MATH524 Assignment 2: Solution by Zino Meyer ID 132611593

October 30, 2024

2.8)

Given the following values:

Singular values:

$$D = \begin{bmatrix} 1.414214 & 0 \\ 0 & 1.414214 \end{bmatrix}$$

Left singular vectors:

$$U = \begin{bmatrix} -0.7071068 & 0.7071068 \\ 0.7071068 & 0.7071068 \end{bmatrix}$$

Right singular vectors: since V is symmetric, its transpose is the same

$$V = V^T = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}$$

Computing UDV^T:

1. Computing $DV^T = DV$

$$DV = \begin{bmatrix} (1.414214)(-1) + (0)(0) & (1.414214)(0) + (0)(1) \\ (0)(-1) + (1.414214)(0) & (0)(0) + (1.414214)(1) \end{bmatrix} \begin{bmatrix} -1.414214 & 0 \\ 0 & 1.414214 \end{bmatrix}$$

2. Compute $U(DV^T) = U(DV)$

$$\begin{gathered} U(DV) = \begin{bmatrix} -0.7071068 & 0.7071068 \\ 0.7071068 & 0.7071068 \end{bmatrix} \begin{bmatrix} -1.414214 & 0 \\ 0 & 1.414214 \end{bmatrix} \\ = \begin{bmatrix} (-0.7071068)(-1.414214) + (0.7071068)(0) & (-0.7071068)(0) + (0.7071068)(1.414214) \\ (0.7071068)(-1.414214) + (0.7071068)(0) & (0.7071068)(0) + (0.7071068)(1.414214) \end{bmatrix} \\ = \begin{bmatrix} 1,0000003361 & 1,0000003361 \\ -1,0000003361 & 1,0000003361 \end{bmatrix} \\ \approx \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix} = A \end{gathered}$$

2.9)

```
A <- matrix(
    c(
        1, 2, 3, 4, 5,
        6, 7, 8, 9, 10,
        11, 12, 13, 14, 15
    ),
    nrow = 3,
    byrow = TRUE
)

svd_A <- svd(A)

U <- svd_A$u
U
#             [,1]       [,2]       [,3]
# [1,] -0.2016649  0.8903171  0.4082483
# [2,] -0.5168305  0.2573316 -0.8164966
# [3,] -0.8319961 -0.3756539  0.4082483

D <- diag(svd_A$d) # convert from vector to diagonal matrix
D
#           [,1]     [,2]      [,3]
# [1,] 35.12722 0.000000 0.000e+00
# [2,]  0.00000 2.465397 0.000e+00
# [3,]  0.00000 0.000000 4.162e-16

V <- svd_A$v
V
#             [,1]        [,2]         [,3]
# [1,] -0.3545571 -0.68868664 -0.516261856
# [2,] -0.3986964 -0.37555453  0.822628731
# [3,] -0.4428357 -0.06242242 -0.091458089
# [4,] -0.4869750  0.25070970 -0.219922590
# [5,] -0.5311143  0.56384181  0.005013804
```

2.10)

```
# Since I converted D to a diagonal matrix in the previous exercise,
# I use D[1, 1] to get the scalar d1
d1 <- D[1, 1]
u1 <- U[, 1]
v1 <- V[, 1]
```

```
# Then, it is the scalar times the outer product of u1 and v1
B <- d1 * outer(u1, v1) # same as d1 * u1 %*% t(v1) since u1 and v1 are vectors
B
#           [,1]      [,2]      [,3]      [,4]      [,5]
# [1,]  2.511657  2.824337  3.137016  3.449696  3.762376
# [2,]  6.436920  7.238261  8.039602  8.840944  9.642285
# [3,] 10.362183 11.652185 12.942188 14.232191 15.522194
```

Because we only use the first singular vectors, we capture the main trend of the original matrix A, but rather simplified. The approximation smoothes out the original values. The values of each row vector of B are closer to the mean of the original row vector, i.e. the first value of a row is greater than the original, the middle one is almost equal, and the last one is smaller then the original.

2.11)

a)

```
a_a_t <- A %*% t(A)
eig_a_a_t <- eigen(a_a_t)

eig_a_a_t$values
# [1] 5.3834039 3.2895832 0.6170129

eig_a_a_t$vectors
#            [,1]      [,2]       [,3]
# [1,]  0.1586046 0.8912905  0.4247892
# [2,] -0.5272625 0.4402097 -0.7267803
# [3,]  0.8347687 0.1087047 -0.5397634
```

b)

```
a_t_a <- t(A) %*% A
eig_a_t_a <- eigen(a_t_a)

eig_a_t_a$values
# [1] 5.383404e+00 3.289583e+00 6.170129e-01 6.087397e-16

eig_a_t_a$vectors
#            [,1]       [,2]      [,3]       [,4]
# [1,]  0.2171740  0.8204225 0.1388675 -0.5103511
# [2,] -0.5206528 -0.3496772 0.3785963 -0.6806698
# [3,] -0.7280696  0.4410850 0.2426462  0.4652756
# [4,]  0.3894493 -0.1003834 0.8823284  0.2444361
```

c)

```
svd_A <- svd(A)
svd_A$d
# [1] 2.3202163 1.8137208 0.7855017

U <- svd_A$u
U
#             [,1]      [,2]       [,3]
# [1,]  0.1586046 0.8912905  0.4247892
# [2,] -0.5272625 0.4402097 -0.7267803
# [3,]  0.8347687 0.1087047 -0.5397634

V <- svd_A$v
V
#             [,1]       [,2]       [,3]
# [1,] -0.2171740  0.8204225 -0.1388675
# [2,]  0.5206528 -0.3496772 -0.3785963
# [3,]  0.7280696  0.4410850 -0.2426462
# [4,] -0.3894493 -0.1003834 -0.8823284
```

d)

The eigenvalues of AA^t and A^tA are identical (except A^tA has one almost zero eigenvalue more, but since it's e-16, it is negligible). These eigenvalues are the squares of the singular values D obtained from the SVD analysis (eigenvalues: 5.3834039 3.2895832 0.6170129; singular values D: 2.3202163 1.8137208 0.7855017)

The eigenvectors of AA^t are the same as the left singular vectors U. The eigenvectors of A^tA are *almost* the same as the right singular vectors V. This differences cna easily be explained: First, the eigenvectors have one extra column (one extra vector) over the right singular vectors. This vector corresponds to the zero eigenvalue and is therefore not included in the SVD. Second, the signs are flipped in the first and third eigenvector. This is because the signs are ambiguous with eigenvectors (it is basically a scalar multiplication with -1, the direction stays the same)

e)

The singular vectors U and V are orthogonal if their inner products with themselves (U^tU and V^tV) result in the identity matrix.

```
U_orthonormal <- t(U) %*% U
U_orthonormal
#                [,1]          [,2]         [,3]
# [1,]  1.000000e+00 -2.743475e-16 2.729620e-17
# [2,] -2.743475e-16  1.000000e+00 2.145436e-17
# [3,]  2.729620e-17  2.145436e-17 1.000000e+00
```

```
V_orthonormal <- t(V) %*% V
V_orthonormal
#               [,1]          [,2]          [,3]
# [1,]  1.000000e+00 -1.089398e-16  3.797139e-16
# [2,] -1.089398e-16  1.000000e+00 -1.643707e-16
# [3,]  3.797139e-16 -1.643707e-16  1.000000e+00
```

The results show that they are indeed the identity matrix. The very small values (e-16 and e-17) can be interpreted as zeros.

2.19)

a)

If we want to show that the eigenvalues of C are non-negative, we need to show that it is (1) a symmetric matrix and that (2) the quadratic form is also positive.

Hypothesis 1:

$$C = C^T$$

Proof 1:

$$\begin{aligned} C^T &= (AA^T)^T \\ &= A^{TT}A^T \\ &= AA^T = C \end{aligned}$$

→ C and its transpose are the same, thus C is symmetric.

Hypothesis 2:

$$Q = x^T C x > 0$$

Proof 2:

The quadratic form is defined as:

$$Q = x^T C x$$

Substituting $C = AA^T$:

$$Q = x^T (AA^T) x$$

Rewriting since matrices are associative:

$$Q = (x^T A)(A^T x)$$

With $x^T A = (A^T x)^T$:

$$Q = (A^T x)^T (A^T x)$$

This is the dot product of $A^T x$ with itself. With $y^T y = ||y||^2$ (a vector's dot product with itself equals it's squared norm) we get:

$$Q = \mid A^T x \mid^2$$

Conclusion:

→ Since the quadratic form Q of the matrix $C = AA^T$ is always a square, it will also always be positive: $\mid A^T x \mid^2 \geq 0.$

⇒ Since C is symmetric (1) and the quadratic form is always positive (2), we can conclude that the eigenvalues are non-negative.

b)

Given $C = AA^T$ and $v = A^T u$, show that v is an eigenvector of $C^T = A^T A$

Hypothesis: v is an eigenvector of C^T:

$$C^T v = \lambda v$$

Assumption: u is an eigenvector of $C = AA^T$, thus:

$$Cu = AA^T u = \lambda u$$

Proof:

Substitute C^T into hypothesis:

$$C^T v = A^T A v$$

Substitute v in:

$$= A^T A (A^T u)$$

By associativity, we can write:

$$= A^T (AA^T) u$$

Use $C = AA^T$:

$$= A^T (Cu)$$

With eigenvalue assumption $Cu = \lambda u$, we can substitute to:

$$= A^T(\lambda u)$$

With λ being a scalar, we can extract it from the matrix vector product:

$$= \lambda(A^T u)$$

Then finally, we can use the given $v = A^T u$:

$$= \lambda v$$

→ Thus, we proved that $C^T v = \lambda v.\ v = A^T u$ is an eigenvector of $C^T = A^T A$

2.20)

Given:

$$A = \begin{bmatrix} 2 & -1 \\ -1 & 3 \end{bmatrix}, x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, P(x_1, x_2) = x^T A A^T x$$

First, calculate the matrix AA^T:

$$\begin{aligned} AA^T &= \begin{bmatrix} 2 & -1 \\ -1 & 3 \end{bmatrix} \begin{bmatrix} 2 & -1 \\ -1 & 3 \end{bmatrix} \\ &= \begin{bmatrix} (2 \times 2) + (-1 \times -1) & (2 \times -1) + (-1 \times 3) \\ (-1 \times 2) + (3 \times -1) & (-1 \times -1) + (3 \times 3) \end{bmatrix} \\ &= \begin{bmatrix} 4+1 & -2-3 \\ -2-3 & 1+9 \end{bmatrix} \\ &= \begin{bmatrix} 5 & -5 \\ -5 & 10 \end{bmatrix} \end{aligned}$$

Then, we use it in the quadratic form $P(x_1, x_2) = x^T A A^T x$:

$$P(x_1, x_2) = \begin{bmatrix} x_1 & x_2 \end{bmatrix} \begin{bmatrix} 5 & -5 \\ -5 & 10 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

Then, perform the matrix multiplications:

$$\begin{aligned} &= \begin{bmatrix} x_1 & x_2 \end{bmatrix} \begin{bmatrix} 5x_1 - 5x_2 \\ -5x_1 + 10x_2 \end{bmatrix} \\ &= x_1(5x_1 - 5x_2) + x_2(-5x_1 + 10x_2) \\ &= 5x_1^2 - 5x_1x_2 - 5x_1x_2 + 10x_2^2 \\ &= 5x_1^2 - 10x_1x_2 + 10x_2^2 \end{aligned}$$

Result:

$$P(x_1, x_2) = 5x_1^2 - 10x_1x_2 + 10x_2^2$$

2.21)

Given:

The quadratic form in its polynomial form:

$$P(x_1, x_2) = 5x_1^2 - 10x_1x_2 + 10x_2^2$$

And x is a unit vector, meaning its magnitude is 1:

$$x_1^2 + x_2^2 = 1$$

Finding the maximum P_{max} of the quadratic $P(x_1, x_2)$:

We can rewrite the unit vector constraint:

$$\begin{gathered} x_1^2 + x_2^2 = 1 \\ \Rightarrow x_2^2 = 1 - x_1^2 \\ \Rightarrow x_2 = \sqrt{1 - x_1^2} \end{gathered}$$

Substitute x_2 into the quadratic function & simplify:

$$\begin{gathered} P(x_1) = 5x_1^2 - 10x_1\left(\sqrt{1-x_1^2}\right) + 10\left(\sqrt{1-x_1^2}\right)^2 \\ = 5x_1^2 - 10x_1 \times \sqrt{1-x_1^2} + 10\left(1 - x_1^2\right) \\ = 5x_1^2 - 10x_1 \times \sqrt{1-x_1^2} + 10 - 10x_1^2 \\ = -5x_1^2 - 10x_1 \times \sqrt{1-x_1^2} + 10 \end{gathered}$$

Now we can calculate the derivative:

$$P'(x) = -10\sqrt{1-x^2} + \frac{10x^2}{\sqrt{1-x^2}} - 10x$$

Set derivative to zero to find optima:

$$0 = -10\sqrt{1-x^2} + \frac{10x^2}{\sqrt{1-x^2}} - 10x$$

Gives two solutions:

$$\begin{gathered} x_1 = -\sqrt{\frac{5-\sqrt{5}}{10}} \\ x_2 = \sqrt{\frac{5+\sqrt{5}}{10}} \end{gathered}$$

The first one x_1 is the maximum. Plugging it into the function $P(x_1)$ gives the point:

$$P_{max} = (-0.52573, 13.0901699437495)$$

Relation of the maximum of quadratic form to the eigenvalues $C = AA^T$*:*

To find the eigenvalues of C, we use the definition of eigenvalues $det(A - \lambda I) = 0$. We replace A with C in our case:

$$det(\begin{bmatrix} 5 & -5 \\ -5 & 10 \end{bmatrix} - \lambda I) = 0$$
$$\Rightarrow det(\begin{bmatrix} 5-\lambda & -5 \\ -5 & 10-\lambda \end{bmatrix}) = 0$$
$$\Rightarrow (5-\lambda)(10-\lambda) - (-5)(-5) = 0$$
$$\Rightarrow \lambda^2 - 15\lambda + 25 = 0$$

Solving the resulting quadratic gives two values:

$$\lambda = \frac{15 \pm \sqrt{15^2 - 4 \cdot 1 \cdot 25}}{2} = \frac{15 \pm \sqrt{125}}{2}$$
$$\lambda_1 = 13,0901699437$$
$$\lambda_2 = 1,9098300563$$

Conclusion:

We see that the bigger eigenvalue λ_1 is the same as the y-value of the maximum of the quadratic function P:

$$P_{max} = (-0.52573, 13.0901699437495)$$

$$\lambda_1 = 13,0901699437$$

This shows that the eigenvalues respond to the optimisation of the quadratic form.

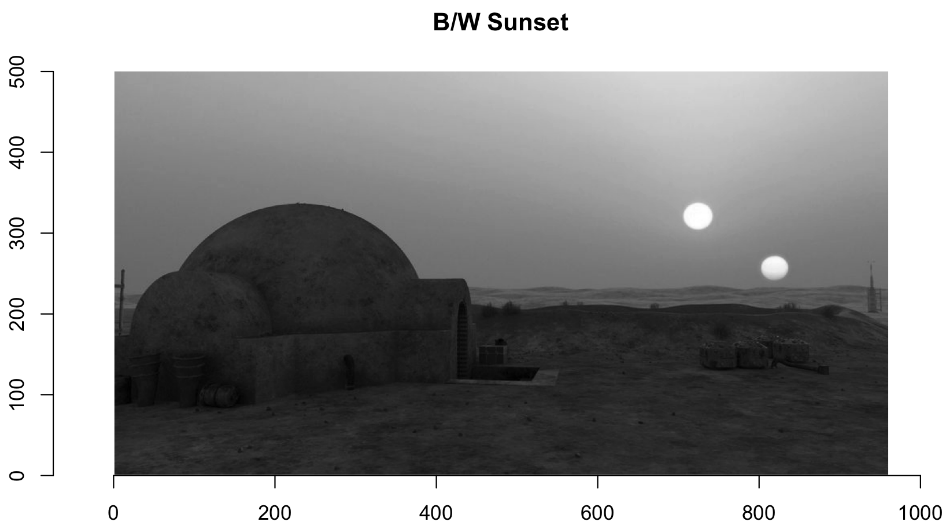
Problem #8

Chosen photo: sunset.jpeg

```
library(imager)
sunset_img <- load.image("images/sunset.jpeg") # 36 KB file size

# a)
A <- grayscale(sunset_img)
A[1:3, 1:5]
# [,1]      [,2]      [,3]      [,4]      [,5]
# [1,] 0.6625882 0.6625882 0.6625882 0.6625882 0.6625882
# [2,] 0.6625882 0.6625882 0.6625882 0.6625882 0.6625882
# [3,] 0.6625882 0.6625882 0.6625882 0.6625882 0.6625882

# b)
plot(A,
    xlim = c(0, 960), ylim = c(0, 500),
    main = "B/W Sunset"
)
dev.off()
```



Problem #9

a)

```
# SVD analysis
svd_A <- svd(A)

D <- svd_A$d
D[1:5]
# [1] 336.85945  31.45088  14.88263  11.22749   9.48810

U <- svd_A$u
U[1:5, 1:5]
# [,1]       [,2]       [,3]       [,4]        [,5]
# [1,] -0.02882415 0.02142493 0.01815178 0.02985490 -0.08097674
# [2,] -0.02889038 0.02141685 0.02113961 0.03609065 -0.09044643
# [3,] -0.02894194 0.02172832 0.02292550 0.03965408 -0.09658090
# [4,] -0.02891078 0.02176661 0.02186279 0.03627760 -0.09376029
# [5,] -0.02891545 0.02132887 0.02201846 0.03589219 -0.10129097

V <- svd_A$v
V[1:5, 1:5]
# [,1]        [,2]        [,3]        [,4]        [,5]
# [1,] -0.07250391 -0.04013077 0.008661080 -0.01334525 0.009426218
# [2,] -0.07248727 -0.04014685 0.008563544 -0.01317204 0.009528562
# [3,] -0.07239671 -0.04021180 0.009082291 -0.01339862 0.009017418
```

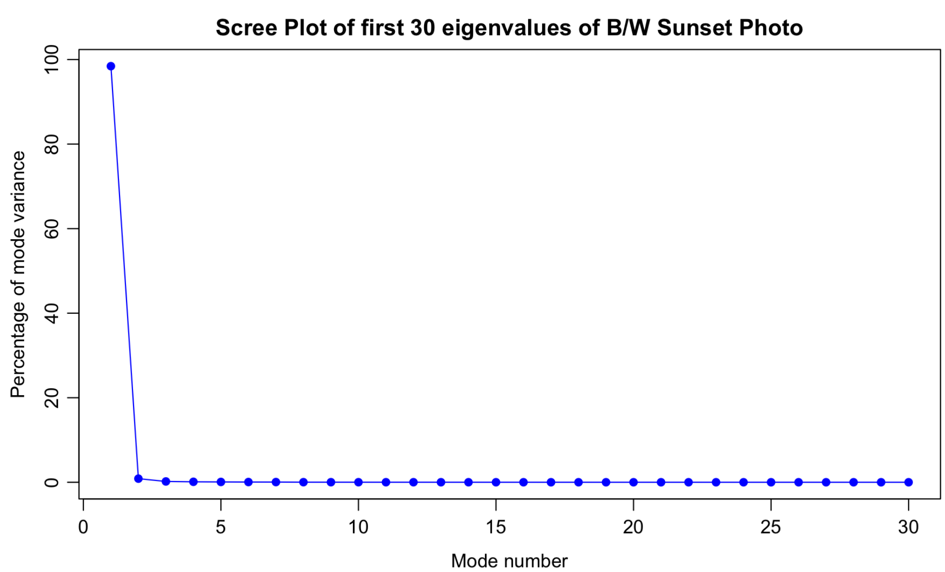
```
# [4,] -0.07229404 -0.04000343 0.009052688 -0.01311987 0.009939243
# [5,] -0.07221855 -0.03992743 0.008515128 -0.01305185 0.009673950
```

b)

```
# Scree plot of first 30 modes of A
variance_percent_D <- 100 * (D^2) / sum(D^2)
cum_percent_D <- cumsum(variance_percent_D)
modeK <- 1:length(D)
K <- 30

plot(modeK[1:K], variance_percent_D[1:K],
    type = "o", col = "blue",
    xlab = "Mode number", pch = 16,
    ylab = "Percentage of mode variance",
    main = "Scree Plot of first 30 eigenvalues of B/W Sunset Photo"
)

dev.off()
```

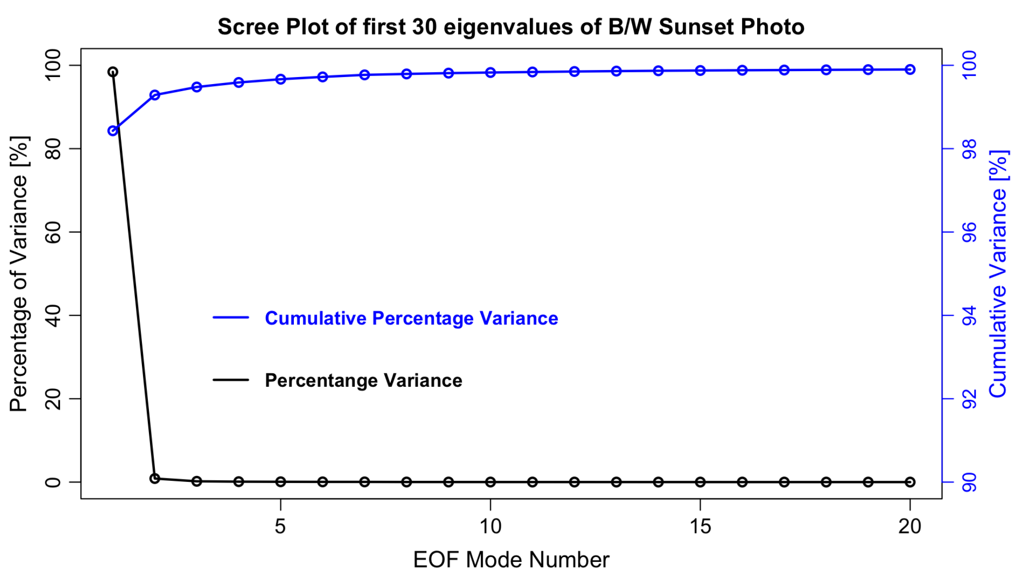



```
# Scree plot of first 30 modes of A, incl. cumulative variance
par(mar = c(4, 4, 2, 4), mgp = c(2.2, 0.7, 0))
plot(1:K,
    variance_percent_D[1:K],
    ylim = c(0, 100),
    type = "o",
    ylab = "Percentage of Variance [%]",
```

```
    xlab = "EOF Mode Number",
    cex.lab = 1.2, cex.axis = 1.1, lwd = 2,
    main = "Scree Plot of first 30 eigenvalues of B/W Sunset Photo"
)
legend(3, 30,
    col = c("black"), lty = 1, lwd = 2.0,
    legend = c("Percentange Variance"), bty = "n",
    text.font = 2, cex = 1.0, text.col = "black"
)

par(new = TRUE)
plot(1:K, cum_percent_D[1:K],
    ylim = c(90, 100), type = "o",
    col = "blue", lwd = 2, axes = FALSE,
    xlab = "", ylab = ""
)
legend(3, 94.5,
    col = c("blue"), lty = 1, lwd = 2.0,
    legend = c("Cumulative Percentage Variance"), bty = "n",
    text.font = 2, cex = 1.0, text.col = "blue"
)
axis(4, col = "blue", col.axis = "blue", mgp = c(3, 0.7, 0))
mtext("Cumulative Variance [%]",
    col = "blue",
    cex = 1.2, side = 4, line = 2
)

dev.off()
```



c)

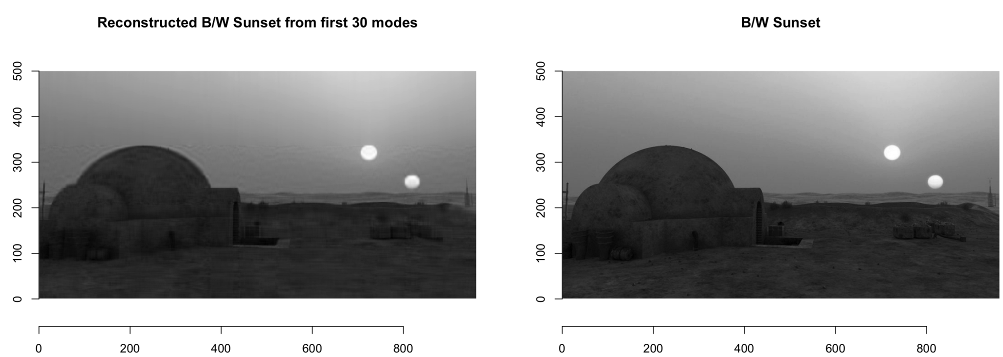
```
# Reconstruct from first 30 modes
A_recon_30 <- U[, 1:30] %*% diag(D)[1:30, 1:30] %*% t(V[, 1:30])
A_recon_30[1:5, 1:5]
# [,1]      [,2]      [,3]      [,4]      [,5]
# [1,] 0.6705306 0.6702721 0.6715154 0.6707233 0.6698029
# [2,] 0.6700016 0.6696815 0.6710111 0.6696757 0.6687013
# [3,] 0.6682787 0.6679999 0.6691361 0.6676661 0.6670194
# [4,] 0.6676286 0.6673559 0.6683376 0.6669393 0.6661647
# [5,] 0.6679623 0.6676991 0.6686842 0.6672425 0.6665847
```

d)

```
# Plot reconstructed next to original image
par(mfrow = c(1, 2), mar = c(3, 3, 3, 3))

plot(as.cimg(A_recon_30),
    xlim = c(0, 960), ylim = c(0, 500),
    main = "Reconstructed B/W Sunset from first 30 modes"
)
plot(A,
    xlim = c(0, 960), ylim = c(0, 500),
    main = "B/W Sunset"
)

dev.off()
```



Problem # 10:

Given:

$$A = UDV^t$$

$$C = \frac{1}{Y} AA^t$$

Hypothesis 1:

The eigenvectors $\mathbf{u}_k$ of the covariance matrix C are the same as the spatial modes from the SVD of A

Hypothesis 2:

Show that the eigenvalues λ_k of the covariance matrix C relate to the SVD eigenvalues d_k of A by $\lambda_k = \frac{d_k^2}{Y}$

Proof:

1. Substituting A in C and rewriting:
   - Definition of Covariance Matrix

$$C = \frac{1}{Y} AA^t$$

   - Substituting A in:

$$C = \frac{1}{Y}(UDV^t)(UDV^t)^t$$

   - with $(UDV^t)^t = (V^t)^t D^t U^t = VDU^t$, we can rewrite by associativity:

$$\begin{aligned} C &= \frac{1}{Y}(UDV^t)VDU^t \\ &= \frac{1}{Y}UD(V^tV)DU^t \end{aligned}$$

   - And since $V^tV = I_Y$, we can rewrite to:

$$\begin{aligned} C &= \frac{1}{Y}UDIDU^t \\ &= \frac{1}{Y}UD^2U^t \end{aligned}$$

   - This is similar to the eigenvalue decomposition, with Q being the matrix of eigenvectors and Λ being the diagonal matrix of eigenvalues:

$$C = Q\Lambda Q^T$$

- Thus, we show that the spatial modes U are the eigenvectors of the covariance matrix C.

2. Substituting C in the eigenvalues problem:
   - We have the following problem:

$$C\mathbf{u}_k = \lambda_k \mathbf{u}_k$$

   - And with $C = \frac{1}{Y}UD^2U^t$, we can rewrite to:

$$\frac{1}{Y}UD^2U^tU = \lambda U$$
$$\frac{1}{Y}UD^2 = \lambda U$$

   - Thus:

$$\lambda = \frac{1}{Y}D^2$$

   - And thus:

$$\lambda_k = \frac{d_k^2}{Y}$$

Thus, we proved that the eigenvectors of the covariance matrix of A are the same as the spatial modes of the SVD of the matrix A, and that the eigenvalues of the covariance matrix of A relate to the singular values of the SVD of the matrix A by square.