

Homework Set 4, Bill Karr

Problem 1: Optimization Theory (20 points)

- (a) Prove that any local minimum of a convex function f on a convex set $\mathcal{S} \subseteq \mathbb{R}^n$ is a global minimum of f on \mathcal{S} .

Proof. Suppose $x \in \mathcal{S}$ is a local minimum of f but not a global minimum. Then, choose $y \in \mathcal{S}$ with $f(y) < f(x)$. Since f is convex, then for any $s, t \in (0, 1)$ with $s + t = 1$, we have

$$f(sx + ty) \leq sf(x) + tf(y) < sf(x) + tf(x) = (s + t)f(x) = f(x).$$

So, f is smaller than $f(x)$ on the whole open line segment between x and y . Therefore, $f(x)$ is *not* a local minimum, a contradiction. \square

- (b) Let $f : \mathbb{R}^n \rightarrow \mathbb{R}$ and $g : \mathbb{R}^n \rightarrow \mathbb{R}^m$. Consider the optimization problem

$$\min_x f(x) \text{ subject to } g(x) = 0.$$

Show that the Hessian of the *Lagrangian function* $\mathcal{L} : \mathbb{R}^{n+m} \rightarrow \mathbb{R}$

$$\mathcal{L}(y) = \mathcal{L}(x, \lambda) = f(x) + \lambda^T g(x)$$

is given by

$$H_{\mathcal{L}}(x, \lambda) = \begin{bmatrix} H_f(x) + \sum_{i=1}^m \lambda_i H_{g_i}(x) & J_g^T(x) \\ J_g(x) & 0 \end{bmatrix}.$$

Proof. We need only compute all of the second partial derivatives.

$$\frac{\partial^2 \mathcal{L}}{\partial x_i \partial x_j} = \frac{\partial^2 f}{\partial x_i \partial x_j} + \sum_{k=1}^m \lambda_k \frac{\partial^2 g_k}{\partial x_i \partial x_j} = (H_f(x))_{ij} + \sum_{k=1}^m \lambda_k (H_{g_k})_{ij},$$

$$\frac{\partial^2 \mathcal{L}}{\partial x_i \partial \lambda_j} = \frac{\partial g_j}{\partial x_i} = (J_g(x))_{ji},$$

$$\frac{\partial^2 \mathcal{L}}{\partial \lambda_i \partial x_j} = \frac{\partial}{\partial \lambda_i} \left(\frac{\partial f}{\partial x_j} + \lambda^T \frac{\partial g}{\partial x_j} \right) = \frac{\partial g_i}{\partial x_j} = (J_g(x))_{ij},$$

and finally

$$\frac{\partial^2 \mathcal{L}}{\partial \lambda_i \partial \lambda_j} = 0$$

since \mathcal{L} only depends linearly on the λ_i . Thus, putting these terms together in a matrix, we see that

$$H_{\mathcal{L}}(x, \lambda) = \begin{bmatrix} H_f(x) + \sum_{k=1}^m \lambda_k H_{g_k}(x) & J_g^T(x) \\ J_g(x) & 0 \end{bmatrix}.$$

\square

- (c) Prove that $H_{\mathcal{L}}(x, \lambda)$ cannot be positive definite.

Proof. For any $\beta \in \mathbb{R}^m \setminus \{0\}$, notice that

$$\begin{bmatrix} 0 \\ \beta \end{bmatrix}^T \begin{bmatrix} H_f(x) + \sum_{k=1}^m \lambda_k H_{g_k}(x) & J_g^T(x) \\ J_g(x) & 0 \end{bmatrix} \begin{bmatrix} 0 \\ \beta \end{bmatrix} = [\beta^T J_g(x) \quad 0] \begin{bmatrix} 0 \\ \beta \end{bmatrix} = 0$$

Therefore, we can find nonzero vectors $v \in \mathbb{R}^{n+m}$ so that $v^T H_{\mathcal{L}}(x, \lambda) v \not\geq 0$, i.e. $H_{\mathcal{L}}(x, \lambda)$ is not positive definite. \square

- (d) What does the non-positive-definiteness of $H_{\mathcal{L}}(x, \lambda)$ mean for the solution of the problem $\nabla \mathcal{L}(x, \lambda) = 0$?

Solution. It means that if we find a critical point of \mathcal{L} , i.e. (x, λ) with $\nabla \mathcal{L}(x, \lambda) = 0$, there is no guarantee that it will be an absolute minimum because \mathcal{L} is not convex. \square

Problem 2: Optimization on quadratic forms (20 points)

- (a) Provide an analytic expression for the gradient ∇f and the Hessian H_f .

Solution. If $f(x_1, \dots, x_n) = \frac{1}{2} \sum_{i,j=1}^n x_i A_{ij} x_j - \sum_{i=1}^n x_i b_i + c$, then

$$\frac{\partial f}{\partial x_k} = \frac{1}{2} \sum_{i,j=1}^n (\delta_{ik} A_{ij} x_j + x_i A_{ij} \delta_{jk}) - b_k = \frac{1}{2} \sum_{i=1}^n (A_{ki} x_i + A_{ik} x_i) - b_k = \sum_{i=1}^n A_{ki} x_i - b_k$$

Thus, $\nabla f(x) = Ax - b$.

In addition,

$$\frac{\partial^2 f}{\partial x_k \partial x_m} = A_{km}.$$

Thus, $H_f = A$. \square

- (b) At what point does f attain a minimum?

Solution. Since A is symmetric and positive-definite, f is convex and achieves its minimum at any critical point. Thus, when

$$\nabla f(x) = Ax - b = 0 \Rightarrow x = A^{-1}b,$$

f achieves its minimum. \square

- (c) Show that Newton's method for finding the minimum converges in one iteration from any starting point, x_0 , by showing that the error after one step, $e_1 = x_1 - x^*$, is zero.

Proof. Since $H_f = A$ and $\nabla f(x) = Ax - b$, then using Newton's method starting with $x_0 \in \mathbb{R}^n$, we obtain

$$x_1 = x_0 - H_f(x_0)^{-1} \nabla f(x_0) = x_0 - A^{-1}(Ax_0 - b) = x_0 - x_0 + A^{-1}b = A^{-1}b$$

which is the minimum point of f . Therefore, Newton's method converges after one iteration. \square

- (d) Show that one step of steepest descend (for this function) can be written as

$$x_1 = x_0 + \frac{r^T r}{r^T A r} r$$

where $r = b - Ax$. What happens to the error after one step if the initial error, $e_0 = x_0 - x^*$, is an eigenvector of A .

Proof. For the steepest descend method, we let $r = -\nabla f(x_0) = b - Ax_0$ and choose $\alpha \in \mathbb{R}$ to minimize $f(x_0 + \alpha r)$. Then, $x_1 = x_0 + \alpha r$.

Notice that $\frac{d}{d\alpha} f(x_0 + \alpha r) = r^T \nabla f(x_0 + \alpha r)$. Thus, $\frac{d}{d\alpha} f(x_0 + \alpha r) = 0$ when

$$r^T (A(x_0 + \alpha r) - b) = 0 \Rightarrow r^T (Ax_0 - b) + \alpha r^T Ar = 0 \Rightarrow \alpha = \frac{r^T r}{r^T Ar}.$$

We conclude that

$$x_1 = x_0 + \frac{r^T r}{r^T Ar} r.$$

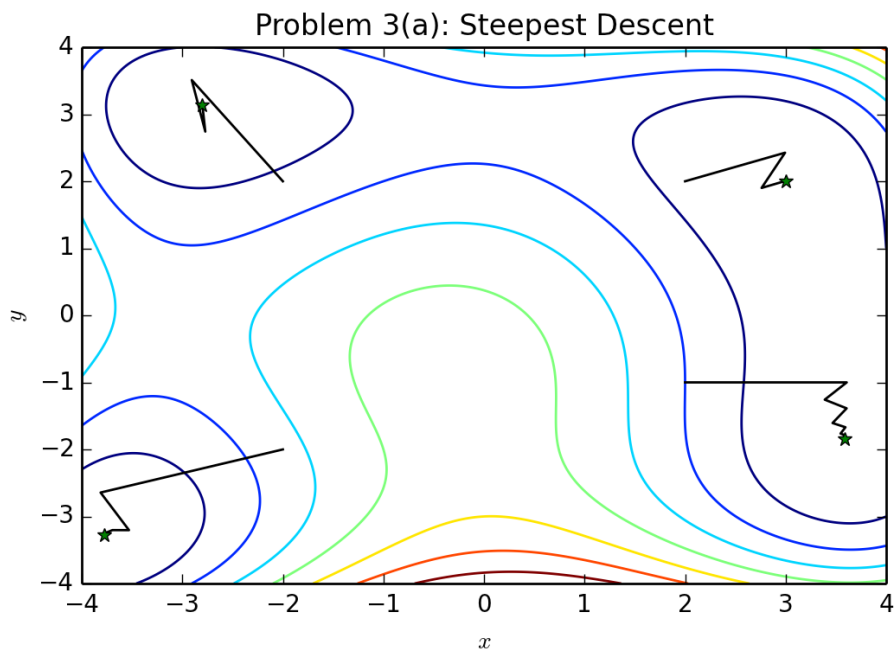
If $e_0 = x_0 - x^* = x_0 - A^{-1}b$ is an eigenvector with eigenvalue λ , then $r = -(Ax_0 - b) = -A(x_0 - A^{-1}b) = -\lambda(x_0 - x^*)$ is also an eigenvector of A with the same eigenvalue and $\alpha = r^T r / (r^T Ar) = \frac{1}{\lambda}$. Thus,

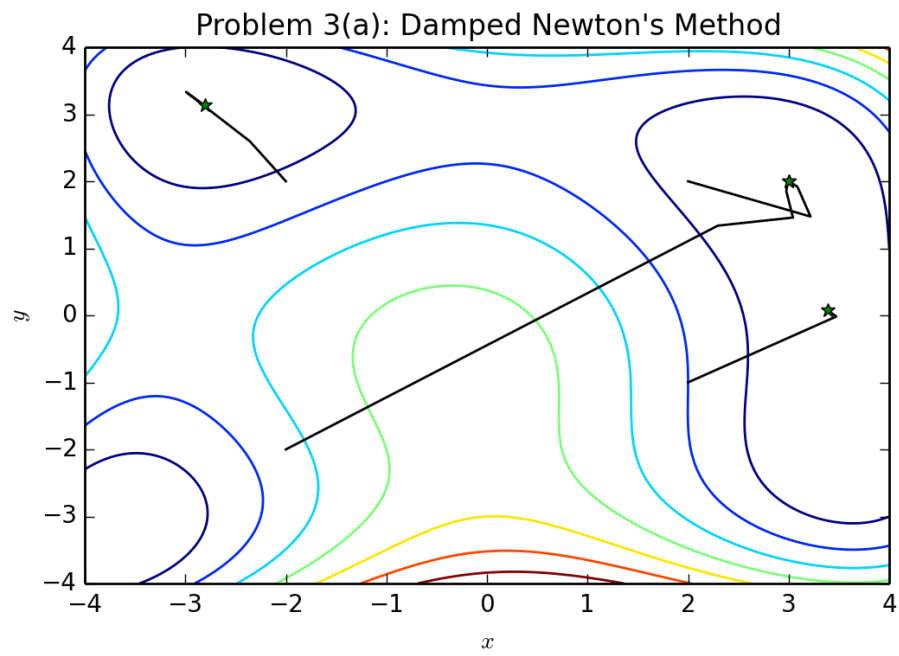
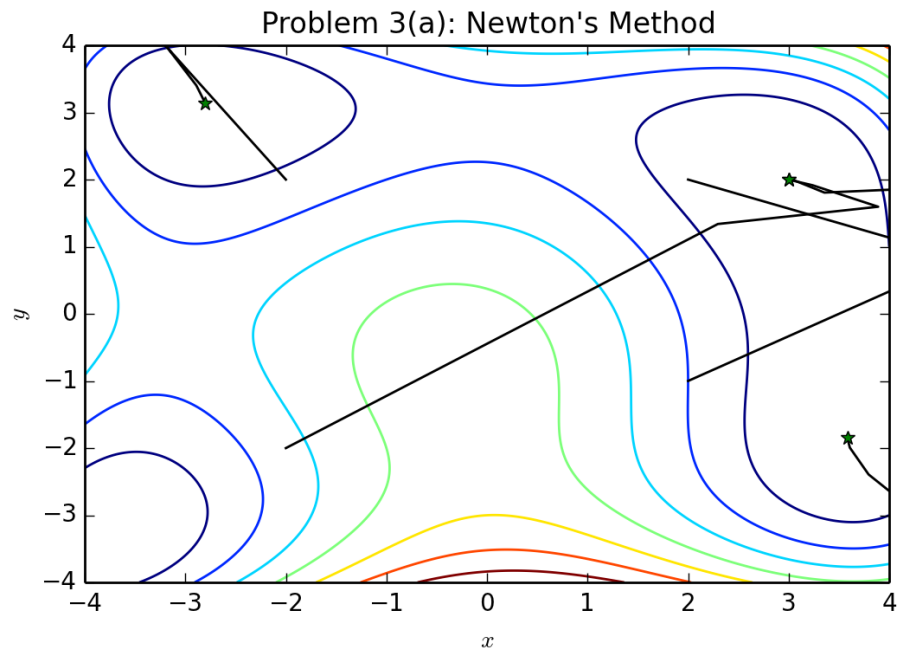
$$x_1 = x_0 + \frac{r}{\lambda}.$$

□

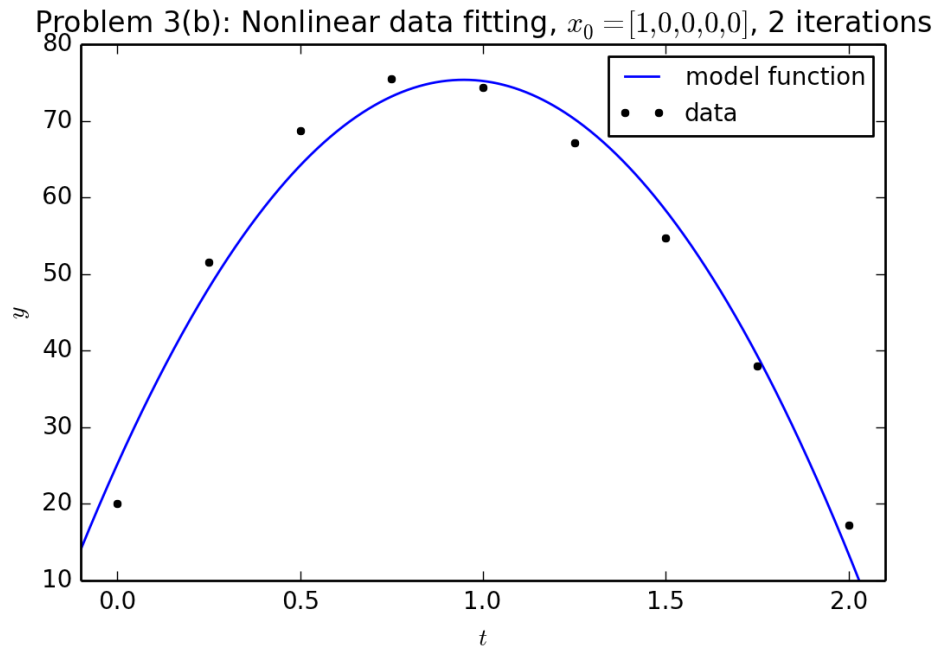
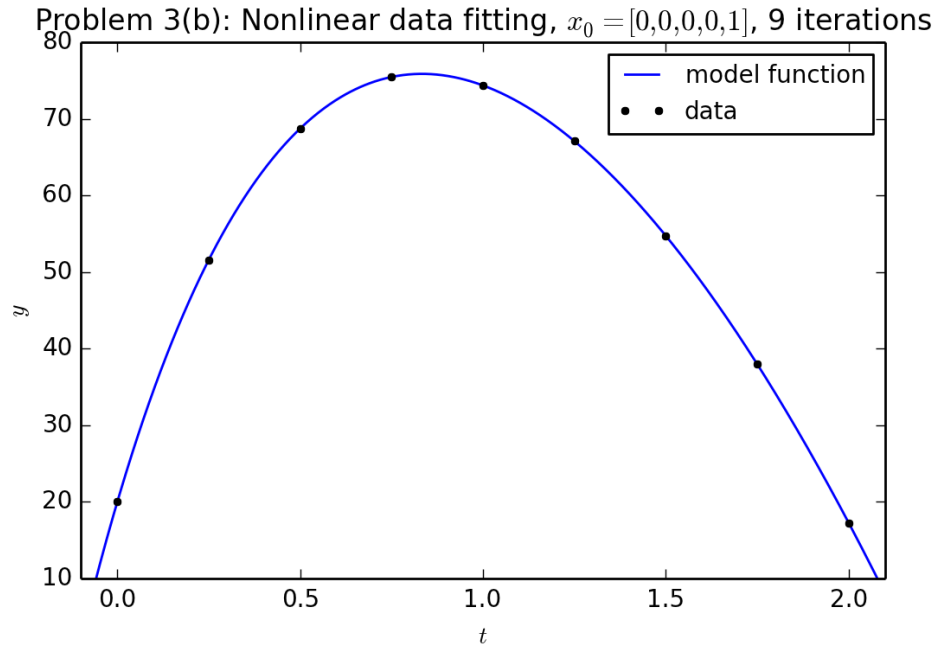
Problem 3: Optimization and Nonlinear Data Fitting (20 points)

- (a) See `cs450hw4p3.py` for my code. The damped Newton's method finished after the fewest iterations and steepest descent took the longest. This makes sense since steepest descent has linear convergence and Newton's method already has quadratic convergence but the line search is only speeding that up by ensuring not to go too far at each step. Here are the plots for each of the three iterative methods with the four starting points:

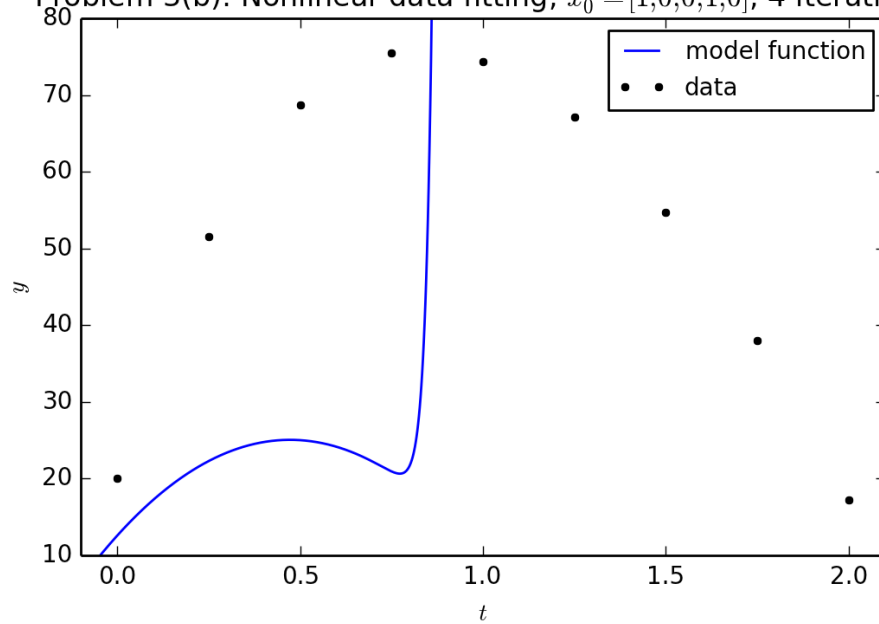




- (b) See `cs450hw4p3.py` for my code. The result from the first starting vector matches almost perfectly, the second one is slightly off but still a good fit, and the third is not a good fit at all. The algorithm terminated for each starting vector. Here are the plots for the three cases:

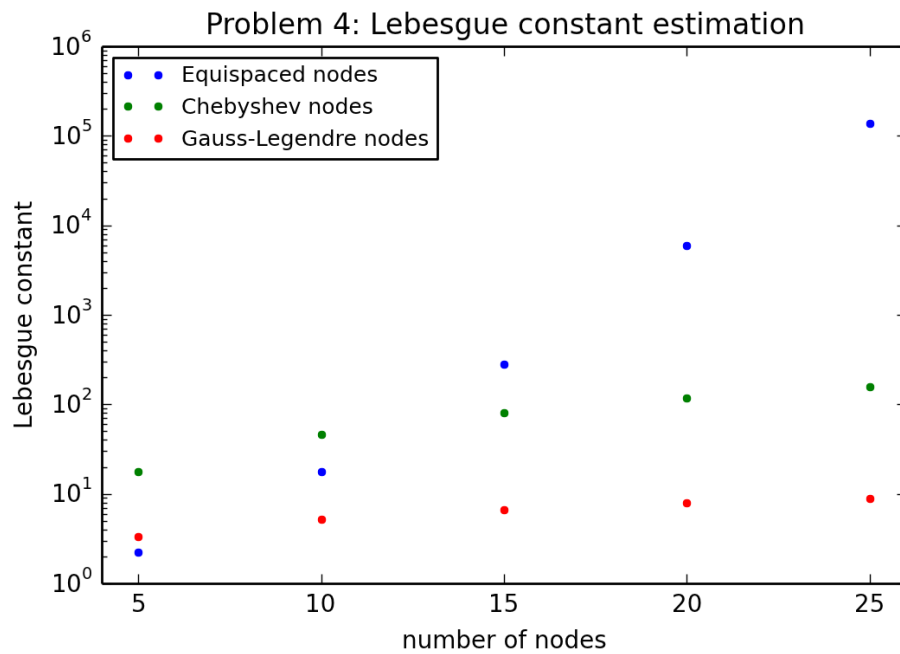


Problem 3(b): Nonlinear data fitting, $x_0 = [1, 0, 0, 1, 0]$, 4 iterations



Problem 4: Stability of Interpolation (20 points)

See `cs450hw4p5.py` for my code. Runge's phenomenon is when interpolated polynomials blow up near the endpoints of an interval when using equispaced nodes. Chebyshev nodes and Gauss-Legendre nodes concentrate near the endpoints to control Runge's phenomenon. Here is my plot:



Problem 5: Chebyshev polynomials, Vandermonde matrices (20 points)

- (a) *Proof.* Since $F_n(t) = \cos(n \arccos(t))$, we clearly have $F_0(t) = 1$ and $F_1(t) = t$. Suppose for $k = 2, \dots, n-1$, we have

$$F_{k+1}(t) = 2tF_k(t) - F_{k-1}(t).$$

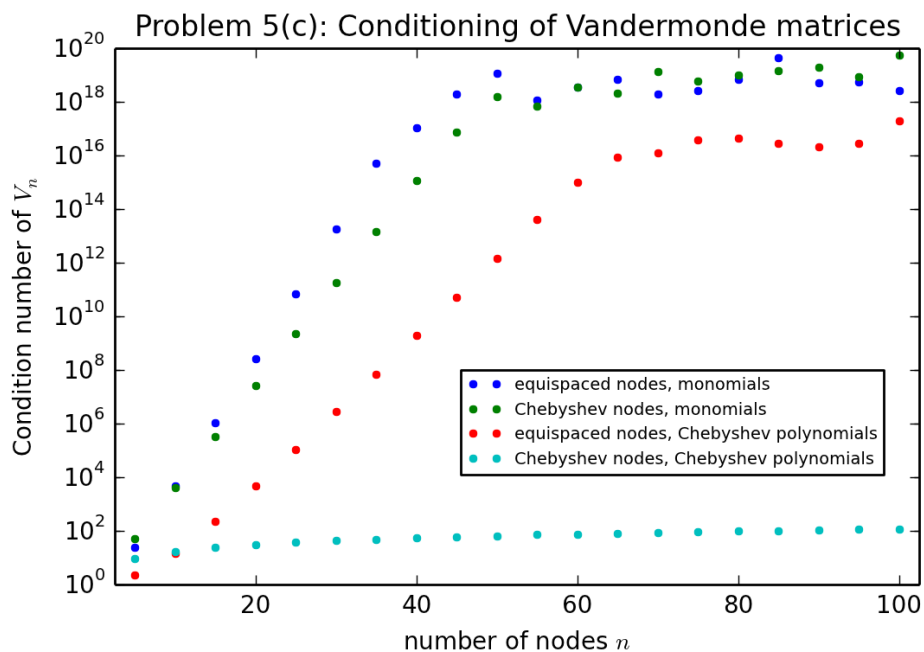
Then,

$$\begin{aligned} F_{n+1}(t) &= \cos((n+1) \arccos(t)) = \cos(n \arccos(t)) \cos(\arccos(t)) - \sin(n \arccos(t)) \sin(\arccos(t)) \\ &= 2 \cos(n \arccos(t)) \cos(\arccos(t)) - (\cos(n \arccos(t)) \cos(\arccos(t)) + \sin(n \arccos(t)) \sin(\arccos(t))) \\ &\quad 2t \cos(n \arccos(t)) - \cos((n-1) \arccos(t)) = 2tF_n(t) - F_{n-1}(t). \end{aligned}$$

By induction, $F_{n+1}(t) = 2tF_n(t) - F_{n-1}(t)$ for $n \geq 2$. \square

- (b) *Proof.* First, $F_0(t) = 1$ and $F_1(t) = t$ are polynomials. Suppose $F_0(t), F_1(t), \dots, F_n(t)$ are polynomials. Then, $F_{n+1}(t) = 2tF_n(t) - F_{n-1}(t)$ is also a polynomial since F_n and F_{n-1} are polynomials. By induction $F_n(t)$ is a polynomial for all n . \square

- (c) See `cs450hw4p5.py` for my code. Here is my plot:



- (d) The combination of Chebyshev nodes and Chebyshev polynomials worked best.