Problem Set 6 Solutions

Problem 1. *Naïve curve fitting and its consequences*

Just like we were able to construct a line of best fit for a collection of data points, we can use the least-squares method to construct a "polynomial of best fit" instead. Unless care is taken, we are going to run into to very nasty problems rather quickly. The following exercises outlines some naïve approaches and some suggestions for how to fix the problems encountered with each.

(a) For any positive integer n, define the set

$$S_n = \{-n, -n+1, \dots, -1, 0, 1, \dots, n-1, n\} \subset \mathbb{R},$$

and take $f(x) = \sin(x)$. Use the model

$$y(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_{2n} x^{2n} = \sum_{k=0}^{2n} a_k t^k$$

to create "parabola of best fit" for $f(x)$ on S_1. Repeat this to get a "quartic of best fit" on S_2 and a "sextic(?) of best fit" on S_3.

In sending the computation to MATLAB, we find

```
b =

         0
    0.8415
         0
```

for $n = 1$,

```
b =

         0
    0.9704
         0
   -0.1289
         0
```

for $n = 2$, and

```
b =

    0.0000
    0.9941
   -0.0000
   -0.1586
    0.0000
    0.0059
   -0.0000
```

for $n = 3$. That is, for $n = 1$,

$$y(x) \approx 0 \cdot 1 + 0.8415x + 0 \cdot x^2;$$

for $n = 2$,

$$y(x) \approx 0 \cdot 1 + 0.9704x + 0 \cdot x^2 - 0.1289x^3 + 0 \cdot x^4;$$

and for $n = 3$,

$$y(x) \approx 0 \cdot 1 + 0.9941x + 0 \cdot x^2 - 0.1586x^3 + 0 \cdot x^4 + 0.0059x^5 + 0 \cdot x^6.$$

(b) Write a MATLAB script that computes the degree-$2n$ curve of best fit for $f(x)$ on S_n. The backslash operator $A\backslash b$ in MATLAB automatically produces least-squares solutions to $A\vec{\mathbf{x}} = \vec{\mathbf{b}}$ when $\vec{\mathbf{b}} \notin \operatorname{im} A$.

Plot your regression curves and $f(x)$ together on the same graph for the cases when $n = 1, 2, 3, 4, 5$. Make a conjecture about what you expect this regression curve to approach as $n \to \infty$?

The script I used to generate my curves of best fit:

```
clc
close all;

for n = 1:5
    A = zeros(2*n+1);
    x = linspace(-n-2,n+2,1000);
    y = zeros(1,length(x));

    for jj = 0:(2*n)
        A(:,jj+1) = [-n:n]'.^jj;
    end

    f = sin([-n:n]');
    b = linsolve(A'*A,A'*f);

    for ii = 1:length(x)
        for jj = 1:(2*n+1)
            y(ii) = y(ii) + b(jj)*x(ii)^(jj-1);
        end
    end

    figure(1);
    subplot(1,3,n);
    plot(x,sin(x),x,y)
    title(['n = ',num2str(n)])
    ylim([-2,2])
end
```

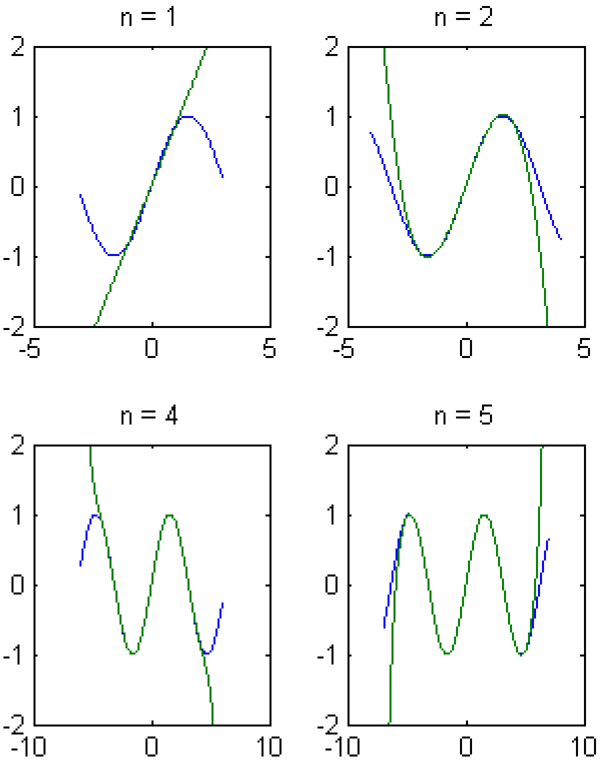

Figure 1: The curves of best fit (green) of $f(x)$ (blue) for the cases $n = 1, 2, 3, 4, 5$.

See Figure 1 for the plots of the curves of best fit for $n = 1, 2, 3, 4, 5$.

It would appear that the regression curve is beginning to align with $f(x) = \sin x$ as we increase n. We might conjecture that in the limit as $n \to \infty$ this approximation becomes exact.

(c) Try justifying your conjecture from part (b) by using your script to plot the degree-$2n$ regression curves for $n = 10, 20, 30$. What do you see in the plots of $y(x)$ compared with $f(x)$ as n gets larger? Does this prove or disprove your conjecture?

Consider the plots in Figure 2. We note the apparent “flailing” behavior of $y(x)$ toward $x = \pm n$ (which might be attributed to poor conditioning of the coefficient matrix A) for the larger values of n. These curves are certainly not making better approximations of $f(x) = \sin x$.

Since we have only plotted examples for relatively small values of n, this figure neither proves nor disproves our conjecture above; but it's certainly not looking good for the conjecture.

(d) We can try to fix the problem found in part (c) by using a different set of (smaller) sample points

$$T_n = \left\{ \frac{-n+j}{n} \right\}_{j=0}^{2n} \subset \mathbb{R}.$$

Append to your script from (b) another script that computes the degree-$2n$ curve of best fit for $f(x)$ on T_n instead and plots it with $f(x)$. Use this script for the

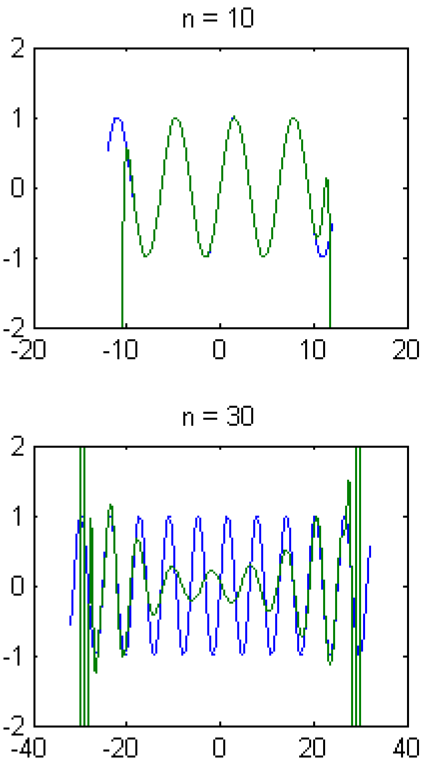

Figure 2: The degree-$2n$ regression curves $y(x)$ fitting $f(x) = \sin x$ on S_n for $n = 10, 20, 30$.

cases when $n = 10, 20, 30$. Does this new script fix the problem in part (c)? Did we introduce any new problems with this new script?

Here is the script used to generate the plots in Figure 3:

```
n = 10;

while n <= 30
    B = zeros(2*n+1);
    x = linspace(-10,10,1000);
    y = zeros(1,length(x));

    for jj = 0:(2*n)
        B(:,jj+1) = linspace(-1,1,2*n+1)'.^jj;
    end

    f = sin(linspace(-1,1,2*n+1)');
    b = linsolve(B,f);

    for ii = 1:length(x)
        for jj = 1:(2*n+1)
            y(ii) = y(ii) + b(jj)*x(ii)^(jj-1);
        end
```

```
    end

    figure(2);
    subplot(2,2,n/10)
    plot(x,sin(x),x,y)
    title(['n = ',num2str(n)])
    ylim([-2,2])

    n = n+10;
end
```

The $y(x)$ curve is now extraordinarily accurate in fitting the points on $f(x) = \sin x$ in the interval $-1 \leq x \leq 1$, but the curve now immediately deviates from $f(x)$ beyond this interval. Indeed, we've fixed one problem but introduced another.

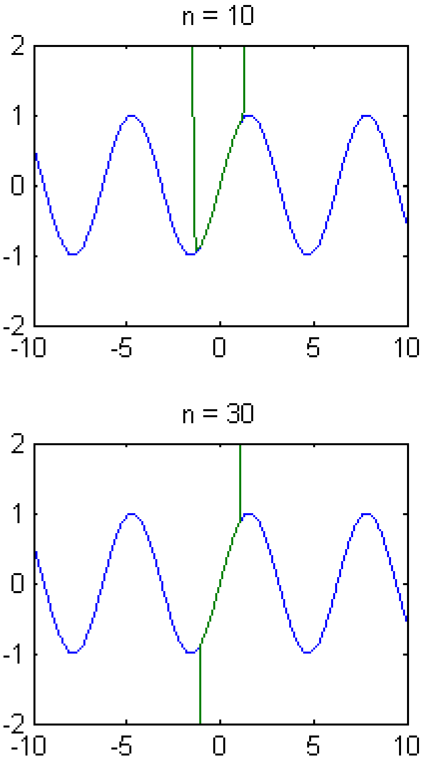

Figure 3: The degree-$2n$ regression curves $y(x)$ fitting $f(x) = \sin x$ on T_n for $n = 10, 20, 30$.

(e) We can try to fix the problem found in part (d) by fixing the degree of our model to be 10, say. That is, let's fix our model as

$$y(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_{10} x^{10} = \sum_{k=0}^{10} a_k x^k.$$

Now append to your script yet another script that computes the degree-10 curve of best fit for $f(x)$ on T_n when $n > 10$ via the normal equations for the resulting linear system on the a_k variables (don't use MATLAB's backslash operation!). Use your script to create the degree-10 curve of best fit for $f(x)$ on T_n in the cases when $n = 20, 30, 40$. Does this new script fix the problem in part (d)? Did we introduce yet more problems with this third script?

Here is the script use to generate the plots in Figure 4:

```
n = 15;

while n <= 60
    C = zeros(2*n+1,11);
    y = zeros(1,length(x));

    for jj = 0:10
        C(:,jj+1) = linspace(-1,1,2*n+1)'.^jj;
    end

    f = sin(linspace(-1,1,2*n+1)');
    b = linsolve(C'*C,C'*f);

    for ii = 1:length(x)
        for jj = 1:10
            y(ii) = y(ii) + b(jj)*x(ii)^(jj-1);
        end
    end

    figure(3);
    subplot(2,2,n/15);
    plot(x,sin(x),x,y)
    title(['n = ',num2str(n)])
    ylim([-2,2])

    n = n+15;
end
```

It would appear that we can indeed *extrapolate* a little bit beyond the previous interval of $[-1, 1]$! This doesn't quite fix the original problem, but it's a good first step in determining how we could proceed further in gaining better approximations. And no, it doesn't look like any new problems were introduced in the process!

(f) Suggest another change that we could make to our approximation scheme that you think would help prove your conjecture in part (b).

There are many acceptable responses. Some ideas might include, but are not limited to:

- Increasing n to larger values and consider how much further along $f(x) = \sin x$ we can approximate by only using the points in T_n.
- Choosing a larger, fixed degree for the polynomial model, say degree-20 or higher.

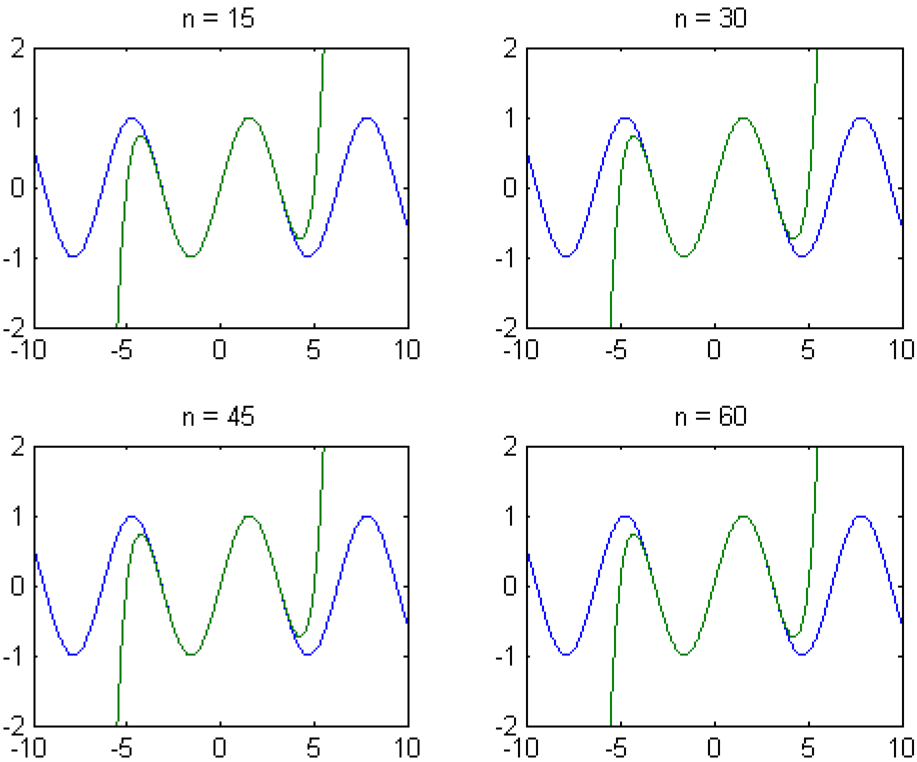

Figure 4: The degree-10 regression curves $y(x)$ fitting $f(x) = \sin x$ on T_n for $n = 15, 30, 45, 60$.

- Going back to using S_n to see what the curve of best fit looks like when we over-determine the regression curve.

Please submit all three parts of your script.

Problem 2. *The uncanny sameness of finite-dimensional vector spaces*
One of the reasons we like finite-dimensional vector spaces so much is because of how similar they are to one another. Functions, matrices, signals... They all act like column vectors when we use them. They sure "look" and "feel" like column vectors, don't they? Let's formalize this a bit, shall we?

(a) Prove the following theorem:

Theorem 1. Let $(V, \mathbb{F})$ be a vector space with $\dim(V) = n < \infty$. Then $V \simeq \mathbb{F}^n$.

This shows that "all n-dimensional vector spaces are really the same."

The proof looks very similar, if not identical, to the one for *real* vector spaces shown in lecture:

Proof. Suppose $\dim(V) = n$ and that $\beta = \{\vec{\mathbf{v}}_1, \ldots, \vec{\mathbf{v}}_n\} \subset V$ is a basis for V. We claim that the coordinate map $[\cdot]_\beta \colon V \to \mathbb{F}^n$ is an isomorphism.

Consider the function $L\colon \mathbb{F}^n \to V$ given by

$$L[\vec{\mathbf{c}}] = \sum_{k=1}^{n} c_k \vec{\mathbf{v}}_k, \quad \text{for} \quad \vec{\mathbf{c}} = \begin{bmatrix} c_1 \\ \vdots \\ c_n \end{bmatrix}.$$

We claim that $L = [\cdot]_\beta^{-1}$.

Indeed, if $\vec{\mathbf{v}} = \sum_{k=1}^{n} c_k \vec{\mathbf{v}}_k \in V$, then consider the composition

$$L \circ [\vec{\mathbf{v}}]_\beta = L \begin{bmatrix} c_1 \\ \vdots \\ c_n \end{bmatrix} = \sum_{k=1}^{n} c_k \vec{\mathbf{v}}_k = \vec{\mathbf{v}}$$

so that $L \circ [\cdot]_\beta = Id_V$; and for any $\vec{\mathbf{c}} = (c_1, \ldots, c_n)^T \in \mathbb{F}^n$ we have

$$[L[\vec{\mathbf{c}}]]_\beta = \left[\sum_{k=1}^{n} c_k \vec{\mathbf{v}}_k \right]_\beta = \begin{bmatrix} c_1 \\ \vdots \\ c_n \end{bmatrix} = \vec{\mathbf{c}}$$

so that $[\cdot]_\beta \circ L = Id_{\mathbb{F}^n}$. Therefore, $[\cdot]_\beta$ is indeed a linear isomorphism from V to $\mathbb{F}^n$, and this shows that $V \simeq \mathbb{F}^n$.

$\square$

(b) Show that this is false in the case where $\dim(V) = \infty$; that is, find two vector spaces V and W that both have infinite dimension but are not isomorphic. [*Hint*: You may use the fact that the integers $\mathbb{Z}$ and the real numbers $\mathbb{R}$ have different sizes, even though they are both infinite.]

Consider the function spaces

$$\mathcal{F}(\mathbb{Z}) = \{f\colon \mathbb{Z} \to \mathbb{R}\} \quad \text{and} \quad \mathcal{F}(\mathbb{R}) = \{f\colon \mathbb{R} \to \mathbb{R}\}.$$

It has been shown in class that these are both infinite-dimensional vector spaces.

First, consider the functions

$$f_\omega(x) = \begin{cases} 1, & \text{if } x = \omega \\ 0, & \text{if } x \neq \omega \end{cases}.$$

We define the sets

$$\beta = \{f_k\}_{k\in\mathbb{Z}} \quad \text{and} \quad \gamma = \{f_c\}_{c\in\mathbb{R}}.$$

Observe that these are *minimal spanning sets* for $\mathcal{F}(\mathbb{Z})$ and $\mathcal{F}(\mathbb{R})$, respectively, since any function $f \in \mathcal{F}(\mathbb{Z})$ can be represented as the formal sum

$$f(x) = \sum_{k=-\infty}^{\infty} f(k) f_k(x)$$

and any function $f \in \mathcal{F}(\mathbb{R})$ can be represented as the formal sum

$$f(x) = \sum_{c \in \mathbb{R}} f(c) f_c(x).$$

In particular, we cannot omit any functions from either β or γ, lest we are unable to recreate the function $f(x)$ at one point (this is what makes these sets *minimal*). In this sense, β and γ are "bases" $\mathcal{F}(\mathbb{Z})$ and $\mathcal{F}(\mathbb{R})$, respectively.

Because of the allowed assumption that $\mathbb{Z}$ and $\mathbb{R}$ are different cardinalities (sizes) of infinity, we see that these sets β and γ have different sizes as well. Therefore, there is no way to map the "basis" β to the "basis" γ in an isomorphic fashion. Therefore, $\mathcal{F}(\mathbb{Z}) \not\cong \mathcal{F}(\mathbb{R})$.

Problem 3. *Sturm-Liouville, part II*

Recall the general Sturm-Liouville equation from the previous Problem Set:

$$-\frac{d}{dx}\left(p(x)\frac{dy}{dx}\right) + q(x)y = \lambda r(x)y,$$

where $p, q, r \in C^1[a, b]$ with $p(x), q(x), r(x) > 0$ for $a < x < b$ and $\lambda \in \mathbb{R}$ (note the leading minus sign now). For simplicity, we also stipulate that the *solutions* to this differential equation must satisfy the "pinned" *Dirichlet boundary condition*

$$y(a) = y(b) = 0.$$

Our goal for this exercise is to build up to the *weak formulation* of this differential equation. This allows us to frame the *solutions* to this differential equation as those functions that minimize a particular quadratic function. That is, we are going to show that this equation *always* has a *unique* solution satisfying the boundary conditions.

We first define the linear function $L\colon C_0^\infty[a, b] \to (C_0^\infty[a, b])^2$ as

$$L[u] = \begin{pmatrix} u' \\ u \end{pmatrix};$$

that is, L sends a smooth function (that is pinned at $x = a$ and $x = b$) to the vector containing its derivative and itself (in that order). If we take

$$\vec{\mathbf{f}} = \begin{pmatrix} f_1(x) \\ f_2(x) \end{pmatrix} \quad \text{and} \quad \vec{\mathbf{g}} = \begin{pmatrix} g_1(x) \\ g_2(x) \end{pmatrix},$$

define the weighted inner product on $(C_0^\infty[a, b])^2$ as

$$\langle \vec{\mathbf{f}}, \vec{\mathbf{g}} \rangle = \int_a^b [p(x) f_1(x) g_1(x) + q(x) f_2(x) g_2(x)]\, dx.$$

We use $\langle\langle \cdot, \cdot \rangle\rangle$ to denote the standard L^2-inner product on $C_0^\infty[a, b]$.

(a) Show that the adjoint L^* of L with respect to these inner products is given by

$$\begin{aligned} L^* \begin{pmatrix} h_1 \\ h_2 \end{pmatrix} &= -\frac{d(p(x)h_1(x))}{dx} + q(x)h_2(x) \\ &= -p(x)h_1'(x) + p'(x)h_1(x) + q(x)h_2(x). \end{aligned}$$

Let $\vec{\mathbf{f}} = (f_1(x), f_2(x))^T \in (C_0^\infty[a,b])^2$ and $g \in C_0^\infty[a,b]$. Then consider

$$\begin{aligned}
\langle\langle L^*[\vec{\mathbf{f}}], g\rangle\rangle &= \int_a^b \left(-\frac{d}{dx}(p(x)f_1(x)) + q(x)f_2(x)\right) g(x)\, dx \\
&= \int_a^b -\frac{d}{dx}(p(x)f_1(x))g(x)\, dx + \int_a^b q(x)f_2(x)g(x)\, dx \\
&= -[p(x)f_1(x)g(x)]_a^b + \int_a^b p(x)f_1(x)g'(x)\, dx + \int_a^b q(x)f_2(x)g(x)\, dx \\
&= \int_a^b p(x)f_1(x)g'(x) + q(x)f_2(x)g(x)\, dx \\
&= \langle \vec{\mathbf{f}}, L[g]\rangle,
\end{aligned}$$

where we used the condition $g(a) = g(b) = 0$ after the integration-by-parts step. Therefore, the linear function L^* defined above behaves like the adjoint for L. Because $\vec{\mathbf{f}}$ and g were arbitrarily chosen, this means that the defined function L^* is indeed the adjoint of L.

(b) We know that $S = L^* \circ L$ is self-adjoint. Show that

$$S[u] = -\frac{d}{dx}\left(p(x)\frac{du}{dx}\right) + q(x)u(x)$$

so that we see the Sturm-Liouville equation as the linear equation

$$S[u] = \lambda r(x)u(x).$$

We merely show that the composition $L^* \circ L$ produces the left-hand side of the Sturm-Liouville equation. Indeed, for $u \in C_0^\infty[a,b]$, we have

$$(L^* \circ L)[u] = L^* \begin{bmatrix} u' \\ u \end{bmatrix} = -\frac{d}{dx}\left(p(x)u'(x)\right) + q(x)u(x).$$

(c) Use the Minimization Theorem for linear functions to show that the solution to the Sturm-Liouville equation is given as the unique minimizer of the quadratic function

$$Q[u] = \frac{1}{2}\|L[u]\|^2 - \langle\langle \lambda r u, u\rangle\rangle,$$

where $\|\cdot\|$ is induced by $\langle\cdot,\cdot\rangle$, but $\langle\langle\cdot,\cdot\rangle\rangle$ is the standard L^2 inner product on $[a,b]$.

Suppose indeed that $u \in C_0^\infty[a,b]$ is a solution to the Sturm-Liouville equation. To meet the conditions of the General Minimization Theorem for the function Q, we need to guarantee that (1) $S = L^* \circ L$ is positive-definite and (2) $r(x)u(x) \in \mathrm{im} S$.

Condition (1) happens if and only if $\ker L$ is trivial. A function $f \in C_0^\infty[a,b]$ is in $\ker L$ if $L[f] = (f', f)^T = (0,0)^T$. This is the case only if $f(x) = \alpha + \beta x$, and the conditions $f(a) = f(b) = 0$ implies that $\alpha = \beta = 0$. Hence, $f \in \ker L$ if and only if $f \equiv 0$, and we have shown $\ker L$ is trivial. This shows that the minimizer of Q is unique when it exists.

Condition (2) requires $S[u] = \lambda r(x)u(x)$. This is guaranteed because u was assumed to be a solution to the Sturm-Liouville equation.

The General Minimization Theorem then shows that a minimizer exists for Q and that it is unique. We conclude that the (unique) solution to the "pinned" Sturm-Liouville equation is the unique minimizer of the quadratic *energy functional* Q.