Morton John Canty
mort.canty@gmail.com

Image Analysis, Classification and Change Detection in Remote Sensing

Fifth Revised Edition

Solutions to Selected Exercises

CRC PRESS
Boca Raton Ann Arbor London Tokyo

Contents

1

Images, Arrays and Matrices

Exercise 1

The components of $\boldsymbol{x}$ and $\boldsymbol{y}$ are

$$x_1 = \|\boldsymbol{x}\| \cos \theta_x, \quad x_2 = \|\boldsymbol{x}\| \sin \theta_x$$
$$y_1 = \|\boldsymbol{y}\| \cos \theta_y, \quad y_2 = \|\boldsymbol{y}\| \sin \theta_y.$$

The angle between the two vectors is $\theta_x - \theta_y$. The inner product is therefore

$$\begin{aligned} \boldsymbol{x}^\top \boldsymbol{y} &= \|\boldsymbol{x}\| \|\boldsymbol{y}\| \cos(\theta_x - \theta_y) \\ &= \|\boldsymbol{x}\| \|\boldsymbol{y}\| (\cos \theta_x \cos \theta_y + \sin \theta_x \sin \theta_y) \\ &= x_1 y_1 + x_2 y_2. \end{aligned}$$

Exercise 2

The outer product of $\boldsymbol{x}$ and $\boldsymbol{y}$ is

$$\boldsymbol{x}\boldsymbol{y}^\top = \begin{pmatrix} x_1 y_1 & x_1 y_2 \\ x_2 y_1 & x_2 y_2 \end{pmatrix},$$

the determinant of which is

$$|\boldsymbol{x}\boldsymbol{y}^\top| = x_1 y_1 x_2 y_2 - x_1 y_2 x_2 y_1 = 0.$$

We can express the second column as in terms of the first:

$$y_2 \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = c y_1 \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$

by choosing $c = y_2/y_1$, so thr rank is 1.

Exercise 3

Verifying the identity with random 2×2 matrices:

```
import numpy as np
A = np.mat(np.random.rand(2,2))
B = np.mat(np.random.rand(2,2))
print (A*B).T
```

```
print B.T*A.T

[[0.58298781 0.40052227]
 [0.26627182 0.18432174]]
[[0.58298781 0.40052227]
 [0.26627182 0.18432174]]
```

Exercise 4

Let the colinear vectors be $\boldsymbol{x}$, $\boldsymbol{y}$ and $\boldsymbol{z}$. Then clearly the difference vectors $\boldsymbol{z}-\boldsymbol{x}$ and $\boldsymbol{y}-\boldsymbol{x}$ point in the same direction. Therefore

$$\boldsymbol{z}-\boldsymbol{x}=\rho(\boldsymbol{y}-\boldsymbol{x})$$

or

$$(1-\rho)\boldsymbol{x}+\rho\boldsymbol{y}+(-1)\boldsymbol{z}=\boldsymbol{0}$$

for some value of ρ and the vectors are linearly dependent.

Exercise 5

The determinant of

$$\boldsymbol{A}=\begin{pmatrix}a_{11} & a_{12}\\ a_{12} & a_{22}\end{pmatrix}$$

is

$$|\boldsymbol{A}|=a_{11}a_{22}-a_{12}^2.$$

The eigenvalues are

$$\begin{aligned}\lambda^{(1)}&=\frac{1}{2}\left(a_{11}+a_{22}+\sqrt{(a_{11}+a_{22})^2-4(a_{11}a_{22}-a_{12}^2)}\right)\\ \lambda^{(2)}&=\frac{1}{2}\left(a_{11}+a_{22}-\sqrt{(a_{11}+a_{22})^2-4(a_{11}a_{22}-a_{12}^2)}\right).\end{aligned}$$

with product

$$\lambda^{(1)}\lambda^{(2)}=\frac{1}{4}((a_{11}+a_{22})^2-((a_{11}+a_{22})^2-4(a_{11}a_{22}-a_{12}^2)))=|\boldsymbol{A}|.$$

Exercise 6

Let $\boldsymbol{A}$ be symmetric. Its inverse satisfies

$$\boldsymbol{A}^{-1}\boldsymbol{A}=\boldsymbol{I}.$$

Transposing both sides:

$$\boldsymbol{A}^\top(\boldsymbol{A}^{-1})^\top=\boldsymbol{I}.$$

But $\boldsymbol{A}^\top=\boldsymbol{A}$ and therefore $\boldsymbol{A}(\boldsymbol{A}^{-1})^\top=\boldsymbol{I}$ or $(\boldsymbol{A}^{-1})^\top=\boldsymbol{A}^{-1}$, so $\boldsymbol{A}^{-1}$ is also symmetric.

Exercise 7

Let the eigenvalues and eigenvectors of $\boldsymbol{A}$ be λ_k and $\boldsymbol{u}_k$, respectively, for $k = 1, 2$. Then

$$\boldsymbol{A}\boldsymbol{u}_k = \lambda_k \boldsymbol{u}_k.$$

Multiply both sides with $\boldsymbol{A}^{-1}$. Then

$$\boldsymbol{u}_k = \lambda_k \boldsymbol{A}^{-1}\boldsymbol{u}_k$$

or

$$\boldsymbol{A}^{-1}\boldsymbol{u}_k = \lambda_k^{-1}\boldsymbol{u}_k.$$

Exercise 8

Starting from

$$y = \boldsymbol{x}^\top \boldsymbol{A}\boldsymbol{x},$$

we can write

$$\boldsymbol{x}y\boldsymbol{x}^\top = \boldsymbol{x}\boldsymbol{x}^\top \boldsymbol{A}\boldsymbol{x}\boldsymbol{x}^\top,$$

and therefore, since y is scalar,

$$\mathrm{tr}(\boldsymbol{x}\boldsymbol{x}^\top)y = \mathrm{tr}(\boldsymbol{x}\boldsymbol{x}^\top \boldsymbol{A}\boldsymbol{x}\boldsymbol{x}^\top).$$

With the second of Equations (1.30),

$$\mathrm{tr}(\boldsymbol{x}\boldsymbol{x}^\top)y = \mathrm{tr}(\boldsymbol{A}\boldsymbol{x}\boldsymbol{x}^\top \boldsymbol{x}\boldsymbol{x}^\top).$$

But since $(\boldsymbol{x}^\top \boldsymbol{x})$ is also scalar,

$$\mathrm{tr}(\boldsymbol{x}\boldsymbol{x}^\top)y = \mathrm{tr}(\boldsymbol{A}\boldsymbol{x}\boldsymbol{x}^\top)(\boldsymbol{x}^\top \boldsymbol{x}).$$

The identity then follows from $\mathrm{tr}(\boldsymbol{x}\boldsymbol{x}^\top) = (\boldsymbol{x}^\top \boldsymbol{x})$.

Exercise 9

Let A be a Hermitian matrix, λ be an eigenvalue of A, and v be the corresponding eigenvector. We'll prove that λ is real.

1. By definition of an eigenvector:

$$Av = \lambda v$$

2. Take the conjugate transpose of both sides:

$$(Av)^H = (\lambda v)^H$$

$$v^H A^H = \lambda^* v^H$$

Where λ^* denotes the complex conjugate of λ.

3. Since A is Hermitian, we know that

$$A^H = A$$

. Substituting this:

$$v^H A = \lambda^* v^H$$

4. Now, multiply both sides of the original equation by v^H from the left:

$$v^H A v = v^H \lambda v = \lambda v^H v$$

5. The left side of this equation is equal to $\lambda^* v^H v$ (from step 3), so we have:

$$\lambda^* v^H v = \lambda v^H v$$

6. $v^H v$ is a positive real number (it's the dot product of v with itself), so we can divide both sides by it:

$$^* =$$

7. For a complex number to be equal to its own conjugate, it must be real.

Exercise 10

Let the eigenvectors of $\boldsymbol{A}$ be $\boldsymbol{u}_1$ and $\boldsymbol{u}_2$. Then

$$\begin{aligned}\boldsymbol{u}_2^\top \boldsymbol{A}\boldsymbol{u}_1 &= \lambda_1 \boldsymbol{u}_2^\top \boldsymbol{u}_1\\ \boldsymbol{u}_1^\top \boldsymbol{A}\boldsymbol{u}_2 &= \lambda_2 \boldsymbol{u}_1^\top \boldsymbol{u}_2.\end{aligned}$$

Subtracting gives, since $u_2^\top \boldsymbol{u}_1 = u_1^\top \boldsymbol{u}_2$,

$$(\lambda_1 - \lambda_2)\boldsymbol{u}_1^\top \boldsymbol{u}_2 = \boldsymbol{u}_2^\top \boldsymbol{A}\boldsymbol{u}_1 - \boldsymbol{u}_1^\top \boldsymbol{A}\boldsymbol{u}_2 = 0,$$

since $\boldsymbol{A}$ is symmetric. Hence $\boldsymbol{u}_1$ and $\boldsymbol{u}_2$ are orthogonal.

Exercise 11

For a 2×2 symmetric matrix $\boldsymbol{A}$, Equation (1.45) can be written

$$\boldsymbol{A} = (\boldsymbol{u}_1, \boldsymbol{u}_2)\begin{pmatrix}\lambda_1 & 0\\ 0 & \lambda_2\end{pmatrix}\begin{pmatrix}\boldsymbol{u}_1^\top\\ \boldsymbol{u}_2^\top\end{pmatrix}.$$

Multiplying this out gives Equation (1.46).

Exercise 12

With Equation (1.46),

$$\mathrm{Tr}(\boldsymbol{A}) = \sum_i \lambda_i \mathrm{Tr}\left(\boldsymbol{u}^{(i)}\boldsymbol{u}^{(i)\top}\right) = \sum_i \lambda_i \sum_j u_j^{(i)} u_j^{(i)} = \sum_i \lambda_i \boldsymbol{u}^{(i)\top}\boldsymbol{u}^{(i)} = \sum_i \lambda_i.$$

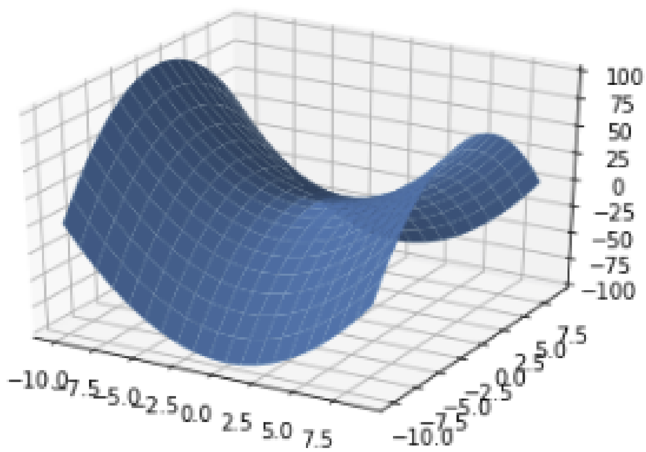

FIGURE 1.1
Plot of $f(\boldsymbol{x}) = x_1^2 - x_2^2$.

Exercise 13

$$\frac{\partial}{\partial \boldsymbol{y}}\left(\frac{1}{\boldsymbol{x}^\top \boldsymbol{A}\boldsymbol{y}}\right) = -\frac{1}{(\boldsymbol{x}^\top \boldsymbol{A}\boldsymbol{y})^2}\frac{\partial}{\partial \boldsymbol{y}}(\boldsymbol{x}^\top \boldsymbol{A}\boldsymbol{y}) = -\frac{1}{(\boldsymbol{x}^\top \boldsymbol{A}\boldsymbol{y})^2}\boldsymbol{A}^\top \boldsymbol{x}.$$

Exercise 14

This calculates the right eigenvectors:

```
import numpy as np
A = [[1,2,3],[4,5,6],[7,8,9]]
print np.linalg.eig(A)[1]

[[-0.23197069 -0.78583024  0.40824829]
 [-0.52532209 -0.08675134 -0.81649658]
 [-0.8186735   0.61232756  0.40824829]]
```

Exercise 15

The following code plots the function, see Figure 1.1.

```
from mpl_toolkits.mplot3d import Axes3D
import matplotlib.pyplot as plt
import numpy as np
fig = plt.figure()
ax = fig.gca(projection='3d')
X1 = range(-10,10)
```

```
X2 = range(-10,10)
X1, X2 = np.meshgrid(X1, X2)
Z = X1**2 - X2**2
surf = ax.plot_surface(X1, X2, Z)
```

The Lagrange function is

$$L = x_1^2 - x_2^2 + \lambda(x_1^2 + x_2^2 - 1).$$

Differentiating:

$$\begin{aligned}
\frac{\partial L}{\partial x_1} &= 2x_1 + 2\lambda x_1 = 0 \Rightarrow x_1(1+\lambda) = 0 \\
\frac{\partial L}{\partial x_2} &= -2x_2 + 2\lambda x_2 = 0 \Rightarrow x_2(-1+\lambda) = 0 \\
\frac{\partial L}{\partial \lambda} &= x_1^2 + x_2^2 = 0.
\end{aligned}$$

Possible values for λ are therefore $\lambda = 1$ and $\lambda = -1$. If $\lambda = 1$, then necessarily $x_1 = 0$ and therefore $x_2 = \pm 1$. Similarly if $\lambda = -1$ then $x_2 = 0,\ x_1 = \pm 1$. It is clear from the plot that the constrained maxima are at $(\pm 1, 0)$ and the minima at $(0, \pm 1)$.

2

Image Statistics

Exercise 1

From the definition of variance,

$$\begin{aligned}\mathrm{var}(G) &= \langle (G - \langle G \rangle)^2 \rangle \\ &= \langle G^2 - 2G\langle G \rangle + \langle G \rangle^2 \rangle \\ &= \langle G^2 \rangle - 2\langle G \rangle^2 + \langle G \rangle^2 \\ &= \langle G^2 \rangle - \langle G \rangle^2.\end{aligned}$$

Using this result,

$$\begin{aligned}\mathrm{var}(a_0 + a_1 G) &= \mathrm{var}(a_1 G) = \langle a_1^2 G^2 \rangle - \langle a_1 G \rangle^2 \\ &= a_1^2(\langle G^2 \rangle - \langle G \rangle^2) = a_1^2 \mathrm{var}(G).\end{aligned}$$

Exercise 2

Let $Y = |X|$ where X has the distribution $\Phi(x)$. For $y > 0$ we have

$$P(y) = \Pr(Y \le y) = \Pr(|X| \le y) = \Pr(-y \le X \le y) = \Phi(y) - \Phi(-y).$$

Hence

$$p(y) = \frac{d}{dy} P(y) = \phi(y) - \phi(-y) = 2\phi(y),$$

and clearly for $y \le 0$, $p(y) = 0$.

Exercise 3

(a) The chi square distribution is

$$P_{\chi^2;m}(z) = \frac{1}{2^{m/2}\Gamma(m/2)} \int_0^z x^{(m-2)/2} e^{x/2} dx.$$

Let $x \to 2t$,

$$\begin{aligned}P_{\chi^2;m}(z) &= \frac{1}{2^{m/2}\Gamma(m/2)} \int_0^{z/2} (2t)^{(m-2)/2} e^{-t} 2dt \\ &= \frac{1}{\Gamma(m/2)} \int_0^{z/2} t^{m/2-1} e^{-t} dt = \gamma(m/2, z/2).\end{aligned}$$

(b) In the GEE Python API,

```
import ee
def chi2cdf(chi2,df):
    ''' Chi square cumulative distribution function '''
    return ee.Image(chi2.divide(2)). \
               gammainc(ee.Number(df).divide(2))
```

Exercise 4

Let X be standard normally distributed, i.e. with density function

$$p(x) = \phi(x) = \frac{1}{\sqrt{2\pi}} e^{-x^2/2}.$$

We can't apply Theorem 2.1 directly to get the density function for X^2 because the function $u(x) = x^2$ is not monotonic wherever $p(x) \neq 0$. But consider the random variable

$$Y = Z^2, \quad \text{where} \quad Z = |X|.$$

We saw in Exercise 2 that Z has the density function

$$f(z) = \begin{cases} 2\phi(z) & \text{for } z > 0 \\ 0 & \text{elsewhere.} \end{cases}$$

The function $y = u(z) = z^2$ is monotonic wherever $f(z) > 0$, so Theorem 2.1 holds. Inverting, $z = w(y) = y^{1/2}$. So for $y > 0$

$$g(y) = f(y^{1/2}) \left| \frac{1}{2} y^{-1/2} \right| = \frac{2}{\sqrt{2\pi}} e^{-y/2} \frac{1}{2} y^{-1/2} = \frac{1}{\sqrt{2\pi}} y^{-1/2} e^{-y/2},$$

which is the chi-square distribution, Equation (2.38), for $m = 1$.

Exercise 5

Given X_1 and X_2 independent and $\sim \mathcal{N}(0,1)$, we want the distribution of $Y = X_1 + X_2$. The joint density function for the random vector $\boldsymbol{X} = (X_1, X_2)^\top$ is, from independence,

$$f(x_1, x_2) = \frac{1}{\sqrt{2\pi}} e^{-x_1^2/2} \frac{1}{\sqrt{2\pi}} e^{-x_2^2/2} = \frac{1}{2\pi} e^{-x_1^2/2 - x_2^2/2}$$

or

$$f(x_1, x_2) = \frac{1}{2\pi} e^{-(x_1+x_2)^2/2 + x_1 x_2}.$$

For fixed X_1, we can write $x_2 = w(y) = y - x_1$. Applying Theorem 2.1, Y has the density

$$g(y, x_1) = f(x_1, y - x_1) \left| \frac{\partial w}{\partial y} \right| = \frac{1}{2\pi} e^{-y^2/2 + (y - x_1)x_1}.$$

The complete density for Y is then obtained by integrating over x_1. Dropping the subscript on x_1, this is

$$p(y) = \int_{-\infty}^{\infty} g(y,x)dx = \frac{1}{2\pi} e^{-y^2/2} \int_{-\infty}^{\infty} e^{(y-x)x} dx.$$

The integral is

$$\int_{-\infty}^{\infty} e^{(y-x)x} dx = \int_{-\infty}^{\infty} e^{xy-x^2+y^2/4-y^2/4} dx = e^{y^2/4} \int_{-\infty}^{\infty} e^{-(x-y/2)^2} dx.$$

Let $u = \sqrt{2}(x - y/2)$. Then the last expression above is

$$e^{y^2/4} \int_{-\infty}^{\infty} e^{-u^2/2} \frac{1}{\sqrt{2}} du = e^{y^2/4} \sqrt{\pi}.$$

Therefore

$$p(y) = \frac{1}{\sqrt{2\pi}\sqrt{2}} e^{-y^2/4},$$

that is, $Y \sim \mathcal{N}(0, 2)$.

Exercise 6

We have, with Theorem 2.3,

$$\langle \bar{Z} \rangle = \left\langle \frac{1}{m} \sum_{i=1}^{m} Z_i \right\rangle = \frac{1}{m} m\mu = \mu,$$

and

$$\mathrm{var}(\bar{Z}) = \left(\frac{1}{m}\right)^2 \mathrm{var}\left(\sum_{i=1}^{m} Z_i\right) = \frac{1}{m^2} m\sigma^2 = \frac{1}{m}\sigma^2.$$

Exercise 7

The gamma function is

$$\Gamma(\alpha) = \int_0^{\infty} x^{\alpha-1} e^{-x} dx.$$

Integrating by parts ($\int v du = uv - \int u dv$) with

$$v = x^{\alpha-1}, \quad du = e^{-x}dx \Rightarrow u = -e^{-x},$$

we have

$$\begin{aligned} \Gamma(\alpha) &= -e^{-x} x^{\alpha-1} \Big|_0^{\infty} + \int_0^{\infty} e^{-x} (\alpha-1) x^{\alpha-2} dx \\ &= (\alpha-1) \int_0^{\infty} x^{\alpha-2} e^{-x} = (\alpha-1)\Gamma(\alpha-1). \end{aligned}$$

Exercise 8

(a) From Equation (2.33)

$$\langle X\rangle = \int_0^\infty x\frac{1}{\beta^\alpha\Gamma(\alpha)}x^{\alpha-1}e^{-x/\beta}dx.$$

Let $y = x/\beta$. Then

$$\begin{aligned}\langle X\rangle &= \int_0^\infty \beta y\frac{1}{\beta^\alpha\Gamma(\alpha)}\beta^{\alpha-1}y^{\alpha-1}e^{-y}\beta dy\\ &= \beta\frac{1}{\Gamma(\alpha)}\int_0^\infty y^\alpha e^{-y}dy = \beta\frac{1}{\Gamma(\alpha)}\Gamma(\alpha+1) = \alpha\beta.\end{aligned}$$

Similarly

$$\begin{aligned}\langle X^2\rangle &= \int_0^\infty (\beta y)^2\frac{1}{\beta^\alpha\Gamma(\alpha)}\beta^{\alpha-1}y^{\alpha-1}e^{-y}\beta dy\\ &= \beta^2\frac{1}{\Gamma(\alpha)}\int_0^\infty y^{\alpha+1}e^{-y}dy\\ &= \beta^2\frac{1}{\Gamma(\alpha)}\Gamma(\alpha+2) = \beta^2\frac{1}{\Gamma(\alpha)}(\alpha+1)\Gamma(\alpha+1) = \alpha(\alpha+1)\beta^2.\end{aligned}$$

Therefore

$$\mathrm{var}(X^2) = \langle X^2\rangle - \langle X\rangle^2 = \alpha(\alpha+1)\beta^2 - (\alpha\beta)^2 = \alpha\beta^2.$$

(b) For exponentially distributed Z_1 and Z_2, let $Z = Z_1 + Z_2$. Then the distribution function for Z is

$$\begin{aligned}P(z) = \Pr(Z_1+Z_2<z) &= \int_0^z\int_0^{z-z_2}\frac{1}{\beta^2}e^{z_1/\beta}e^{z_2/\beta}dz_1dz_2\\ &= \frac{1}{\beta^2}\int_0^z e^{z_2/\beta}\int_0^{z-z_2}e^{z_1/\beta}dz_1\ dz_2\\ &= \frac{1}{\beta^2}\int_0^z e^{z_2/\beta}\left[-\beta e^{z_1/\beta}\Big|_0^{z-z_2}\right]dz_2\\ &= \frac{1}{\beta^2}\int_0^z e^{z_2/\beta}\left[-\beta e^{-z/\beta}e^{z_2/\beta}+\beta\right]dz_2\\ &= \frac{1}{\beta^2}\int_0^z\left[-\beta e^{-z/\beta}+\beta e^{-z_2/\beta}\right]dz_2\\ &= \frac{1}{\beta^2}\left[-\beta z e^{-z/\beta}+\beta\left[-\beta e^{z_2/\beta}\right]_0^z\right]\\ &= \frac{1}{\beta^2}\left[-\beta z e^{-z/\beta}+\beta\left[-\beta(e^{-z/\beta}-1)\right]\right]\\ &= -\frac{1}{\beta}ze^{-z/\beta} - e^{-z/\beta}+1.\end{aligned}$$

Differentiating,

$$p(z) = \frac{d}{dz}P(z) = -\frac{1}{\beta}\left[e^{-z/\beta} - \frac{z}{\beta}e^{-z/\beta}\right] + \frac{1}{\beta}e^{-z/\beta}$$
$$= \frac{z}{\beta^2}e^{-z/\beta} = \frac{1}{\beta^2\Gamma(2)}ze^{-z/\beta}$$

since $\Gamma(2) = 1$.

Exercise 9

From the inverse transformations $x = su,\ y = s(1-u)$ we get with Theorem 2.2 the Jacobian

$$\boldsymbol{J} = \begin{vmatrix} \frac{\partial x}{\partial s} & \frac{\partial x}{\partial u} \\ \frac{\partial y}{\partial s} & \frac{\partial y}{\partial u} \end{vmatrix} = \begin{vmatrix} u & s \\ 1-u & -s \end{vmatrix} = -s$$

and $|J| = s$. Therefore the joint density function of S and U is

$$p(s,u) = \frac{1}{\Gamma(m)\Gamma(n)}\beta^{-m-n}(su)^{n-1}(s(1-u))^{m-1}e^{-su/\beta}e^{-s(1-u)/\beta}s.$$

This can be manipulated into the form

$$p(s,u) = \frac{\Gamma(m+n)}{\Gamma(m)\Gamma(n)}u^{n-1}(1-u)^{m-1}\frac{1}{\Gamma(m+n)}\frac{1}{\beta^{m+n}}s^{n+m-1}e^{-s\beta}.$$

This is just the product of the beta and gamma density functions. We get the marginal distributions for s by integrating over u, so the random variable S is $(m+n,\beta)$-gamma distributed. Similarly U is (m,n)-beta distributed. Since the joint probability density function of S and U is the product of the separate density functions, they are independent.

Exercise 10

For two dimensions, the covariance is

$$\mathrm{cov}(\boldsymbol{a}^\top\boldsymbol{G}, \boldsymbol{b}^\top\boldsymbol{G}) = \mathrm{cov}(a_1G_1 + a_2G_2, b_1G_1 + b_2G_2)$$
$$= a_1b_1\mathrm{var}(G_1) + a_1b_2\mathrm{cov}(G_1G_2) + a_2b_1\mathrm{cov}(G_2,G_1) + a_2b_2\mathrm{var}(G_2),$$

which by inspection is the same as

$$(a_1,a_2)\begin{pmatrix} \mathrm{var}(G_1) & \mathrm{cov}(G_1,G_2) \\ \mathrm{cov}(G_2,G_1) & \mathrm{var}(G_2) \end{pmatrix}\begin{pmatrix} b_1 \\ b_2 \end{pmatrix} = \boldsymbol{a}^\top\boldsymbol{\Sigma}\boldsymbol{b}.$$

More generally,

$$\mathrm{cov}(\boldsymbol{a}^\top\boldsymbol{G}, \boldsymbol{b}^\top\boldsymbol{G}) = \langle \boldsymbol{a}^\top\boldsymbol{G}(\boldsymbol{b}^\top\boldsymbol{G})^\top\rangle = \boldsymbol{a}^\top\langle\boldsymbol{G}\boldsymbol{G}^\top\rangle\boldsymbol{b} = \boldsymbol{a}^\top\boldsymbol{\Sigma}\boldsymbol{b}.$$

Exercise 11

For $\boldsymbol{\Sigma} = \sigma^2 \boldsymbol{I}$, $|\boldsymbol{\Sigma}| = (\sigma^2)^N$, and the multivariate density simplifies as follows

$$\begin{aligned}
p(\boldsymbol{g}) &= \frac{1}{(2\pi)^{N/2}\sqrt{|\boldsymbol{\Sigma}|}} \exp\left(-\frac{1}{2}(\boldsymbol{g}-\boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1}(\boldsymbol{g}-\boldsymbol{\mu})\right) \\
&= \frac{1}{(\sqrt{2\pi}\sigma)^N} \exp\left(-\|\boldsymbol{g}-\boldsymbol{\mu}\|^2/2\sigma^2\right) \\
&= \prod_{i=1}^{N} \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-(g_i-\mu_i)^2/2\sigma^2\right).
\end{aligned}$$

For $N = 1$, therefore,

$$p(g) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-(g-\mu)^2/2\sigma^2\right).$$

The mean of G is given by

$$\langle G \rangle = \frac{1}{\sqrt{2\pi}\sigma} \int_{-\infty}^{\infty} g \cdot e^{-\frac{1}{2}\left(\frac{g-\mu}{\sigma}\right)^2} dg$$

which we can write in the form

$$\langle G \rangle = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \frac{g-\mu}{\sigma} e^{-\frac{1}{2}\left(\frac{g-\mu}{\sigma}\right)^2} dg + \frac{\mu}{\sqrt{2\pi}\sigma} \int_{-\infty}^{\infty} e^{-\frac{1}{2}\left(\frac{g-\mu}{\sigma}\right)^2} dg.$$

The second term is just $\mu \int_{-\infty}^{\infty} p(g)dg = \mu$. The first term vanishes. That is, making the change of variable substitution

$$y = \frac{1}{2}\left(\frac{g-\mu}{\sigma}\right),$$

we have

$$\int_{\mu}^{\infty} \frac{g-\mu}{\sigma} e^{-\frac{1}{2}\left(\frac{g-\mu}{\sigma}\right)^2} dg = \sigma \int_{0}^{\infty} e^{-y} dy = \sigma$$

and similarly

$$\int_{-\infty}^{\mu} \frac{g-\mu}{\sigma} e^{-\frac{1}{2}\left(\frac{g-\mu}{\sigma}\right)^2} dg = -\sigma.$$

Thus $\langle G \rangle = \mu$.

Exercise 12

The covariance matrix estimate can be written as

$$\boldsymbol{s} = \frac{m}{m-1} \cdot \frac{1}{m} \sum_{i=1}^{m} (\boldsymbol{z}_i - \bar{\boldsymbol{z}})(\boldsymbol{z}_i - \bar{\boldsymbol{z}})^\top.$$

Expanding,

$$\begin{aligned}
\boldsymbol{s} &= \frac{m}{m-1}\left[\frac{1}{m}\sum_i \boldsymbol{z}_i\boldsymbol{z}_1^\top - 2\bar{\boldsymbol{z}}\bar{\boldsymbol{z}}^\top + \frac{1}{m}\cdot m\bar{\boldsymbol{z}}\bar{\boldsymbol{z}}^\top\right] \\
&= \frac{m}{m-1}\left[\frac{1}{m}\boldsymbol{\mathcal{Z}}^\top\boldsymbol{\mathcal{Z}} - \bar{\boldsymbol{z}}\bar{\boldsymbol{z}}^\top\right]
\end{aligned}$$

With Equation (2.52) we can write

$$\begin{aligned}
\boldsymbol{s} &= \frac{m}{m-1}\left[\frac{1}{m}\boldsymbol{\mathcal{Z}}^\top\boldsymbol{\mathcal{Z}} - \frac{1}{m^2}\boldsymbol{\mathcal{Z}}^\top\mathbf{1}_m\mathbf{1}_m^\top\boldsymbol{\mathcal{Z}}\right] \\
&= \frac{1}{m-1}\left[\boldsymbol{\mathcal{Z}}^\top\boldsymbol{\mathcal{Z}} - \frac{1}{m}\boldsymbol{\mathcal{Z}}^\top\mathbf{1}_m\mathbf{1}_m^\top\boldsymbol{\mathcal{Z}}\right] \\
&= \frac{1}{m-1}\boldsymbol{\mathcal{Z}}^\top\left[\boldsymbol{I}_{mm} - \frac{1}{m}\mathbf{1}_m\mathbf{1}_m^\top\right]\boldsymbol{\mathcal{Z}} = \frac{1}{m-1}\boldsymbol{\mathcal{Z}}\boldsymbol{H}\boldsymbol{\mathcal{Z}}.
\end{aligned}$$

Clearly, $\boldsymbol{H}^\top = \boldsymbol{H}$, so $\boldsymbol{H}$ is symmetric. Also

$$\begin{aligned}
\boldsymbol{H}\boldsymbol{H} &= \left[\boldsymbol{I}_{mm} - \frac{1}{m}\mathbf{1}_m\mathbf{1}_m^\top\right]\left[\boldsymbol{I}_{mm} - \frac{1}{m}\mathbf{1}_m\mathbf{1}_m^\top\right] \\
&= \boldsymbol{I}_{mm} - \frac{1}{m}\mathbf{1}_m\mathbf{1}_m^\top - \frac{1}{m}\mathbf{1}_m\mathbf{1}_m^\top + \frac{1}{m^2}\mathbf{1}_m\mathbf{1}_m^\top\mathbf{1}_m\mathbf{1}_m^\top \\
&= \boldsymbol{I}_{mm} - \frac{1}{m}\mathbf{1}_m\mathbf{1}_m^\top = \boldsymbol{H},
\end{aligned}$$

since $\mathbf{1}_m^\top\mathbf{1}_m = m$. Let $\boldsymbol{x}$ be any m-component vector. Then with the above result

$$\boldsymbol{x}^\top\boldsymbol{s}\boldsymbol{x} = \frac{1}{m-1}\boldsymbol{x}^\top\boldsymbol{\mathcal{Z}}^\top\boldsymbol{H}\boldsymbol{H}\boldsymbol{\mathcal{Z}}\boldsymbol{x} = \frac{1}{m-1}\boldsymbol{y}^\top\boldsymbol{y} \geq 0,$$

where $\boldsymbol{y} = \boldsymbol{H}\boldsymbol{\mathcal{Z}}\boldsymbol{x}$. Therefore $\boldsymbol{s}$ is positive semi-definite.

Exercise 13

With $s_i = \sqrt{s_{ii}}$ Equation (2.58) is

$$\begin{aligned}
\boldsymbol{d}^{-1/2}\boldsymbol{s}\boldsymbol{d}^{-1/2} &= \begin{pmatrix} 1/s_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 1/s_m \end{pmatrix}\begin{pmatrix} s_{11} & \cdots & s_{1m} \\ \vdots & \ddots & \vdots \\ s_{m1} & \cdots & s_{mm} \end{pmatrix}\begin{pmatrix} 1/s_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 1/s_m \end{pmatrix} \\
&= \begin{pmatrix} s_{11}/s_1^2 & \cdots & s_{1m}/s_1s_m \\ \vdots & \ddots & \vdots \\ s_{m1}/s_ms_1 & \cdots & s_{mm}/s_m^2 \end{pmatrix} = \boldsymbol{r}.
\end{aligned}$$

Exercise 14

Let A_i represent the situation "auto is behind door i". The *a priori* probabilities are $\Pr(A_i) = 1/3$, $i = 1, 2, 3$. Let O_i be the observation "quizmaster opens door i". Suppose the contestant chooses door 2 and the quizmaster opens door 1. Then we have the following conditional probabilities:

$$\begin{aligned}
\Pr(O_1 \mid A_1) &= 0 \quad \text{quizmaster won't give the car away.}\\
\Pr(O_1 \mid A_2) &= 1/2 \quad \text{quizmaster is indifferent.}\\
\Pr(O_1 \mid A_3] &= 1 \quad \text{quizmaster has no choice.}
\end{aligned}$$

Now apply Bayes' Theorem to find the *a posteriori* probability that the auto is behind door 2, given the observation:

$$\begin{aligned}
\Pr(A_2 \mid O_1) &= \frac{\Pr(O_1 \mid A_2)\Pr(A_2)}{\Pr(O_1 \mid A_2)\Pr(A_2) + \Pr(O_1 \mid A_3)\Pr(A_3)}\\
&= \frac{(1/2)(1/3)}{(1/2)(1/3) + (1)(1/3)} = 1/3,
\end{aligned}$$

whereas, with the same argument,

$$\Pr(A_3 \mid O_1) = 2/3.$$

The contestant would therefore be well-advised to switch to door 3.

Exercise 15

From the monotonicity of F_0,

$$\Pr(Q \geq q \mid H_0) = \Pr(F_0(Q) \geq F_0(q)) = 1 - \Pr(F_0(Q) < F_0(q)).$$

Therefore,

$$\Pr(F_0(Q) < F_0(q)) = 1 - \Pr(Q \geq q \mid H_0) = \Pr(Q < q \mid H_0) = F_0(q).$$

Exercise 16

```
import numpy as np
import scipy.stats as stats
s1 = np.random.randn(100)
s2 = np.random.randn(100)

stat, p = stats.ttest_ind(s1,s2)
print p

stat, p = stats.bartlett(s1,s2)
print p

0.35275438625619526
0.42889408182589617
```

Exercise 17

$$\begin{aligned}\frac{L(\mu_0)}{L(\hat{\mu})} &= \exp\left(-\sum_\nu (z(\nu)-\mu_0)^2/2\sigma^2\right) \Bigg/ \exp\left(-\sum_\nu (z(\nu)-\bar{z})^2/2\sigma^2\right) \\ &= \exp\left(-\frac{1}{2\sigma^2}\sum_\nu (z(\nu)-\mu_0)^2 + (z(\nu)-\bar{z})^2\right) \\ &= \exp\left(-\frac{1}{2\sigma^2/m}(\bar{z}-\mu_0)^2\right) \le k\end{aligned}$$

Exercise 18

Differentiating Equation (2.88):

$$0 = \frac{\partial z}{\partial a} = -\frac{2}{\sigma^2}\sum_\nu (y(\nu) - a - bx(\nu))$$
$$0 = \frac{\partial z}{\partial b} = -\frac{2}{\sigma^2}\sum_\nu x(\nu)(y(\nu) - a - bx(\nu)).$$

From the first equation we get immediately

$$a = \bar{y} - b\bar{x},$$

where $\bar{x}$ and $\bar{y}$ are given by Equation (2.90). Substituting for a in the second equation, we have

$$0 = \sum_\nu x(\nu)y(\nu) - (\bar{y} - b\bar{x})\sum_\nu x(\nu) - b\sum_\nu x(\nu)^2.$$

Re-arranging:

$$b\left(\sum_\nu x(\nu)^2 - \bar{x}\sum_\nu x(\nu)\right) = \sum_\nu x(\nu)y(\nu) - \bar{y}\sum_\nu x(\nu).$$

The expression in brackets on the left is just

$$\sum_\nu x(\nu)^2 - m\bar{x}^2 = \sum_\nu (x(\nu)-\bar{x})^2 = ms_{xx},$$

as can easily be seen by expansion. Similarly the right hand side is

$$\sum_\nu (x(\nu)-\bar{x})(y(\nu)-\bar{y}) = ms_{xy}.$$

Hence

$$b = \frac{s_{xy}}{s_{xx}}.$$

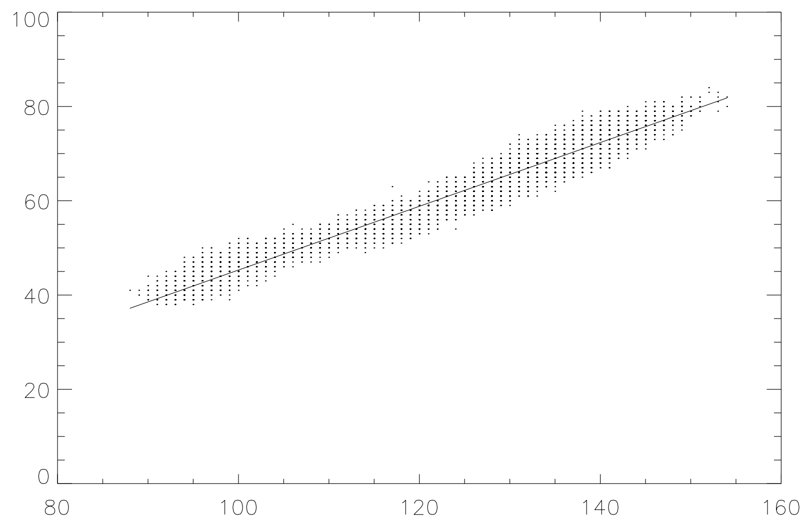

FIGURE 2.1
Regression of band 2 on band 1.

To show that these values minimize Equation (2.88), we require the Hessian matrix:

$$
\begin{aligned}
\frac{\partial^2 z}{\partial a^2} &= \frac{2m}{\sigma^2}\\
\frac{\partial^2 z}{\partial b^2} &= \frac{2}{\sigma^2}\sum_\nu x(\nu)^2\\
\frac{\partial^2 z}{\partial b \partial a} &= \frac{\partial^2 z}{\partial a \partial b} = \frac{2}{\sigma^2}\sum_\nu x(\nu)
\end{aligned}
$$

The Hessian matrix is thus

$$
\boldsymbol{H} = \frac{2m}{\sigma^2}\begin{pmatrix} 1 & \bar{x} \\ \bar{x} & s_{xx} + \bar{x}^2 \end{pmatrix}.
$$

For any vector $\boldsymbol{y} \neq 0$ we then have

$$
\boldsymbol{y}^\top \boldsymbol{H} \boldsymbol{y} = (y_1 + \bar{x} y_2)^2 + s_{xx} y_2^2 > 0,
$$

so $\boldsymbol{H}$ is positive definite and (a, b) is a minimum of Equation (2.88).

Exercise 19

We require the derivative of a with respect to y_i, where

$$a = \bar{y} - b\bar{x} = \frac{1}{m}\sum_\nu y(\nu) - \frac{s_{xy}}{s_{xx}}\bar{x}.$$

We have

$$\frac{\partial a}{\partial y(\nu)} = \frac{1}{m} - \frac{\bar{x}}{s_{xx}}\frac{\partial s_{xy}}{\partial y(\nu)}$$

and

$$\begin{aligned}\frac{\partial s_{xy}}{\partial y(\nu)} &= \frac{1}{m}\left(x(\nu) - \bar{x} - \sum_j (x_j - \bar{x})\frac{\partial \bar{y}}{\partial y(\nu)}\right)\\ &= \frac{1}{m}(x(\nu) - \bar{x}).\end{aligned}$$

Hence

$$\frac{\partial a}{\partial y(\nu)} = \frac{1}{m}\left(1 - \frac{\bar{x}}{s_{xx}}(x(\nu) - \bar{x})\right)$$

and

$$\left(\frac{\partial a}{\partial y(\nu)}\right)^2 = \frac{1}{m^2}\left(1 - \frac{2\bar{x}}{s_{xx}}(x(\nu) - \bar{x}) + \frac{\bar{x}^2}{s_{xx}^2}(x(\nu) - \bar{x})^2\right).$$

Summing over i gives

$$\begin{aligned}\sum_\nu \left(\frac{\partial a}{\partial y(\nu)}\right)^2 &= \frac{1}{m^2}\left(m + 0 + \frac{\bar{x}^2}{s_{xx}^2} m s_{xx}\right)\\ &= \frac{1}{m}\left(\frac{s_{xx} + \bar{x}^2}{s_{xx}}\right) = \frac{\sum_\nu x(\nu)^2}{m^2 s_{xx}},\end{aligned}$$

from which the expression for σ_a^2 in Equation (2.91) readily follows.

Exercise 20

From Equation (2.98),

$$z(\boldsymbol{w}) = \frac{1}{\sigma^2}(\boldsymbol{y}^\top\boldsymbol{y} - \boldsymbol{y}^\top\boldsymbol{\mathcal{X}}\boldsymbol{w} - \boldsymbol{w}^\top\boldsymbol{\mathcal{X}}\boldsymbol{y} + \boldsymbol{w}^\top\boldsymbol{\mathcal{X}}^\top\boldsymbol{\mathcal{X}}\boldsymbol{w}),$$

so that

$$\frac{\partial z(\boldsymbol{w})}{\partial \boldsymbol{w}} = 0 = \frac{1}{\sigma^2}(-\boldsymbol{\mathcal{X}}^\top\boldsymbol{y} - \boldsymbol{\mathcal{X}}^\top\boldsymbol{y} + 2\boldsymbol{\mathcal{X}}^\top\boldsymbol{\mathcal{X}}\boldsymbol{w})$$

from which Equation (2.99) follows.

Exercise 21

```
import  numpy as np
from osgeo import gdal
from osgeo.gdalconst import GA_ReadOnly
import matplotlib.pyplot as plt
import scipy.stats as stats

gdal.AllRegister()
inDataset = gdal.Open('imagery/AST_20010409',GA_ReadOnly)
cols = inDataset.RasterXSize
rows = inDataset.RasterYSize
bands = inDataset.RasterCount
image = np.zeros((bands,rows*cols))
for b in range(bands):
    band = inDataset.GetRasterBand(b+1)
    image[b,:]=band.ReadAsArray(0,0,cols,rows).ravel()
inDataset = None
b, a,_,_,_  = stats.linregress(image[0,:],image[1,:])
plt.plot(image[0,:],image[1,:],'.',
[50,250], [a+b*50,a+b*250],'r')
```

Exercise 22

The numerator of the right hand side of Equation (2.94) becomes

$$
\begin{aligned}
&\sum_\nu (y(\nu) + \hat{y}(\nu) - \hat{y}(\nu) - \bar{y})(\hat{y}(\nu) - \bar{y}) \\
&= \sum_\nu (y(\nu) - \hat{y}(\nu))(\hat{y}(\nu) - \bar{y}) + \sum_\nu (\hat{y}(\nu) - \bar{y})^2 \\
&= \sum_\nu (\hat{y}(\nu) - \bar{y})^2,
\end{aligned}
$$

since the term $\sum_\nu (y(\nu) - \hat{y}(\nu))(\hat{y}(\nu) - \bar{y})$ vanishes, see Section 2.6.2. Including the denominator then gives the desired result.

Exercise 23

From Equations (2.107) and (2.108),

$$
\hat{\boldsymbol{w}} = \frac{1}{\lambda}(\boldsymbol{\mathcal{X}}^\top \boldsymbol{y} - \boldsymbol{\mathcal{X}}^\top \boldsymbol{\mathcal{X}} \boldsymbol{w})
$$

Solving for $\hat{\boldsymbol{w}}$ gives Equation (2.109).

Exercise 24

We want to show that

$$-\int p(x) \ln \left[\frac{q(x)}{p(x)}\right] dx \geq 0$$

when $q(x) \neq p(x)$. From Jensen's inequality, Equation (2.121), identify $f(x)$ with the convex function $-\ln(x)$ and $g(x)$ with the ratio $q(x)/p(x)$ to get

$$-\int p(x) \ln \left[\frac{q(x)}{p(x)}\right] dx \geq -\ln \left(\int q(x) dx\right).$$

But, since $q(x)$ is also a probability density function, the integral on the right hand side is one, and its logarithm vanishes.

Exercise 25

This is simply because $\mathrm{KL}(p, q) \neq \mathrm{KL}(q, p)$, whereas a distance measure must be symmetric.

3

Transformations

Exercise 1

Substituting $\hat{g}(-1) = -1/2\mathbf{i}$, $\hat{g}(1) = 1/2\mathbf{i}$ with $\hat{g}(f) = 0$ elsewhere into (3.1),

$$\begin{aligned}\int_{-\infty}^{\infty} \hat{g}(f) e^{-\mathbf{i}2\pi f x} dx &= -\frac{1}{2\mathbf{i}} e^{-\ i2\pi x} + \frac{1}{2\mathbf{i}} e^{\mathbf{i}2\pi x} \\ &= \frac{1}{2\mathbf{i}} \left(-(\cos 2\pi x - \mathbf{i} \sin 2\pi x) + (\cos 2\pi x + \mathbf{i} \sin 2\pi x)\right) \\ &= \sin(2\pi x).\end{aligned}$$

Exercise 2

Multiply Equation (3.5) by $e^{-\mathbf{i}2\pi k' j}$ and sum over j to get, with (3.7),

$$\sum_j e^{-\mathbf{i}2\pi k' j} g(j) = \sum_k \hat{g}(k) \sum_j e^{\mathbf{i}2\pi (k-k')j/c} = \sum_k \hat{g}(k) c \delta_{k,k'} = c\hat{g}(k'),$$

which gives Equation (3.6).

Exercise 3

From Equation (3.5) we have the equations

$$\begin{aligned} j = 0: &\quad 2 = \hat{g}(0) + \hat{g}(1) + \hat{g}(2) + \hat{g}(3) \\ j = 1: &\quad 4 = \hat{g}(0) + \mathbf{i}\hat{g}(1) - \hat{g}(2) - \mathbf{i}\hat{g}(3) \\ j = 2: &\quad 6 = \hat{g}(0) - \hat{g}(1) + \hat{g}(2) - \hat{g}(3) \\ j = 3: &\quad 8 = \hat{g}(0) - \mathbf{i}\hat{g}(1) - \hat{g}(2) + \mathbf{i}\hat{g}(3) \end{aligned}$$

which we can solve by substitution to get

$$\hat{g}(0) = 5,\ \hat{g}(1) = -1 + \mathbf{i},\ \hat{g}(2) = -1,\ \hat{g}(3) = -1 - \mathbf{i}.$$

In Python:

```
>>> import numpy.fft as fft
>>> fft.fft([2,4,6,8])/4
array([ 5.+0.j, -1.+1.j, -1.+0.j, -1.-1.j])
```

Exercise 4

From Equation (3.8)

$$\begin{aligned} &g(i,j)\exp\left(\mathbf{i}2\pi(ik_0/c+j\ell_0/r)\right) \\ &= \sum_{k,\ell}\hat{g}(k,\ell)\exp\left(\mathbf{i}2\pi(ik/c+ik_0/c+j\ell/r+j\ell_0/r)\right). \end{aligned}$$

Let $k' = k + k_0$ and $\ell' = \ell + \ell_0$. Then

$$\begin{aligned} &g(i,j)\exp\left(\mathbf{i}2\pi(ik_0/c+j\ell_0/r)\right) \\ &= \sum_{k',\ell'}\hat{g}(k'-k_0,\ell'-\ell_0)\exp\left(\mathbf{i}2\pi(ik'/c+j\ell'/r)\right), \end{aligned}$$

which is the translation property

$$g(i,j)\exp\left(\mathbf{i}2\pi(ik_0/c+j\ell_0/r)\right) \Leftrightarrow \hat{g}(k'-k_0,\ell'-\ell_0).$$

Exercise 5

Starting with Equation (3.6) we have

$$|\hat{g}(k)|^2 = \hat{g}(k)\hat{g}^*(k) = \frac{1}{c^2}\sum_{j,j'} g(j)g^*(j')e^{-\mathbf{i}2\pi k(j-j')/c},$$

where the $*$ denotes the complex conjugate. Therefore, with Equation (3.7),

$$\begin{aligned} \sum_k |\hat{g}(k)|^2 &= \frac{1}{c^2}\sum_{j,j'} g(j)g^*(j')\sum_k e^{-\mathbf{i}2\pi k(j-j')/c} \\ &= \frac{1}{c^2}\sum_{j,j'} g(j)g^*(j')c\delta_{j,j'} = \frac{1}{c}\sum_j |g(j)|^2. \end{aligned}$$

Exercise 6

The standard Haar basis for V_2 is $\{\phi_{2,0},\phi_{2,1},\phi_{2,2},\phi_{2,3}\}$. According to the Orthogonal Decomposition Theorem, $\phi_{2,0}$ can be expanded as

$$\phi_{2,0}(x) = \frac{\langle\phi_{2,0},\phi_{1,0}\rangle}{\langle\phi_{1,0},\phi_{1,0}\rangle}\phi_{1,0}(x) + \frac{\langle\phi_{2,0},\phi_{1,1}\rangle}{\langle\phi_{1,1},\phi_{1,1}\rangle}\phi_{1,1}(r) + r(x) = \frac{1}{2}\phi_{1,0}(x) + r(x),$$

where $r(x)$ is in the residual space $V_1^{\perp}$. Hence

$$\begin{aligned} r(x) &= \phi_{2,0}(x) - \frac{1}{2}\phi_{1,0}(x) = \phi(4x) - \frac{1}{2}(\phi(4x)+\phi(4x-1)) = \frac{1}{2}(\phi(4x)-\phi(4x-1)) \\ &= \frac{1}{2}\psi_{1,0}(x). \end{aligned}$$

Similarly the residuals for other V_2 basis functions can be expressed in terms of $\psi_{1,0}$ or $\psi_{1,1}$. Therefore $\{\psi_{1,0},\psi_{1,1}\}$ is an orthogonal basis for $V_1^{\perp}$.

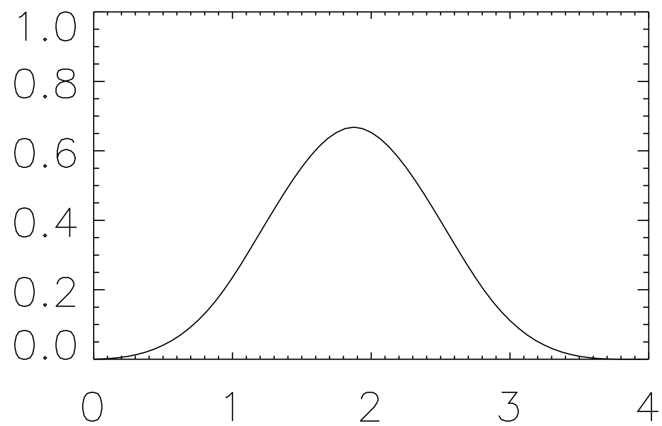

FIGURE 3.1
Cascade algorithm approximation to the B-spline scaling function.

Exercise 7

The dilation equation for $\phi(x-j)$ is

$$\phi(x-j) = \sum_k c_k \phi(2x - 2j - k),$$

so we have, for $j \neq 0$,

$$\begin{aligned} 0 = \langle \phi(x), \phi(x-j) \rangle &= \left\langle \sum_k c_k \phi(2x-k), \sum_{k'} c_k \phi(2x - 2j - k') \right\rangle \\ &= \sum_{k,k'} c_k c_{k'} \langle \phi(2x-k) \phi(2x-2j-k') \rangle \\ &= \sum_{k,k'} c_k c_{k'} \frac{1}{2} \delta_{k,k'+2j} \\ &= \frac{1}{2} \sum_k c_k c_{k-2j}. \end{aligned}$$

Exercise 8

From the dilation equation,

$$\int \phi(x)dx = \sum_k c_k \int \phi(2x-k)dx = \sum_k c_k \int \phi(u)\frac{du}{2} = \frac{1}{2}\sum_k c_k \int \phi(x)dx,$$

and since the integral is not zero,

$$\sum_k c_k = 2.$$

Exercise 9

The modified program is (see Figure 3.1):

```
def F(x,i,c):
    if i==0:
        if x==0:
            return 1.0
        else:
            return 0.0
    else:
        return c[0]*F(2*x,i-1,c)+c[1]*F(2*x-1,i-1,c) \
            +c[2]*F(2*x-2,i-1,c)+c[3]*F(2*x-3,i-1,c) \
            +c[4]*F(2*x-4,i-1,c)

# B-spline wavelet refinement coefficients
c = np.zeros(5)
c[0]=1./8; c[4]=1./8; c[1]=1./2; c[3]=1./2; c[2]=3./4

# fourth order approximation
n = 4
x = np.array(range(4*2**n))/float(2**n)
FF = np.zeros(4*2**n)
for i in range(4*2**n):
    FF[i] = F(x[i],n,c)

plt.plot(x,FF)
plt.ylim(0,1)
```

Exercise 10

(a) This can be seen from the cascade algorithm, which approximates a scaling function to any degree of accuracy: $\phi(x) = \lim_{i\to\infty} f_i(x)$, where

$$\begin{aligned} f_0(x) &= \delta_{x,0} \\ f_i(x) &= c_0 f_{i-1}(2x) + c_1 f_{i-1}(2x-1) + \ldots + c_n f_{i-1}(2x-n). \end{aligned}$$

Clearly no iteration will lead to a value of $f(x) \neq 0$ for $x < 0$. For $i = 0$, $f_0(x) \neq 0$ only for $x = 0$. For $i = 1$ the largest value at which $f_1(x)$ can be non-vanishing is $x = n/2$, for $i = 2$ it is $x = 3n/4$, and for any i it is $\frac{(2^i-1)n}{2^i}$. Hence the scaling function vanishes outside the interval $[0, n]$.

(b) We know from the above that the scaling function $\phi(x)$, which appears on the left hand side of the dilation Equation (3.25), is supported on a finite interval $[a, b]$ included within $[0, n]$. The shifted functions $\phi(2x - k)$ on the right hand side are supported on $[(a + k)/2, (b + k)/2]$ and the k ranges from 0 to n. Equating the intervals for left and right hand sides,

$$[a, b] = \left[\frac{a}{2}, \frac{b+n}{2}\right],$$

leads to the solution $a = 0,\ b = n$.

Exercise 11

After line 40 insert the code

```
G_r = reshape(G_r,(rows,cols,r))
outfile,fmt = auxil.select_outfilefmt()
driver = gdal.GetDriverByName(fmt)
outDataset = driver.Create('imagery/recon',
                    cols,rows,bands,GDT_Float32)
projection = inDataset.GetProjection()
geotransform = inDataset.GetGeoTransform()
if geotransform is not None:
    outDataset.SetGeoTransform(geotransform)
if projection is not None:
    outDataset.SetProjection(projection)
for k in range(r):
    outBand = outDataset.GetRasterBand(k+1)
    outBand.WriteArray(G_r[:,:,k],0,0)
    outBand.FlushCache()
outDataset = None
```

Exercise 12

The SVD of the $m \times N$ data matrix $\boldsymbol{\mathcal{G}}$ is

$$\boldsymbol{\mathcal{G}} = \boldsymbol{U}\boldsymbol{W}\boldsymbol{V}^\top, \quad \boldsymbol{U}^\top\boldsymbol{U} = \boldsymbol{V}^\top\boldsymbol{V} = \boldsymbol{I},\ \boldsymbol{W} \text{ is diagonal.}$$

Then

$$\begin{aligned}(m-1)\boldsymbol{\Sigma} &= \boldsymbol{\mathcal{G}}^\top\boldsymbol{\mathcal{G}} = \boldsymbol{V}\boldsymbol{W}\boldsymbol{U}^\top\boldsymbol{U}\boldsymbol{W}\boldsymbol{V}^\top \\ &= \boldsymbol{V}\boldsymbol{W}^2\boldsymbol{V}^\top\end{aligned}$$

or

$$\boldsymbol{V}^\top\boldsymbol{\Sigma}\boldsymbol{V} = \frac{1}{m-1}\boldsymbol{W}^2 \text{ is diagonal,}$$

so $\boldsymbol{V}$ is the matrix of eigenvectors and the eigenvalues are proportional to the squares of the diagonal elements of $\boldsymbol{W}$.

Exercise 13

(a) The difference of the class means of the projections v is

$$\bar{v}_1 - \bar{v}_2 = \frac{1}{n_1}\sum_{\boldsymbol{g}\in B_1} \boldsymbol{w}^\top \boldsymbol{g} - \frac{1}{n_2}\sum_{\boldsymbol{g}\in B_2} \boldsymbol{w}^\top \boldsymbol{g} = \boldsymbol{w}^\top(\boldsymbol{m}_1 - \boldsymbol{m}_2).$$

Hence

$$(\bar{v}_1 - \bar{v}_2)^2 = \|\boldsymbol{w}^\top(\boldsymbol{m}_1 - \boldsymbol{m}_2)\|^2 = \|(\boldsymbol{m}_1 - \boldsymbol{m}_2)^\top \boldsymbol{w}\|^2 = \boldsymbol{w}^\top \boldsymbol{\mathcal{C}}_B \boldsymbol{w}$$

and, since the overall variance is

$$\mathrm{var}(v) = \boldsymbol{w}^\top\ \boldsymbol{\mathcal{C}}_1 \boldsymbol{w} + \boldsymbol{w}^\top \boldsymbol{\mathcal{C}}_2 \boldsymbol{w},$$

we get Equation (3.81).

(b) By differentiating $J(\boldsymbol{w})$ with respect to $\boldsymbol{w}$ and setting the result equal to zero, we get the generalized eigenvalue problem

$$\boldsymbol{\mathcal{C}}_B \boldsymbol{w} = \lambda(\boldsymbol{\mathcal{C}}_1 + \boldsymbol{\mathcal{C}}_2)\boldsymbol{w}.$$

But

$$\boldsymbol{\mathcal{C}}_B \boldsymbol{w} = (\boldsymbol{m}_1 - \boldsymbol{m}_2)[(\boldsymbol{m}_1 - \boldsymbol{m}_2)^\top \boldsymbol{w}] \propto (\boldsymbol{m}_1 - \boldsymbol{m}_2)$$

from which the result follows.

(c) The following script calculates and plots Fisher's linear discriminant for the simulated data:

```
%matplotlib inline
import numpy as np
import matplotlib.pyplot as plt
# generate a 2-band image with 2 classes
n1 = np.random.randn(1000)
n2 = n1 + np.random.randn(1000)
B1 = np.stack((n1,n2),1)
B2 = np.stack((n1+4,n2),1)
image = np.zeros((2000,2))
image[:1000,:] = B1
image[1000:,:] = B2
center_x = np.mean(image[:,0])
# determine direction W
m1 = np.mat([np.mean(n1),np.mean(n2)])
m2 = np.mat([np.mean(n1)+4,np.mean(n2)])
C1 = np.mat(np.cov(B1,rowvar=False))
C2 = np.mat(np.cov(B2,rowvar=False))
```

```
W = (C1+C2).I*(m1-m2).T
a = W[1,0]/W[0,0]
# scatterplot and Fischer discriminant
plt.xlim((-8,8))
plt.ylim((-8,8))
plt.axes().set_aspect(1)
plt.scatter(image[:,0],image[:,1])
plt.plot([center_x-10,center_x+10],[-10*a,10*a],'r')
```

Exercise 14

The code is available from the author on request.

Exercise 15

We have $\boldsymbol{w} = \boldsymbol{\mathcal{G}}^\top \boldsymbol{\alpha} = \lambda^{-1/2}\boldsymbol{\mathcal{G}}^\top \boldsymbol{v}$. Therefore

$$\begin{aligned}
\mathrm{var}(\boldsymbol{w}^\top \boldsymbol{g}) &= \boldsymbol{w}^\top \boldsymbol{\mathcal{C}} \boldsymbol{w} = \boldsymbol{w}^\top \boldsymbol{\mathcal{G}}^\top \boldsymbol{\mathcal{G}} \boldsymbol{w}/(m-1) \\
&= \lambda^{-1} \boldsymbol{v}^\top \boldsymbol{\mathcal{G}}\boldsymbol{\mathcal{G}}^\top \boldsymbol{\mathcal{G}}\boldsymbol{\mathcal{G}}^\top \boldsymbol{v}/(m-1) \\
&= \lambda^{-1}\lambda^2 \boldsymbol{v}^\top \boldsymbol{v}/(m-1) = \lambda/(m-1).
\end{aligned}$$

Exercise 16

Primal formulation: with Equation (3.54), we maximize

$$\frac{\boldsymbol{a}^\top \boldsymbol{\mathcal{G}}^\top \boldsymbol{\mathcal{G}} \boldsymbol{a}}{\boldsymbol{a}^\top \boldsymbol{\mathcal{G}}_N^\top \boldsymbol{\mathcal{G}}_N \boldsymbol{a}} - 1$$

with respect to $\boldsymbol{a}$. This leads to the generalized eigenvalue problem

$$\boldsymbol{\mathcal{G}}_N^\top \boldsymbol{\mathcal{G}}_N \boldsymbol{a} = \lambda \boldsymbol{\mathcal{G}}^\top \boldsymbol{\mathcal{G}} \boldsymbol{a}.$$

Dual formulation: with $\boldsymbol{a} \propto \boldsymbol{\mathcal{G}}^\top \boldsymbol{b}$, we maximize

$$\frac{\boldsymbol{b}^\top \boldsymbol{\mathcal{G}}\boldsymbol{\mathcal{G}}^\top \boldsymbol{\mathcal{G}}\boldsymbol{\mathcal{G}}^\top \boldsymbol{b}}{\boldsymbol{b}^\top \boldsymbol{\mathcal{G}}\boldsymbol{\mathcal{G}}_N^\top \boldsymbol{\mathcal{G}}_N \boldsymbol{\mathcal{G}}^\top \boldsymbol{b}} - 1.$$

Using the notation $\boldsymbol{\mathcal{K}} = \boldsymbol{\mathcal{G}}\boldsymbol{\mathcal{G}}^\top$ and $\boldsymbol{\mathcal{K}}_N = \boldsymbol{\mathcal{G}}\boldsymbol{\mathcal{G}}_N^\top$ we get the generalized eigenvalue problem

$$\boldsymbol{\mathcal{K}}_N \boldsymbol{\mathcal{K}}_N^\top \boldsymbol{b} = \lambda \boldsymbol{\mathcal{K}}^2 \boldsymbol{b}.$$

Since the rank of the $m \times m$ Gram matrix $\boldsymbol{\mathcal{K}}$ (and hence of $\boldsymbol{\mathcal{K}}^2$) is $N < m$, it is singular, so it is not positive definite. Therefore solution of the dual problem with Cholesky decomposition (see, e.g., IDL's function `CHOLDC()`) is not possible.

4

Convolutions, Filters and Fields

Exercise 1

For $f = g * h$ we have

$$\begin{aligned}
f(0) &= \sum_{\ell=0}^{2} h(\ell)g(-\ell) = h(0)g(0) + h(1)g(5) + h(2)g(4) \\
f(1) &= \sum_{\ell=0}^{2} h(\ell)g(1-\ell) = h(0)g(1) + h(1)g(0) + h(2)g(5) \\
&\vdots \\
f(5) &= \sum_{\ell=0}^{2} h(\ell)g(5-\ell) = h(0)g(5) + h(1)g(4) + h(2)g(3).
\end{aligned}$$

Exercise 2

```
% matplotlib inline
import numpy as np
import scipy.signal as signal
from numpy import fft
from osgeo import gdal
from osgeo. gdalconst import GA_ReadOnly
import matplotlib.pyplot as plt
# get an image band
gdal . AllRegister ()
infile = 'imagery/AST_20070501'
inDataset = gdal.Open (infile , GA_ReadOnly )
cols = inDataset.RasterXSize
rows = inDataset.RasterYSize
# pick out the middle row of pixels
band = inDataset . GetRasterBand (3)
G = band.ReadAsArray (0, 0, 200 ,200)
# define a 3x3 FIR kernel
h = np.array ([[1,2,1],[2,4,2],[1,2,1]])/16.0
# convolve in the spatial domain
Gs = signal.convolve2d(h,G)
hp = G*0
```

```
hp [0:3,0:3] = h
# convolve in the frequency domain
Gf = fft.ifft2(fft.fft2(G)*fft.fft2(hp))
plt.imshow(Gs)
plt.show()
plt.imshow(np.real(Gf))
```

Exercise 3

```
def butterworthfilter(d0,n,k,l):
    dst = dist(k,l)
    result = []
    for d in dst:
        result.append( 1./(1+(d/d0)**(2*n)) )
    return np.reshape(np.array(result),(k,l))
```

Exercise 4

The row of pixels $\boldsymbol{f} = (f_0, f_1 \dots f_{c-1})$ defines the function $f_n(x)$ in V_n given by

$$f_n(x) = \sum_{j=0}^{c-1} f_j \phi_{n,j}(x)$$

in terms of the standard basis. The basis functions are normalized and orthogonal:

$$\langle \phi_{n,j}(x), \phi_{n,j'}(x) \rangle = \delta_{j,j'}.$$

Now project $f_n(x)$ onto the residual subspace $V_n^\perp$ spanned by the basis functions

$$\psi_{n-1,k}(x) = \psi(2^{n-1}x - k),$$

where, from Equation (3.34),

$$\psi(2^{n-1}x - k) = \sum_{k'} (-1)^k h_{1-k'} \phi(2^n x - 2k - k'). \tag{4.1}$$

(Due to normalization, the factor $\sqrt{2}$ has been absorbed into the basis functions $\phi_{n,k}$.) We get

$$\begin{aligned} r_{n-1}(x) &= \sum_{k=0}^{c/2-1} \langle f_n(x), \psi_{n-1,k}(x) \rangle \, \psi_{n-1,k}(x) \\ &= \sum_{k=0}^{c/2-1} (G\boldsymbol{f})_k \psi_{n-1,k}(x), \end{aligned}$$

where

$$(G\boldsymbol{f})_k = \langle f_n(x), \psi_{n-1,k}(x)\rangle = \sum_{j=0}^{c-1} f_j \langle \phi(2^n x - j), \psi_{n-1,k}(x)\rangle.$$

Substituting from Equation (4.1) above then gives

$$\begin{aligned}(G\boldsymbol{f})_k &= \sum_{j=0}^{c-1} f_j \sum_{k'} g_{k'} \langle \phi(2^n x - j), \phi(2^n x - 2k - k')\rangle \\ &= \sum_{j=0}^{c-1} f_j \sum_{k'} g_{k'} \delta_{j,2k+k'} \\ &= \sum_{j=0}^{c-1} f_j g_{j-2k},\end{aligned}$$

which is Equation (4.13).

Exercise 5

(a) Consider the inner product of the first row of $\boldsymbol{W}$ with the first column of $\boldsymbol{W}^\top$. With (3.35),

$$h_0^2 + h_1^2 + h_2^2 + h_3^2 = \frac{1}{2}(c_0^2 + c_1^2 + c_2^2 + c_3^2) = \frac{1}{2} \cdot 2 = 1$$

and similarly for the other diagonal elements of the product. The inner product of the first row of $\boldsymbol{W}$ with the second column of $\boldsymbol{W}^\top$ is

$$h_0 h_3 - h_1 h_2 + h_2 h_1 - h_3 h_0 = 0$$

and similarly for all other non-diagonal elements.
(b) You will have to install the discrete wavelet package `PyWavelets` (see its documentation), then run the script

```
%matplotlib inline
import pywt
import numpy as np
import matplotlib.pyplot as plt

a = np.zeros(1024,dtype=np.float64)
a[4] = 1.0
r = pywt.idwt(a,None,'db4')
plt.plot(r[:10])
```

Exercise 7

To see this, let $\gamma = 1/2$ and expand

$$\|\boldsymbol{g} - \boldsymbol{g}'\|^2 = \boldsymbol{g}^\top \boldsymbol{g} + \boldsymbol{g}'^\top \boldsymbol{g}' - 2\boldsymbol{g}^\top \boldsymbol{g}',$$

so that

$$k(\boldsymbol{g}, \boldsymbol{g}') = \exp\left(-\frac{\boldsymbol{g}^\top \boldsymbol{g}}{2}\right) \exp(\boldsymbol{g}^\top \boldsymbol{g}') \exp\left(-\frac{\boldsymbol{g}'^\top \boldsymbol{g}'}{2}\right). \tag{4.2}$$

Now expand the central factor in a Taylor series:

$$\exp(\boldsymbol{g}^\top \boldsymbol{g}') = 1 + \boldsymbol{g}^\top \boldsymbol{g}' + \frac{1}{2}(\boldsymbol{g}^\top \boldsymbol{g}')^2 + \ldots .$$

Then the function in Equation (4.2) is of the form $\boldsymbol{\phi}(\boldsymbol{g})^\top \boldsymbol{\phi}(\boldsymbol{g}')$, where

$$\boldsymbol{\phi}(\boldsymbol{g}) = \exp\left(-\frac{\boldsymbol{g}^\top \boldsymbol{g}}{2}\right) \left(1, g_1 \ldots g_n, \frac{g_1^2}{\sqrt{2}} \ldots \frac{g_n^2}{\sqrt{2}}, \frac{g_1 g_2}{\sqrt{2}} ...\right)^\top .$$

Exercise 8

```python
def kernelMatrix(X,Y=None,gma=None,nscale=10,
                  bias=1,degree=2,kernel=0):
    if Y is None:
        Y = X
    if kernel == 0:     # linear kernel
        X = np.mat(X)
        Y = np.mat(Y)
        return (X*(Y.T),0)
    elif kernel == 1:  # Gaussian kernel
        m = X[:,0].size
        n = Y[:,0].size
        onesm = np.mat(np.ones(m))
        onesn = np.mat(np.ones(n))
        K = np.mat(np.sum(X*X,axis=1)).T*onesn
        K = K + onesm.T*np.mat(np.sum(Y*Y,axis=1))
        K = K - 2*np.mat(X)*np.mat(Y).T
        if gma is None:
            scale = np.sum(np.sqrt(abs(K)))/(m**2-m)
            gma = 1/(2*(nscale*scale)**2)
        return (np.exp(-gma*K),gma)
    else:              # polynomial kernel
        X = np.mat(X)
        Y = np.mat(Y)
        if gma is None:
            scale = np.sum(np.sqrt(abs(K)))/(m**2-m)
            gma = 1/(2*(nscale*scale)**2)
        return ((gma*X*Y.T+bias)**degree,gma)
```

Exercise 9

For $m = 10$ and $N = 5$ a program to do this is

```
pro solution4_9
   G = randomu(seed,5,10)
   K = gausskernel_matrix(G)
   print, 'Average␣eiv␣pre-centering:␣'+$
     strtrim(total(eigenql(K))/10,2)
   print, 'Average␣eiv␣post-centering:␣'+$
     strtrim(total(eigenql(center(K)))/10,2)
   K = polykernel_matrix(G)
   print, 'Average␣eiv␣pre-centering:␣'+$
     strtrim(total(eigenql(K))/10,2)
   print, 'Average␣eiv␣post-centering:␣'+$
     strtrim(total(eigenql(center(K)))/10,2)
end
```

Exercise 10

First note that, since $\boldsymbol{P}^\top \boldsymbol{P} = \boldsymbol{I}_r$, we have

$$\boldsymbol{B}^{1/2}\boldsymbol{B}^{1/2} = \boldsymbol{P}\boldsymbol{\Lambda}^{1/2}\boldsymbol{P}^\top\boldsymbol{P}\boldsymbol{\Lambda}^{1/2}\boldsymbol{P}^\top = \boldsymbol{P}\boldsymbol{\Lambda}^{1/2}\boldsymbol{\Lambda}^{1/2}\boldsymbol{P}^\top = \boldsymbol{P}\boldsymbol{\Lambda}\boldsymbol{P}^\top = \boldsymbol{B}.$$

Moreover $\boldsymbol{B}^{-1/2} = (\boldsymbol{P}\boldsymbol{\Lambda}^{1/2}\boldsymbol{P}^\top)^{-1} = \boldsymbol{P}\boldsymbol{\Lambda}^{-1/2}\boldsymbol{P}^\top$, where $\boldsymbol{\Lambda}^{-1/2}$ is an $r \times r$ diagonal matrix of the inverse square roots of the (nonzero) eigenvalues, so the inverse exists and Equation (4.42) can be written in the form

$$(\boldsymbol{B}^{-1/2}\boldsymbol{A}\boldsymbol{B}^{-1/2})(\boldsymbol{B}^{1/2}\boldsymbol{x}) = \lambda(\boldsymbol{B}^{1/2}\boldsymbol{x}),$$

a symmetric, ordinary eigenvalue problem.

Exercise 11

The answer is no, since the pixels in a clique must be mutual neighbors.

5

Image Enhancement and Correction

Exercise 1

See the code for the function `lin2pcstr()` in the `auxil.auxil1.py` module in the accompanying software.

Exercise 2

The decorrelation stretch can be implemented with a simple modification of the script `pca.py`. After the principal components of three bands have been determined in the variable `pcs`, stretch each component

```
import auxil.auxil1 as auxil
for k in range(bands):
    pcs[:,k] = auxil.histeqstr(pcs[:,k])
```

and then proceed with the reconstruction, choosing all three of the components.

Exercise 3

The power spectrum of ∇_1 is generated by

```
import numpy as np
from numpy import fft
import matplotlib.pyplot as plt
from matplotlib import cm
import auxil.auxil1 as auxil

# create filter
g = np.zeros((512,512),dtype=float)
g[0:2,0:2] = np.array([[1,0],[0,-1]])

#  shift Fourier transform to center
a = np.reshape(range(512**2),(512,512))
i = a % 512
j = a / 512
g = (-1)**(i+j)*g

#  compute power spectrum and display
p = np.abs(fft.fft2(g))**2
```

```
plt.imshow(auxil.linstr(p), cmap=cm.jet)
```

Exercise 5

The transformation generates the new coordinates

$$x' = \alpha x, \quad y' = \alpha y.$$

The centralized moment transforms to

$$\mu'_{pq} = \int\int (x' - x'_c)^p (y' - y'_c)^q f'(x', y') dx' dy'.$$

But necessarily $f'(x', y') = f(x, y)$, so we get

$$\mu'_{pq} = \int\int \alpha^p (x - x_c)^p \alpha^q (y - y_c)^q f(x, y) d(\alpha x) d(\alpha y) = \alpha^{p+q+2} \mu_{pq}.$$

Therefore η_{pq} is scale invariant:

$$\eta'_{pq} = \frac{\mu'_{pq}}{{\mu'_{00}}^{(p+q)/2+1}} = \frac{\alpha^{p+q+2}\mu_{pq}}{(\alpha^2)^{(p+q)/2+1}\mu_{00}^{(p+q)/2+1}} = \frac{\mu_{pq}}{\mu_{00}^{(p+q)/2+1}} = \eta_{pq}.$$

Exercise 6

The noise reduction program is given in Listing **??**. Pre- and post-reduction noise covariance matrices for the last three MNF bands of a LANDSAT TM image are given below.

```
Noise covariance matrix, file [Memory6] (600x600x3)
       1.0915236    -0.0063084596    -0.0029047414
   -0.0063084596        1.0775507    -0.0031050553
   -0.0029047414    -0.0031050553        1.0520645

Noise covariance matrix, file [Memory7] (600x600x3)
      0.28040335     -0.021618183    -0.0067004804
    -0.021618183       0.22010893    -0.0018719031
   -0.0067004804    -0.0018719031       0.17734220
```

Exercise 7

The renormalized panchromatic wavelet coefficients are of the form

$$aC + b, \quad b = m_{ms} - a m_{pan}, \; a = \sigma_{ms}/\sigma_{pan}.$$

Therefore,

$$\langle aC + b\rangle = a\langle C\rangle + b = a m_{pan} + m_{ms} - a m_{pan} = m_{ms}$$

and

$$\mathrm{var}(aC + b) = a^2 \mathrm{var}(C) = \frac{\sigma^2_{ms}}{\sigma^2_{pan}} \sigma^2_{pan} = \sigma^2_{ms}.$$

Exercise 9

(a) We have $p_u(u) = e^{-u/2}/2$ and g $= xu/2$. Hence

$$\left|\frac{du}{dg}\right| = \frac{2}{x}$$

and, with Theorem 2.1,

$$p_g(g) = \frac{1}{2}e^{-u/2} \cdot \left|\frac{du}{dg}\right| = \frac{1}{2}e^{-g/x} \cdot \frac{2}{x} = \frac{1}{x}e^{-g/x}, \quad g \geq 0.$$

(b) We have

$$p_g(g) = \frac{1}{(x/m)^m \Gamma(m)} g^{m-1} e^{-gm/x}$$

and $v = g/x$. Hence

$$\left|\frac{dg}{dv}\right| = x.$$

With Theorem (2.1),

$$p_v(v) = \frac{1}{(x/m)^m \Gamma(m)} x^{m-1} v^{m-1} e^{-mv} \cdot x = \frac{m^m}{\Gamma(m)} v^{m-1} e^{-vm}.$$

Exercise 10

(a) For single polarization the ENL estimate reduces to

$$\frac{\langle \bar{G}\rangle^2}{\langle \bar{G}^2\rangle - \langle \bar{G}\rangle^2} = \frac{\langle \bar{G}\rangle^2}{\mathrm{var}(\bar{G})}.$$

(b) ¡to be done¿

(c) The following Python script will simulate an m-look quad polSAR image:

```
def make_simimage(fn,m=5,sigma=1,alpha=0.2,beta=0.2):
    simimage = np.zeros((100**2,9))
    ReSigma = np.zeros((3,3))
    ImSigma = np.zeros((3,3))
    for i in range(3):
        for j in range(3):
            if i==j:
                ReSigma[i,j]=sigma**2
            elif i<j:
                ReSigma[i,j] = alpha*sigma**2
                ImSigma[i,j] = beta*sigma**2
            else:
                ReSigma[i,j] = alpha*sigma**2
                ImSigma[i,j] = -beta*sigma**2
    Sigma = np.mat(ReSigma +1j*ImSigma)
```

```
C = np.linalg.cholesky(Sigma)
for i in range(100**2):
    X = np.mat(np.random.randn(m,3))
    Y = np.mat(np.random.randn(m,3))
    Wr = X.T*X + Y.T*Y
    Wi = X.T*Y - Y.T*X
    W = (Wr - 1j*Wi)/2
    W = C*W*C.H
    simimage[i,0] = np.real(W[0,0])
    simimage[i,1] = np.real(W[0,1])
    simimage[i,2] = np.imag(W[0,1])
    simimage[i,3] = np.real(W[0,2])
    simimage[i,4] = np.imag(W[0,2])
    simimage[i,5] = np.real(W[1,1])
    simimage[i,6] = np.real(W[1,2])
    simimage[i,7] = np.imag(W[1,2])
    simimage[i,8] = np.real(W[2,2])
driver = gdal.GetDriverByName('GTiff')
outDataset = driver.Create(fn,100,100,9,GDT_Float32)
for i in range(9):
    outband = outDataset.GetRasterBand(i+1)
    outband.WriteArray(np.reshape(simimage[:,i],
                                  (100,100)),0,0)
    outband.FlushCache()
outDataset = None
```

Exercise 12

From the definition of gradient, we get

$$\nabla g(x_1,x_2) = \boldsymbol{i}\frac{\partial g(x_1,x_2)}{\partial x_1} + \boldsymbol{j}\frac{\partial g(x_1,x_2)}{\partial x_2}.$$

Therefore

$$\begin{aligned}\|\nabla g(x_1,x_2)\| &= \sqrt{\left(\frac{\partial g(x_1,x_2)}{\partial x_1}\right)^2 + \left(\frac{\partial g(x_1,x_2)}{\partial x_1 2}\right)^2} \\ &\approx \sqrt{\left(\frac{\Delta x}{2}\right)^2 + \left(\frac{\Delta y}{2}\right)^2}.\end{aligned}$$

6

Supervised Classification Part 1

Exercise 1

(a) For the one-dimensional case we have

$$\begin{aligned}\int \sqrt{p(g \mid 1)p(g \mid 2)} &= \frac{1}{\sqrt{2\pi\sigma_1\sigma_2}} \int \exp\left(-\frac{1}{4\sigma_1^2}(g-\mu_1)^2 - \frac{1}{4\sigma_2^2}(g-\mu_2)^2\right) dg \\ &= \frac{1}{\sqrt{2\pi\sigma_1\sigma_2}} \exp\left(-\frac{1}{4}\left(\frac{\mu_1^2}{\sigma_1^2}+\frac{\mu_2^2}{\sigma_2^2}\right)\right) \int \exp(ag - bg^2)dg,\end{aligned}$$

where

$$a = \frac{1}{2}\left(\frac{\mu_1}{\sigma_1^2}+\frac{\mu_2}{\sigma_2^2}\right), \quad b = \frac{1}{4}\left(\frac{1}{\sigma_1^2}+\frac{1}{\sigma_2^2}\right).$$

Therefore, using the definite integral given in the hint,

$$\begin{aligned}\int \sqrt{p(g \mid 1)p(g \mid 2)} &= \frac{1}{\sqrt{2\pi\sigma_1\sigma_2}} \exp\left(-\frac{1}{4}\left(\frac{\mu_1^2}{\sigma_1^2}+\frac{\mu_2^2}{\sigma_2^2}\right)\right) \sqrt{\frac{\pi}{b}} \exp(a^2/4b) \\ &= \frac{1}{\sqrt{2\sigma_1\sigma_2}} \sqrt{\frac{1}{\frac{1}{4}\left(\frac{1}{\sigma_1^2}+\frac{1}{\sigma_2^2}\right)}} \exp\left(-\frac{1}{4}\left(\frac{\mu_1^2}{\sigma_1^2}+\frac{\mu_2^2}{\sigma_2^2}\right)\right) \exp(a^2/4b) \\ &= \left(\frac{\sigma_1\sigma_2}{2}\left(\frac{1}{\sigma_1^2}+\frac{1}{\sigma_2^2}\right)\right)^{-1/2} \exp\left(-\frac{1}{4}\left(\frac{\mu_1^2}{\sigma_1^2}+\frac{\mu_2^2}{\sigma_2^2}\right)\right) \exp(a^2/4b).\end{aligned}$$

The first factor may be written in the form

$$\exp\left(-\frac{1}{2}\log\left(\frac{(\sigma_1^2+\sigma_2^2)/2}{\sigma_1\sigma_2}\right)\right).$$

The exponents in the second two terms can be combined to give

$$-\frac{1}{4}\left(\frac{\mu_1^2}{\sigma_1^2}+\frac{\mu_2^2}{\sigma_2^2}\right) + \frac{1}{\left(\frac{1}{\sigma_1^2}+\frac{1}{\sigma_2^2}\right)} \frac{1}{4b}\left(\frac{\mu_1}{\sigma_1^2}+\frac{\mu_2}{\sigma_2^2}\right),$$

which simplifies straightforwardly to

$$-\frac{1}{8}(\mu_1-\mu_2)^2 \frac{2}{\sigma_1^2+\sigma_2^2}.$$

Combining, we get the one-dimensional Bhattacharyya distance

$$B = \frac{1}{8}(\mu_1 - \mu_2)^2 \frac{2}{\sigma_1^2 + \sigma_2^2} + \frac{1}{2}\log\left(\frac{(\sigma_1^2 + \sigma_2^2)/2}{\sigma_1\sigma_2}\right).$$

(b) For the multivariate case, we had

$$\int_{-\infty}^{\infty} \sqrt{p(\boldsymbol{g} \mid 1)p(\boldsymbol{g} \mid 2)}d\boldsymbol{g} = e^{-B}.$$

Therefore, the Jeffries-Matusita distance is

$$\begin{aligned} J &= \int_{-\infty}^{\infty} \left(p(\boldsymbol{g} \mid 1)^{1/2} - p(\boldsymbol{g} \mid 2)^{1/2}\right)^2 d\boldsymbol{g} \\ &= \int_{-\infty}^{\infty} \left(p(\boldsymbol{g} \mid 1) + p(\boldsymbol{g} \mid 2) - 2\sqrt{p(\boldsymbol{g} \mid 1)p(\boldsymbol{g} \mid 2)}\right) d\boldsymbol{g} \\ &= 2 - 2e^{-B}. \end{aligned}$$

(c) This is a satisfactory separability measure since d_{12} vanishes when the distributions are identical and is positive otherwise. Moreover it is symmetric, i.e., $d_{12} = d_{21}$.

(d) From the definition of the Kullback-Leibler divergence we have

$$\begin{aligned} \mathrm{KL}(p(g \mid 1), p(g \mid 2)) &= \int p(g \mid 1)\log\frac{p(g \mid 2)}{p(g \mid 1)}dg \\ &= \int p(g \mid 1)\left(\log\frac{\sigma_1}{\sigma_2} - \frac{1}{2\sigma_2^2}(g-\mu_2)^2 + \frac{1}{2\sigma_1^2}(g-\mu_1)^2\right)dg \\ &= \log\frac{\sigma_1}{\sigma_2} - \frac{1}{2\sigma_2^2}\int p(g \mid 1)(g-\mu_2)^2 dg + \frac{1}{2}. \end{aligned}$$

The remaining integral can be expanded as follows:

$$\begin{aligned} \int p(g \mid 1)(g-\mu_2)^2 dg &= \int p(g \mid 1)(g - \mu_1 + \mu_1 - \mu_2)^2 dg \\ &= \int p(g \mid 1)\big((g-\mu_1)^2 + 2(g-\mu_1)(\mu_1-\mu_2) + (\mu_1-\mu_2)^2\big)dg \\ &= \sigma_1^2 + 2(\mu_1-\mu_2)\int p(g \mid 1)(g-\mu_1)dg + (\mu_1-\mu_2)^2 \\ &= \sigma_1^2 + (\mu_1-\mu_2)^2. \end{aligned}$$

This gives

$$\mathrm{KL}(p(g \mid 1), p(g \mid 2)) = \log\frac{\sigma_1}{\sigma_2} + \frac{\sigma_2^2 - \sigma_1^2}{2\sigma_2^2} + \frac{(\mu_1-\mu_2)^2}{2\sigma_1^2},$$

and, from symmetry,

$$\mathrm{KL}(p(g \mid 2), p(g \mid 1)) = \log\frac{\sigma_2}{\sigma_1} + \frac{\sigma_1^2 - \sigma_2^2}{2\sigma_1^2} + \frac{(\mu_1-\mu_2)^2}{2\sigma_2^2}.$$

Adding the two expressions gives the result.

Exercise 2

(a) This follows directly by expanding Equation (6.17) and dropping terms that do not involve the class label k.

(b) The classifier will choose $k = 1$ when

$$\log(\Pr(1)) - \boldsymbol{\mu}_1^\top \boldsymbol{\Sigma}^{-1}\boldsymbol{g} + \frac{1}{2}\boldsymbol{\mu}_1^\top \boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_1 > \log(\Pr(2)) - \boldsymbol{\mu}_2^\top \boldsymbol{\Sigma}^{-1}\boldsymbol{g} + \frac{1}{2}\boldsymbol{\mu}_2^\top \boldsymbol{\Sigma}^{-2}\boldsymbol{\mu}_2,$$

which simplifies to

$$h = (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^\top \boldsymbol{\Sigma}^{-1}\left(\boldsymbol{g} - \frac{\boldsymbol{\mu}_1 + \boldsymbol{\mu}_2}{2}\right) > \log\left(\frac{\Pr(2)}{\Pr(1)}\right).$$

(c) The random variable H is

$$H = (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^\top \boldsymbol{\Sigma}^{-1}\left(\boldsymbol{G} - \frac{\boldsymbol{\mu}_1 + \boldsymbol{\mu}_2}{2}\right) = \boldsymbol{a}^\top \boldsymbol{G} - \boldsymbol{b},$$

where

$$\boldsymbol{a}^\top = (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^\top \boldsymbol{\Sigma}^{-1}, \quad \boldsymbol{b} = \frac{1}{2}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^\top \boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}_1 + \boldsymbol{\mu}_2).$$

If the observation in in class 1, then with $H = H_1$,

$$\langle H_1 \rangle = \boldsymbol{a}^\top \boldsymbol{\mu}_1 - \boldsymbol{b} = (\mu_1 - \boldsymbol{\mu}_2)^\top \boldsymbol{\Sigma}^{-1}\frac{(\mu_1 - \boldsymbol{\mu}_2)}{2} = \frac{1}{2}d^2.$$

Similarly, if the observation is in class 2,

$$\langle H_2 \rangle = -\frac{1}{2}d^2$$

In either case, the variance is

$$\begin{aligned} \mathrm{var}(H_1) = \mathrm{var}(H_2) = \boldsymbol{a}^\top \boldsymbol{\Sigma}\boldsymbol{a} &= (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^\top \boldsymbol{\Sigma}^{-1}\boldsymbol{\Sigma}\left((\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^\top \boldsymbol{\Sigma}^{-1}\right)^\top \\ &= (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^\top \boldsymbol{\Sigma}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2) = d^2. \end{aligned}$$

Thus $H_1 \sim \mathcal{N}(d^2/2, d^2)$ and $H_2 \sim \mathcal{N}(-d^2/2, d^2)$.

(d) The probability of misclassification is given by the weighted sum

$$\Pr(1)\cdot\Pr\left(H_1 < \log\left(\frac{\Pr(2)}{\Pr(1)}\right)\right) + \Pr(2)\cdot\Pr\left(H_2 > \log\left(\frac{\Pr(2)}{\Pr(1)}\right)\right).$$

For the first term we can standardize H_1 and write

$$\begin{aligned} \Pr\left(H_1 < \log\left(\frac{\Pr(2)}{\Pr(1)}\right)\right) &= \Pr\left(\frac{H_1 - d^2/2}{d} < \frac{\log\left(\frac{\Pr(2)}{\Pr(1)}\right) - d^2/2}{d}\right) \\ &= \Phi\left(-\frac{1}{2}d + \frac{1}{d}\log\left(\frac{\Pr(2)}{\Pr(1)}\right)\right) \end{aligned}$$

and similarly, for H_2,

$$\Pr\left(H_2 > \log\left(\frac{\Pr(2)}{\Pr(1)}\right)\right) = \Pr\left(\frac{H_2 + d^2/2}{d} > \frac{\log\left(\frac{\Pr(2)}{\Pr(1)}\right) + d^2/2}{d}\right)$$
$$= 1 - \Phi\left(\frac{1}{2}d + \frac{1}{d}\log\left(\frac{\Pr(2)}{\Pr(1)}\right)\right)$$
$$= \Phi\left(-\frac{1}{2}d - \frac{1}{d}\log\left(\frac{\Pr(2)}{\Pr(1)}\right)\right).$$

(e) With $a = \Pr(1) = 1 - \Pr(2)$ the misclassification probability is

$$a\Phi\left(-\frac{d}{2} + \frac{1}{d}\log\left(\frac{a}{1-a}\right)\right) + (1-a)\Phi\left(-\frac{d}{2} - \frac{1}{d}\log\left(\frac{a}{1-a}\right)\right).$$

The first term increases monotonically from 0 to 1 and the second term decreases monotonically from 1 to 0 as a increases from 0 to 1. Therefore the minimum in their sum must occur at their intersection, when both terms are equal. This is at $a = 1/2$. Hence the minimum misclassification probability is $\Phi\left(-\frac{d}{2}\right)$.

Exercise 3

(a) Let $\boldsymbol{g}_0$ and $\boldsymbol{g}$ be any two points in the hyperplane defined by $\boldsymbol{w}$ and w_0. Then $\boldsymbol{w}^\top(\boldsymbol{g} - \boldsymbol{g}_0) = 0$, so that $\boldsymbol{w}$ is orthogonal to any vector $\boldsymbol{g} - \boldsymbol{g}_0$ which lies in the hyperplane.
(b) Since the 3 points are not colinear, a straight line (oriented hyperplane in two dimensions) can always be found which separates any one from the other two. This is not possible for 4 non-colinear points in two dimensions, for example the four corners of a square with two diagonally opposite points belonging to class 1 and the other two to class 2. Therefore the VC dimension for the hyperplane classifier in two dimensions is 3.
(c) See Listing **??** and Figure **??**.

Exercise 4

Taking the logarithm of the likelihood function, we get

$$\sum_\nu \log p(\boldsymbol{\ell}(\nu) \mid \boldsymbol{g}(\nu)) - \sum_\nu \log p(\boldsymbol{g}(\nu).$$

The first term involves the posterior probability density, which is modelled by the network and hence depends on the synaptic weights. The second term doesn't involve the network weights and so can be ignored. Maximizing the likelihood is thus equivalent to minimizing the cost function

$$E = -\sum_\nu \log p(\boldsymbol{\ell}(\nu) \mid \boldsymbol{g}(\nu)).$$

Since the components of $\boldsymbol{\ell}(\nu)$ are statistically independent, we can write this as

$$E = -\sum_{\nu}\sum_{k} \log p(\ell_k(\nu) \mid \boldsymbol{g}(\nu)).$$

Now substituting from Equation (6.80) gives

$$E = \frac{1}{2\sigma^2}\sum_{\nu}\sum_{k}(m_k(\nu) - \ell_k(\nu))^2 + nK\log\sigma + \frac{nK}{2}\log(2\pi),$$

where we have replaced the unknown functions h_k with the network outputs m_k which model them. Dropping the terms which don't depend on the synaptic weights and the factor $1/\sigma^2$,

$$E = \frac{1}{2}\sum_{\nu}\sum_{k}(m_k(\nu) - \ell_k(\nu))^2 = \frac{1}{2}\sum_{\nu}\|\boldsymbol{m}(\nu) - n\ell(\nu)\|^2,$$

which is the quadratic cost function.

Exercise 6

The third step in the backpropagation algorithm is

$$w^o_{jk}(\nu+1) \to w^o_{jk}(\nu) - \eta\frac{\partial E(\nu)}{\partial w^o_{jk}},$$

where, with (6.36),

$$\frac{\partial E(\nu)}{\partial w^o_{jk}} = -\delta^o_k(\nu)n_j(\nu).$$

Therefore

$$w^o_{jk}(\nu+1) \to w^o_{jk}(\nu) + \eta\delta_k n_j(\nu) = w^o_{jk}(\nu) + \eta\left(\boldsymbol{n}(\nu)\boldsymbol{\delta}^o(\nu)^\top\right)_{jk}.$$

Exercise 7

(a) This follows easily from $f(x) = 1/(1+e^{-x})$.
(b) Consider the jth neuron in the hidden layer of Figure 6.10 in the text. Its output signal can be written as

$$n_j(\boldsymbol{g}) = \frac{1}{2}\tanh\left(\frac{\boldsymbol{w}_j^{h\top}\boldsymbol{g}}{2}\right) + \frac{1}{2}$$

Its contribution to the activation of the kth output neuron is therefore

$$n_j(\boldsymbol{g})w^o_{kj} = \left(\frac{1}{2}\tanh\left(\frac{\boldsymbol{w}_j^{h\top}\boldsymbol{g}}{2}\right) + \frac{1}{2}\right)w^o_{kj}.$$

If we reverse the sign of the hidden weight vector $\boldsymbol{w}_j^{h\top}$ and also that of the output weight w_{kj}^o, then this contribution becomes

$$n_j(\boldsymbol{g})w_{kj}^o = \left(\frac{1}{2}\tanh\left(\frac{\boldsymbol{w}_j^{h\top}\boldsymbol{g}}{2}\right) - \frac{1}{2}\right)w_{kj}^o.$$

The difference between the two is just w_{kj}^o. This difference can be exactly compensated by the bias weight w_{k0}^o so that the network output is unchanged. There are thus two equivalent configurations of the network for each of the L hidden neurons, or 2^L in all. It follows that every minimum in the cost function corresponds to at least 2^L different synaptic weight combinations.

Exercise 8

With Δ independent of ν, for the first n updates,

$$\begin{aligned}
&\Delta_1 \to \Delta + \alpha\Delta \\
&\Delta_2 \to \Delta + \alpha(\Delta + \alpha\Delta) = \Delta(1 + \alpha + \alpha^2) \\
&\quad\vdots \\
&\Delta_n \to \Delta(1 + \alpha + \alpha^2 + \ldots + \alpha^{n-1}) = \Delta(1 - \alpha^n)/(1 - \alpha)
\end{aligned}$$

Thus for $n \to \infty$ and $\alpha < 1$

$$\Delta \to \frac{1}{1-\alpha}\Delta.$$

Exercise 9

Combining Equations (6.73) and (6.76) gives

$$\sum_{j\neq i}\frac{\Pr(\ell = i \mid \boldsymbol{g})}{\mu_{ij}} = (K-2)\Pr(\ell = i \mid \boldsymbol{g}) + 1.$$

Solving for $\Pr(\ell = i \mid \boldsymbol{g})$ gives Equation (6.77).

7

Supervised Classification Part 2

Exercise 1

From Equation (7.13) with $Y = y$ the limits on θ occur when

$$\frac{y/n - \theta}{\sigma} = \pm s,$$

where from Equation (7.11)

$$\sigma = \sqrt{\frac{\theta(1-\theta)}{n}}.$$

Therefore the limits for θ are the solutions of

$$(y - n\theta)^2 = s^2 n\theta(1-\theta),$$

a quadratic equation for θ. The solutions are

$$\theta = \frac{y + s^2/2 \pm \sqrt{s^2/4 + y(n-y)/n}}{s^2 + n}$$

which gives (7.14) for $s = 1.960$.

Exercise 4

The log discriminant function for $\boldsymbol{x} = m\bar{\mathbf{c}}$, where $\bar{\mathbf{c}}$ is a multi-look polarimetric SAR image pixel in complex covariance matrix format, is

$$\begin{aligned} d_k(\boldsymbol{x}) &= \log(Pr(k) + \log(p_{W_c}(\boldsymbol{x}|\boldsymbol{\Sigma}_k)) \\ &= \log(Pr(k)) - m\log|\boldsymbol{\Sigma}_k| - \mathrm{tr}(\boldsymbol{\Sigma}_k^{-1}\boldsymbol{x}) + \text{ terms independent of } k. \end{aligned}$$

In terms of $\bar{\mathbf{c}}$, therefore,

$$\begin{aligned} d_k(\bar{\mathbf{c}}) &= \log(Pr(k)) - m\log|\boldsymbol{\Sigma}_k| - \mathrm{tr}(\boldsymbol{\Sigma}_k^{-1} m\bar{\mathbf{c}}) \\ &= \log(Pr(k)) - m\left[\log|\boldsymbol{\Sigma}_k| + \mathrm{tr}(\boldsymbol{\Sigma}_k^{-1}\bar{\mathbf{c}})\right], \end{aligned}$$

which is Equation (7.34).

Exercise 5

We have

$$L = (\boldsymbol{G} - \boldsymbol{M}\boldsymbol{\alpha})^\top \boldsymbol{\Sigma}_R^{-1}(\boldsymbol{G} - \boldsymbol{M}\boldsymbol{\alpha}) + 2\lambda(\boldsymbol{\alpha}^\top \boldsymbol{1}_K - 1).$$

Expanding,

$$L = \boldsymbol{G}^\top \boldsymbol{\Sigma}_R^{-1}\boldsymbol{G} + \boldsymbol{\alpha}^\top \boldsymbol{M}^\top \boldsymbol{\Sigma}_R^{-1}\boldsymbol{M}\boldsymbol{\alpha} - 2\boldsymbol{\alpha}^\top \boldsymbol{M}^\top \boldsymbol{\Sigma}_R^{-1}\boldsymbol{G} + 2\lambda(\boldsymbol{\alpha}^\top \boldsymbol{1}_K - 1)$$

and applying the rules for vector differentiation,

$$\frac{\partial L}{\partial \boldsymbol{\alpha}} = 2\boldsymbol{M}^\top \boldsymbol{\Sigma}_R^{-1}\boldsymbol{M}\boldsymbol{\alpha} - 2\boldsymbol{M}^\top \boldsymbol{\Sigma}_R^{-1}\boldsymbol{G} + 2\lambda \boldsymbol{1}_k = 0.$$

Therefore,

$$\boldsymbol{\alpha} = (\boldsymbol{M}^\top \boldsymbol{\Sigma}_R^{-1}\boldsymbol{M})^{-1}\left(\boldsymbol{M}^\top \boldsymbol{\Sigma}_R^{-1}\boldsymbol{G} - \lambda \boldsymbol{1}_K\right).$$

Together with the constraint $\boldsymbol{\alpha}^\top \boldsymbol{1}_K = 1$, the substitution $\boldsymbol{G} \to \boldsymbol{g}$ gives the result.

Exercise 6

(a) The matrix

$$\boldsymbol{P} = \boldsymbol{I} - \boldsymbol{U}(\boldsymbol{U}^\top \boldsymbol{U})^{-1}\boldsymbol{U}^\top$$

"projects out" the undesirable components, since

$$\boldsymbol{P}\boldsymbol{G} = \boldsymbol{P}\boldsymbol{d}\alpha_d + \boldsymbol{I}\boldsymbol{U}\boldsymbol{\beta} - \boldsymbol{U}\boldsymbol{\beta} + \boldsymbol{P}\boldsymbol{R} = \boldsymbol{P}\boldsymbol{d}\alpha_d + \boldsymbol{P}\boldsymbol{R}.$$

(b) A suitable function for OSP is

```
def osp(G, U):

    N = G.shape[1]
    P = np.eye(N) - U*(U.T*U).I*U.T
    return P*G.T
```

(c) Assuming the mean to be close to zero, we can write

$$\langle(\boldsymbol{w}^\top \boldsymbol{G})^2\rangle \approx \langle(\boldsymbol{w}^\top \boldsymbol{G} - 0)^2\rangle = \mathrm{var}(\boldsymbol{w}^\top \boldsymbol{G}).$$

Therefore we must minimize $\mathrm{var}(\boldsymbol{w}^\top \boldsymbol{G}) = \boldsymbol{w}^\top \boldsymbol{\Sigma}_R \boldsymbol{w}$ under the constraint $\boldsymbol{w}^\top \boldsymbol{d} = 1$. The corresponding Lagrange function is

$$L = \boldsymbol{w}^\top \boldsymbol{\Sigma}_R \boldsymbol{w} - 2\lambda(\boldsymbol{w}^\top \boldsymbol{d} - 1).$$

We have

$$\frac{\partial L}{\partial \boldsymbol{w}} = 2\boldsymbol{\Sigma}_R \boldsymbol{w} - 2\lambda \boldsymbol{d} = 0,$$

so that

$$\boldsymbol{w} = \lambda \boldsymbol{\Sigma}_R^{-1} \boldsymbol{d}.$$

The constraint requires

$$\lambda (\boldsymbol{\Sigma}_R^{-1} \boldsymbol{d})^\top \boldsymbol{d} = 1$$

or

$$\lambda = \frac{1}{\boldsymbol{d}^\top \boldsymbol{\Sigma}_R^{-1} \boldsymbol{d}}$$

and therefore

$$\boldsymbol{w} = \frac{\boldsymbol{\Sigma}_R^{-1} \boldsymbol{d}}{\boldsymbol{d}^\top \boldsymbol{\Sigma}_R^{-1} \boldsymbol{d}}. \tag{7.1}$$

(d) The SAM angle θ is given by

$$\theta = \cos^{-1} \left(\frac{\boldsymbol{g}^\top \boldsymbol{d}}{\|\boldsymbol{g}\| \|\boldsymbol{d}\|} \right)$$

so we project $\boldsymbol{g}$ along $\boldsymbol{d}$. In CEM with $\boldsymbol{\Sigma}_R = \sigma^2 \boldsymbol{I}$ we do precisely the same thing, i.e., we project $\boldsymbol{g}$ along $\boldsymbol{w}$ given by, see Equation (7.1) above,

$$\boldsymbol{w} = \boldsymbol{d} / \|\boldsymbol{d}\|^2.$$

Exercise 7

With Equation (7.38) we can write

$$\begin{aligned}
\boldsymbol{\mu}_\phi^\top \tilde{\boldsymbol{\Phi}}^\top &= \frac{1}{m} \sum_\nu \boldsymbol{\phi}^\top(\nu) \left[(\boldsymbol{\phi}(1) \dots \boldsymbol{\phi}(m)) - \frac{1}{m} \sum_{\nu'} \boldsymbol{\phi}(\nu') \right] \\
&= \frac{1}{m} \sum_\nu \Big(\boldsymbol{\phi}^\top(\nu) \boldsymbol{\phi}(1) \dots \boldsymbol{\phi}^\top(\nu) \boldsymbol{\phi}(m) \Big) - \frac{1}{m^2} \sum_{\nu\nu'} \boldsymbol{\phi}^\top(\nu) \boldsymbol{\phi}(\nu').
\end{aligned}$$

Replacing inner products $\boldsymbol{\phi}^\top(\nu)\boldsymbol{\phi}(\nu')$ by the kernels $k(\boldsymbol{g}(\nu), \boldsymbol{g}(\nu'))$ gives Equation (7.52).

8

Unsupervised Classification

Exercise 1

(a) Consider only the kth term in the cost function Equation (8.12) and write $n = n_k$ and $u_i = u_{ki}$. The term is then

$$\sum_i u_i \left(\boldsymbol{g}_i - \frac{1}{n} \sum_{i'} u_{i'} \boldsymbol{g}_{i'} \right)^\top \left(\boldsymbol{g}_i - \frac{1}{n} \sum_{i''} u_{i''} \boldsymbol{g}_{i''} \right).$$

Expanding, we get

$$\sum_i u_i \|\boldsymbol{g}_i\|^2 - \frac{2}{n} \sum_{i,i'} u_i u_{i'} \boldsymbol{g}_i^\top \boldsymbol{g}_{i'} + \frac{1}{n} \sum_{i',i''} u_{i'} u_{i''} \boldsymbol{g}_{i'}^\top \boldsymbol{g}_{i''},$$

which simplifies to

$$\sum_i u_i \|\boldsymbol{g}_i\|^2 - \frac{1}{n} \sum_{i,i'} u_i u_{i'} \boldsymbol{g}_i^\top \boldsymbol{g}_{i'}.$$

This can be rewritten in the equivalent form

$$\begin{aligned}
&\frac{1}{2n} \left(n \sum_i \|\boldsymbol{g}_i\|^2 + n \sum_{i'} \|\boldsymbol{g}_{i'}\|^2 - 2 \sum_{i,i'} u_i u_{i'} \boldsymbol{g}_i^\top \boldsymbol{g}_{i'} \right) \\
&\quad = \frac{1}{2n} \left(\sum_{i,i'} u_i u_{i'} \|\boldsymbol{g}_i\|^2 + \sum_{i,i'} u_i u_{i'} \|\boldsymbol{g}_i\|^2 - 2 \sum_{i,i'} u_i u_{i'} \boldsymbol{g}_i^\top \boldsymbol{g}_{i'} \right) \\
&\quad = \frac{1}{2n} \sum_{i,i'} u_i u_{i'} \|\boldsymbol{g}_i - \boldsymbol{g}_{i'}\|^2.
\end{aligned}$$

Reintroducing the index k and summing over it establishes Equation (8.62).
(b) The expression

$$\boldsymbol{D} = \boldsymbol{\mathcal{G}}^\top - \boldsymbol{\mathcal{G}}^\top \boldsymbol{U}^\top \boldsymbol{M} \boldsymbol{U} = \boldsymbol{\mathcal{G}}^\top (\boldsymbol{I}_m - \boldsymbol{U}^\top \boldsymbol{M} \boldsymbol{U})$$

is a $N \times m$ matrix whose νth column is the distance of observation $\boldsymbol{g}(\nu)$ from its associated mean. Therefore the diagonal elements of $\boldsymbol{D}^\top \boldsymbol{D}$ are the squares of those distances, and the sum of squares is its trace:

$$E(C) = \mathrm{tr}(\boldsymbol{D}^\top \boldsymbol{D}) = \mathrm{tr} \Big((\boldsymbol{I}_m - \boldsymbol{U}^\top \boldsymbol{M} \boldsymbol{U}) \boldsymbol{\mathcal{G}} \boldsymbol{\mathcal{G}}^\top (\boldsymbol{I}_m - \boldsymbol{U}^\top \boldsymbol{M} \boldsymbol{U}) \Big)$$

or, expanding and using the identity $\mathrm{tr}(A) = \mathrm{tr}(A^\top)$,

$$E(C) = \mathrm{tr}\left(\boldsymbol{\mathcal{G}}\boldsymbol{\mathcal{G}}^\top - 2\boldsymbol{U}^\top\boldsymbol{M}\boldsymbol{U}\boldsymbol{\mathcal{G}}\boldsymbol{\mathcal{G}}^\top + U^\top\boldsymbol{M}\boldsymbol{U}\boldsymbol{\mathcal{G}}\boldsymbol{\mathcal{G}}^\top U^\top\boldsymbol{M}\boldsymbol{U}\right).$$

Now, with the identity $\mathrm{tr}(AB) = \mathrm{tr}(BA)$, we can write

$$E(C) = \mathrm{tr}\left(\boldsymbol{\mathcal{G}}\boldsymbol{\mathcal{G}}^\top - 2\boldsymbol{U}^\top\boldsymbol{M}\boldsymbol{U}\boldsymbol{\mathcal{G}}\boldsymbol{\mathcal{G}}^\top + \boldsymbol{U}^\top\boldsymbol{M}\boldsymbol{U}\boldsymbol{U}^\top\boldsymbol{M}\boldsymbol{U}\boldsymbol{\mathcal{G}}\boldsymbol{\mathcal{G}}^\top\right).$$

But

$$\boldsymbol{U}\boldsymbol{U}^\top\boldsymbol{M} = \begin{pmatrix} m_1 & 0 & \dots & 0 \\ 0 & m_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & m_K \end{pmatrix}\begin{pmatrix} 1/m_1 & 0 & \dots & 0 \\ 0 & 1/m_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1/m_K \end{pmatrix} = \boldsymbol{I}_k$$

so we have finally

$$\begin{aligned} E(C) &= \mathrm{tr}\left(\boldsymbol{\mathcal{G}}\boldsymbol{\mathcal{G}}^\top - 2\boldsymbol{U}^\top\boldsymbol{M}\boldsymbol{U}\boldsymbol{\mathcal{G}}\boldsymbol{\mathcal{G}}^\top + \boldsymbol{U}^\top\boldsymbol{M}\boldsymbol{U}\boldsymbol{\mathcal{G}}\boldsymbol{\mathcal{G}}^\top\right) \\ &= \mathrm{tr}\left(\boldsymbol{\mathcal{G}}\boldsymbol{\mathcal{G}}^\top - \boldsymbol{U}^\top\boldsymbol{M}\boldsymbol{U}\boldsymbol{\mathcal{G}}\boldsymbol{\mathcal{G}}^\top\right). \end{aligned}$$

Exercise 2

(b) This is a matter of experimentation. A good discrimination can be obtained with an `NSCALE` value of around 0.3.

Exercise 3

(a) The symmetry of $\boldsymbol{L}$ follows from the symmetry of $\boldsymbol{D}$ and $\boldsymbol{\mathcal{K}}$. For any m-dimensional vector $\boldsymbol{x}$,

$$\begin{aligned} \boldsymbol{x}^\top\boldsymbol{L}\boldsymbol{x} &= \boldsymbol{x}^\top\boldsymbol{D}\boldsymbol{x} - \boldsymbol{x}^\top\boldsymbol{\mathcal{K}}\boldsymbol{x} \\ &= \sum_i d_i x_i^2 - \sum_{ij} x_i x_j (\boldsymbol{\mathcal{K}})_{ij} \\ &= \frac{1}{2}\left(\sum_i d_i x_i^2 - 2\sum_{ij} x_i x_j (\boldsymbol{\mathcal{K}})_{ij} + \sum_j d_j x_j^2\right) \\ &= \frac{1}{2}\left(\sum_{ij} (\boldsymbol{\mathcal{K}})_{ij} x_i^2 - 2\sum_{ij} x_i x_j (\boldsymbol{\mathcal{K}})_{ij} + \sum_{ij} (\boldsymbol{\mathcal{K}})_{ij} x_j^2\right) \\ &= \frac{1}{2}\sum_{ij} (\boldsymbol{\mathcal{K}})_{ij}(x_i - x_j)^2 \geq 0, \end{aligned}$$

so that $\boldsymbol{L}$ is positive semi-definite. In particular,

$$\boldsymbol{1}_m^\top\boldsymbol{L}\boldsymbol{1}_m = \frac{1}{2}\sum_{ij} (\boldsymbol{\mathcal{K}})_{ij}(1-1)^2 = 0,$$

so that $\boldsymbol{L}\mathbf{1}_m = 0\mathbf{1}_m$ and $\mathbf{1}_m$ is an eigenvector with eigenvalue 0. Since $\boldsymbol{L}$ is symmetric positive semi-definite, all the other eigenvalues must be ≥ 0.
(c) Generalization could for example be accomplished by using the clustered samples to train a supervised classifier such as a neural network.

Exercise 7

Starting from the first equality in Equations (8.27):

$$\begin{aligned}
\Delta(k,j) &= \sum_{i\in C_k\cup C_j}\|\boldsymbol{g}_i-\bar{\boldsymbol{m}}\|^2-\sum_{i\in C_k}\|\boldsymbol{g}_i-\boldsymbol{m}_k\|^2-\sum_{i\in C_j}\|\boldsymbol{g}_i-\boldsymbol{m}_j\|^2\\
&= \sum_{i\in C_k}(\|\boldsymbol{g}_i-\bar{\boldsymbol{m}}\|^2-\|\boldsymbol{g}_i-\boldsymbol{m}_k\|^2)+\sum_{i\in C_j}(\|\boldsymbol{g}_i-\bar{\boldsymbol{m}}\|^2-\|\boldsymbol{g}_i-\boldsymbol{m}_j\|^2)\\
&= \sum_{i\in C_k}(2\boldsymbol{g}_i^\top(\boldsymbol{m}_k-\bar{\boldsymbol{m}})-\|\bar{\boldsymbol{m}}\|^2-\|\boldsymbol{m}_k\|^2)\\
&\quad+\sum_{i\in C_j}(2\boldsymbol{g}_i^\top(\boldsymbol{m}_j-\bar{\boldsymbol{m}})-\|\bar{\boldsymbol{m}}\|^2-\|\boldsymbol{m}_j\|^2)\\
&= 2n_k\boldsymbol{m}_k^\top(\boldsymbol{m}_k-\bar{\boldsymbol{m}})+n_k(\|\bar{\boldsymbol{m}}\|^2-\|\boldsymbol{m}_k\|^2)\\
&\quad+2n_j\boldsymbol{m}_j^\top(\boldsymbol{m}_j-\bar{\boldsymbol{m}})+n_j(\|\bar{\boldsymbol{m}}\|^2-\|\boldsymbol{m}_j\|^2)\\
&= n_k(\|\boldsymbol{m}_k\|^2+\|\bar{\boldsymbol{m}}\|^2)+n_j(\|\boldsymbol{m}_j\|^2+\|\bar{\boldsymbol{m}}\|^2)-2\bar{\boldsymbol{m}}^\top(n_k\boldsymbol{m}_k+n_j\boldsymbol{m}_j).
\end{aligned}$$

Now multiply the last term above by $(n_k+n_j)/(n_k+n_j)$ to give

$$\begin{aligned}
\Delta(k,j) &= n_k(\|\boldsymbol{m}_k\|^2+\|\bar{\boldsymbol{m}}\|^2)+n_j(\|\boldsymbol{m}_j\|^2+\|\bar{\boldsymbol{m}}\|^2)-2\|\bar{\boldsymbol{m}}\|^2(n_k+n_j)\\
&= n_k\|\boldsymbol{m}_k\|^2+n_k\|\boldsymbol{m}_j\|^2-\|\bar{\boldsymbol{m}}\|^2(n_k+n_j)\\
&= n_k\|\boldsymbol{m}_k\|^2+n_k\|\boldsymbol{m}_j\|^2-\frac{\|n_k\boldsymbol{m}_k+n_j\boldsymbol{m}_j\|^2}{n_k+n_j}\\
&= \frac{n_kn_j}{n_k+n_j}(\|\boldsymbol{m}_k\|^2-2\boldsymbol{m}_k^\top\boldsymbol{m}_j+\|\boldsymbol{m}_j\|^2)\\
&= \frac{n_kn_j}{n_k+n_j}\|\boldsymbol{m}_k-\boldsymbol{m}_j\|^2.
\end{aligned}$$

Exercise 8

The proposed cost function is

$$E(C)=\sum_k\sqrt{|\boldsymbol{\mathcal{C}}_k|}.$$

Under the linear transformation $\boldsymbol{T}$, the covariance matrix becomes

$$\boldsymbol{\mathcal{C}}_k\to\boldsymbol{\mathcal{C}}_k'=\boldsymbol{T}\boldsymbol{\mathcal{C}}_k\boldsymbol{T}^\top.$$

From the properties of the determinant, Chapter 1,

$$|\boldsymbol{\mathcal{C}}'_k| = |\boldsymbol{T}|^2 |\boldsymbol{\mathcal{C}}_k|,$$

so that

$$E'(C) = \sum_k \sqrt{|\boldsymbol{\mathcal{C}}'_k|} = \sum_k |\boldsymbol{T}| \sqrt{|\boldsymbol{\mathcal{C}}_k|} = |\boldsymbol{T}| E(C).$$

Therefore $E'(C)$ has the same minimum as $E(C)$.

Exercise 9

(a) Differentiating the log-likelihood with respect to μ_k,

$$\begin{aligned}
0 = \frac{\partial \mathcal{L}}{\partial \mu_k} &= \sum_i \frac{\partial}{\partial \mu_k} \log\left(\sum_{k'} p(g_i \mid k')\Pr(k')\right) \\
&= \sum_i \frac{1}{p(g_i)} \frac{\partial}{\partial \mu_k} p(g_i \mid k)\Pr(k) \\
&= -\sum_i \frac{1}{p(g_i)} \Pr(k) p(g_i \mid k) \frac{g_i - \mu_k}{\sigma_k^2} \\
&= -\sum_i \Pr(k \mid g_i) \frac{g_i - \mu_k}{\sigma_k^2} \text{ (using Bayes' Theorem).}
\end{aligned}$$

Solving for μ_k gives the required result, namely

$$\mu_k = \frac{\sum_i \Pr(k \mid g_i) g_i}{\sum_i \Pr(k \mid g_i)}.$$

The expression for σ_k^2 follows in the same way by setting the derivative of $\mathcal{L}$ with respect to σ_k equal to zero. The parameter $\Pr(k)$ is constrained by $\sum_k \Pr(k) = 1$, so to get it we must use the Lagrange function

$$L = \sum_i \log(\sum_k p(g_i \mid k)\Pr(k)) - \lambda(\sum_k \Pr(k) - 1).$$

Thus

$$\begin{aligned}
0 = \frac{\partial L}{\partial \Pr(k)} &= \sum_i \frac{1}{p(g_i)} p(g_i \mid k) - \lambda \\
&= \sum_i \frac{\Pr(k \mid g_i)}{\Pr(k)} - \lambda,
\end{aligned}$$

from which

$$\Pr(k) = \frac{1}{\lambda} \sum_i \Pr(k \mid g_i).$$

Finally, summing the last equation over k,

$$1 = \frac{1}{\lambda}\sum_i \sum_k \Pr(k \mid g_i) = \frac{1}{\lambda}\sum_i 1 = \frac{1}{\lambda}n,$$

so that $\lambda = n$.
(b) In the FMLE algorithm for one-dimensional data, replace the class membership u_{ki} by $\Pr(k \mid g_i)$ and it becomes identical to the EM algorithm.

Exercise 10

For a cubic array of pixels in a Kohonen network,

$$\begin{aligned} d^2(k^*, k) \; &= [(k^* - 1) \bmod m - (k-1) \bmod m]^2 \\ &\quad + [((k^*-1)/m - (k-1)/m) \bmod m]^2 \\ &\quad + [(k^*-1)/m^2 - (k-1)/m^2]^2. \end{aligned}$$

Exercise 12

(a) The mean shift preserves edges because each pixel is assigned to the basin of attraction of the nearest mode and the basins are unambiguous, that is, they have well-defined edges.
(b) Writing the Gaussian kernel density estimate as

$$p(\boldsymbol{w}) = \frac{1}{Z}\sum_\nu e^{-\|\boldsymbol{w}-\boldsymbol{y}_\nu\|^2/2\sigma^2},$$

where Z is the normailization, we have

$$\begin{aligned} \frac{\partial p(\boldsymbol{w})}{\partial \boldsymbol{w}} &= -\frac{1}{Z\sigma^2}\sum_\nu (\boldsymbol{w} - \boldsymbol{y}_\nu) e^{-\|\boldsymbol{w}-\boldsymbol{y}_\nu\|^2/2\sigma^2} \\ &= \frac{1}{Z\sigma^2}\sum_\nu e^{-\|\boldsymbol{w}-\boldsymbol{y}_\nu\|^2/2\sigma^2}\left[\frac{\sum_\nu \boldsymbol{y}_\nu e^{-\|\boldsymbol{w}-\boldsymbol{y}_\nu\|^2/2\sigma^2}}{\sum_\nu e^{-\|\boldsymbol{w}-\boldsymbol{y}_\nu\|^2/2\sigma^2}} - \boldsymbol{w}\right] \\ &= \frac{1}{\sigma^2}p(\boldsymbol{w})\left[\frac{\sum_\nu \boldsymbol{y}_\nu e^{-\|\boldsymbol{w}-\boldsymbol{y}_\nu\|^2/2\sigma^2}}{\sum_\nu e^{-\|\boldsymbol{w}-\boldsymbol{y}_\nu\|^2/2\sigma^2}} - \boldsymbol{w}\right]. \end{aligned}$$

The expression in square brackets is the difference between the weighted mean as calculated with the kernel and the center of the window, i.e., the mean shift. Thus the mean shift is proportional to the estimated gradient of the density.

9

Change Detection

Exercise 1

(a) Multiply the first equation in Equations (9.7) from the left by $\mathbf{\Sigma}_{11}^{-1}$ and the second by ρ to give

$$\begin{aligned}\mathbf{\Sigma}_{11}^{-1}\mathbf{\Sigma}_{12}\boldsymbol{b} - \rho\boldsymbol{a} &= \mathbf{0}\\ \rho\mathbf{\Sigma}_{12}^{\top}\boldsymbol{a} - \rho^2\mathbf{\Sigma}_{22}\boldsymbol{b} &= \mathbf{0}.\end{aligned}$$

Eliminate $\boldsymbol{a}$ by multiplying the first equation above from the left by $\mathbf{\Sigma}_{12}^{\top}$ and adding. This gives Equation (9.11).

Exercise 2

(a) The vector of canonical variates can be written as

$$\boldsymbol{U} = \boldsymbol{A}^{\top}\boldsymbol{G}.$$

Therefore the covariance matrix with the original variates $\boldsymbol{G}$ is

$$\begin{pmatrix} \mathrm{cov}(G_1, U_1) & \cdots & \mathrm{cov}(G_1, U_N) \\ \vdots & \ddots & \vdots \\ \mathrm{cov}(G_N, U_1) & \cdots & \mathrm{cov}(G_N, U_N) \end{pmatrix} = \langle \boldsymbol{G}\boldsymbol{U}^{\top}\rangle = \langle \boldsymbol{G}\boldsymbol{G}^{\top}\rangle\boldsymbol{A} = \mathbf{\Sigma}_{11}\boldsymbol{A}.$$

Multiplying this expression from the left with the matrix $\boldsymbol{D}$ will generate the matrix

$$\boldsymbol{D}\mathbf{\Sigma}_{11}\boldsymbol{A} = \begin{pmatrix} \mathrm{cov}(G_1, U_1)/\sqrt{\mathrm{var}(G_1)} & \cdots & \mathrm{cov}(G_1, U_N)/\sqrt{\mathrm{var}(G_1)} \\ \vdots & \ddots & \vdots \\ \mathrm{cov}(G_N, U_1)/\sqrt{\mathrm{var}(G_N)} & \cdots & \mathrm{cov}(G_N, U_N)/\sqrt{\mathrm{var}(G_N)} \end{pmatrix},$$

but since $\mathrm{var}(U_i) = 1$, $i = 1 \dots N$, this is just the correlation matrix

$$\boldsymbol{D}\mathbf{\Sigma}_{11}\boldsymbol{A} = \begin{pmatrix} \mathrm{corr}(G_1, U_1) & \cdots & \mathrm{corr}(G_1, U_N) \\ \vdots & \ddots & \vdots \\ \mathrm{corr}(G_N, U_1) & \cdots & \mathrm{corr}(G_N, U_N) \end{pmatrix}.$$

(b) Multiplying the above correlation matrix from the right with the diagonal matrix $\boldsymbol{S}$ gives

$$\boldsymbol{D}\boldsymbol{\Sigma}_{11}\boldsymbol{A}\boldsymbol{S} = \begin{pmatrix} \mathrm{corr}(G_1, U_1)\frac{s_1}{|s_1|} & \cdots & \mathrm{corr}(G_1, U_N)\frac{s_N}{|s_N|} \\ \vdots & \ddots & \vdots \\ \mathrm{corr}(G_N, U_1)\frac{s_1}{|s_1|} & \cdots & \mathrm{corr}(G_N, U_N)\frac{s_N}{|s_N|} \end{pmatrix}$$

with column sums

$$\frac{s_1^2}{|s_1|} \cdots \frac{s_N^2}{|s_N|},$$

all of which are positive as required.

Exercise 3

The answer is of course no. The MAD transformation is invariant under any linear transformation.

Exercise 5

(a) For the null hypothesis

$$\log(L(x)) = -2m \log x - (g_1 + g_2)\frac{m}{x} + \text{terms independent of x}.$$

Differentiating,

$$\frac{d\log(L(x))}{dx} = -2m\frac{1}{x} + (g_1 + g_2)\frac{m}{x^2} = 0.$$

Therefore

$$\hat{x} = \frac{g_1 + g_2}{2}.$$

The result for the alternative hypothesis follows just as easily.

(b) Taking logarithms:

$$\ln L_0 = -mj \ln x - \frac{m}{x}\sum_{i=1}^{i} g_i + \text{terms indpendent of } x.$$

Setting $\partial \ln_0 / \partial x = 0$ gives the result.

Exercise 6

The exponent in L_0 is obviously $-mj$. For L_1 we have

$$-m\frac{\sum_{i=1}^{j-1} g_i}{\frac{1}{j-1}\sum_{i=1}^{j-1} g_i} - m\frac{g_j}{g_j} = -m(j-1) - m = -mj$$

as well.

Exercise 8

Setting $N = 2$ the likelihood ratio test becomes

$$Q = 2^{4m} \frac{|\boldsymbol{x}_1|^m |\boldsymbol{x}_2|^m}{|\boldsymbol{x}_1 + \boldsymbol{x}_2|^{2m}} \leq k.$$

The quantity

$$-2\log Q = -2m(4\log 2 + \log|\boldsymbol{x}_1| + \log|\boldsymbol{x}_2| - 2\log|\boldsymbol{x}_1 + \boldsymbol{x}_2|)$$

is chi-square distributed with 4 degrees of freedom as $m \to \infty$.

Exercise 10

The second of Equations (9.50) follows from the scaling of the y-axis:

$$\frac{Y_R - \tilde{y}}{Y_T - y} = \frac{L_R}{L_T}.$$

The x-axis scales as

$$\frac{L_R/a_R}{L_T/a_T},$$

and hence

$$\frac{\tilde{x} - X_R}{x - X_T} = \frac{L_R a_T}{L_T/a_R}$$

from which we get the first of Equations (9.50).