1. Consider data points in the plane $(x_1, y_1), \dots, (x_n, y_n)$, and assume you are searching for parameters of a function $f(x)$ such that,

$$\sum_{i=1}^{n} (f(x_i) - y_i)^2,$$

is minimized.

   (a) When $f(x) = \beta_0 + \beta_1 x$, there are simple formulas for optimizing $\beta_0$ and $\beta_1$. See for example formulas (8.18) - (8.20) in [SWJ - draft version not for circulation available via BB]. Use the design matrix $A$ as in formula (8.8) in [SWJ], and the pseudo-inverse $A^\dagger = (A^T A)^{-1} A^T$ to derive the formulas in (8.18).

   (b) Under what conditions on $x_1, \dots, x_n$ does the inverse $(A^T A)^{-1}$ above exist?

   (c) See now formula (8.35) in Chapter 8 of [SWJ]. It presents an expression for $H_{ii}$, the diagonal elements of the projection matrix $A(A^T A)^{-1} A^T$. Derive this formula.

   (d) Consider Listing 8.8 in [SWJ]. Modify the code in this listing to find the estimates of $\beta_0$ and $\beta_1$, directly via the pseudo-inverse via: $\hat{\beta} = A^\dagger y$ (where $\hat{\beta}$ is the vector of length 2 and $y$ is the vectors of $y_1, \dots, y_n$). For each of the four datasets, find the point with the highest leverage (highest $H_{ii}$).

   (e) Assume now that you wish to fit $f(x) = \beta_0 + \beta_1 x + \beta_2 x^2$. Here the design matrix $A$ is $n \times 3$. Repeat the fitting of Listing 8.8 using this type of model. Display your fit model graphically.

   (f) Assume now that $f(x) = \beta_1 x$ (no intercept term). Derive a formula for the optimal $\beta_1$, again using the pseudo inverse, this time with the matrix $A = [x_1 \quad \dots \quad x_n]^T$.

2. Consider a polynomial of degree $n$ with real valued coefficients:

$$p(u) = a_0 + a_1 u + a_2 u^2 + \ldots + a_{n-1} u^{n-1} + u^n.$$

The $n \times n$ *companion matrix* associated with this polynomial is,

$$C_p = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \cdots & \cdots & \cdots & \ddots & \cdots \\ 0 & 0 & 0 & \cdots & 1 \\ -a_0 & -a_1 & -a_2 & \cdots & -a_{n-1} \end{bmatrix}.$$

   (a) Present the companion matrix for the polynomial $p(u) = (2-u)(7-u)(9-u)$.

   (b) Compute the characteristic polynomial of the companion matrix for this polynomial. Show that it equals $p(u)$.

   (c) Try to prove that in general, the characteristic polynomial of a companion matrix $C_p$ equals the polynomial $p(u)$. If you are not able to do so in general do it for $n = 1, 2, 3, 4$.

   (d) If you were given a method to efficiently compute the eigenvalues of any matrix, then the companion matrix allows you to use the eigenvalue algorithm for find roots of polynomials. Use now Julia's `eigvals()` to find the roots of the polynomial in (a).

   (e) Consider now the polynomial $p(u) = \prod_{i=1}^{10}(u - i)$. Its roots are clearly $1, 2, \dots, 10$. Expand $p(u)$ analytically to obtain the coefficients $p(u) = a_0 + a_1 u + \ldots + a_9 u^9 + u^{10}$. Then compute the eigenvalues of the companion matrix $C_p$ to verify they are $1, \dots, 10$.

3. Consider $x_1(n)$ as the population of owls (in hundreds) at time $n$ and $x_2(n)$ as the population of mice (in tens of thousands) at time $n$. Assume the following model:

$$x_1(n) = \frac{2}{5}x_1(n-1) + \frac{3}{5}x_2(n-1)$$
$$x_2(n) = -\frac{3}{10}x_1(n-1) + \frac{13}{10}x_2(n-1)$$

Assume that at $n = 0$ we have $x_1 = 2$ and $x_2 = 3$ (this is 200 owls and 30,000 mice).

(a) Represent this as the linear dynamical system $x(n) = Ax(n-1)$.
What is the matrix $A$?

(b) Determine the eigenvalues of $A$.

(c) Find corresponding eigenvectors $v_1$ and $v_2$.

(d) Represent $x_0 = [2 \quad 3]^T$ as $x_0 = \alpha_1 v_1 + \alpha_2 v_2$.

(e) Now use diagonalization to compute explicit expressions for $x_1(n)$ and $x_2(n)$ for any time $n$ based on the expansion of $x_0$ in the basis $\{v_1, v_2\}$ as in (d).

(f) Determine $\lim_{n\to\infty}$ of the vector $x(n)$? What is the meaning of this vector.

(g) Plot the trajectory of $x(n)$ both in a manner similar to Figure 10.1 in [SWJ] (on the $x_1$, $x_2$ plane), and as functions of $n$ (both plots for $x_1(n)$ and $x_2(n)$ on the same plot). Present the two alternatives plots of this dynamical system, side by side. Make sure that the plots are neatly labeled and formatted. The plots should show convergence to the limiting point found in (f).

4. Consider the matrix

$$S = \begin{bmatrix} a & a & a \\ a & a+b & a-b \\ a & a-b & a+b \end{bmatrix}.$$

(a) Prove that the eigenvectors of $S$ are real valued (not complex).

(b) Show that for any vector $x = [x_1 \quad x_2 \quad x_3]^T$,

$$x^T S x = a(x_1 + x_2 + x_3)^2 + b(x_2 - x_3)^2.$$

(c) Use as many methods as you can to determine for which values of $a$ and $b$, the matrix $S$ is positive semidefinite (and positive definite).

(d) Assume now that you have a function $f : \mathbb{R}^3 \to \mathbb{R}$ and that $S$ is the Hessian matrix of this function around the point $x^{(0)}$ at which $\nabla f(x^{(0)}) = 0$. What are the conditions on $a$ and $b$ for $x^{(0)}$ to be a local minimum? How about a local maximum?

5. Take a "Selfie" of yourself and transform it to a $300 \times 200$ monochrome matrix with elements in the range $[0, 1]$ (if you prefer a different picture use that instead - but make sure that it is a picture that you took).

(a) Plot your selfie using `heatmap()`.

(b) Now present low-rank, SVD based approximations of your selfie, including rank $= 1, 5, 10, 15, 20, 40, 80, 160, 200$ (the last one being full rank). Use the Julia `svd()` function for this purpose.

(c) Determine what you believe is the "optimal" rank. Compute the storage savings with this rank approximation (how many numbers are needed in comparison to $200 \times 300$).