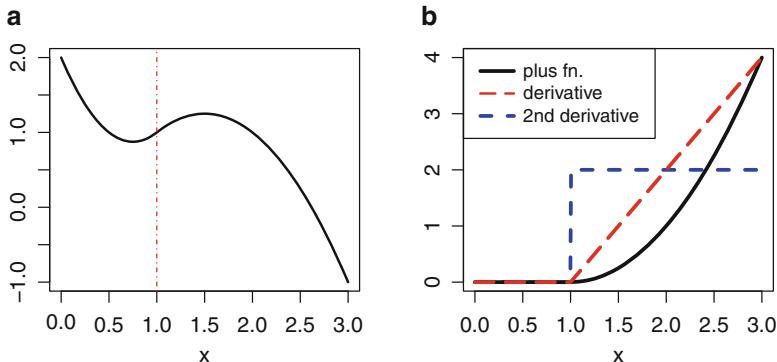
$$s(x) = \beta_0 + \beta_1 x + b_1(x - t_1)_+ + b_2(x - t_2)_+ + \dots + b_K(x - t_K)_+.$$

Because the plus functions are continuous, the spline is the sum of continuous functions and is therefore continuous itself.

### 21.4.3 Quadratic Splines

A linear spline is continuous but has “kinks” at its knots, where its first derivative jumps. If we want a function without these kinks, we cannot use a linear spline. A quadratic spline is a function obtained by piecing together quadratic polynomials. More precisely,  $s(x)$  is a quadratic spline with knots  $t_1 < \dots < t_K$  if  $s(x)$  equals one quadratic polynomial to the left of  $t_1$  and equals a second quadratic polynomial between  $t_1$  and  $t_2$ , and so on. The quadratic polynomials are pieced together, so that the spline is continuous and, to guarantee no kinks, its first derivative is also continuous. Figure 21.6a shows a quadratic spline with a knot at 1. Notice that the function does not have a kink at the knot but changes from convex to concave there.

As with linear splines, continuity can be enforced by using plus functions. Define the quadratic plus function



**Fig. 21.6.** (a) Quadratic spline with a knot at 1. The dotted red vertical line marks the knot's location. (b) The quadratic plus function  $(x-1)_+^2$  with a knot at 1 (black) and its first (red) and second (blue) derivatives.

$$(x-t)_+^2 = \begin{cases} 0, & x < t, \\ (x-t)^2, & x \geq t. \end{cases}$$

Notice that  $(x-t)_+^2$  equals  $\{(x-t)_+\}^2$ , not  $\{(x-t)^2\}_+ = (x-t)^2$ .

Figure 21.6b shows a quadratic plus function and its first and second derivatives. One can see that for  $x > t$

$$\frac{d}{dx}(x-t)_+^2 = 2(x-t)_+$$

and

$$\frac{d^2}{dx^2}(x-t)_+^2 = 2(x-t)_+^0,$$

where  $(x-t)_+^0 = \{(x-t)_+\}^0$ , so that  $(x-t)_+^0$  is the 0th-degree plus function

$$(x-t)_+^0 = \begin{cases} 0, & x < t, \\ 1, & x \geq t. \end{cases}$$

Therefore, the second derivative of  $(x-t)_+^2$  jumps from 0 to 2 at the knot  $t$ .

A quadratic spline with knots  $t_1 < \dots < t_K$  can be written as

$$s(x) = \beta_0 + \beta_1 x + \beta_2 x^2 + b_1(x-t_1)_+^2 + b_2(x-t_2)_+^2 + \dots + b_K(x-t_K)_+^2.$$

The second derivative of  $s$  jumps by the amount  $2b_k$  at knot  $t_k$  for  $k = 1, \dots, K$ .

#### 21.4.4 $p$ th Degree Splines

The way to define a general  $p$ th-degree spline with knots  $t_1 < \dots < t_K$  should now be obvious:

$$s(x) = \beta_0 + \beta_1 x + \cdots + \beta_p x^p + b_1(x - t_1)_+^p + \cdots + b_K(x - t_K)_+^p, \quad (21.12)$$

where, as we have seen for the specific case of  $p = 2$ ,  $(x - t)_+^p$  equals  $\{(x - t)_+^p\}$ . The first  $p - 1$  derivatives of  $s$  are continuous while the  $p$ th derivative takes a jump equal to  $p! b_k$  at the  $k$ th knot.

### 21.4.5 Other Spline Bases

Given a degree  $p$  and knots  $\kappa_1, \dots, \kappa_K$ , the polynomials  $1, x, \dots, x^p$  and plus functions  $(x - \kappa_1)_+^p, \dots, (x - \kappa_K)_+^p$  form a spline basis. What this means is that any  $p$ th degree spline with knots  $\kappa_1, \dots, \kappa_K$  is a linear combination of these basis functions. The basis of polynomials and plus functions is simple to understand, but is known to be numerically unstable if the number of knots is large. For this reason, other bases are often used for numerical computations. The B-spline basis is particularly popular. It is assumed here that the reader will not be programming spline estimators from scratch but rather will be using spline software. Therefore, B-splines and other bases will not be covered here, but see Sect. 21.6 for further reading.

## 21.5 Penalized Splines

Because a  $p$ th degree spline with  $K$  knots has  $1 + p + K$  parameters, an ordinary least-squares fit will usually overfit the data unless both  $p$  and  $K$  are kept small, for instance,  $1 + p + K \leq 6$ . (There is nothing especially about the number 6 and it is just being used as a rule of thumb. Any number between 5 and 10 would be equally good.) An example is the quadratic spline with one knot (so  $1 + p + K = 4$ ) used as a forward-rate curve in Example 11.3. However, a spline with  $p$  and  $K$  both small is essentially a parametric model. To have the flexibility of a nonparametric model, that is, a wide range of potential values of  $p_{\text{eff}}$ , we need to have  $K$  large and find another way to avoid overfitting. Penalized least-squares estimation does this.

Let  $\mu(x; \boldsymbol{\beta}) = \mathbf{B}(x)^T \boldsymbol{\beta}$  be a spline, where  $\boldsymbol{\beta}$  is a vector of coefficients and  $\mathbf{B}(x) = (B_1(x), \dots, B_{1+p+K}(x))^T$  is a spline basis. For example,  $\mathbf{B}(x) = (1, x, \dots, x_p, (x - \kappa_1)_+^p, \dots, (x - \kappa_K)_+^p)$  if we use model (21.12). A penalized least-squares estimator minimizes over  $\boldsymbol{\beta}$  the penalized sum of squares

$$\sum_{i=1}^n \{Y_i - \mu(X_i; \boldsymbol{\beta})\}^2 + \lambda \boldsymbol{\beta}^T \mathbf{D} \boldsymbol{\beta}, \quad (21.13)$$

where  $\mathbf{D}$  is a positive semidefinite matrix and  $\lambda > 0$  is a penalty parameter.

A common choice of  $\mathbf{D}$  has the  $i, j$ th element equal to

$$\int_a^b B_i^{(2)}(x) B_j^{(2)}(x) dx \quad (21.14)$$

for some  $a < b$ , such as,  $a = \min(X_i)$  and  $b = \max(X_i)$ . Here  $B_i^{(2)}(x)$  is the second derivative of  $B_i(x)$ . With this  $\mathbf{D}$ ,

$$\lambda \boldsymbol{\beta}^\top \mathbf{D} \boldsymbol{\beta} = \lambda \int_a^b \left\{ \mu^{(2)}(x; \boldsymbol{\beta}) \right\}^2 dx, \quad (21.15)$$

Since  $\mu^{(2)}(x)$  is the amount of curvature of  $\mu$  at  $x$ , this choice of  $\mathbf{D}$  penalizes wiggly functions and, if  $\lambda$  is chosen appropriately, prevents overfitting. If  $\lambda = 0$ , then there is no penalization and the effective number of parameters is  $1 + p + K$ . With this  $\mathbf{D}$ , in the limit as  $\lambda \rightarrow \infty$ , any curvature at all receives an infinite penalty, so the estimator converges to a linear polynomial fit and the effective number of parameters converges to 2. Any value of  $p_{\text{eff}}$  between 2 and  $1 + p + K$  is achievable by the some value of  $\lambda$  between the extremes of 0 and  $\infty$ .

Let  $\mathbf{X}$  be the  $n \times (1 + p + K)$  matrix with  $i, j$ th element  $B_j(X_i)$  and let  $\mathbf{Y} = (Y_1, \dots, Y_n)^\top$ . The penalized least-squares estimate is

$$\hat{\boldsymbol{\beta}}(\lambda) = \left( \mathbf{X}^\top \mathbf{X} + \lambda \mathbf{D} \right)^{-1} \mathbf{X}^\top \mathbf{Y}, \quad (21.16)$$

which is obtained by setting the gradient of (21.13) equal to zero and solving. The fitted values are

$$\hat{\mathbf{Y}}(\lambda) = \mathbf{X} \hat{\boldsymbol{\beta}}(\lambda) = \left\{ \mathbf{X} (\mathbf{X}^\top \mathbf{X} + \lambda \mathbf{D})^{-1} \mathbf{X}^\top \right\} \mathbf{Y} = \mathbf{H}(\lambda) \mathbf{Y}, \quad (21.17)$$

where  $\mathbf{H}(\lambda) = \left\{ \mathbf{X} (\mathbf{X}^\top \mathbf{X} + \lambda \mathbf{D})^{-1} \mathbf{X}^\top \right\}$  is the smoother matrix.

### 21.5.1 Cubic Smoothing Splines

A very widely used nonparametric regression estimator is the cubic smoothing spline. This estimator uses a knot at each unique value of  $\{X_1, \dots, X_n\}$  and the second-derivative penalty in (21.15). Using this many knots is not really necessary, and a variation on the cubic smoothing spline also uses penalty (21.15) but fewer knots. The knots could be equally-spaced or at selected quantiles of  $\{X_1, \dots, X_n\}$ .

The function `smooth.spline()` in R fits a cubic smoothing spline if the argument `all.knots` is `TRUE` or if  $n < 50$ . If `all.knots = FALSE` and  $n > 49$ , then it uses less than the full set of knots.

### 21.5.2 Selecting the Amount of Penalization

The penalty parameter  $\lambda$  determines the amount of smoothing and can be chosen by AIC or GCV. Another popular method for choosing  $\lambda$  is REML (restricted maximum likelihood). REML is based on a so-called mixed model, where some of the spline coefficients are random variables. A description of mixed models and REML is beyond the scope of this book, but the interested reader may consult the references in Sect. 21.6.

*Example 21.2. Estimating the drift and volatility for the evolution of the risk-free returns*

In this example, we return to estimating the drift and squared volatility functions for the evolution of the risk-free returns. Three estimators will be used: local linear, local quadratic, and a penalized spline. The R code is:

```

1 library(Ecdat)
2 library(KernSmooth)
3 library(locfit)
4 library(mgcv)
5 data(Capm)
6 attach(Capm)
7 n = length(rf)
8 year = seq(1960.125,2003,length=n)
9 diffrrf=diff(Capm$rf)
10 rf_lag = rf[1:(n-1)]
11 log_rf_lag = log(rf_lag)
12 ll_mu <- locpoly(rf_lag, diffrrf, bandwidth = dpill(rf_lag,diffrrf))
13 muhat = spline(ll_mu$x, ll_mu$y, xout = rf_lag)$y
14 epsilon_sqr = (diffrrf - muhat)^2
15 ll_sig <- locpoly(rf_lag, epsilon_sqr,
16   bandwidth = dpill(rf_lag, epsilon_sqr) )
17 gam_mu = gam(diffrrf ~ s(rf_lag, bs = "cr"), method = "REML")
18 epsilon_sqr = (diffrrf-gam_mu$fit)^2
19 gam_sig = gam(epsilon_sqr ~ s(rf_lag, bs = "cr"), method = "REML")
20 locfit_mu = locfit(diffrrf ~ rf_lag)
21 epsilon_sqr = (diffrrf - fitted(locfit_mu))^2
22 locfit_sig = locfit(epsilon_sqr ~ rf_lag)
23 std_res = (diffrrf - fitted(locfit_mu)) / sqrt(fitted(locfit_sig))
24 min(rf_lag[(gam_mu$fit < 0)])
25 orrf = order(rf_lag)
26 pdf("riskfree02.pdf", width = 8, height = 4)
27 par(mfrow=c(1, 2))
28 plot(rf_lag[orrf], gam_mu$fit[orrf], type = "l", lwd = 3, lty = 1,
29   xlab = "lagged rate", ylab = "change in rate", main = "(a)")
30 lines(ll_mu$x,ll_mu$y, lwd = 3, lty = 2, col = "red")
31 lines(locfit_mu, lwd = 3, lty = 3, col = "blue")
32 legend(0.1, -0.05, c("spline", "local linear", "local quadratic"),
33   lty = c(1, 2, 3), cex = 0.85, lwd = 3,
34   col = c("black", "red", "blue"))
35 rug(rf_lag)
36 abline(h = 0, lwd = 2)
37 plot(rf_lag[orrf], gam_sig$fit[orrf], type="l", lwd = 3, lty = 1,
38   ylim = c(0, 0.03), xlab = "lagged rate",
39   ylab = "squared residual", main = "(b)")
40 lines(ll_sig$x,ll_sig$y, lwd = 3, lty = 2, col = "red")
41 lines(locfit_sig, lwd = 3, lty = 3, col = "blue")
42 abline(h = 0, lwd = 2)
```

```

43 legend("topleft", c("spline", "local linear", "local quadratic"),
44   lty = c(1, 2, 3), cex = 0.85, lwd = 3,
45   col = c("black", "red", "blue"))
46 rug(rf_lag)
47 graphics.off()

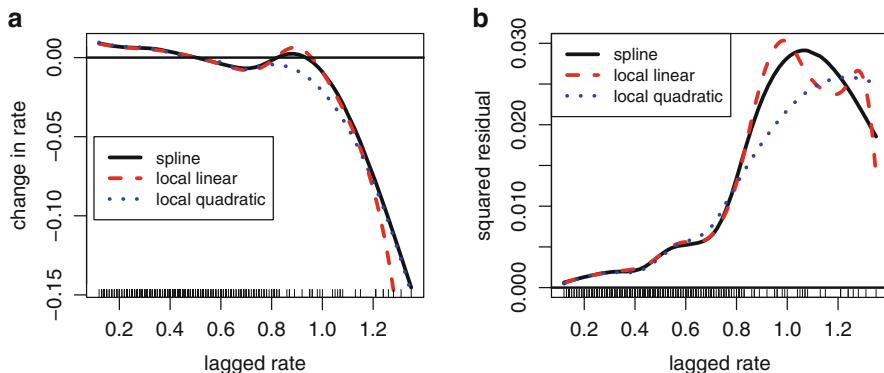
```

The first estimator, local linear, is computed at line 12 using the function `locpoly()` in R's `KernSmooth` package. The dpi plug-in bandwidth selector is computed using the function `dpill()` in this package.<sup>2</sup>

In the R code, the changes in the risk-free returns (`diffrrf`) are regressed on the lagged returns (`rf_lag`) to estimate the drift. The local linear estimator is computed on an equally-spaced grid, and to compute residuals the function `spline()` is used at line 13 to interpolate the fit to the observed values of `rf_lag`. Finally, the squared residuals (`epsilon_sqr`) computed at line 14 are regressed at lines 15–16 on the lagged returns to estimate the squared volatility function. The estimated drift function is in the object `ll_mu` and the estimated squared volatility function is in `ll_sig`.

The penalized spline estimator is computed at line 17 by the `gam()` function in the `mgcv` package. The specification `bs = "cr"` requests a cubic spline fit with penalty (21.15). The REML method is used to select the amount of smoothing.

The local quadratic estimator is computed at line 20 with the function `locfit()` in R's `locfit` package. Spline interpolation is not necessary here, since with `locfit()` the fitted values can be computed with the `fitted` function.



**Fig. 21.7.** Risk-free monthly returns. (a) Estimates of the drift function. (b) Estimates of the squared volatility function.

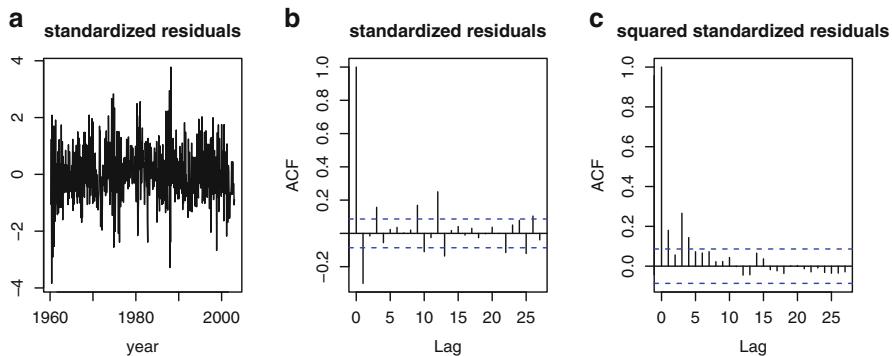
<sup>2</sup> “dpill” means “direct plug-in, local linear.”

All three estimated drift functions are shown in Fig. 21.7a and the squared volatility function estimates are in Fig. 21.7b.

The drift functions have a general decreasing trend and are negative to the right of 0.51 (approximately), except that the estimates have humps around 0.9–1.0 and the spline and local linear estimates are slightly positive at this hump. It is likely that the hump is due to random variation, which increases as one moves from left to right (see Fig. 21.1). If we use the local quadratic fit, then the estimated drift is positive to the left of 0.51 and negative to the right of 0.51. The drift will cause reversion to a mean of 0.51, which is an annual rate of  $6.12\% = (12)(0.51)\%$ . The Chan et al. (1992) drift function,  $\mu(r) = \beta(r - \alpha)$ , is also mean-reverting, but linear. In contrast, the local quadratic estimated drift function in Fig. 21.7 is nonlinear and shows much faster reversion to the mean when the rate is high.

The squared volatility estimates show that volatility increases with the rate, at least to a point. For very high rates, the estimated volatility function becomes decreasing. There is not enough data with extremely high rates to tell if this phenomenon is “real” or due to random estimation error. The extremely high rates occurred only for the brief period in the early 1980s; see Fig. 21.1a.

The standardized residuals  $\{\Delta r_t - \hat{\mu}(r_{t-1})\}/\hat{\sigma}(r_{t-1})$  show negative serial correlation and GARCH-type volatility clustering; see Fig. 21.8. Neither of these is surprising. Negative lag-1 autocorrelation is common in a differenced series and volatility clustering is certainly to be expected in any financial time series. This case study could be continued by fitting an ARMA/GARCH model to the standardized residuals.  $\square$



**Fig. 21.8.** Risk-free monthly returns. Residual analysis. (a) Time series plot of standardized residuals. (b) ACF of standardized residuals. (c) ACF of squared standardized residuals.

*Example 21.3.* Spline estimation of a forward rate

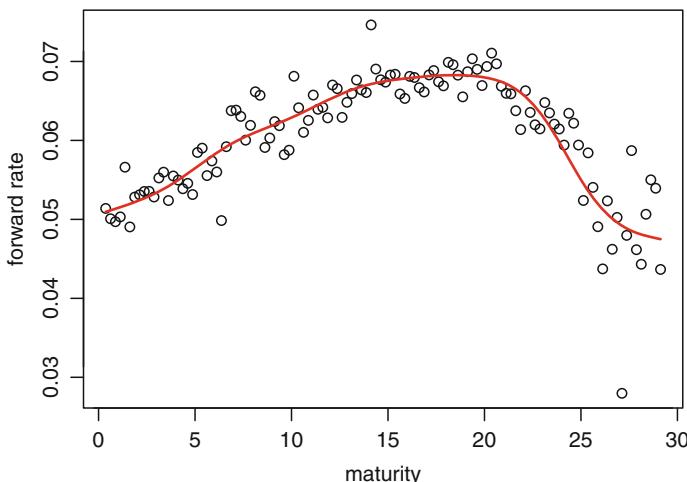
This example used the STRIPS data that were already analyzed in Example 11.3. In that example, an unpenalized spline was fit to the bond prices by nonlinear regression, and smoothness was controlled by using only knot.

The function `gam()` in the `mgcv` is a powerful tool that can fit a wide variety of spline models with penalties. In this example, `gam()` is used to fit a cubic spline to the empirical forward rates that are defined at the beginning of Sect. 11.3. The code is below.

```

1 dat = read.table("strips_dec95.txt", header = T)
2 T = dat$T
3 n = length(T)
4 ord = order(T)
5 T = T[ord]
6 price = dat$price[ord]
7 Int_F = - log(price) + log(100)
8 emp_forward = diff(Int_F)/diff(T)
9 library(mgcv)
10 X = T[-1]
11 fit_gam = gam(emp_forward ~ s(X, bs = "cr"))
12 pred_gam = predict(fit_gam, as.data.frame(X) )
13 pdf("forward_spline.pdf", width = 6, height = 5)
14 plot(X, emp_forward, xlab = "maturity", ylab = "forward rate")
15 lines(X, pred_gam, col = "red", lwd = 2)
16 graphics.off()

```



**Fig. 21.9.** Empirical forward rates (circle) with a spline fit.

Figure 21.9 shows the empirical forward rates and the cubic spline fit to them. Notice that the spline goes through the empirical forward rates with little sign of bias and yet is smooth.  $\square$

## 21.6 Bibliographic Notes

Ruppert, Wand, and Carroll (2003) and Wood (2006) offer comprehensive introductions to nonparametric and semiparametric modeling and their applications. Wand and Jones (1995) and Fan and Gijbels (1996) are good sources of information about local polynomial regression. REML is discussed in detail by Ruppert, Wand, and Carroll (2003) and Wood (2006). Wasserman (2006) is an interesting modern synthesis of nonparametric estimation. Wood (2006) is a good introduction to his `mgcv` package.

## 21.7 R Lab

### 21.7.1 Additive Model for Wages, Education, and Experience

This section uses the Current Population Survey data in the `CPS1988` data set introduced in Sect. 10.4.1. We will fit spline effects for both predictors, `education` and `experience`. This is easily done with the `gam()` function in the `mgcv` package. The model being fit is

$$\log(\text{wage}) = \beta_0 + s_1(\text{education}) + s_2(\text{experience}) + \beta_1 \text{ethnicity} + \epsilon_i,$$

where  $\beta_0$  is the intercept,  $s_1$  and  $s_2$  are splines, `ethnicity` is 0 for Caucasians and 1 for African Americans, and  $\epsilon_i$  is white noise. To fit this model, print its summary, and plot the estimates of  $s_1$  and  $s_2$ , run:

```
library(AER)
library(mgcv)
data(CPS1988)
attach(CPS1988)
fitGam = gam(log(wage)~s(education)+s(experience)+ethnicity)
summary(fitGam)
par(mfrow=c(1,2))
plot(fitGam)
```

**Problem 1** What are the estimates of  $\beta_0$  and  $\beta_1$ ?

**Problem 2** Describe the shapes of  $s_1$  and  $s_2$ .

### 21.7.2 An Extended CKLS Model for the Short Rate

In this section, we use splines to extend the CKLS model in Sect. 11.12 by letting the drift parameters  $a$  and  $\theta$  vary with time so that

$$\mu(t, r) = a(t) \{ \theta(t) - r \}. \quad (21.18)$$

One could also let the volatility parameters  $\sigma$  and  $\gamma$  vary as well with  $t$ , but, for simplicity, we will not do that here. We will fit this model with  $a(t)$  being linear in time and  $\theta(t)$  being a piecewise linear spline. [Letting both  $a(t)$  and  $\theta(t)$  be splines can lead to unstable estimates, so we will restrict  $a(t)$  to be linear.] First, read in the data, and then create the knots and the truncated line basis functions.

```
# CKLS, extended
library(Ecdat)
data(Irates)
r1 = Irates[,1]
n = length(r1)
lag_r1 = lag(r1)[-n]
delta_r1 = diff(r1)
n = length(lag_r1)
knots = seq(from=1950,to=1985,length=10)
t = seq(from=1946,to =1991+2/12,length=n)
X1 = outer(t,knots,FUN="-")
X2 = X1 * (X1>0)
X3 = cbind(rep(1,n), (t - 1946),X2)
m2 = dim(X3)[2]
m = m2 - 1
```

**Problem 3** How many knots are being used here? What does the `outer()` function do here? What is done by the statement `X2 = X1 * (X1>0)`? Describe what is in the variable `X3`.

Now fit the CKLS model with time-varying drift.

```
nlmod_CKLS_ext = nls(delta_r1 ~ X3[,1:2] %*% a *
                      (X3 %*% theta-lag_r1),
                      start=list(theta = c(10,rep(0,m)),
                      a=c(.01,0)),control=list(maxiter=200))
AIC(nlmod_CKLS_ext)
param4 = summary(nlmod_CKLS_ext)$parameters[,1]
par(mfrow=c(1,3))
plot(t,X3 %*% param4[1:m2],ylim=c(0,16),ylab="rate",
     main="(a)",col="red",type="l",lwd=2)
lines(t,lag_r1)
legend("topleft",c("theta(t)","lagged rate"),lwd=c(2,1),
       col=c("red","black"))
```

```

plot(t,X3[,1:2] %*% param4[(m2+1):(m2+2)],ylab="a(t)",
      col="red",type="l",lwd=2,main="(b)")

res_sq = residuals(nlmod_CKLS_ext)^2
nlmod_CKLS_ext_res <- nls(res_sq ~ A*lag_r1^B,
      start=list(A=.2,B=1/2) )

plot(lag_r1,sqrt(res_sq),pch=5,ylim=c(0,6),ylab="",main="(c)")
lines(lag_r1,sqrt(fitted(nlmod_CKLS_ext_res)),
      lw=3,col="red",type="l")
legend("topleft",c("abs res","volatility fn"),lty=c(NA,1),
      pch=c(5,NA),col=c("black","red"),lwd=1:2)

```

**Problem 4** Explain why  $X3[,1:2] %*% a$  is a linear function but  $X3 %*% \theta$  is a spline.

**Problem 5** What is the interpretation of a time-varying  $\theta$ ? Note that in panel (a),  $\theta$  seems to track the interest rate. Does this make sense? Why or why not?

**Problem 6** Would you accept or reject the null hypothesis that  $a(t)$  is constant, that is, that the slope of the linear function  $a(t)$  is zero? Justify your answer.

## 21.8 Exercises

- A linear spline  $s(t)$  has knots at 1, 2, and 3. Also,  $s(0) = 1$ ,  $s(1) = 1.3$ ,  $s(2) = 5.5$ ,  $s(4) = 6$ , and  $s(5) = 6$ .
  - What is  $s(0.5)$ ?
  - What is  $s(3)$ ?
  - What is  $\int_2^4 s(t) dt$ ?
- Suppose that (21.1) holds with  $\mu(r) = 0.1(0.035 - r)$  and  $\sigma(r) = 2.3r$ .
  - What is the expected value of  $r_t$  given that  $r_{t-1} = 0.04$ ?
  - What is the variance of  $r_t$  given that  $r_{t-1} = 0.02$ ?
- Let the spline  $s(x)$  be defined as

$$s(x) = (x)_+ - 3(x-1)_+ + (x-2)_+.$$

- Is  $s(x)$  either a probability density function (pdf) or a cumulative distribution function (cdf)? Explain your answer.
- If  $X$  is a random variable and  $s$  is its pdf or cdf [whichever is the correct answer in (a)], then what is the 90th percentile of  $X$ ?

- Let  $s$  be the spline

$$s(x) = 1 + 0.65x + x^2 + (x-1)_+^2 + 0.6(x-2)_+^2.$$

- What are  $s(1.5)$  and  $s'(1.5)$ ?
- What is  $s''(2.2)$ ?

## References

- Chan, K. C., Karolyi, G. A., Longstaff, F. A., and Sanders, A. B. (1992) An empirical comparison of alternative models of the short-term interest rate. *Journal of Finance*, **47**, 1209–1227.
- Cox, J. C., Ingersoll, J. E., and Ross, S. A. (1985) A theory of the term structure of interest rates. *Econometrica*, **53**, 385–407.
- Fan, J., and Gijbels, I. (1996) *Local Polynomial Modelling and Its Applications*, Chapman & Hall, London.
- Merton, R. C. (1973) Theory of rational option pricing. *Bell Journal of Economics and Management Science*, **4**, 141–183.
- Ruppert, D., Sheather, S., and Wand, M. P. (1995) An effective bandwidth selector for local least squares kernel regression, *Journal of the American Statistical Association*, **90**, 1257–1270.
- Ruppert, D., Wand, M. P., and Carroll, R. J. (2003) *Semiparametric Regression*, Cambridge University Press, Cambridge.
- Vasicek, O. A. (1977) An equilibrium characterization of the term structure. *Journal of Financial Economics*, **5**, 177–188.
- Wand, M. P., and Jones, M. C. (1995) *Kernel Smoothing*, Chapman & Hall, London.
- Wasserman, L. (2006) *All of Nonparametric Statistics*, Springer, New York.
- Wood, S. (2006) *Generalized Additive Models: An Introduction with R*, Chapman & Hall, Boca Raton, FL.
- Yau, P., and Kohn, R. (2003) Estimation and variable selection in nonparametric heteroskedastic regression. *Statistics and Computing*, **13**, 191–208.

# A

---

## Facts from Probability, Statistics, and Algebra

### A.1 Introduction

It is assumed that the reader is already familiar with the basics of probability, statistics, matrix algebra, and other mathematical topics needed in this book, and so the goal of this appendix is merely to provide a quick review and cover some more advanced topics that may not be familiar.

### A.2 Probability Distributions

#### A.2.1 Cumulative Distribution Functions

The *cumulative distribution function (CDF)* of  $Y$  is defined as

$$F_Y(y) = P\{Y \leq y\}.$$

If  $Y$  has a PDF  $f_Y$ , then

$$F_Y(y) = \int_{-\infty}^y f_Y(u) du.$$

Many CDFs and PDFs can be calculated by computer software packages, for instance, `pnorm()`, `pt()`, and `pbinom()` in R calculate, respectively, the CDF of a normal,  $t$ , and binomial random variable. Similarly, `dnorm()`, `dt()`, and `dbinom()` calculate the PDFs of these distributions.

### A.2.2 Quantiles and Percentiles

If the CDF  $F(y)$  of a random variable  $Y$  is continuous and strictly increasing, then it has an inverse function  $F^{-1}$ . For each  $q$  between 0 and 1,  $F^{-1}(q)$  is called the  $q$ -quantile or 100 $q$ th percentile.

The median is the 50% percentile or 0.5-quantile. The 25% and 75% percentiles (0.25- and 0.75-quantiles) are called the first and third quartiles and the median is the second quartile. The three quartiles divide the range of a continuous random variable into four groups of equal probability. Similarly, the 20%, 40%, 60%, and 80% percentiles are called quintiles and the 10%, 20%, ..., 90% percentiles are called deciles.

For any CDF  $F$ , invertible or not, the *pseudo-inverse* is defined as

$$F^-(x) = \inf\{y : F(y) \geq x\}.$$

Here “inf” is the infimum or greatest lower bound of a set; see Appendix A.5. For any  $q$  between 0 and 1, the  $q$ th quantile will be defined as  $F^-(q)$ . If  $F$  is invertible, then  $F^{-1} = F^-$ , so this definition of quantile agree with the one for invertible CDFs.  $F^-$  is often called the *quantile function*.

Sometimes a  $(1 - \alpha)$ -quantile is called an  $\alpha$ -upper quantile, to emphasize the amount of probability above the quantile. In analogy, a quantile might also be referred to as lower quantile.

Quantiles are said to “respect transformations” in the following sense. If  $Y$  is a random variable whose  $q$ -quantile equals  $y_q$ , if  $g$  is a strictly increasing function, and if  $X = g(Y)$ , then  $g(y_q)$  is the  $q$ -quantile of  $X$ ; see (A.5).

### A.2.3 Symmetry and Modes

A probability density function (PDF)  $f$  is said to be *symmetric* about  $\mu$  if  $f(\mu - y) = f(\mu + y)$  for all  $y$ . A *mode* of a PDF is a local maximum, that is a value  $y$  such that for some  $\epsilon > 0$ ,  $f(y) > f(x)$  if  $y - \epsilon < x < y + \epsilon$ . A PDF with one mode is called *unimodal*, with two modes *bimodal*, and with two or more modes *multimodal*.

### A.2.4 Support of a Distribution

The support of a *discrete* distribution is the set of all  $y$  that have a positive probability. More generally, a point  $y$  is in the support of a distribution if, for every  $\epsilon > 0$ , the interval  $(y - \epsilon, y + \epsilon)$  has positive probability. For example, the support of a normal distribution is  $(-\infty, \infty)$ , the support of a gamma or log-normal distribution is  $[0, \infty)$ , and the support of a binomial( $n, p$ ) distribution is  $\{0, 1, 2, \dots, n\}$  provided  $p \neq 0, 1$ .<sup>1</sup>

---

<sup>1</sup> It is assumed that most readers are already familiar with the normal, gamma, log-normal, and binomial distributions. However, these distributions will be discussed in some detail later.

### A.3 When Do Expected Values and Variances Exist?

The expected value of a random variable could be infinite or not exist at all. Also, a random variable need not have a well-defined and finite variance. To appreciate these facts, let  $Y$  be a random variable with density  $f_Y$ . The expectation of  $Y$  is

$$\int_{-\infty}^{\infty} y f_Y(y) dy$$

provided that this integral is defined. If

$$\int_{-\infty}^0 y f_Y(y) dy = -\infty \text{ and } \int_0^{\infty} y f_Y(y) dy = \infty, \quad (\text{A.1})$$

then the expectation is, formally,  $-\infty + \infty$ , which is not defined, so the expectation does not exist. If integrals in (A.1) are both finite, then  $E(Y)$  exists and equals the sum of these two integrals. The expectation can exist but be infinite, because if

$$\int_{-\infty}^0 y f_Y(y) dy = -\infty \text{ and } \int_0^{\infty} y f_Y(y) dy < \infty,$$

then  $E(Y) = -\infty$ , and if

$$\int_{-\infty}^0 y f_Y(y) dy > -\infty \text{ and } \int_0^{\infty} y f_Y(y) dy = \infty,$$

then  $E(Y) = \infty$ .

If  $E(Y)$  is not defined or is infinite, then the variance that involves  $E(Y)$  cannot be defined either. If  $E(Y)$  is defined and finite, then the variance is also defined. The variance is finite if  $E(Y^2) < \infty$ ; otherwise the variance is infinite.

The nonexistence of finite expected values and variances is of importance for modeling financial markets data, because, for example, the popular GARCH models discussed in Chap. 14 need not have finite expected values and variances. Also,  $t$ -distributions that, as demonstrated in Chap. 5, can provide good fits to equity returns may have nonexistent means or variances.

One could argue that any variable  $Y$  derived from financial markets will be bounded, that is, that there is a constant  $M < \infty$  such that  $P(|Y| \leq M) = 1$ . In this case, the integrals in (A.1) are both finite, in fact at most  $M$ , and  $E(Y)$  exists and is finite. Also,  $E(Y^2) \leq M^2$ , so the variance of  $Y$  is finite. So should we worry at all about the mathematically niceties of whether expected values and variances exist and are finite? The answer is that we should. A random variable might be bounded in absolute value by a very large constant  $M$  and yet, if  $M$  is large enough, behave much like a random variable that does not have an expected value or has an expected value that is infinite or has a finite expected value but an infinite variance. This can be seen in the simulations

of GARCH processes. Results from computer simulations are bounded by the maximum size of a number in the computer. Yet these simulations behave as if the variance were infinite.

## A.4 Monotonic Functions

The function  $g$  is increasing if  $g(x_1) \leq g(x_2)$  whenever  $x_1 < x_2$  and strictly increasing if  $g(x_1) < g(x_2)$  whenever  $x_1 < x_2$ . Decreasing and strictly decreasing are defined similarly, and  $g$  is (strictly) monotonic if it is either (strictly) increasing or (strictly) decreasing.

## A.5 The Minimum, Maximum, Infimum, and Supremum of a Set

The minimum and maximum of a set are its smallest and largest values, if these exists. For example, if  $A = \{x : 0 \leq x \leq 1\}$ , then the minimum and maximum of  $A$  are 0 and 1. However, not all sets have a minimum or a maximum, for example,  $B = \{x : 0 < x < 1\}$  has neither a minimum nor a maximum. Every set as an infimum (or inf) and a supremum (or sup). The inf of a set  $C$  is the largest number that is less than or equal to all elements of  $C$ . Similarly, the sup of  $C$  is the smallest number that is greater than or equal to every element of  $C$ . The set  $B$  just defined has an inf of 0 and a sup of 1. The following notation is standard:  $\min(C)$  and  $\max(C)$  are the minimum and maximum of  $C$ , if these exist, and  $\inf(C)$  and  $\sup(C)$  are the infimum and supremum.

## A.6 Functions of Random Variables

Suppose that  $X$  is a random variable with PDF  $f_X(x)$  and  $Y = g(X)$  for  $g$  a strictly increasing function. Since  $g$  is strictly increasing, it has an inverse, which we denote by  $h$ . Then  $Y$  is also a random variable and its CDF is

$$F_Y(y) = P(Y \leq y) = P\{g(X) \leq y\} = P\{X \leq h(y)\} = F_X\{h(y)\}. \quad (\text{A.2})$$

Differentiating (A.2), we find the PDF of  $Y$ :

$$f_Y(y) = f_X\{h(y)\}h'(y). \quad (\text{A.3})$$

Applying a similar argument to the case, where  $g$  is strictly decreasing, one can show that whenever  $g$  is strictly monotonic, then

$$f_Y(y) = f_X\{h(y)\}|h'(y)|. \quad (\text{A.4})$$

Also from (A.2), when  $g$  is strictly increasing, then

$$F_Y^{-1}(p) = g\{F_X^{-1}(p)\}, \quad (\text{A.5})$$

so that the  $p$ th quantile of  $Y$  is found by applying  $g$  to the  $p$ th quantile of  $X$ . When  $g$  is strictly decreasing, then it maps the  $p$ th quantile of  $X$  to the  $(1-p)$ th quantile of  $Y$ .

**Result A.1** Suppose that  $Y = a + bX$  for some constants  $a$  and  $b \neq 0$ . Let  $g(x) = a + bx$ , so that the inverse of  $g$  is  $h(y) = (y - a)/b$  and  $h'(y) = 1/b$ . Then

$$\begin{aligned} F_Y(y) &= F_X\{b^{-1}(y - a)\}, \quad b > 0, \\ &= 1 - F_X\{b^{-1}(y - a)\}, \quad b < 0, \\ f_Y(y) &= |b|^{-1} f_X\{b^{-1}(y - a)\}, \end{aligned}$$

and

$$\begin{aligned} F_Y^{-1}(p) &= a + bF_X^{-1}(p), \quad b > 0 \\ &= a + bF_X^{-1}(1 - p), \quad b < 0. \end{aligned}$$

## A.7 Random Samples

We say that  $\{Y_1, \dots, Y_n\}$  is a *random sample* from a probability distribution if they each have that probability distribution and are independent. In this case, we also say that they are *independent and identically distributed* or simply i.i.d. The probability distribution is often called the population and its expected value, variance, CDF, and quantiles are called the *population mean*, *population variance*, *population CDF*, and *population quantiles*. It is worth mentioning that the population is, in effect, infinite. There is a statistical theory of sampling, usually without replacement, from finite populations, but sampling of this type will not concern us here. Even in cases where the population is finite, such as, when sampling house prices, the population is usually large enough, so that it can be treated as infinite.

If  $Y_1, \dots, Y_n$  is a sample from an unknown probability distribution, then the population mean can be estimated by the *sample mean*

$$\bar{Y} = n^{-1} \sum_{i=1}^n Y_i, \quad (\text{A.6})$$

and the population variance can be estimated by the *sample variance*

$$s_Y^2 = \frac{\sum_{i=1}^n (Y_i - \bar{Y})^2}{n - 1}. \quad (\text{A.7})$$

The reason for the denominator of  $n - 1$  rather than  $n$  is discussed in Sect. 5.9. The *sample standard deviation* is  $s_Y$ , the square root of  $s_Y^2$ .

## A.8 The Binomial Distribution

Suppose that we conduct  $n$  experiments for some fixed (nonrandom) integer  $n$ . On each experiment there are two possible outcomes called “success” and “failure”; the probability of a success is  $p$ , and the probability of a failure is  $q = 1 - p$ . It is assumed that  $p$  and  $q$  are the same for all  $n$  experiments. Let  $Y$  be the total number of successes, so that  $Y$  will equal 0, 1, 2, …, or  $n$ . If the experiments are independent, then

$$P(Y = k) = \binom{n}{k} p^k q^{n-k} \quad \text{for } k = 0, 1, 2, \dots, n,$$

where

$$\binom{n}{k} = \frac{n!}{k!(n-k)!}.$$

The distribution of  $Y$  is called the *binomial distribution* and denoted  $\text{Binomial}(n, p)$ . The expected value of  $Y$  is  $np$  and its variance is  $npq$ . The  $\text{Binomial}(1, p)$  distribution is also called the Bernoulli distribution and its density is

$$P(Y = y) = p^y(1-p)^{1-y}, \quad y = 0, 1. \quad (\text{A.8})$$

Notice that  $p^y$  is equal to either  $p$  (when  $y = 1$ ) or 1 (when  $y = 0$ ), and similarly for  $(1-p)^{1-y}$ .

The functions `pbinom()`, `dbinom()`, `qbinom()`, and `rbinom()` compute binomial CDFs, pdfs, quantiles, and random numbers, respectively. For example,

```
> pbinom(3, 6, 0.5)
[1] 0.65625
```

shows that the probability of 3 or less heads in 6 tosses of a fair coin is 0.65625.

## A.9 Some Common Continuous Distributions

### A.9.1 Uniform Distributions

The uniform distribution on the interval  $(a, b)$  is denoted by  $\text{Uniform}(a, b)$  and has PDF equal to  $1/(b-a)$  on  $(a, b)$  and equal to 0 outside this interval. It is easy to check that if  $Y$  is  $\text{Uniform}(a, b)$ , then its expectation is

$$E(Y) = \frac{1}{b-a} \int_a^b Y dY = \frac{a+b}{2},$$

which is the midpoint of the interval. Also,

$$E(Y^2) = \frac{1}{b-a} \int_a^b Y^2 dY = \frac{Y^3|_a^b}{3(b-a)} = \frac{b^2 + ab + a^2}{3}.$$

Therefore,

$$\sigma_Y^2 = E(Y^2) - \{E(Y)\}^2 = \frac{b^2 + ab + a^2}{3} - \left(\frac{a+b}{2}\right)^2 = \frac{(b-a)^2}{12}.$$

*Reparameterization* means replacing the parameters of a distribution by an equivalent set. The uniform distribution can be reparameterized by using  $\mu = (a+b)/2$  and  $\sigma = (b-a)/\sqrt{12}$  as the parameters. Then  $\mu$  is a location parameter and  $\sigma$  is the scale parameter. Which parameterization of a distribution is used depends upon which aspects of the distribution one wishes to emphasize. The parameterization  $(a, b)$  of the uniform specifies its endpoints while the parameterization  $(\mu, \sigma)$  gives the mean and standard deviation. One is free to move back and forth between two or more parameterizations, using whichever is most useful in a given context. The uniform distribution does not have a shape parameter since the shape of its density is always rectangular.

The functions `punif()`, `dunif()`, `qunif()`, and `runeif()` compute uniform CDFs, pdfs, quantiles, and random numbers, respectively. For example,

```
> runif(3,0,5)
[1] 1.799252 4.003232 3.978002
```

are three random numbers uniformly distributed between 0 and 5.

### A.9.2 Transformation by the CDF and Inverse CDF

If  $Y$  has a continuous CDF  $F$ , then  $F(Y)$  has a Uniform(0,1) distribution.  $F(Y)$  is often called the *probability transformation* of  $Y$ . This fact is easy to see if  $F$  is strictly increasing, since then  $F^{-1}$  exists, so that

$$P\{F(Y) \leq y\} = P\{Y \leq F^{-1}(y)\} = F\{F^{-1}(y)\} = y. \quad (\text{A.9})$$

The result holds even if  $F$  is not strictly increasing, but the proof is slightly more complicated. It is only necessary that  $F$  be continuous.

If  $U$  is Uniform(0,1) and  $F$  is a CDF, then  $Y = F^{-1}(U)$  has  $F$  as its CDF. Here  $F^{-1}$  is the pseudo-inverse of  $F$ . This can be proved easily when  $F$  is continuous and strictly increasing, since then  $F^{-1} = F^{-}$  and

$$P(Y \leq y) = P\{F^{-1}(U) \leq y\} = P\{Y \leq F(y)\} = F(y).$$

In fact, the result holds for any CDF  $F$ , but it is more difficult to prove in the general case.  $F^{-1}(U)$  is often called the *quantile transformation* since  $F^{-1}$  is the quantile function.

### A.9.3 Normal Distributions

The *standard normal distribution* has the familiar bell-shaped density

$$\phi(y) = \frac{1}{\sqrt{2\pi}} \exp(-y^2/2), \quad -\infty < y < \infty.$$

The standard normal has mean 0 and variance 1. If  $Z$  is standard normal, then the distribution of  $\mu + \sigma Z$  is called the *normal distribution with mean  $\mu$  and variance  $\sigma^2$*  and denoted by  $N(\mu, \sigma^2)$ . By Result A.1, the  $N(\mu, \sigma^2)$  density is

$$\frac{1}{\sigma} \phi\left(\frac{y-\mu}{\sigma}\right) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left\{-\frac{(y-\mu)^2}{2\sigma^2}\right\}. \quad (\text{A.10})$$

The parameter  $\mu$  is a location parameter and  $\sigma$  is a scale parameter. The normal distribution does not have a shape parameter since its density is always the same bell-shaped curve.<sup>2</sup> The standard normal CDF is

$$\Phi(y) = \int_{-\infty}^y \phi(u) du.$$

$\Phi$  can be evaluated using software such as R's `pnorm` function. If  $Y$  is  $N(\mu, \sigma^2)$ , then since  $Y = \mu + \sigma Z$ , where  $Z$  is standard normal, by Result A.1,

$$F_Y(y) = \Phi\{(y-\mu)/\sigma\}. \quad (\text{A.11})$$

Normal distribution are also called Gaussian distributions after the great German mathematician Carl Friedrich Gauss.

### Normal Quantiles

The  $q$ -quantile of the  $N(0, 1)$  distribution is  $\Phi^{-1}(q)$  and, more generally, the  $q$ -quantile of an  $N(\mu, \sigma^2)$  distribution is  $\mu + \sigma\Phi^{-1}(q)$ . The  $\alpha$ -upper quantile of  $\Phi$ , that is,  $\Phi^{-1}(1 - \alpha)$ , is denoted by  $z_\alpha$ . As shown later,  $z_\alpha$  is widely used for confidence intervals.

For example,  $z_{0.1}$  and  $z_{0.01}$  are 1.282 and 2.326, respectively, as can be seen in the following R output:

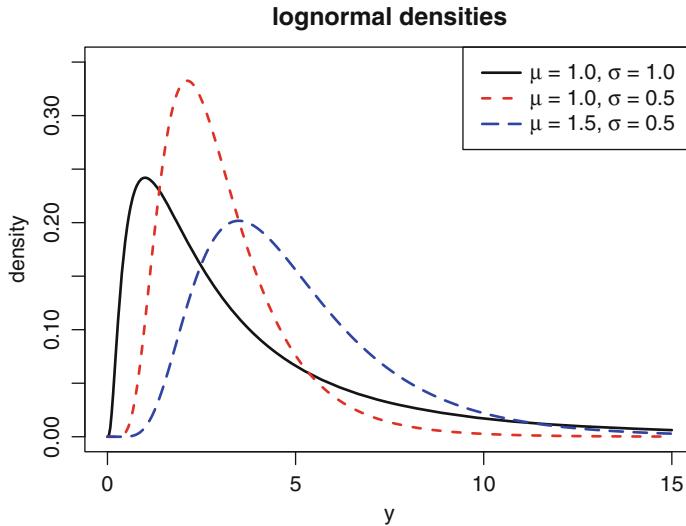
```
> round(qnorm(c(0.1, 0.01), lower.tail = FALSE), 3)
[1] 1.282 2.326
```

### A.9.4 The Lognormal Distribution

If  $Z$  is distributed  $N(\mu, \sigma^2)$ , then  $Y = \exp(Z)$  is said to have a  $\text{Lognormal}(\mu, \sigma^2)$  distribution. In other words,  $Y$  is *lognormal* if its logarithm is normally

---

<sup>2</sup> In contrast, a  $t$ -density is also a bell curve, but the exact shape of the bell depends on a shape parameter, the degrees of freedom which is a tail index.



**Fig. A.1.** Examples of lognormal probability densities. Here  $\mu$  and  $\sigma$  are the log-mean and log-standard deviation, that is, the mean and standard deviation of the logarithm of the lognormal random variable.

distributed. We will call  $\mu$  the log-mean and  $\sigma$  the log-standard deviation. Also,  $\sigma^2$  will be called the log-variance.

The median of  $Y$  is  $\exp(\mu)$  and the expected value of  $Y$  is  $\exp(\mu + \sigma^2/2)$ .<sup>3</sup> The expectation is larger than the median because the lognormal distribution is right skewed, and the skewness is more extreme with larger values of  $\sigma$ . Skewness is discussed further in Sect. 5.4. The probability density functions of several lognormal distributions are shown in Fig. A.1.

The log-mean  $\mu$  is a scale parameter and the log-standard deviation  $\sigma$  is a shape parameter. The lognormal distribution does not have a location parameter since its support is fixed to start at 0.

Use the functions `plnorm()`, `dlnorm()`, `qlnorm()`, and `rlnorm()` for the lognormal distribution. For example,

```
> options(digits = 3)
> dlnorm(0.5, meanlog = 1, sdlog = 2)
[1] 0.279
```

computes the lognormal density at 0.5 when the log-mean is 1 and the log-standard deviation is 2.

---

<sup>3</sup> It is important to remember that if  $Y$  is lognormal( $\mu, \sigma$ ), then  $\mu$  is the expected value of  $\log(Y)$ , not of  $Y$ .

### A.9.5 Exponential and Double-Exponential Distributions

The *exponential distribution* with scale parameter  $\theta > 0$ , which we denote by  $\text{Exponential}(\theta)$ , has CDF

$$F(y) = 1 - e^{-y/\theta}, \quad y > 0.$$

The  $\text{Exponential}(\theta)$  distribution has PDF

$$f(y) = \frac{e^{-y/\theta}}{\theta}, \quad (\text{A.12})$$

expected value  $\theta$ , and standard deviation  $\theta$ . The inverse CDF is

$$F^{-1}(y) = -\theta \log(1 - y), \quad 0 < y < 1.$$

Use the functions `pexp()`, `dexp()`, `qexp()`, and `rexp()` for the exponential distribution.

The *double-exponential* or *Laplace distribution* with mean  $\mu$  and scale parameter  $\theta$  has PDF

$$f(y) = \frac{e^{-|y-\mu|/\theta}}{2\theta}. \quad (\text{A.13})$$

If  $Y$  has a double-exponential distribution with mean  $\mu$ , then  $|Y - \mu|$  has an exponential distribution. A double-exponential distribution has a standard deviation of  $\sqrt{2}\theta$ . The mean  $\mu$  is a location parameter and  $\theta$  is a scale parameter.

### A.9.6 Gamma and Inverse-Gamma Distributions

The *gamma distribution* with scale parameter  $b > 0$  and shape parameter  $\alpha > 0$  has density

$$\frac{y^{\alpha-1}}{\Gamma(\alpha)b^\alpha} \exp(-y/b),$$

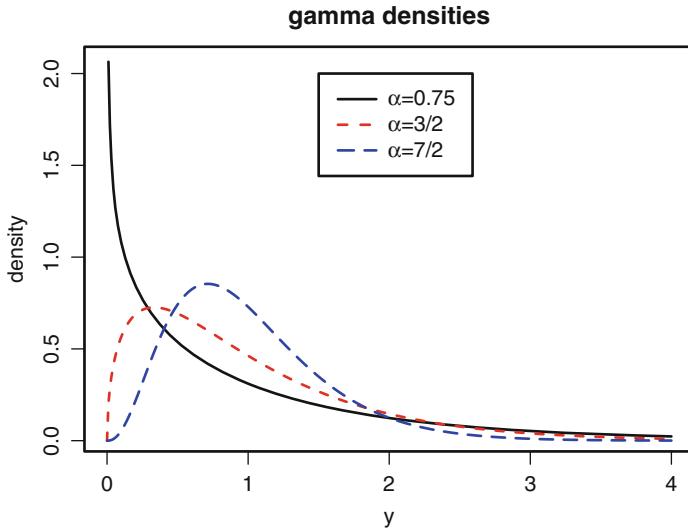
where  $\Gamma$  is the gamma function defined in Sect. 5.5.2. The mean, variance, and skewness coefficient of this distribution are  $b\alpha$ ,  $b^2\alpha$ , and  $2\alpha^{-1/2}$ , respectively. Figure A.2 shows gamma densities with shape parameters equal to 0.75, 3/2, and 7/2 and each with a mean equal to 1.

The gamma distribution is often parameterized using  $\beta = 1/b$ , so that the density is

$$\frac{\beta^\alpha y^{\alpha-1}}{\Gamma(\alpha)} \exp(-\beta y).$$

With this form of the parameterization,  $\beta$  is an *inverse-scale parameter* and the mean and variance are  $\alpha/\beta$  and  $\alpha/\beta^2$ . Also,  $\beta$  is often called the rate parameter, e.g., in R.

Use the functions `pgamma()`, `dgamma()`, `qgamma()`, and `rgamma()` for the gamma distribution. For example, the median of the gamma distribution with  $\alpha = 2$  and  $\beta = 3$  can be computed in two equivalent ways:



**Fig. A.2.** Examples of gamma probability densities with differing shape parameters. In each case, the scale parameter has been chosen so that the expectation is 1.

```
> qgamma(0.5, shape = 2, rate = 3)
[1] 0.559
> qgamma(0.5, shape = 2, scale = 1/3)
[1] 0.559
```

If  $X$  has a gamma distribution with inverse-scale parameter  $\beta$  and shape parameter  $\alpha$ , then we say that  $1/X$  has an *inverse-gamma distribution* with scale parameter  $\beta$  and shape parameter  $\alpha$ . The mean of this distribution is  $\beta/(\alpha - 1)$  provided  $\alpha > 1$  and the variance is  $\beta^2/[(\alpha - 1)^2(\alpha - 2)]$  provided that  $\alpha > 2$ .

### A.9.7 Beta Distributions

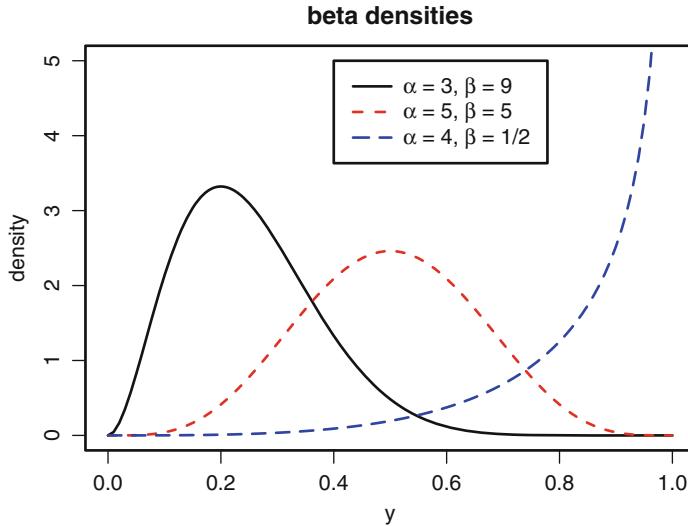
The beta distribution with shape parameters  $\alpha > 0$  and  $\beta > 0$  has density

$$\frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)\Gamma(\beta)} y^{\alpha-1}(1-y)^{\beta-1}, \quad 0 < y < 1. \quad (\text{A.14})$$

The mean and variance are  $\alpha/(\alpha + \beta)$  and  $(\alpha\beta)/[(\alpha + \beta)^2(\alpha + \beta + 1)]$ , and if  $\alpha > 1$  and  $\beta > 1$ , then the mode is  $(\alpha - 1)/(\alpha + \beta - 2)$ .

Figure A.3 shows beta densities for several choices of shape parameters. A beta density is right-skewed, symmetric about 1/2, or left-skewed depending on whether  $\alpha < \beta$ ,  $\alpha = \beta$ , or  $\alpha > \beta$ .

Use the functions `pbeta()`, `dbeta()`, `qbeta()`, and `rbeta()` for the beta distribution. For example, the code below created Fig. A.3.



**Fig. A.3.** Examples of beta probability densities with differing shape parameters.

```
pdf("beta_densities.pdf", width = 6, height = 5) ## Figure A.3
par(lwd = 2)
x = seq(0, 1, 0.01)
plot(x, dbeta(x, 3, 9), type = "l", lty = 1, xlab = "y",
      ylab = "density", main = "beta densities", ylim = c(0, 5))
lines(x, dbeta(x, 5, 5), type = "l", lty = 2, col = "red")
lines(x, dbeta(x, 4, 1/2), type = "l", lty = 5, col = "blue")
legend(0.4, 5, c(
  expression(paste(alpha," = 3, ",beta," = 9")),
  expression(paste(alpha," = 5, ",beta," = 5")),
  expression(paste(alpha," = 4, ",beta," = 1/2"))),
  lty = c(1, 2, 5), col = c("black", "red", "blue"), lwd = 2)
graphics.off()
```

### A.9.8 Pareto Distributions

A random variable  $X$  has a Pareto distribution, named after the Swiss economics professor Vilfredo Pareto (1848–1923), if its CDF for some  $a > 0$

$$F(x) = 1 - \left(\frac{c}{x}\right)^a, \quad x > c, \tag{A.15}$$

where  $c > 0$  is the minimum possible value of  $X$ .

The PDF of the distribution in (A.15) is

$$f(x) = \frac{ac^a}{x^{a+1}}, \quad x > c, \tag{A.16}$$

so a Pareto distribution has polynomial tails and  $a$  is the *tail index*. It is also called the *Pareto constant*.

## A.10 Sampling a Normal Distribution

A common situation is that we have a random sample from a normal distribution and we wish to have confidence intervals for the mean and variance or test hypotheses about these parameters. Then, the following distributions are very important, since they are the basis for many commonly used confidence intervals and tests.

### A.10.1 Chi-Squared Distributions

Suppose that  $Z_1, \dots, Z_n$  are i.i.d.  $N(0, 1)$ . Then, the distribution of  $Z_1^2 + \dots + Z_n^2$  is called the *chi-squared distribution* with  $n$  degrees of freedom. This distribution has an expected value of  $n$  and a variance of  $2n$ . The  $\alpha$ -upper quantile of this distribution is denoted by  $\chi_{\alpha,n}^2$  and is used in tests and confidence intervals about variances; see Appendix A.10.1 for the latter. Also, as discussed in Sect. 5.11,  $\chi_{\alpha,n}^2$  is used in likelihood ratio testing. As an example,  $\chi_{0.05,10}^2$  is 18.31 and can be computed in two ways:

```
> qchisq(0.05, 10, lower.tail = FALSE)
[1] 18.31
> qchisq(0.95, 10)
[1] 18.31
```

So far, the degrees-of-freedom parameter has been an integer-valued, but this can be generalized. The chi-squared distribution with  $\nu$  degrees of freedom is equal to the gamma distribution with scale parameter equal to 2 and shape parameter equal to  $\nu/2$ . Thus, since the shape parameter of a gamma distribution can be any positive value, the chi-squared distribution can be defined for any positive value of  $\nu$  as the gamma distribution with scale and shape parameters equal to 2 and  $\nu/2$ , respectively.

### A.10.2 F-Distributions

If  $U$  and  $W$  are independent and chi-squared-distributed with  $n_1$  and  $n_2$  degrees of freedom, respectively, then the distribution of

$$\frac{U/n_1}{W/n_2}$$

is called the *F-distribution* with  $n_1$  and  $n_2$  degrees of freedom. The  $\alpha$ -upper quantile of this distribution is denoted by  $F_{\alpha,n_1,n_2}$ .  $F_{\alpha,n_1,n_2}$  is used as a critical value for *F*-tests in regression. For example,  $F_{0.95,3,7}$  is 4.347:

```
> qf(0.95, 3, 7)
[1] 4.347
> qf(0.05, 3, 7, lower.tail = FALSE)
[1] 4.347
```

The degrees-of-freedom parameters of the chi-square, *t*-, and *F*-distributions are shape parameters.

## A.11 Law of Large Numbers and the Central Limit Theorem for the Sample Mean

Suppose that  $\bar{Y}_n$  is the mean of an i.i.d. sample  $Y_1, \dots, Y_n$ . We assume that their common expected value  $E(Y_1)$  exists and is finite and call it  $\mu$ . The *law of large numbers* states that

$$P(\bar{Y}_n \rightarrow \mu \text{ as } n \rightarrow \infty) = 1.$$

Thus, the sample mean will be close to the population mean for large enough sample sizes. However, even more is true. The famous *central limit theorem* (CLT) states that if the common variance  $\sigma^2$  of  $Y_1, \dots, Y_n$  is finite, then the probability distribution of  $\bar{Y}_n$  gets closer to a normal distribution as  $n$  converges to  $\infty$ . More precisely, the CLT states that

$$P\{\sqrt{n}(\bar{Y}_n - \mu) \leq y\} \rightarrow \Phi(y/\sigma) \text{ as } n \rightarrow \infty \text{ for all } y. \quad (\text{A.17})$$

Stated differently, for large  $n$ ,  $\bar{Y}$  is approximately  $N(\mu, \sigma^2/n)$ .

Students often misremember or misunderstand the CLT. A common misconception is that a large *population* is approximately normally distributed. The CLT says nothing about the distribution of a population; it is only a statement about the distribution of a sample mean. Also, the CLT does not assume that the population is large; it is the size of the sample that is converging to infinity. Assuming that the sampling is with replacement, the population could be quite small, in fact, with only two elements.

When the variance of  $Y_1, \dots, Y_n$  is infinite, then the limit distribution of  $\bar{Y}_n$  may still exist but will be a nonnormal stable distribution.

Although the CLT was first discovered for the sample mean, other estimators are now known to also have approximate normal distributions for large sample sizes. In particular, there are central limit theorems for the maximum likelihood estimators of Sect. 5.9 and the least-squares estimators discussed in Chap. 9. This is very important, since most estimators we use will be maximum likelihood estimators or least-squares estimators. So, if we have a reasonably large sample, we can assume that these estimators have an approximately normal distribution and the normal distribution can be used for testing and constructing confidence intervals.

## A.12 Bivariate Distributions

Let  $f_{Y_1, Y_2}(y_1, y_2)$  be the joint density of a pair of random variables  $(Y_1, Y_2)$ . Then, the *marginal density* of  $Y_1$  is obtained by “integrating out”  $Y_2$ :

$$f_{Y_1}(y_1) = \int f_{Y_1, Y_2}(y_1, y_2) dy_2,$$

and similarly  $f_{Y_2}(y_2) = \int f_{Y_1, Y_2}(y_1, y_2) dy_1$ .

The *conditional density* of  $Y_2$  given  $Y_1$  is

$$f_{Y_2|Y_1}(y_2|y_1) = \frac{f_{Y_1,Y_2}(y_1, y_2)}{f_{Y_1}(y_1)}. \quad (\text{A.18})$$

Equation (A.18) can be rearranged to give the joint density of  $Y_1$  and  $Y_2$  as the product of a marginal density and a conditional density:

$$f_{Y_1,Y_2}(y_1, y_2) = f_{Y_1}(y_1)f_{Y_2|Y_1}(y_2|y_1) = f_{Y_2}(y_2)f_{Y_1|Y_2}(y_1|y_2). \quad (\text{A.19})$$

The *conditional expectation* of  $Y_2$  given  $Y_1$  is just the expectation calculated using  $f_{Y_2|Y_1}(y_2|y_1)$ :

$$E(Y_2|Y_1 = y_1) = \int y_2 f_{Y_2|Y_1}(y_2|y_1) dy_2,$$

which is, of course, a function of  $y_1$ . The conditional variance of  $Y_2$  given  $Y_1$  is

$$\text{Var}(Y_2|Y_1 = y_1) = \int \{y_2 - E(Y_2|Y_1 = y_1)\}^2 f_{Y_2|Y_1}(y_2|y_1) dy_2.$$

A formula that is important elsewhere in this book is

$$f_{Y_1, \dots, Y_n}(y_1, \dots, y_n) = f_{Y_1}(y_1)f_{Y_2|Y_1}(y_2|y_1) \cdots f_{Y_n|Y_1, \dots, Y_{n-1}}(y_n|y_1, \dots, y_{n-1}), \quad (\text{A.20})$$

which follows from repeated use of (A.19).

The marginal mean and variance are related to the conditional mean and variance by

$$E(Y) = E\{E(Y|X)\} \quad (\text{A.21})$$

and

$$\text{Var}(Y) = E\{\text{Var}(Y|X)\} + \text{Var}\{E(Y|X)\}. \quad (\text{A.22})$$

Result (A.21) has various names, especially the *law of iterated expectations* and the *tower rule*.

Another useful formula is that if  $Z$  is a function of  $X$ , then

$$E(ZY|X) = ZE(Y|X). \quad (\text{A.23})$$

The idea here is that, given  $X$ ,  $Z$  is constant and can be factored outside the conditional expectation.

## A.13 Correlation and Covariance

Expectations and variances summarize the individual behavior of random variables. If we have two random variables,  $X$  and  $Y$ , then it is convenient to have some way to summarize their joint behavior—correlation and covariance do this.

The *covariance* between two random variables  $X$  and  $Y$  is

$$\text{Cov}(X, Y) = \sigma_{XY} = E\left[\{X - E(X)\}\{Y - E(Y)\}\right].$$

The two notations  $\text{Cov}(X, Y)$  and  $\sigma_{XY}$  will be used interchangeably. If  $(X, Y)$  is continuously distributed, then using (A.36), we have

$$\sigma_{XY} = \int \{x - E(X)\}\{y - E(Y)\}f_{XY}(x, y) dx dy.$$

The following are useful formulas:

$$\sigma_{XY} = E(XY) - E(X)E(Y), \quad (\text{A.24})$$

$$\sigma_{XY} = E[\{X - E(X)\}Y], \quad (\text{A.25})$$

$$\sigma_{XY} = E[\{Y - E(Y)\}X], \quad (\text{A.26})$$

$$\sigma_{XY} = E(XY) \text{ if } E(X) = 0 \text{ or } E(Y) = 0. \quad (\text{A.27})$$

The covariance between two variables measures the linear association between them, but it is also affected by their variability; all else equal, random variables with larger standard deviations have a larger covariance. Correlation is covariance after this size effect has been removed, so that correlation is a pure measure of how closely two random variables are related, or more precisely, linearly related. The *Pearson correlation coefficient* between  $X$  and  $Y$  is

$$\text{Corr}(X, Y) = \rho_{XY} = \sigma_{XY}/\sigma_X \sigma_Y. \quad (\text{A.28})$$

The Pearson correlation coefficient is sometimes called simply the correlation coefficient, though there are other types of correlation coefficients; see Sect. 8.5.

Given a bivariate sample  $\{(X_i, Y_i)\}_{i=1}^n$ , the sample covariance, denoted by  $s_{XY}$  or  $\hat{\sigma}_{XY}$ , is

$$s_{XY} = \hat{\sigma}_{XY} = (n - 1)^{-1} \sum_{i=1}^n (X_i - \bar{X})(Y_i - \bar{Y}), \quad (\text{A.29})$$

where  $\bar{X}$  and  $\bar{Y}$  are the sample means. Often the factor  $(n - 1)^{-1}$  is replaced by  $n^{-1}$ , but this change has little effect relative to the random variation in  $\hat{\sigma}_{XY}$ . The *sample correlation* is

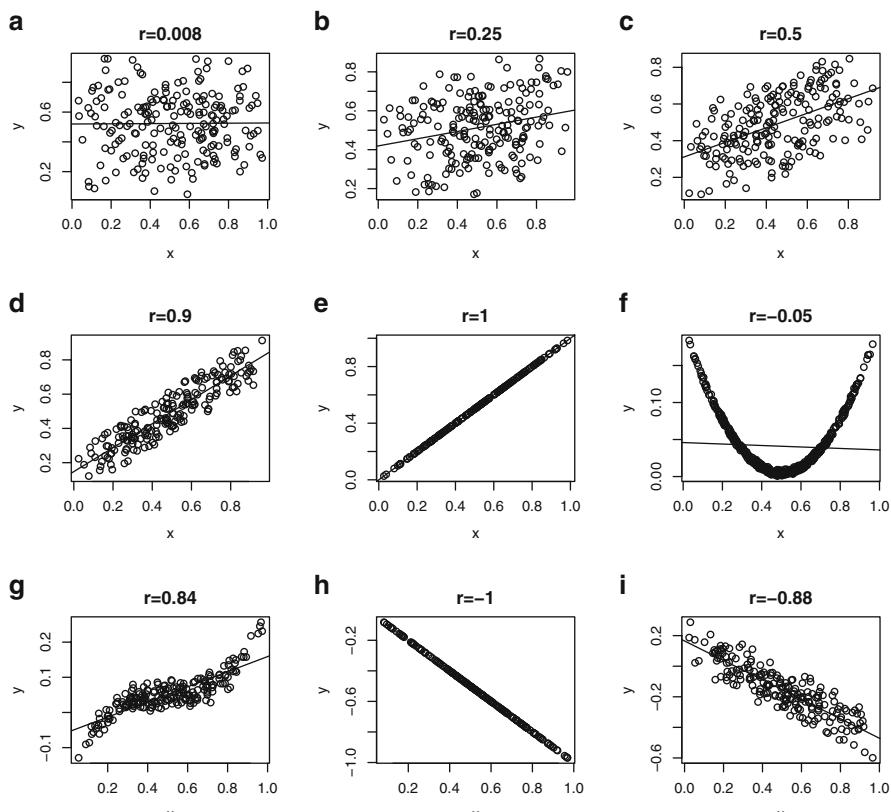
$$\hat{\rho}_{XY} = r_{XY} = \frac{s_{XY}}{s_X s_Y}, \quad (\text{A.30})$$

where  $s_X$  and  $s_Y$  are the sample standard deviations.

To provide the reader with a sense of what particular values of a correlation coefficient imply about the relationship between two random variables, Fig. A.4 shows scatterplots and the sample correlation coefficients for nine bivariate random samples. A *scatterplot* is just a plot of a bivariate sample,  $\{(X_i, Y_i)\}_{i=1}^n$ . Each plot also contains the *linear least-squares fit* (Chap. 9) to illustrate the linear relationship between  $y$  and  $x$ . Notice that

- an absolute correlation of 0.25 or less is weak—see panels (a) and (b);
- an absolute correlation of 0.5 is only moderately strong—see (c);
- an absolute correlation of 0.9 is strong—see (d);
- an absolute correlation of 1 implies an exact linear relationship—see (e) and (h);
- a strong nonlinear relationship may or may not imply a high correlation—see (f) and (g);
- positive correlations imply an increasing relationship (as  $X$  increases,  $Y$  increases on average)—see (b)–(e) and (g);
- negative correlations imply a decreasing relationship (as  $X$  increases,  $Y$  decreases on average)—see (h) and (i).

If the correlation between two random variables is equal to 0, then we say that they are *uncorrelated*.



**Fig. A.4.** Sample correlation coefficients for nine random samples. Each plot also contains the linear regression line of  $y$  on  $x$ .

If  $X$  and  $Y$  are independent, then for all functions  $g$  and  $h$ ,

$$E\{g(X)h(Y)\} = E\{g(X)\}E\{h(Y)\}. \quad (\text{A.31})$$

This fact can be used to prove that if  $X$  and  $Y$  are independent, then  $\sigma_{XY} = 0$ , so the variables are uncorrelated. The opposite is not true. For example, if  $X$  is uniformly distributed on  $[-1, 1]$  and  $Y = X^2$ , then a simple calculation shows that  $\sigma_{XY} = 0$ , but the two random variables are not independent. The key point here is that  $Y$  is related to  $X$ , in fact, completely determined by  $X$ , but the relationship is highly nonlinear and correlation measures linear association.

Another example of random variables that are uncorrelated but dependent is the bivariate  $t$ -distribution. For this distribution, the two variates are dependent even when their correlation is 0; see Sect. 7.6.

If  $E(Y|X) = 0$ , then  $Y$  and  $X$  are uncorrelated, since

$$E(Y) = E\{E(Y|X)\} = 0 \quad (\text{A.32})$$

by the law of iterated expectations, and then

$$\text{Cov}(Y, X) = E(YX) = E\{E(YX|X)\} = E\{XE(Y|X)\} = 0 \quad (\text{A.33})$$

by (A.27), a second application of the law of iterated expectations, (A.23) with  $Z = X$ , and (A.32).

Result (A.22) has an important interpretation. If  $X$  is known and one needs to predict  $Y$ , then  $E(Y|X)$  is the best predictor in that it minimizes the expected squared prediction error. If the best predictor is used, then the prediction error is  $Y - E(Y|X)$  and  $E\{Y - E(Y|X)\}^2$  is the expected squared prediction error. From the law of iterated expectations, that latter is

$$E\{Y - E(Y|X)\}^2 = E\left(E\{[Y - E(Y|X)]^2|X\}\right) = E\{\text{Var}(Y|X)\}, \quad (\text{A.34})$$

the first summand on the right-hand side of (A.22). Also,  $\text{Var}\{E(Y|X)\}$ , the second summand there, is the variability of the best predictor and a measure of how well  $E(Y|X)$  can track  $Y$ —the more  $E(Y|X)$  can vary, the better it can track  $Y$ . Therefore, the sum of the tracking ability and the expected squared prediction error is the constant  $\text{Var}(Y)$ —increasing the tracking ability decreases the expected squared prediction error.

Some insight can be gained by looking at the worst and best cases. The worst case is when  $X$  is independent of  $Y$ . Then,  $E(Y|X) = E(Y)$ , the tracking ability is  $\text{Var}\{E(Y|X)\} = 0$ , and the expected squared prediction takes on its maximum value,  $\text{Var}(Y)$ . The best case is when  $Y$  is a function of  $X$ , say  $y = g(X)$  for some  $g$ . Then,  $E(Y|X) = g(X) = Y$ , the prediction error is 0, and the tracking ability is  $\text{Var}(Y)$ , its maximum possible value.

### A.13.1 Normal Distributions: Conditional Expectations and Variance

The calculation of conditional expectations and variances can be difficult for some probability distributions, but it is quite easy for a pair  $(Y_1, Y_2)$  that has a bivariate normal distribution.

For a bivariate normal pair, the conditional expectation of  $Y_2$  given  $Y_1$  equals the best linear predictor<sup>4</sup> of  $Y_2$  given  $Y_1$ :

$$E(Y_2|Y_1 = y_1) = E(Y_2) + \frac{\sigma_{Y_1, Y_2}}{\sigma_{Y_1}^2} \{y_1 - E(Y_1)\}.$$

Therefore, for normal random variables, best linear prediction is the same as best prediction. Also, the conditional variance of  $Y_2$  given  $Y_1$  is the expected squared prediction error:

$$\text{Var}(Y_2|Y_1 = y_1) = \sigma_{Y_2}^2(1 - \rho_{Y_1, Y_2}^2). \quad (\text{A.35})$$

In general,  $\text{Var}(Y_2|Y_1 = y_1)$  is a function of  $y_1$  but we see in (A.35) that for the special case of a bivariate normal distribution,  $\text{Var}(Y_2|Y_1 = y_1)$  is constant, that is, independent of  $y_1$ .

## A.14 Multivariate Distributions

Multivariate distributions generalized the bivariate distributions of Appendix A.12. A *random vector* is a vector whose elements are random variable. A random vector of continuously distributed random variables,  $\mathbf{Y} = (Y_1, \dots, Y_d)$ , has a *multivariate probability density function*  $f_{Y_1, \dots, Y_d}(y_1, \dots, y_d)$  if

$$P\{(Y_1, \dots, Y_d) \in A\} = \int \int_A f_{Y_1, \dots, Y_d}(y_1, \dots, y_d) dy_1 \cdots dy_d$$

for all sets  $A \subset \Re^p$ .

The PDF of  $Y_j$  is obtained by integrating the other variates out of  $f_{Y_1, \dots, Y_d}$ :

$$f_{Y_j}(y_j) = \int_{y_1} \cdots \int_{y_{j-1}} \int_{y_{j+1}} \cdots \int_{y_d} f_{Y_1, \dots, Y_d}(y_1, \dots, y_d) dy_1 \cdots dy_{j-1} dy_{j+1} \cdots dy_d.$$

Similarly, the PDF of any subset of  $(Y_1, \dots, Y_d)$  is obtained by integrating the other variables out of  $f_{Y_1, \dots, Y_d}(y_1, \dots, y_d)$ .

The expectation of a function  $g$  of  $Y_1, \dots, Y_d$  is given by the formula

$$E\{g(Y_1, \dots, Y_d)\} = \int_{y_1} \cdots \int_{y_d} g(y_1, \dots, y_d) f_{Y_1, \dots, Y_d}(y_1, \dots, y_d) dy_1 \cdots dy_d. \quad (\text{A.36})$$

---

<sup>4</sup> See Sect. 11.9.

If  $Y_1, \dots, Y_d$  are discrete, then their joint probability distribution specifies  $P\{Y_1 = x_1, \dots, Y_d = y_d\}$  for all values of  $y_1, \dots, y_d$ . If  $Y_1, \dots, Y_d$  are discrete and independent, then

$$P\{Y_1 = y_1, \dots, Y_d = y_d\} = P\{Y_1 = y_1\} \cdots P\{Y_d = y_d\}. \quad (\text{A.37})$$

The joint CDF of  $Y_1, \dots, Y_d$ , whether they are continuous or discrete, is

$$F_{Y_1, \dots, Y_d}(x_1, \dots, y_d) = P(Y_1 \leq y_1, \dots, Y_d \leq y_d).$$

Suppose there is a sample of size  $n$  of  $d$ -dimensional random vectors,  $\{\mathbf{Y}_i = (Y_{i,1}, \dots, Y_{i,d}) : i = 1, \dots, n\}$ . Then the empirical CDF is

$$F_n(y_1, \dots, y_d) = \frac{\sum_{i=1}^n I\{Y_{i,j} \leq y_j, \text{ for } j = 1, \dots, d\}}{n}. \quad (\text{A.38})$$

### A.14.1 Conditional Densities

The conditional density of  $Y_1, \dots, Y_q$  given  $Y_{q+1}, \dots, Y_d$ , where  $1 \leq q < d$ , is

$$f_{Y_1, \dots, Y_q | Y_{q+1}, \dots, Y_d}(y_1, \dots, y_q | y_{q+1}, \dots, y_d) = \frac{f_{Y_1, \dots, Y_d}(y_1, \dots, y_d)}{f_{Y_{q+1}, \dots, Y_d}(y_{q+1}, \dots, y_d)}. \quad (\text{A.39})$$

Since  $Y_1, \dots, Y_d$  can be arranged in any order that is convenient, (A.39) provides a formula for the conditional density of any subset of the variables, given the other variables. Also, (A.39) can be rearranged to give the *multiplicative formula*

$$\begin{aligned} & f_{Y_1, \dots, Y_d}(y_1, \dots, y_d) \\ &= f_{Y_1, \dots, Y_q | Y_{q+1}, \dots, Y_d}(y_1, \dots, y_q | y_{q+1}, \dots, y_d) f_{Y_{q+1}, \dots, Y_d}(y_{q+1}, \dots, y_d). \end{aligned} \quad (\text{A.40})$$

Repeated use of (A.40) gives a formula that will be useful later for calculating likelihoods for dependent data

$$\begin{aligned} & f_{Y_1, \dots, Y_d}(y_1, \dots, y_d) \\ &= f_{Y_1}(y_1) f_{Y_2 | Y_1}(y_2 | y_1) f_{Y_3 | Y_1, Y_2}(y_3 | y_1, y_2) \cdots f_{Y_d | Y_1, \dots, Y_{d-1}}(y_d | y_1, \dots, y_{d-1}). \end{aligned} \quad (\text{A.41})$$

If  $Y_1, \dots, Y_d$  are independent, then

$$f_{Y_1, \dots, Y_d}(y_1, \dots, y_d) = f_{Y_1}(y_1) \cdots f_{Y_d}(y_d). \quad (\text{A.42})$$

## A.15 Stochastic Processes

A discrete-time stochastic process is a sequence of random variables  $\{Y_1, Y_2, Y_3, \dots\}$ . The distribution of  $Y_n$  is called its marginal distribution. The process is said to be Markov, or Markovian, if the conditional distribution of  $Y_{n+1}$

given  $\{Y_1, Y_2, \dots, Y_n\}$  equals the conditional distribution of  $Y_{n+1}$  given  $Y_n$ , so  $Y_{n+1}$  depends only on the previous value of the process. The AR(1) process in Sect. 12.4 is a simple example of a Markov process. A process generated by computer simulation will be Markov if only  $Y_n$  and random numbers independent of  $\{Y_1, Y_2, \dots, Y_{n-1}\}$  are used to generate  $Y_{n+1}$ . An important example is Markov chain Monte Carlo, the topic of Sect. 20.7.

A distribution  $\pi$  is a stationary distribution for a Markov process if, for all  $n$ ,  $Y_{n+1}$  has distribution  $\pi$  whenever  $Y_n$  has distribution  $\pi$ .

Stochastic processes can also have a continuous-time parameter. Examples are Brownian motion and geometric Brownian motion, which are used, *inter alia*, to model the log-prices and prices of equities, respectively, in continuous time.

## A.16 Estimation

### A.16.1 Introduction

One of the major areas of statistical inference is estimation of unknown parameters, such as a population mean, from data. An estimator is defined as any function of the observed data. The key question is which of many possible estimators should be used. If  $\theta$  is an unknown parameter and  $\hat{\theta}$  is an estimator, then  $E(\hat{\theta}) - \theta$  is called the *bias* and  $E\{\hat{\theta} - \theta\}^2$  is called the *mean-squared error* (MSE). One seeks estimators that are efficient, that is, having the smallest possible value of the MSE (or of some other measure of inaccuracy). It can be shown from simple algebra that the MSE is the squared bias plus the variance, that is,

$$E\{\hat{\theta} - \theta\}^2 = \{E(\hat{\theta}) - \theta\}^2 + \text{Var}(\hat{\theta}), \quad (\text{A.43})$$

so an efficient estimator will have both a small bias and a small variance. An estimator with a zero bias is called *unbiased*. However, it is not necessary to use an unbiased estimator—we only want the bias to be small, not necessarily exactly zero. One should be willing to accept a small bias if this leads to a significant reduction in variance.

The most popular methods of estimation are least squares (Sect. 9.2.1), maximum likelihood (Sects. 5.9 and 5.14), and Bayes estimation (Chap. 20).

### A.16.2 Standard Errors

When an estimator is calculated from a random sample, it is a random variable, but this fact is often not appreciated by beginning students. When first exposed to statistical estimation, students tend not to think of estimators such as a sample mean as random. If we have only a single sample, then the sample mean does not *appear* random. However, if we realize that the observed

sample is only one of many possible samples that could have been drawn, and that each sample has a different sample mean, then we see that the mean is in fact random.

Since an estimator is a random variable, it has an expectation and a standard deviation. We have already seen that the difference between its expectation and the parameter is called the bias. The standard deviation of an estimator is called its *standard error*. If there are unknown parameters in the formula for this standard deviation, then they can be replaced by estimates. If  $\hat{\theta}$  is an estimator of  $\theta$ , then  $s_{\hat{\theta}}$  will denote its standard error with any unknown parameters replaced by estimates.

*Example A.1. The standard error of the mean*

Suppose that  $Y_1, \dots, Y_n$  are i.i.d. with mean  $\mu$  and variance  $\sigma^2$ . Then, it follows from (7.13) that the standard deviation of  $\bar{Y}$  is  $\sigma/\sqrt{n}$ . Thus,  $\sigma/\sqrt{n}$ , or when  $\sigma$  is unknown  $s_Y/\sqrt{n}$ , is called the standard error of the sample mean. That is,  $s_{\bar{Y}}$  is  $\sigma/\sqrt{n}$  or  $s_Y/\sqrt{n}$  depending on whether or not  $\sigma$  is known.  $\square$

## A.17 Confidence Intervals

Instead of estimating an unknown parameter by a single number, it is often better to provide a range of numbers that gives a sense of the uncertainty of the estimate. Such ranges are called *interval estimates*. One type of interval estimate, the Bayesian credible interval, is introduced in Chap. 20. Another type of interval estimate is the confidence interval. A *confidence interval* is defined by the requirement that the probability that the interval will include the true parameter is a specified value called the *confidence coefficient*, so, for example, if a large number of independent 90 % intervals are constructed, then approximately 90 % of them will contain the parameter.

### A.17.1 Confidence Interval for the Mean

If  $\bar{Y}$  is the mean of a sample from a normal population, then

$$\bar{Y} \pm t_{\alpha/2, n-1} s_{\bar{Y}} \quad (\text{A.44})$$

is a confidence interval with  $(1 - \alpha)$  confidence. This confidence interval is derived in Sect. 6.3.2. If  $\alpha = 0.05$  (0.95 or 95 % confidence) and if  $n$  is reasonably large, then  $t_{\alpha/2, n-1}$  is approximately 2, so  $\bar{Y} \pm 2 s_{\bar{Y}}$  is often used as an approximate 95 % confidence interval. Since  $s_{\bar{Y}} = s_Y/\sqrt{n}$ , the confidence can also be written as  $\bar{Y} \pm 2 s_Y/\sqrt{n}$ . When  $n$  is reasonably large, say 20 or more, then  $\bar{Y}$  will be approximately normally distributed by the central limit theorem, and the assumption that the population itself is normal can be dropped.

*Example A.2. Confidence interval for a normal mean*

Suppose we have a sample of size 25 from a normal distribution,  $s_Y^2 = 2.7$ ,  $\bar{Y} = 16.1$ , and we want a 99 % confidence interval for  $\mu$ . We need  $t_{0.005,24}$ . This quantile can be found, for example, using the R function `qt` and  $t_{0.005,24} = 2.797$ . Then, the 99 % confidence interval for  $\mu$  is

$$16.1 \pm \frac{(2.797)\sqrt{2.7}}{\sqrt{25}} = 16.1 \pm 0.919 = [15.18, 17.02].$$

Since  $n = 25$  is reasonably large, this interval should have approximately 99 % confidence even if the population is not normally distributed. The exception would be if the population was extremely heavily skewed or had very heavy tails; in such cases a sample size larger than 25 might be necessary for this confidence interval to have near 99 % coverage.

Just how large a sample is needed for  $\bar{Y}$  to be nearly normally distributed depends on the population. If the population is symmetric and the tails are not extremely heavy, then approximate normality is often achieved with  $n$  around 10. For skewed populations, 30 observations may be needed, and even more in extreme cases. If the data appear to come from a highly skewed or heavy-tailed population, it might be better to assume a parametric model and compute the MLE as discussed in Chap. 5 and perhaps to use the bootstrap (Chap. 6) for finding the confidence interval.

The function `t.test()` computes a confidence interval for a normal mean. The output below gives a 99 % confidence interval for daily log-returns on Ford using `t.test()` and then using (A.44). The interval is  $(-0.000417, 0.003407)$

```
> ford = read.csv("RecentFord.csv")
> returns = diff(log(ford[, 7]))
> options(digits = 3)
> t.test(returns, conf.level = 0.99)

One Sample t-test

data:  returns
t = 2.02, df = 1256, p-value = 0.04388
alternative hypothesis: true mean is not equal to 0
99 percent confidence interval:
-0.000417  0.003407
sample estimates:
mean of x
0.00149

> n = length(returns)
> mean(returns) + c(-1, 1) *
  qt(0.995, n - 1) * sd(returns) / sqrt(n)
[1] -0.000417  0.003407
```

□

### A.17.2 Confidence Intervals for the Variance and Standard Deviation

A  $(1 - \alpha)$  confidence interval for the variance of a normal distribution is given by

$$\left[ \frac{(n-1)s_Y^2}{\chi_{\alpha/2, n-1}^2}, \frac{(n-1)s_Y^2}{\chi_{1-\alpha/2, n-1}^2} \right], \quad (\text{A.45})$$

where  $n$  is the sample size,  $s_Y^2$  is the sample variance given by equation (A.7), and, as defined in Appendix A.10.1,  $\chi_{\gamma, n-1}^2$  is the  $(1 - \gamma)$ -quantile of the chi-square distribution with  $n - 1$  degrees of freedom.

*Example A.3. Confidence interval for a normal standard deviation*

Suppose we have a sample of size 25 from a normal distribution,  $s_Y^2 = 2.7$ , and we want a 90 % confidence interval for  $\sigma^2$ . The quantiles we need for constructing the interval are  $\chi_{0.95, 24}^2 = 13.848$  and  $\chi_{0.05, 24}^2 = 36.415$ . These values can be found using software such as `qchisq()` in R. The 90 % confidence interval for  $\sigma^2$  is

$$\left[ \frac{(2.7)(24)}{36.415}, \frac{(2.7)(24)}{13.848} \right] = [1.78, 4.68].$$

Taking square roots of both endpoints, we get  $1.33 < \sigma < 2.16$  as a 90 % confidence interval for the standard deviation.

As another example, confidence intervals for the variance and standard deviation of daily Ford returns are calculate below. The confidence interval for the standard deviation is (0.0253, 0.0273).

```
> ford = read.csv("RecentFord.csv")
> returns = diff(log(ford[,7]))
> n = length(returns)
> options(digits = 3)
> ci = (n - 1) * var(returns) / qchisq(c(0.025, 0.975), n - 1,
  lower.tail = FALSE)
> ci
[1] 0.000639 0.000748
> sqrt(ci)
[1] 0.0253 0.0273
```

□

Unfortunately, the assumption that the population is normally distributed cannot be dispensed with, even if the sample size is large. If a normal probability plot or test of normality (see Sect. 4.4) suggests that the population might be nonnormally distributed, then one might instead construct a confidence interval for  $\sigma$  using the bootstrap; see Chap. 6. Another possibility is to assume a nonnormal parametric model such as the  $t$ -model if the data are symmetric and heavy-tailed; see Example 5.3.

### A.17.3 Confidence Intervals Based on Standard Errors

Many estimators are approximately unbiased and approximately normally distributed. Then, an approximate 95 % confidence interval is the estimator plus or minus twice its standard error; that is,

$$\hat{\theta} \pm 2 s_{\hat{\theta}}$$

is an approximate 95 % confidence interval for  $\theta$ .

## A.18 Hypothesis Testing

### A.18.1 Hypotheses, Types of Errors, and Rejection Regions

Statistical hypothesis testing uses data to decide whether a certain statement called the *null hypothesis* is true. The negation of the null hypothesis is called the *alternative hypothesis*. For example, suppose that  $Y_1, \dots, Y_n$  are i.i.d.  $N(\mu, 1)$  and  $\mu$  is unknown. The null hypothesis could be that  $\mu$  is 1. Then, we write  $H_0: \mu = 1$  and  $H_1: \mu \neq 1$  to denote the null and alternative hypotheses.

There are two types of errors that we hope to avoid. If the null hypothesis is true but we reject it, then we are making a *type I error*. Conversely, if the null hypothesis is false and we accept it, then we are making a *type II error*.

The *rejection region* is the set of possible samples that lead us to reject  $H_0$ . For example, suppose that  $\mu_0$  is a hypothesized value of  $\mu$  and the null hypothesis is  $H_0: \mu = \mu_0$  and the alternative is  $H_1: \mu \neq \mu_0$ . One rejects  $H_0$  if  $|\bar{Y} - \mu_0|$  exceeds an appropriately chosen cutoff value  $c$  called a *critical value*. The rejection region is chosen to keep the probability of a type I error below a prespecified small value called the *level* and often denoted by  $\alpha$ . Typical values of  $\alpha$  used in practice are 0.01, 0.05, or 0.1. As  $\alpha$  is made smaller, the rejection region must be made smaller. In the example, since we reject the null hypothesis when  $|\bar{Y} - \mu_0|$  exceeds  $c$ , the critical value  $c$  gets larger as the  $\alpha$  gets smaller. The value of  $c$  is easy to determine. Assuming that  $\sigma$  is known,  $c$  is  $z_{\alpha/2} \sigma / \sqrt{n}$ , where, as defined in Appendix A.9.3,  $z_{\alpha/2}$  is the  $\alpha/2$ -upper quantile of the standard normal distribution. If  $\sigma$  is unknown, then  $\sigma$  is replaced by  $s_X$  and  $z_{\alpha/2}$  is replaced by  $t_{\alpha/2, n-1}$ , where, as defined in Sect. 5.5.2,  $t_{\alpha/2, n-1}$  is the  $\alpha/2$ -upper quantile of the  $t$ -distribution with  $n - 1$  degrees of freedom. The test using the  $t$ -quantile is called the *one-sample t-test*.

### A.18.2 p-Values

Rather than specifying  $\alpha$  and deciding whether to accept or reject the null hypothesis at that  $\alpha$ , we might ask “for what values of  $\alpha$  do we reject the null hypothesis?” The *p-value* for a sample is defined as the smallest value of  $\alpha$  for

which the null hypothesis is rejected. Stated differently, to perform the test using a given sample, we first find the  $p$ -value of that sample, and then  $H_0$  is rejected if we decide to use  $\alpha$  larger than the  $p$ -value and  $H_0$  is accepted if we use  $\alpha$  smaller than the  $p$ -value. Thus,

- a small  $p$ -value is evidence *against* the null hypothesis while
- a large  $p$ -value shows that the *data are consistent* with the null hypothesis.

*Example A.4. Interpreting p-values*

If the  $p$ -value of a sample is 0.033, then we reject  $H_0$  if we use  $\alpha$  equal to 0.05 or 0.1, but we accept  $H_0$  if we use  $\alpha = 0.01$ .  $\square$

The  $p$ -value not only tells us whether the null hypothesis should be accepted or rejected, but it also tells us whether or not the decision to accept or reject  $H_0$  is a close call. For example, if we are using  $\alpha = 0.05$  and the  $p$ -value were 0.047, then we would reject  $H_0$  but we would know the decision was close. If instead the  $p$ -value were 0.001, then we would know the decision was not so close.

When performing hypothesis tests, statistical software routinely calculates  $p$ -values. Doing this is much more convenient than asking the user to specify  $\alpha$ , and then reporting whether the null hypothesis is accepted or rejected for that  $\alpha$ .

### A.18.3 Two-Sample $t$ -Tests

Two-sample  $t$ -tests are used to test hypotheses about the difference between two population means. The independent-samples  $t$ -test is used when we sample independently from the two populations. Let  $\mu_i$ ,  $\bar{Y}_i$ ,  $s_i$ , and  $n_i$  be the population mean, sample mean, sample standard deviation, and sample size for the  $i$ th sample,  $i = 1, 2$ , respectively. Let  $\Delta_0$  be a hypothesized value of  $\mu_1 - \mu_2$ . We assume that the two populations have the same standard deviation and estimate this parameter by the *pooled standard deviation*, which is

$$s_{\text{pool}} = \left\{ \frac{(n_1 - 1)s_1^2 + (n_2 - 1)s_2^2}{n_1 + n_2 - 2} \right\}^{1/2}. \quad (\text{A.46})$$

The independent-samples  $t$ -statistic is

$$t = \frac{\bar{Y}_1 - \bar{Y}_2 - \Delta_0}{s_{\text{pool}} \sqrt{\frac{1}{n_1} + \frac{1}{n_2}}}.$$

If the hypotheses are  $H_0: \mu_1 - \mu_2 = \Delta_0$  and  $H_1: \mu_1 - \mu_2 \neq \Delta_0$ , then  $H_0$  is rejected if  $|t| > t_{\alpha/2|n_1+n_2-2}$ . If the hypotheses are  $H_0: \mu_1 - \mu_2 \leq \Delta_0$  and  $H_1: \mu_1 - \mu_2 > \Delta_0$ , then  $H_0$  is rejected if  $t > t_{\alpha|n_1+n_2-2}$  and if they are  $H_0: \mu_1 - \mu_2 \geq \Delta_0$  and  $H_1: \mu_1 - \mu_2 < \Delta_0$ , then  $H_0$  is rejected if  $t < -t_{\alpha|n_1+n_2-2}$ .

Sometimes the samples are paired rather than independent. For example, suppose we wish to compare returns on small-cap versus large-cap<sup>5</sup> stocks and for each of  $n$  years we have the returns on a portfolio of small-cap stocks and on a portfolio of large-cap stocks. For any year, the returns on the two portfolios will be correlated, so an independent-samples test is not valid. Let  $d_i = X_{i,1} - X_{i,2}$  be the difference between the observations from populations 1 and 2 for the  $i$ th pair, and let  $\bar{d}$  and  $s_d$  be the sample mean and standard deviation of  $d_1, \dots, d_n$ . The paired-sample  $t$ -statistics is

$$t = \frac{\bar{d} - \Delta_0}{s_d/\sqrt{n}}. \quad (\text{A.47})$$

The rejection regions are the same as for the independent-samples  $t$ -tests except that the degrees-of-freedom parameter for the  $t$ -quantiles is  $n-1$  rather than  $n_1 + n_2 - 2$ .

The power of a test is the probability of correctly rejecting  $H_0$  when  $H_1$  is true. Paired samples are often used to obtain more power. In the example of comparing small- and large-cap stocks, the returns on both portfolios will have high year-to-year variation, but the  $d_i$  will be free of this variation, so that  $s_d$  should be relatively small compared to  $s_1$  and  $s_2$ . A small variation in the data means that  $\mu_1 - \mu_2$  can be more accurately estimated and deviations of this parameter from  $\Delta_0$  are more likely to be detected.

Since  $\bar{d} = \bar{Y}_1 - \bar{Y}_2$ , the numerators in (A.46) and (A.47) are equal. What differs are the denominators. The denominator in (A.47) will be smaller than in (A.46) when the correlation between observations  $(Y_{i,1}, Y_{i,2})$  in a pair is positive. It is the smallness of the denominator in (A.47) that gives the paired  $t$ -test increased power.

Suppose someone had a paired sample but incorrectly used the independent-samples  $t$ -test. If the correlation between  $Y_{i,1}$  and  $Y_{i,2}$  is zero, then the paired samples behave the same as independent samples and the effect of using the incorrect test would be small. Suppose that this correlation is positive. The result of using the incorrect test would be that if  $H_0$  is false, then the true  $p$ -value would be overestimated and one would be less likely to reject  $H_0$  than if the paired-sample test had been used. However, if the  $p$ -value is small, then one can be confident in rejecting  $H_0$  because the  $p$ -value for the paired-sample test would be even smaller.<sup>6</sup> Unfortunately, statistical methods are often used

<sup>5</sup> The market capitalization of a stock is the product of the share price and the number of shares outstanding. If stocks are ranked based on market capitalization, then all stocks below some specified quantile would be small-cap stocks and all above another specified quantile would be large-cap.

<sup>6</sup> An exception would be the rare situation, where  $Y_{i,1}$  and  $Y_{i,2}$  are *negatively* correlated.

by researchers without a solid understanding of the underlying theory, and this can lead to misapplications. The hypothetical use just described of an incorrect test is often a reality, and it is sometimes necessary to evaluate whether the results that are reported can be trusted.

Confidence intervals can also be constructed for the difference between the two means and are

$$\bar{Y}_1 - \bar{Y}_2 \pm t_{\alpha/2|n_1+n_2-2} s_{\text{pool}} \sqrt{\frac{1}{n_1} + \frac{1}{n_2}} \quad (\text{A.48})$$

for unpaired samples and

$$\bar{d} \pm t_{\alpha/2|n_1+n_2-2} s_d / \sqrt{n}. \quad (\text{A.49})$$

for paired samples.

#### *Example A.5. A Paired Two-sample t-test and Confidence Interval*

In the next example, a 95 % confidence interval is created for the difference between the mean daily log-returns on Merck and Pfizer. Since the prices were taken over the same time intervals, the daily log-returns are highly correlated ( $\hat{\rho} = 0.547$ ), so a paired test and interval were used. The confidence interval was also calculated using (A.49).

```
> prices = read.csv("Stock_Bond.csv")
> prices_Merck = prices[ , 11]
> return_Merck = diff(log(prices_Merck))
> prices_Pfizer = prices[ , 13]
> return_Pfizer = diff(log(prices_Pfizer))
> cor(return_Merck,return_Pfizer)
[1] 0.547
> t.test(return_Merck, return_Pfizer, paired = TRUE)

Paired t-test

data: return_Merck and return_Pfizer
t = -0.406, df = 4961, p-value = 0.6849
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
-0.000584 0.000383
sample estimates:
mean of the differences
-1e-04

> differences = return_Merck - return_Pfizer
> n = length(differences)
> mean(differences) + c(-1,1) * qt(0.025, n - 1,
+ lower.tail = FALSE) * sd(differences) / sqrt(n)
[1] -0.000584 0.000383
```

□

### A.18.4 Statistical Versus Practical Significance

When we reject a null hypothesis, we often say there is a *statistically significant effect*. In this context, the word “significant” is easily misconstrued. It does *not* mean that there is an effect of practical importance. For example, suppose we were testing the null hypothesis that the means of two populations are equal versus the alternative that they are unequal. Statistical significance simply means that the two sample means are sufficiently different that this difference cannot reasonably be attributed to mere chance. Statistical significance does *not* mean that the population means are so dissimilar that their difference is of any practical importance. When large samples are used, small and unimportant effects are likely to be statistically significant.

When determining practical significance, confidence intervals are more useful than tests. In the case of the comparison between two population means, it is important to construct a confidence interval and to conclude that there is an effect of practical significance only if *all* differences in that interval are large enough to be of practical importance. How large is “large enough” is *not* a statistical question but rather must be answered by a subject-matter expert. For an example, suppose a difference between the two population means that exceeds 0.2 is considered important, at least for the purpose under consideration. If a 95 % confidence interval were [0.23, 0.26], then with 95 % confidence we could conclude that there is an important difference. If instead the interval were [0.13, 0.16], then we could conclude with 95 % confidence that there is no important difference. If the confidence interval were [0.1, 0.3], then we could not state with 95 % confidence whether the difference is important or not.

## A.19 Prediction

Suppose that  $Y$  is a random variable that is unknown at the present time, for example, a future change in an interest rate or stock price. Let  $\mathbf{X}$  be a known random vector that is useful for predicting  $Y$ . For example, if  $Y$  is a future change in a stock price or a macroeconomic variable,  $\mathbf{X}$  might be the vector of recent changes in that stock price or macroeconomic variable.

We want to find a function of  $\mathbf{X}$ , which we will call  $\hat{Y}(\mathbf{X})$ , that best predicts  $Y$ . By this we mean that the mean-squared error  $E[\{Y - \hat{Y}(\mathbf{X})\}^2]$  is made as small as possible. The function  $\hat{Y}(\mathbf{X})$  that minimizes the mean-squared error will be called the best predictor of  $Y$  based on  $\mathbf{X}$ . Note that  $\hat{Y}(\mathbf{X})$  can be any function of  $\mathbf{X}$ , not necessarily a linear function as in Sect. 11.9.1. The *best predictor* is theoretically simple—it is the conditional expectation of  $Y$  given  $\mathbf{X}$ . That is,  $E(Y|\mathbf{X})$  is the best predictor of  $Y$  in the sense of minimizing  $E[\{Y - \hat{Y}(\mathbf{X})\}^2]$  among *all* possible choices of  $\hat{Y}(\mathbf{X})$  that are arbitrary functions of  $\mathbf{X}$ .

If  $Y$  and  $\mathbf{X}$  are independent, then  $E(Y|\mathbf{X}) = E(Y)$ . If  $\mathbf{X}$  were unobserved, then  $E(Y)$  would be used to predict  $Y$ . Thus, when  $Y$  and  $\mathbf{X}$  are independent, the best predictor of  $Y$  is the same as if  $\mathbf{X}$  were unknown, because  $\mathbf{X}$  contains no information that is useful for prediction of  $Y$ .

In practice, using  $E(Y|\mathbf{X})$  for prediction is not trivial. The problem is that  $E(Y|\mathbf{X})$  may be difficult to estimate whereas the best linear predictor can be estimated by linear regression as described in Chap. 9. However, the newer technique of *nonparametric regression* can be used to estimate  $E(Y|\mathbf{X})$ . Nonparametric regression is discussed in Chap. 21.

## A.20 Facts About Vectors and Matrices

The norm of the vector  $\mathbf{x} = (x_1, \dots, x_p)^\top$  is  $\|\mathbf{x}\| = (\sum_{i=1}^p x_i^2)^{1/2}$ .

A square matrix  $\mathbf{A}$  is diagonal if  $A_{i,j} = 0$  for all  $i \neq j$ . We use the notation  $\text{diag}(d_1, \dots, d_p)$  for a  $p \times p$  diagonal matrix  $\mathbf{A}$  such that  $A_{i,i} = d_i$ .

A matrix  $\mathbf{O}$  is orthogonal if  $\mathbf{O}^\top = \mathbf{O}^{-1}$ . This implies that the columns of  $\mathbf{O}$  are mutually orthogonal (perpendicular) and that their norms are all equal to 1.

Any symmetric matrix  $\boldsymbol{\Sigma}$  has an *eigenvalue-eigenvector decomposition*, eigen-decomposition for short, which is

$$\boldsymbol{\Sigma} = \mathbf{O} \text{diag}(\lambda_i) \mathbf{O}^\top, \quad (\text{A.50})$$

where  $\mathbf{O}$  is an orthogonal matrix whose columns are the eigenvectors of  $\boldsymbol{\Sigma}$  and  $\lambda_1, \dots, \lambda_p$  are the eigenvalues of  $\boldsymbol{\Sigma}$ . Also, if all of  $\lambda_1, \dots, \lambda_p$  are nonzero, then  $\boldsymbol{\Sigma}$  is nonsingular and

$$\boldsymbol{\Sigma}^{-1} = \mathbf{O} \text{diag}(1/\lambda_i) \mathbf{O}^\top.$$

Let  $\mathbf{o}_1, \dots, \mathbf{o}_p$  be the columns of  $\mathbf{O}$ . Then, since  $\mathbf{O}$  is orthogonal,

$$\mathbf{o}_j^\top \mathbf{o}_k = 0 \quad (\text{A.51})$$

for any  $j \neq k$ . Moreover,

$$\mathbf{o}_j^\top \boldsymbol{\Sigma} \mathbf{o}_k = 0 \quad (\text{A.52})$$

for  $j \neq k$ . To see this, let  $\mathbf{e}_j$  be the  $j$ th unit vector, that is, the vector with a one in the  $j$ th coordinate and zeros elsewhere. Then,  $\mathbf{o}_j^\top \mathbf{O} = \mathbf{e}_j^\top$  and  $\mathbf{O}^\top \mathbf{o}_k = \mathbf{e}_k$ , so that for  $j \neq k$ ,

$$\mathbf{o}_j^\top \boldsymbol{\Sigma} \mathbf{o}_k = \mathbf{o}_j^\top \left\{ \mathbf{O} \text{diag}(\lambda_i) \mathbf{O}^\top \right\} \mathbf{o}_k = \lambda_j \lambda_k \mathbf{e}_j^\top \mathbf{e}_k = 0.$$

The eigenvalue-eigenvector decomposition of a covariance matrix is used in Sect. 7.8 to find the orientation of elliptically contoured densities. This decomposition can be important even if the density is not elliptically contoured and is the basis of principal components analysis (PCA).

*Example A.6. An Eigendecomposition*

In the next example, a  $3 \times 3$  symmetric matrix `Sigma` is created and its eigenvalues and eigenvectors are computed using the function `eigen()`. The eigenvalues are in the vector `decomp$values` and the eigenvectors are in the matrix `decomp$vectors`. It is also verified that `decomp$vectors` is an orthogonal matrix.

```
> Sigma = matrix(c(1, 3, 4, 3, 6, 2, 4, 2, 8), nrow = 3,
+                 byrow = TRUE)
> Sigma
     [,1] [,2] [,3]
[1,]    1    3    4
[2,]    3    6    2
[3,]    4    2    8
> decomp = eigen(Sigma)
> decomp
$values
[1] 11.59  4.79 -1.37

$vectors
     [,1]      [,2]      [,3]
[1,] -0.426 -0.0479  0.903
[2,] -0.499 -0.8203 -0.279
[3,] -0.754  0.5699 -0.326

> round(decomp$vectors %*% t(decomp$vectors), 5)
     [,1] [,2] [,3]
[1,]    1    0    0
[2,]    0    1    0
[3,]    0    0    1
```

□

## A.21 Roots of Polynomials and Complex Numbers

The roots of polynomials play an important role in the study of ARMA processes. Let  $p(x) = b_0 + b_1x + \dots + b_px^p$ , with  $b_p \neq 0$ , be a  $p$ th-degree polynomial. The fundamental theorem of algebra states that  $p(x)$  can be factored as

$$b_p(x - r_1)(x - r_2) \cdots (x - r_p),$$

where  $r_1, \dots, r_p$  are the roots of  $p(x)$ , that is, the solutions to  $p(x) = 0$ . The roots need not be distinct and they can be complex numbers. In R, the roots of a polynomial can be found using the function `polyroot()`.

A complex number can be written as  $a + bi$ , where  $i = \sqrt{-1}$ . The absolute value or magnitude of  $a + bi$  is  $\sqrt{a^2 + b^2}$ . The complex plane is the set of all two-dimensional vectors  $(a, b)$ , where  $(a, b)$  represents the complex number  $a + bi$ . The unit circle is the set of all complex number with magnitude 1.

A complex number is inside or outside the unit circle depending on whether its magnitude is less than or greater than 1.

### *Example A.7. Roots of a Cubic Polynomial*

As an example, the roots of the cubic polynomial  $1 + 2x + 3x^2 + 4x^3$  are computed below. We see that there is one real root,  $-0.606$  and two complex roots,  $-0.072 \pm 0.638i$ . It is also verified that these are roots.

```
> roots = polyroot(c(1, 2, 3, 4))
> roots
[1] -0.072+0.638i -0.606-0.000i -0.072-0.638i
> fn = function(x){1 + 2 * x + 3 * x^2 + 4 * x^3}
> round(fn(roots), 5)
[1] 0+0i 0+0i 0+0i
```

□

## A.22 Bibliographic Notes

Casella and Berger (2002) covers in greater detail most of the statistical theory in this chapter and elsewhere in the book. Wasserman (2004) is a modern introduction to statistical theory and is also recommended for further study. Alexander (2001) is a recent introduction to financial econometrics and has a chapter on covariance matrices; her technical appendices cover maximum likelihood estimation, confidence intervals, and hypothesis testing, including likelihood ratio tests. Evans, Hastings, and Peacock (1993) provides a concise reference for the basic facts about commonly used distributions in statistics. Johnson, Kotz, and Kemp (1993) discusses most of the common discrete distributions, including the binomial. Johnson, Kotz, and Balakrishnan (1994, 1995) contain a wealth of information and extensive references about the normal, lognormal, chi-square, exponential, uniform,  $t$ ,  $F$ , Pareto, and many other continuous distributions. Together, these works by Johnson, Kotz, Kemp, and Balakrishnan are essentially an encyclopedia of statistical distributions.

## References

- Alexander, C. (2001) *Market Models: A Guide to Financial Data Analysis*, Wiley, Chichester.
- Casella, G. and Berger, R. L. (2002) *Statistical Inference*, 2nd ed., Duxbury / Thomson Learning, Pacific Grove, CA.
- Evans, M., Hastings, N., and Peacock, B. (1993) *Statistical Distributions*, 2nd ed., Wiley, New York.
- Gouriéroux, C., and Jasick, J. (2001) *Financial Econometrics*, Princeton University Press, Princeton, NJ.

- Johnson, N. L., Kotz, S., and Balakrishnan, N. (1994) *Continuous Univariate Distributions, Vol. 1*, 2nd ed., Wiley, New York.
- Johnson, N. L., Kotz, S., and Balakrishnan, N. (1995) *Continuous Univariate Distributions, Vol. 2*, 2nd ed., Wiley, New York.
- Johnson, N. L., Kotz, S., and Kemp, A. W. (1993) *Discrete Univariate Distributions*, 2nd ed., Wiley, New York.
- Wasserman, L. (2004) *All of Statistics*, Springer, New York.

---

# Index

- $\cap$ , xxv  
 $\cup$ , xxv  
 $\imath$ , 699  
 $\rho_{XY}$ , xxv, 64, 684  
 $\sigma_{XY}$ , xxv, 684  
 $\sim$ , xxvi  
 $x_+$ , 41
- bias-variance tradeoff, 483  
package in R, 664
- A-C skewed distributions, 102, 117  
`abcnon()` function in R, 147  
`abcpars()` function in R, 147  
Abramson, I., 77  
absolute residual plot, 258, 276  
absolute value  
    of a complex number, 699  
ACF, *see* autocorrelation function  
`acf()` function in R, 387  
ADF test, 340  
`adf.test()` function in R, 340, 341, 371  
adjust parameter, 598  
adjustment matrix (of a VECM), 457  
`AER` package in R, 77, 243, 263, 286, 403  
AIC, 109, 110, 199, 232, 350, 654  
    corrected, 112, 126, 342  
    theory behind, 126  
    underlying statistical theory, 126  
Alexander, C., 352, 433, 443, 460, 575, 700
- Alexander, G., 10, 36, 510  
alpha, 507, 509  
analysis of variance table, 227, 229  
Anderson, D. R., 126  
Anderson-Darling test, 64  
ANOVA table, *see* analysis of variance table  
AOV table, *see* analysis of variance table  
APARCH, 421  
`ar()` function in R, 349, 385  
AR process, 325  
    multivariate, 384  
    potential need for many parameters, 326  
AR(1) process, 314  
    checking assumptions, 319  
    nonstationary, 317  
AR(1)+ARCH(1) process, 409  
AR( $p$ ) process, 325, 330  
ARCH process, 405  
ARCH(1) process, 407  
ARCH( $p$ ) process, 411  
ARFIMA, 391  
`arima()` function in R, 318, 325, 337, 348, 378  
ARIMA model  
    automatic selection, 342  
ARIMA process, 104, 331, 343  
`arima.sim()` function in R, 335, 640

- ARMA process, 328, 331, 332, 343, 405  
 multivariate, 384
- ARMAacf()** function in R, 325, 330
- Artzner, P., 573
- ask price, 278, 298
- asymmetric power ARCH, *see* APARCH
- asymmetry  
 of a distribution, 87
- Atkinson, A., 77, 300
- attach()** function in R, 12
- auto.arima()** function in R, 326, 328, 337–339, 342, 350, 369, 396
- autocorrelation function, 308  
 of a GARCH process, 408  
 of an ARCH(1) process, 407  
 sample, 312
- autocovariance function, 308  
 sample, 312
- autoregressive process, *see* AR(1)  
 process and AR( $p$ ) process
- Azzalini–Capitanio skewed distributions, *see* A-C skewed distributions
- $B$  (MCMC diagnostic), 606
- Bühlmann, P., 395
- back-testing, 112
- backwards operator, 331, 333
- bad data, 293
- Bagasheva, B. S., 635
- Bailey, J., 10, 36, 510
- Balakrishnan, N., 700
- bandwidth, 48  
 automatic selection, 50
- BARRA Inc., 540
- Bates, D., 300
- Bauwens, L., 443
- Bayes estimator, 584
- Bayes' law, 484
- Bayes's rule or law, *see* Bayes's theorem
- Bayes's theorem, 582, 583
- Bayesian calculations  
 simulation methods, 595
- Bayesian statistics, 581
- bcanon()** function in R, 147
- bcPower()** function in R, 303
- Belsley, D., 262
- BE/ME, *see* book-equity-to-market-equity
- Bera, A., 443
- Beran, J., 395
- Berger, J. O., 635
- Berger, R., 700
- Bernardo, J., 635
- Bernoulli distribution, 674
- Bernstein–von Mises Theorem, 593
- Best, N. G., 635
- beta, 499, 500  
 estimation of, 507  
 portfolio, 503
- beta distribution, 586–588, 679
- bias, 139, 689  
 bootstrap estimate of, 139  
 bias–variance tradeoff, 3, 49, 50, 86, 110, 536, 612
- BIC, 109, 110, 232, 235, 350
- bid price, 278, 298
- bid–ask spread, 278, 298
- bimodal, 670
- binary regression, 286
- binary response, 286
- binomial distribution, 674  
 kurtosis of, 90  
 skewness of, 88
- Binomial( $n, p$ ), 674
- Black Monday, 3, 47  
 unlikely under a  $t$  model, 62
- Black–Scholes formula, 10
- block resampling, 394, 395
- Bluhm, C., 274–276, 282
- Bodie, Z., 36, 488, 510
- Bolance, C., 77
- Bollerslev, T., 439, 443
- book value, 529
- book-equity-to-market-equity, 528
- book-to-market value, 529
- Boos, D. D., 126
- boot** package in R, 150, 394
- bootstrap, 137, 139, 368, 560  
 block, 394  
 multivariate data, 175  
 origin of name, 137
- bootstrap approximation, 138
- bootstrap confidence interval  
 ABC, 147  
 basic, 146  
 $BC_a$ , 147
- bootstrap- $t$  interval, 143–146

- normal approximation, 143
- percentile, 146, 147
- bootstrap()** function in R, 141
- bootstrap** package in R, 141, 147, 150
- Box test, 313
- Box, G., 4, 126, 284, 352, 395, 635
- Box–Cox power transformation, 67, 69
- Box–Cox transformation model, 284
- Box–Jenkins model, 352
- Box.test()** function in R, 320
- boxcox()** function in R, 121, 284, 303
- BoxCox.Arima()** function in R, 366
- boxplot, 65, 67
- boxplot()** function in R, 65
- Britten-Jones, M., 488
- Brockwell, P., 352
- Brooks, S., 635
- Brownian motion, 689
  - geometric, 9
- BUGS**, 595
- Burg, D., 11
- Burnham, K. P., 126
- buying on margin, *see* margin, buying on
- bwNeweyWest()** function in R, 375
- ca.jo()** function in R, 458, 460
- calibration
  - of Gaussian copula, 200
  - of t-copula, 201
- Campbell, J., 10, 36, 510
- capital asset pricing model, *see* CAPM
- capital market line, *see* CML
- CAPM, 2, 159, 495, 498–500, 509, 527
  - testing, 507
- car** package in R, 245, 368
- Carlin, B. P., 635, 636
- Carlin, J., 635, 636
- Carroll, R., 77, 262, 300, 443, 664
- Casella, G., 635, 700
- CCF, *see* cross-correlation function
- ccf()** function in R, 381
- CDF, 669, 670
  - calculating in R, 669
  - population, 673
- center
  - of a distribution, 87
- center parameters
  - A-C distributions, 103
- centering
  - variables, 242
- central limit theorem, 89, 682, 690
  - for least-squares estimator, 258
  - for sample quantiles, 54, 77, 561
  - for the maximum likelihood estimator, 105, 107, 126, 139, 142, 177, 594
  - for the posterior, 592, 594, 636
- infinite variance, 682
- multivariate for the maximum likelihood estimator, 175, 594
- Chan, K., 646, 662
- Change Dir**, 11
- change-of-variables formula, 76
- characteristic line, *see* security characteristic line
- Chernick, M., 150
- chi-squared distribution, 681
- $\chi_{\alpha,n}^2$ , 681
- Chib, S., 635
- Chou, R., 443
- CKLS model, 300
  - extended, 665
- Clayton copula, *see* copula, Clayton
- CML (capital market line), 496, 497, 506, 507
  - comparison with SML (security market line), 500
- co-monotonicity copula, *see* copula, co-monotonicity
- coda** package in R, 599, 607
- coefficient of tail dependence
  - co-monotonicity copula, 197
  - Gaussian copula, 197
  - independence copula, 197
  - lower, 196
  - t-copula, 197
  - upper, 197
- coefficient of variation, 283
- coherent risk measure, *see* risk measure, coherent
- cointegrating vector, 453, 457
- cointegration, 453
- collinearity, 234
- collinearity diagnostics, 262
- components
  - of a mixture distribution, 96

- compounding
  - continuous, 32
- concordant pair, 194
- conditional least-squares estimator, 324
- confidence coefficient, 138, 690
- confidence interval, 138, 559, 560, 690
  - accuracy of, 143
  - for determining practical significance, 697
  - for mean using  $t$ -distribution, 143, 690
  - for mean using bootstrap, 144
  - for variance of a normal distribution, 692
  - profile likelihood, 119
- confidence level
  - of VaR, 553
- Congdon, P., 635
- conjugate prior, 586
- consistent estimator, 370
- contaminant, 91, 293
- Cook, R. D., 262
- Cook's D, 251
- Cook's D, 253, 254
- copula, 183, 193
  - Archimedean, 187
  - Clayton, 189, 190, 199, 204
  - co-monotonicity, 185, 188, 189, 214
  - counter-monotonicity, 185, 188, 189
  - Frank, 187, 189
  - Gaussian, 197, 200, 205
  - Gumbel, 191, 199, 204
  - independence, 185
  - Joe, 192, 199, 204
  - nonexchangeable Archimedean, 207
  - $t$ , 197, 201
- copula package in R, 187, 189, 205, 208, 210, 211
- `cor()` function in R, 12
- `CORR`, xxv
- correlation, xxv, 683
  - effect on efficient portfolio, 472
- correlation coefficient, 162, 684
  - interpretation, 685
  - Kendall's tau, 194
  - Pearson, 64, 193, 684
  - rank, 193
  - sample, 684, 685
  - sample Kendall's tau, 195
- sample Spearman's, 195
- Spearman's, 194, 195
- correlation matrix, xxv, 157
  - Kendall's tau, 195
  - sample, 158
  - sample Spearman's, 196
  - Spearman's, 196
- corrlation
  - partial, 226
- $\text{Corr}(X, Y)$ , xxv
- counter-monotonicity copula, *see* copula, counter-monotonicity
- coupon bond, 22, 25
- coupon rate, 23
- COV, xxv
- covariance, xxv, 64, 160, 683, 684
  - sample, 219, 684
- covariance matrix, xxv, 157, 160
  - between two random vectors, 162
  - of standardized variables, 158
  - sample, 158
- coverage probability
  - actual, 142
  - nominal, 142
- `covRob()` function in R, 533
- $\text{Cov}(X, Y)$ , xxv, 684
- Cox, D., 284
- Cox, D. R., 126
- Cox, J., 646
- $C_p$ , 232
- `cp2dp()` function in R, 117
- Cramér–von Mises test, 64
- credible interval, 585, 690
- credit risk, 553
- critical value, 693
  - exact, 108
- cross-correlation, 533
- cross-correlation function, 380, 382
- cross-correlations
  - of principal components, 524
- cross-sectional data, 263
- cross-validation, 111, 654
  - $K$ -fold, 111
  - leave-one-out, 112
- Crouhy, M., 575
- `cumsum()` function in R, 335
- cumulative distribution function, *see* CDF

- current yield, 23  
*CV*, *see* cross-validation
- Dalgaard, P., 11  
 Daniel, M. J., 635  
 data sets  
   air passengers, 309, 365  
   Berndt's monthly equity returns, 529, 539  
   BMW log returns, 320, 322, 323, 350, 413, 415, 422, 427  
   CPI, 381, 385, 389, 528  
   CPS1988, 263, 664  
   Credit Cards, 286, 289, 291  
   CRSP daily returns, 158, 163, 166, 168, 169, 172–174, 176, 564, 620  
   CRSP monthly returns, 531, 537  
   daily midcap returns, 110, 111, 149, 167, 460, 551, 613, 618  
   default frequencies, 274, 276, 281, 283  
   DM/dollar exchange rate, 45, 58, 62, 65  
   Dow Jones, 526  
   Earnings, 75, 76  
   Equity funds, 524, 526, 543, 545  
   EuStockMarkets, 77, 129  
   excess returns on the food industry and the market, 221, 222  
   Fama–French factors, 531, 537  
   Flows in pipelines, 69, 117, 120, 202  
   HousePrices, 302, 303  
   housing starts, 361, 362, 364, 365  
   ice cream consumption, 377, 379  
   Industrial Production (IP), 337, 381, 385, 389, 528  
   inflation rate, 309, 313, 323, 326, 328, 330, 339, 340, 342, 346, 351, 391  
   mk.maturity, 38  
   mk.zero2, 38  
   Nelson–Plosser U.S. Economic Time Series, 235, 241, 424  
   risk-free interest returns, 45, 62, 65, 67, 74, 113, 116, 124, 333, 646  
   S&P 500 daily log returns, 45, 47, 62, 65, 556, 558, 569  
   Treasury yield curves, 455, 520, 522, 523  
   USMacroG, 243, 397, 403
- weekly interest rates, 219, 224–228, 230, 232–234, 240  
 data transformation, 67, 69–71  
 Davis, R., 352  
 Davison, A., 150, 395  
 decile, 53, 670  
 decreasing function, 672  
 default probability  
   estimation, 274–276  
 degrees of freedom, 229  
   of a *t*-distribution, 61  
   residual, 229  
 Delbaen, F., 573  
 $\Delta$ , *see* differencing operator and Delta, of an option price  
 density  
   bimodal, 141  
   trimodal, 58  
   unimodal, 141  
 determinant, xxvi  
 deviance, 110, 111  
 df, *see* degrees of freedom  
 dgfd() function in R, 100  
 diag( $d_1, \dots, d_p$ ), xxv, 698  
 DIC, 609  
 dic.samples() function in R, 601, 611, 612  
 Dickey–Fuller test, 341  
   augmented, 340, 341  
 diffdic() function in R, 628  
 differencing operator, 333  
   kth-order, 334  
 diffseries() function in R, 393  
 diffusion function, 646  
 dimension reduction, 517, 519  
 direct parameters  
   A-C distributions, 103  
 discordant pair, 194  
 discount bond, *see* zero-coupon bond  
 discount function, 33, 34  
   relationship with yield to maturity, 34  
 dispersion, 122  
 distribution  
   full conditional, 596, 597  
   marginal, 46  
   meta-Gaussian, 205  
   symmetric, 89  
   unconditional, 47

- disturbances
  - in regression, 217
- diversification, 495, 503
- dividends, 7
- double-exponential distribution, 678
  - kurtosis of, 90
- Dowd, K., 575
- `dpi11()` function in R, 661
- Draper, N., 243
- drift
  - of a random walk, 9
  - of an ARIMA process, 337, 338
- `dstd()` function in R, 100
- $D_t$ , 8
- Duan, J.-C., 443
- Dunson, D. B., 635
- DUR, *see* duration
- duration, 35, 36
- duration analysis, 553
- Durbin–Watson test, 368
- `DurbinWatsonTest()` function in R, 368
- `dwtest()` function in R, 368
- Eber, J-M., 573
- `Ecdat` package in R, 46, 47, 52, 58, 76, 124, 140, 158, 221, 222, 309, 361, 428, 531
- `ecdf()` function in R, 52
- EDF, *see* sample CDF
- Edwards, W., 584
- effective number of parameters, 610, 653
- `effectiveSize()` function in R, 599, 607
- efficient frontier, 468, 469, 472, 485
- efficient portfolio, 468, 470, 485
- Efron, B., 150
- `eigen()` function in R, 171, 172, 385, 699
- eigenvalue-eigenvector decomposition, 171, 698
- ellipse, 170
- elliptically contoured density, 170, 171
- empirical CDF, *see* sample CDF
- empirical copula, 200, 206
- empirical distribution, 145
- Enders, W., 352, 460
- Engle, R., 439, 443
- equi-correlation model, 200
- Ergashev, B., 635
- ES, *see* expected shortfall
- estimation
  - interval, 690
- estimator, 689
  - efficient, 689
  - unbiased, 689
- Evans, M., 700
- excess expected return, 496, 500
- excess return, 221, 507
- exchangeable, 187
- expectation
  - conditional, 645, 683
  - normal distribution, 687
- expectation vector, 157
- expected loss given a tail event, *see* expected shortfall
- expected shortfall, 1, 65, 554, 555, 557–560
- expected value
  - nonexistent, 671
- exponential distribution, 678
  - kurtosis of, 90
  - skewness of, 90
- exponential random walk, *see* geometric random walk
- exponential tail, 94, 99
- F*-distribution, 681
- F*-test, 488, 681
- F-S skewed distributions, 102, 132
- Fabozzi, F. J., 635
- face value, *see* par value
- `factanal()` function in R, 541–543
- factor, 517, 527
- factor model, 504, 527, 530
  - BARRA, 540
  - cross-sectional, 538, 539
  - fundamental, 528, 529
  - macroeconomic, 528
  - of Fama and French, 529, 530
  - time series, 538, 539
- $F_{\alpha,n_1,n_2}$ , 681
- Fama, E., 528, 529, 546
- Fan, J., 664
- `faraway` package in R, 234, 245
- FARIMA, 391
- `fdHess()` function in R, 175
- `fEcofin` package in R, 38, 110

- Federal Reserve Bank of Chicago, 219  
 Fernandez–Steel skewed distributions,  
   see F-S skewed distributions  
**fGarch** package in R, 100–102  
 $f_{\text{ged}}^{\text{std}}(y|\mu, \sigma^2, \nu)$ , 101  
 Fisher information, 105  
   observed, 106  
 Fisher information matrix, 106, 174  
**FitAR** package in R, 366  
**fitCopula()** function in R, 205, 213  
**fitdistr()** function in R, 113  
**fitMvdc()** function in R, 211  
 fitted values, 218, 223  
   standard error of, 251  
 fixed-income security, 19  
**forecast()** function in R, 396  
**forecast** package in R, 326, 396  
 forecasting, 342, 343  
   AR(1) process, 343  
   AR(2) process, 343  
   MA(1) process, 343  
 forward rate, 29, 30, 33, 34  
   continuous, 33  
   estimation of, 276  
**fracdiff** package in R, 393  
 fractionally integrated, 391  
 Frank copula, *see* copula, Frank  
 French, K., 528, 529, 546  
 $f_{\text{ged}}^{\text{std}}(y|\nu)$ , 99  
 full conditional, *see* distribution, full  
   conditional  
 fundamental factor model, *see* factor  
   model, fundamental  
 fundamental theorem of algebra, 699  
  
 Galai, D., 575  
**gam()** function in R, 661, 664  
 gamma distribution, 678  
   inverse, 679  
 gamma function, 95, 678  
 $\gamma(h)$ , 308, 310  
 $\hat{\gamma}(h)$ , 312  
 GARCH model, 294  
 GARCH process, 99, 104, 405–409, 411,  
   413  
   as an ARMA process, 418  
   fitting to data, 413  
   heavy tails, 413  
   integrated, 408  
 GARCH( $p, q$ ) process, 411  
  
 GARCH(1,1), 419  
 GARCH-in-mean model, 448  
 Gauss, Carl Friedrich, 676  
 Gaussian distribution, 676  
 GCV, 654  
 GED, *see* generalized error distribution  
 Gelman, A., 635, 636  
**gelman.diag()** function in R, 599, 606,  
   607  
**gelman.plot()** function in R, 599, 607  
 generalized cross-validation, *see* GCV,  
   654  
 generalized error distribution, 99, 116  
 generalized linear models, 286  
 generalized Pareto distribution, 575  
 generator  
   Clayton copula, 189  
   Frank copula, 187  
   Gumbel copula, 191  
   Joe copula, 192  
   non-strict of an Archimedean copula,  
     207  
   strict of an Archimedean copula, 187  
 geometric Brownian motion, 689  
 geometric random walk, 9  
   lognormal, 9  
 geometric series, 316  
   summation formula, 23  
 Gibbs sampling, 596  
 Giblin, I., 460  
 Gijbels, I., 664  
 GLM, *see* generalized linear model  
**glm()** function in R, 286, 288  
 Gourieroux, C., 352, 443, 575  
 Gram–Schmidt orthogonalization  
   procedure, 243  
 Greenberg, E., 635  
 growth stock, 531  
 Guillén, R., 77  
 Gumbel copula, *see* copula, Gumbel  
  
 half-normal plot, 254  
 Hamilton, J. D., 352, 395, 443, 460  
 Harrell, F. E., Jr., 243  
 Hastings, N., 700  
 hat diagonals, 251  
 hat matrix, 251, 270, 653  
 Heath, D., 573  
 heavy tails, 57, 257  
 heavy-tailed distribution, 93, 413

- hedge portfolio, 531  
 hedging, 299  
 Hessian matrix, 106, 174  
     computation by finite differences, 175  
 Heston, S., 443  
 heteroskedasticity, 258, 276, 405  
     conditional, 67, 406  
 hierarchical prior, 612, 613  
 Higgins, M., 443  
 high-leverage point, 250  
 Hill estimator, 567, 568, 570, 571  
 Hill plot, 568, 570, 571  
 Hinkley, D., 150, 395  
 histogram, 47  
 HML (high minus low), 529  
 Hoaglin, D., 77  
 holding period, 5, 466  
 homoskedasticity  
     conditional, 407  
 horizon  
     of VaR, 553  
 Hosmer, D., 300  
 Hsieh, K., 443  
 Hsu, J. S. J., 635  
 Hull, J., 575  
 hyperbolic decay, 390  
 hypothesis  
     alternative, 693  
     null, 693  
 hypothesis testing, 137, 693
- I**, xxv  
 $I(0)$ , 335  
 $I(1)$ , 335  
 $I(2)$ , 335  
 $I(d)$ , 335  
 i.i.d., 673  
 Ieno, E., 11  
 illiquid, 298  
 importance sampling, 635  
 increasing function, 672  
 independence  
     of random variables, 160, 162  
     relationship with correlation, 686  
 index fund, 495, 556  
 indicator function, xxvi, 52  
 inf, *see* infinum  
 infinum, 670, 672
- influence.measures()** function in R, 253  
 information set, 342  
 Ingersoll, J., 646  
 integrating  
     as inverse of differencing, 335  
 interest-rate risk, 35  
 interest-rate spread, 527  
 interquartile range, 65, 103  
 intersection  
     of sets, xxv  
 interval estimate, 690  
 inverse Wishart distribution, 619  
**iPsi()** function in R, 187  
 IQR, 65
- Jackson, C., 635  
 JAGS, 595  
 James, J., 36  
 Jarque–Bera test, 64, 91  
**jarque.bera.test()** function in R, 92  
 Jarrow, R., 36, 443  
 Jasiak, J., 352, 443, 575  
 Jenkins, G., 352, 395  
 Jobson, J., 488  
 Joe copula, *see* copula, Joe  
 Johnson, N., 700  
 Jones, M. C., 77, 664  
 Jorion, P., 575
- Kane, A., 36, 488, 510  
 Karolyi, G., 646, 662  
 Kass, R. E., 635  
 KDE, *see* kernel density estimator  
 Kemp, A., 700  
 Kendall's tau, *see* correlation coefficient,  
     Kendall's tau, 194  
 kernel density estimator, 48, 49, 52  
     two-dimensional, 213  
     with transformation, 75  
**KernSmooth()** package in R, 647  
**KernSmooth** package in R, 649, 661  
 Kim, S., 635  
 Kleiber, C., 77  
 knot, 655, 656  
     of a spline, 654  
 Kohn, R., 646, 647  
 Kolmogorov–Smirnov test, 64  
 Korkie, B., 488

- Kotz, S., 700  
 KPSS test, 340  
`kpss.test()` function in R, 340  
 Kroner, K., 443  
 Kuh, E., 262  
 kurtosis, 87, 89, 90  
   binomial distribution, 90  
   excess, 91  
   sample, 91  
   sensitivity to outliers, 91  
 Kutner, M., 243
- lag, 308  
   for cross-correlation, 381  
 lag operator, 331  
 Lahiri, S. N., 395  
 Lange, N., 294  
 Laplace distribution, *see* double exponential distribution  
 large-cap stock, 695  
 large-sample approximation  
   ARMA forecast errors, 345  
 Laurent, S., 443  
 law of iterated expectations, 683  
 law of large numbers, 682  
`leaps()` function in R, 239  
`leaps` package in R, 232, 239  
 least-squares estimator, 218, 221, 682  
   generalized, 271  
   weighted, 259, 424  
 least-squares line, 219, 298  
 least-trimmed sum of squares estimator,  
   *see* LTS estimator  
 Ledoit, O., 636  
 Lehmann, E., 77, 636  
 Lemeshow, S., 300  
 level  
   of a test, 693  
 leverage, 13  
   in estimation, 653  
   in regression, 251  
 leverage effect, 421  
 Li, W. K., 441  
 Liang, K., 109  
 likelihood function, 104  
 likelihood ratio test, 108, 681  
 linear combination, 165  
 linear programming, 490  
`linprog` package in R, 490
- Lintner, J., 510  
 liquidity risk, 553  
 Little, R., 294  
 Ljung–Box test, 312, 320, 336, 383  
`lm()` function in R, 224, 226, 531  
`lmtree` package in R, 368  
 Lo, A., 10, 36, 510  
 loading  
   in a factor model, 530  
 loading matrix (of a VECM), 457, 458  
 location parameter, 86, 88, 89, 675, 676  
   quantile based, 103  
`locfit()` function in R, 661  
`locfit` package in R, 651, 661  
`locpoly()` function in R, 647, 649, 661  
 loess, 245, 259, 652  
 log, xxv  
 $\log_{10}$ , xxv  
 log-drift, 9  
 log-mean, 9, 677  
 log price, 6  
 log return, *see* return, log  
 log-likelihood, 104  
 log-standard deviation, 9, 677  
 log-variance, 677  
 $\text{Lognormal}(\mu, \sigma)$ , 676  
 lognormal distribution, 676  
   skewness of, 91  
 long position, 474  
 longitudinal data, 263  
 Longstaff, F., 646, 662  
 Louis, T. A., 635, 636  
 lower quantile, *see* quantile, lower  
 lowess, 245, 652  
 LTS estimator, 293, 294  
`ltsreg()` function in R, 294  
 Lunn, D. J., 635
- MA(1) process, 328  
 MA( $q$ ) process, 330  
 MacKinlay, A., 10, 36, 510  
 macroeconomic factor model, *see* factor model, macroeconomic  
 MAD, 51, 55, 65, 87, 122, 123  
`mad()` function in R, 52, 79, 123  
 magnitude  
   of a complex number, *see* absolute value, of a complex number  
 MAP estimator, 585

- Marcus, A., 36, 488, 510
- margin  
  buying on, 498
- marginal distribution function, 46
- Mark, R., 575
- market capitalization, 695
- market equity, 529
- market maker, 298
- market risk, 553
- Markov chain Monte Carlo, *see* MCMC
- Markov process, 324, 688
- Markowitz, H., 488
- Marron, J. S., 77
- MASS package in R, 244, 284
- matrix  
  diagonal, 698  
  orthogonal, 698  
  positive definite, 161  
  positive semidefinite, 161
- Matteson, D. S., 436
- maximum likelihood estimator, 85, 104, 108, 246, 324, 325, 682  
  not robust, 122  
  standard error, 105
- MCMC, 137
- mean  
  population, 673  
  sample, 673  
    as a random variable, 137, 690
- mean deviance, 611
- mean-reversion, 309, 453
- mean-squared error, 689
- mean sum of squares, 229
- mean-squared error, 139  
  bootstrap estimate of, 139
- mean-variance efficient portfolio, *see* efficient portfolio
- median, 53, 670
- median absolute deviation, *see* MAD
- Meesters, E., 11
- Merton, R., 488, 510, 646
- meta-Gaussian distribution, 186
- Metropolis–Hastings algorithm, 597
- `mfcoll()` function in R, 12
- `mfrw()` function in R, 12
- `mgcv` package in R, 661, 664
- Michaud, R., 479
- mixed model, 659
- mixing  
  of an MCMC sample, 602
- mixing distribution, 99
- mixture distribution  
  normal scale, 98
- mixture model, 96  
  continuous, 99  
  continuous scale, 99  
  finite, 99
- MLE, *see* maximum likelihood estimator
- mode, 102, 670
- model  
  full, 108  
  parametric, 85  
  reduced, 108  
  semiparametric, 566
- model averaging, 126
- model complexity  
  penalties of, 109
- model selection, 231
- moment, 92  
  absolute, 92  
  central, 92
- momentum  
  in a time series, 335
- monotonic function, 672
- Morgan Stanley Capital Index, 479
- Mossin, J., 510
- Mosteller, 77
- moving average process, *see* MA(1) and MA( $q$ ) processes
- moving average representation, 315
- MSCI, *see* Morgan Stanley Capital Index
- MSE, *see* mean-squared error
- `mst.mple()` function in R, 173
- multicollinearity, *see* collinearity
- multimodal, 670
- multiple correlation, 228
- multiplicative formula  
  for densities, 688
- $N_{\text{eff}}$ , 607
- $N(\mu, \sigma^2)$ , 676
- Nachtsheim, C., 243
- Nandi, S., 443
- Nelson, C. R., 243, 300
- Nelson, D., 443
- Nelson–Siegel model, 278, 281

- net present value, 25
- Neter, J., 243
- Newey, W., 375
- `NeweyWest()` function in R, 375, 424
- Nielsen, J. P., 77
- `nlme` package in R, 175
- `nlminb()` function in R, 104
- `nls()` function in R, 273
- nominal value
  - of a coverage probability, 259
- nonconstant variance
  - problems caused by, 259
- nonlinearity
  - of effects of predictor variables, 259
- nonparametric, 555
- nonrobustness, 71
- nonstationarity, 408
- norm
  - of a vector, 698
- normal distribution, 676
  - bivariate, 687
  - kurtosis of, 90
  - multivariate, 164, 165
  - skewness of, 90
  - standard, 676
- normal mixture distribution, 96
- normal probability plot, 54, 98, 276
  - learning to use, 256
- normality
  - tests of, 64
- OpenBUGS, 598, 635
- OpenBUGS, 595
- operational risk, 553
- `optim()` function in R, 104, 106, 113, 180, 211
- `order()` function in R, 650
- order statistic, 52, 53, 555
- orthogonal polynomials, 243
- `outer()` function in R, 665
- outlier, 256
  - extreme, 257
  - problems caused by, 258
  - rules of thumb for determining, 257
- outlier-prone, 57
- outlier-prone distribution, *see* heavy-tailed distribution
- Overbeck, L., 274–276, 282
- overdifferencing, 393, 394
- overdispersed, 596
- overfit
  - density function, 50
- overfitting, 109, 110, 649
- oversmoothing, 50, 649
- $p_D$ , 610
- $p$ -value, 64, 226, 693, 694
- PACF, *see* partial autocorrelation function
- pairs trading, 459
- Palma, W., 409, 420
- panel data, 263
- `par()` function in R, 12
- par value, 20, 22, 23
- Pareto, Vilfredo, 680
- Pareto constant, *see* tail index
- Pareto distribution, 571, 680
- Pareto tail, *see* polynomial tail, 571
- parsimony, 3, 86, 307, 309, 312, 314, 316, 325
- partial autocorrelation function, 349–351
- PCA, *see* principal components analysis
- `pca()` function in R, 519
- $p_D$ , 609
- Peacock, B., 700
- Pearson correlation coefficient, *see* correlation coefficient, Pearson
- penalized deviance, 611
- percentile, 53, 670
- Pfaff, B., 352, 460
- Phillips–Ouliaris test, 454, 455
- Phillips–Perron test, 340
- $\phi(x)$ , 676
- $\Phi(y)$ , 676
- Pindyck, R., 443
- `plogis()` function in R, 304
- Plosser, C., 243
- plus function, 656
  - linear, 656
  - quadratic, 656
  - 0th-degree, 657
- `pnorm()` function in R, 16
- `po.test()` function in R, 455
- Poisson distribution, 283
- Pole, A., 460
- polynomial regression, *see* regression, polynomial

- polynomial tail, 94, 99
- polynomials
  - roots of, 699
- polyroot()** function in R, 340, 699
- pooled standard deviation, 694
- portfolio, 159
  - efficient, 470, 472, 475, 496
  - market, 496, 500, 504, 505
  - minimum variance, 468
- positive part function, 41
- posterior CDF, 586
- posterior distribution, 582
- posterior interval, 585, 593
- posterior probability, 584
- potential scale reduction factor, 606
- power
  - of a test, 695
- power transformations, 67
- pp.test()** function in R, 340
- practical significance, 697
- prcomp()** function in R, 521
- precision, 589, 618
- precision matrix, 618
- prediction, 295
  - best, 687, 697
  - best linear, 295, 297, 499, 687
    - relationship with regression, 298
  - error, 297, 687
    - unbiased, 297
  - linear, 295
  - multivariate linear, 298
- price
  - stale, 278
- pricing anomaly, 529
- principal axis, 518
- principal components analysis, 517, 519, 521, 523, 525–527, 698
- princomp()** function in R, 548
- prior
  - noninformative, 582
- prior distribution, 582
- prior probability, 584
- probability density function
  - conditional, 683
  - elliptically contoured, 164
  - marginal, 682
  - multivariate, 687
- probability distribution
  - multivariate, 157
- probability transformation, 196, 675
- profile likelihood, 119
- profile log-likelihood, 119
- pseudo-inverse
  - of a CDF, 670, 675
- pseudo-maximum likelihood
  - for copulas, 199
  - parametric for copulas, 200
  - semiparametric for copulas, 200
- $P_t$ , 5
- $p_t$ , 6
- qchisq()** function in R, 692
- QQ plot, *see* quantile–quantile plot
- qqline()** function in R, 55
- qqnorm()** function in R, 54, 55
- qqplot()** function in R, 62
- quadratic programming, 475
- quantile, 53, 54, 670
  - lower, 670
  - population, 673
  - respects transformation, 670
  - upper, 108, 670
- quantile function, 670, 675
- quantile()** function in R, 53
- quantile transformation, 675
- quantile–quantile plot, 61, 62
- quartile, 53, 670
- quintile, 53, 670
- $\Re$ , xxv
- $R$ -squared, 228, 298
- $R^2$  adjusted, 231, 232
- $R^2$ , *see*  $R$ -squared
- Rachev, S. T., 635
- rally
  - bond, 19
- random sample, 673
- random variables
  - linear function of, 159
- random vector, 157, 687
- random walk, 9, 317
  - normal, 9
- random walk hypothesis, 1
- rank, 194
- rank correlation, 194
- rational person
  - definition within economics, 484
- rCopula()** function in R, 189, 209

- `read.csv()` function in R, 11
- regression, 645
  - ARMA disturbances, 377
  - ARMA+GARCH disturbances, 424
  - cubic, 243
  - geometrical viewpoint, 229
  - linear, 645
  - local linear, 647
  - local polynomial, 647, 648
  - logistic, 286, 303
  - multiple linear, 217, 224, 298, 325
  - multivariate, 528
  - no-intercept model, 509
  - nonlinear, 271, 274, 277, 300
  - nonlinear parametric, 274, 645
  - nonparametric, 259, 274, 645, 698
  - polynomial, 225, 242, 243, 246, 259, 274
    - is a linear model, 274
  - probit, 286
  - spurious, 372
  - straight-line, 218
  - transform-both-sides, 281
  - with high-degree polynomials, 243
- regression diagnostics, 251
- regression hedging, 298, 299
- `regsubsets()` function in R, 232
- Reinsel, G., 352, 395
- rejection region, 693
- REML, 659
- reparameterization, 675
- resampling, 54, 137, 138, 144, 559, 560
  - block, 394
  - model-based, 138
    - for time series, 394, 395
  - model-free, 138, 560
  - multivariate data, 175
    - time series, 394
- residual error MS, 537
- residual error SS, 227
- residual mean sum of squares, 229, 653
- residual outlier, 250
- residuals, 218, 255, 274, 318–320
  - correlation, 256, 368
    - effect on confidence intervals and standard errors, 368, 373, 424
  - externally studentized, 253, 255
  - externally studentized (`rstudent`), 250
  - internally studentized, 253
- nonconstant variance, 255, 258
- nonnormality, 255, 256
- raw, 252, 255
- return
  - adjustment for dividends, 7
  - continuously compounded, *see return, log*, 6
  - log, 6
  - multiperiod, 7
  - net, 1, 5
  - simple gross, 6
- return-generating process, 502
- reversion
  - to the mean, 335
- $\hat{R}$ , 607
- $\rho(h)$ , 308, 311
- $\hat{\rho}(h)$ , 312
- $\rho_{XY}$ , 64, 684
- $\hat{\rho}_{XY}$ , 684
- risk, 1
  - market or systematic component, 503
  - unique, nonmarket, or unsystematic component, 503, 504, 509
- risk aversion, 484
  - index of, 498
- risk factor, 517, 527, 538
- risk management, 553
- risk measure
  - coherent, 573
- risk premium, 465, 495, 496, 500
- risk-free asset, 465, 467, 495
- Ritchken, P., 443
- `rjags` package in R, 598, 611, 628
- `rnorm()` function in R, 13
- Robert, C. P., 635
- robust estimation, 294
- robust estimator, 52
- robust estimator of dispersion, 122
- robust modeling, 294
- `robust` package in R, 294, 533
- Rombouts, J. V., 443
- root finder
  - nonlinear, 38
- Ross, S., 646
- Rossi, P., 443
- $\Re^p$ , xxv
- `rstudent`, 250, 251, 253
- $R_t$ , 5
- $r_t$ , 6