MATH661 Homework 2 - Multivariate approximation

Posted: Sep 6 Due: 11:55PM, Sep 20

1 Problem statement

Many physical phenomena are determined by the shape of surfaces separating one domain from another. Examples include:

- flow around a body;
- inclusions of one phase in another, e.g., air bubble in water;
- accumulation of defects in a crystal at boundaries, e.g., formation of metal grains.

A first step in modeling such phenomena is to approximate the bounding surface in an accurate and computationally efficient manner. The parametric form of the surface is $f\colon \mathbb{R}^2 \to \mathbb{R}^3$, with components $(f_1(u,v), f_2(u,v), f_3(u,v))$, and the goal is to seek an approximation $(g_1(u,v), g_2(u,v), g_3(u,v))$. The approximation will be constructed as a superposition of basis functions $B_{jk}(u,v)$

$$g(u,v) = \sum_{l=0}^{p} \sum_{m=0}^{q} c_{lm} B_{lm}(u,v). \tag{1}$$

Formula (1) generalizes the univariate case, e.g., in Newton's form of the global interpolating polynomial

$$p_n(x) = \sum_{i=0}^{n} c_i B_i(x), c_i = [f_0, \ldots, f_i], B_i(x) = \prod_{j=0}^{i-1} (x - x_j). \tag{2}$$

A commonly encountered problem is time evolution of a bounding surface, which can be described through a separation of variables approach as

$$g(t,u,v) = \sum_{l=0}^{p} \sum_{m=0}^{q} c_{lm}(t) B_{lm}(u,v). \tag{3}$$

A simple example is a sphere with time varying radius $r(t)$,

$$f(t,\theta,\varphi) = r(t)(\cos\theta \sin\varphi, \sin\theta \sin\varphi, \cos\varphi),$$

for which $p = q = 0$, $c_{00}(t) = r(t)$, $B_{00}(\theta,\varphi) = (\cos\theta \sin\varphi, \sin\theta \sin\varphi, \cos\varphi)$. A more interesting example is a superposition of spherical harmonics as would occur in the oscillation of a liquid droplet, or the change in an equipotential surface in quantum mechanical descriptions of an atom

$$r(t,\theta,\varphi) = \sum_{l=0}^{p} \sum_{m=-l}^{l} c_{lm}(t) Y_l^m(\theta,\varphi), Y_l^m(\theta,\varphi) = e^{im\varphi} P_l^m(\cos\theta)$$

with $P_l^m(\cos\theta)$ denoting the associated Legendre polynomials defined in terms of the ordinary Legendre polynomials $P_l(x)$ by

$$P_l^m(\cos\theta) = (-1)^m (1-x^2)^{m/2} \frac{\mathrm{d}^m}{\mathrm{d}x^m} P_l(x).$$

This homework will investigate ways of approximating the spherical harmonics $Y_l^m(\theta,\varphi)$, in conjunction with the univariate approximation of univariate functions $c_{lm}(t)$. The theoretical exercises build up the foundation of construction of orthogonal polynomial bases. The computational part uses these results to approximate the spherical harmonics, and produce physically relevant predi- tions.

2 Theoretical exercises

1. K&C, 6.8.1, p.404

   Solution. A series of plots for various values of λ is helpful:

   Mathematica

```
In[8]:= lambda=Table[lam,{lam,0,1,.2}];
        p=Plot[{Sin[x],lambda x},{x,0,Pi/2}];
        SetDirectory["/home/student/courses/MATH661/homework"];
        Export["HW2Fig1.png",p]
```

   HW2Fig1.png

```
In[9]:=
```

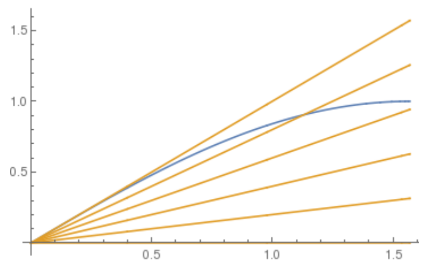

Figure 1. $\sin(x)$ (blue), λx (yellow) for $\lambda = 0, 0.2, ..., 1$

Introduce $F(t,\lambda) = \sin t - \lambda t$, and apply Chebyshev alternation theorem to obtain system

$$\begin{array}{lll} F(\xi,\lambda) = -F\left(\frac{\pi}{2},\lambda\right) \Rightarrow & \sin\xi - \lambda\xi = -\left(1-\lambda\frac{\pi}{2}\right) \\ \frac{\partial F}{\partial t}(\xi,\lambda) = 0 \Rightarrow & \cos\xi - \lambda = 0 \end{array} \Rightarrow \sin\xi - \xi\cos\xi = \frac{\pi}{2}\cos\xi - 1 \Rightarrow$$

$$\left(\xi + \frac{\pi}{2}\right)\cos\xi - \sin\xi = 1 \tag{4}$$

(Full credit for obtaining above system of equations). Solution of (4) can be determined numerically (Lesson 12)

```
In[17]:= csi = csi /. FindRoot[(csi+Pi/2) Cos[csi] - Sin[csi] == 1,{csi,Pi/
         3}][[1]]
```

0.760326

```
In[18]:= lambda = Cos[csi]
```

0.724611

```
In[24]:= p=Plot[{Sin[x],lambda x},{x,0,Pi/2},AxesLabel->{"x","sin(x),
         lambda  x"},GridLines->{{csi,Pi/2},None}];
         SetDirectory["/home/student/courses/MATH661/homework"];
         Export["HW2Fig2.png",p]
```

HW2Fig2.png

```
In[25]:=
```

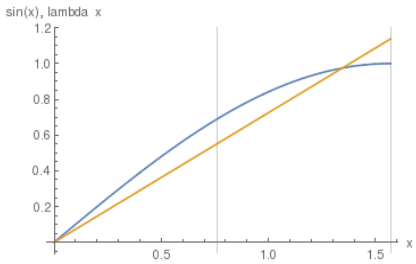

Figure 2. Best approximant of $\sin x$ by linear function is $0.724611x$.

2. K&C, 6.8.2, p.404 (include a plot comparing exact function and approximations from exercises 1 and 2)

   Solution. Using same notation as above, the quadratic norm best approximant is the solution of the problem

$$\min_{\lambda} g, \text{ with}$$
$$g(\lambda) = \|F\|_2 = \int_0^{\pi/2} (\sin x - \lambda x)^2 \, \mathrm{d}x \text{ ,}$$

   Solve $g'(\lambda) = 0$,

$$\int_0^{\pi/2} -x(\sin x - \lambda x) \, \mathrm{d}x = 0 \Rightarrow \lambda = \left[\int_0^{\pi/2} x \sin x \, \mathrm{d}x\right] \left[\int_0^{\pi/2} x^2 \, \mathrm{d}x\right]^{-1} \Rightarrow$$

$$\lambda = 24/\pi^3$$

```
In[27]:= p=Plot[{Sin[x],lambda x, 24 x/Pi^3},{x,0,Pi/2},AxesLabel->{"x",
         "sin(x), lambda  x"},GridLines->{{csi,Pi/2},None}];
         SetDirectory["/home/student/courses/MATH661/homework"];
         Export["HW2Fig3.png",p]
```

HW2Fig3.png

```
In[28]:=
```

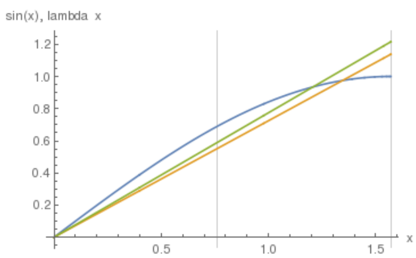

Figure 3. Comparison of $\| \|_2$, $\| \|_\infty$ best approximants of $\sin x$.

3. K&C, 6.8.5, p.404

Solution. Consider the expansions of f, g on the orthonormal set $\mathcal{U} = \{u_1, ..., u_n\}$

$$f = a_1 u_1 + ... + a_n u_n, g = b_1 u_1 + ..., + b_n u_n, a_i, b_i \in \mathbb{R}$$

Since $\mathcal{U}$ is an orthonormal set obtain

$$\begin{array}{rcl} \langle f, u_j \rangle &=& \langle a_1 u_1 + ... + a_n u_n, u_j \rangle &=& a_j, \\ \langle g, u_j \rangle &=& \langle b_1 u_1 + ..., + b_n u_n, u_j \rangle &=& b_j, \\ \langle f, g \rangle &=& \left\langle \sum_{j=1}^{n} a_j u_j, \sum_{k=1}^{n} b_k u_k \right\rangle &=& \sum_{j=1}^{n} \sum_{k=1}^{n} a_j b_k \langle u_j, u_k \rangle \\ &=& \sum_{j=1}^{n} a_j b_j & & \checkmark \end{array}$$

4. K&C, 6.8.21, p.405

Solution. Apply the Gram-Schmidt orthogonalization procedure on $\{1, x, x^2, x^3, x^4, x^5\}$ with scalar product

$$\langle f, g \rangle = \int_{-1}^{1} f(x)\, g(x)\, \mathrm{d}x$$

Using code from Lesson 6

```
In[29]:= ScProd[f_,g_,w_,x_,a_,b_]:=Integrate[w f g,{x,a,b}];
         GS[u_,w_,x_,a_,b_]:=Module[{q={},n=Length[u],v=u,r},
           r=IdentityMatrix[n];
           For [i=1, i<=n, i++,
             r[[i,i]] = ScProd[v[[i]],v[[i]],w,x,a,b]^(1/2); AppendTo[q,
         v[[i]]/r[[i,i]] ];
             For [j=i+1, j<=n, j++,
               r[[i,j]]=ScProd[q[[i]],v[[j]],w,x,a,b]; v[[j]]=v[[j]]-r[[i,
         j]] q[[i]] ]];
           Return[Simplify[q]] ];
         Legendre = GS[{1,x,x^2,x^3,x^4,x^5},1,x,-1,1]
```

$$\left\{\frac{1}{\sqrt{2}}, \sqrt{\frac{3}{2}}\, x, \frac{1}{2}\sqrt{\frac{5}{2}}\,(3\,x^2-1), \frac{1}{2}\sqrt{\frac{7}{2}}\, x\,(5\,x^2-3), \frac{3\,(35\,x^4-30\,x^2+3)}{8\sqrt{2}}, \frac{1}{8}\sqrt{\frac{11}{2}}\, x\,(63\,x^4 - 70\,x^2+15)\right\}$$

```
In[30]:= p=Plot[Legendre,{x,-1,1}];
         SetDirectory["/home/student/courses/MATH661/homework"];
         Export["HW2Fig4.png",p]
```

HW2Fig4.png

```
In[31]:=
```

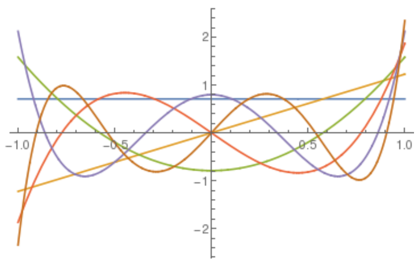

Figure 4. Orthonormal Legendre polynomials

The monic form of the Legendre polynomials from the text is

```
In[33]:= TableForm[Expand[Legendre / Table[Coefficient[Legendre[[k+1]],x,
         k],{k,0,5}]]]
```

$$1$$
$$x$$
$$x^2-\frac{1}{3}$$
$$x^3-\frac{3\,x}{5}$$
$$x^4-\frac{6\,x^2}{7}+\frac{3}{35}$$
$$x^5-\frac{10\,x^3}{9}+\frac{5\,x}{21}$$

```
In[34]:=
```

5. K&C, 6.9.8, p.420. Use both elementary calculus and application of Chebyshev alternation theorem

   Solution. Since $\cosh(x)$ is an even function, the problem can be stated as

$$\min_{a,b}\max_{0\leqslant x\leqslant 1}|\varepsilon(x)|, \varepsilon(x)=a+bx^2-\cosh(x).$$

   Extrema of $\varepsilon\colon[0,1]\to\mathbb{R}$ are either endpoint values $\varepsilon(0)$,$\varepsilon(1)$ (since $[0,1]$ is compact), or local maxima/minima $\varepsilon(t)$, with t solutions within (0,1) of

$$\varepsilon'(x)=2bx-\sinh(x)=0.$$

   The possible extrema are therefore

$$y_1(a)=\varepsilon(0)=a-1, y_2(a,b)=\varepsilon(1)=a+b-\cosh(1),$$

$$y_3(a,b)=\varepsilon(t)=a+bt^2-\cosh t, 2bt-\sinh(t)=0.$$

   From the Chebyshev alternating theorem (equioscillation theorem)

$$y_1(a)=-y_3(a,b)=y_2(a,b),$$

   and the best inf-norm approximant is $b=\cosh(1)-1$, and a from solution of system

$$\begin{cases} 2a=1-bt^2+\cosh t\\ 2bt=\sinh(t)\\ b=\cosh(1)-1\end{cases}.$$

6. K&C, 6.10.17, p. 438.

   Solution. Introduce basis sets $\{a_1(x),...,a_p(x)\}$, $\{b_1(y),...,b_q(y)\}$ of U,V respectively. The dimension of a vector (linear) space is equal to the number of basis vectors (functions). A basis set for $U\otimes V$ is $\mathcal{B}=\{a_1(x)b_1(y),...,a_1(x)b_q(y),a_2(x)b_1(y),...,a_p(x)b_q(y\}$, with pq elements, since elements of $\mathcal{B}$ are independent, and any $w(x,y)=\sum_{i=1}^p\sum_{j=1}^q w_{ij}a_i(x)b_j(y)$

3 Computational application

The following tasks will be fully formulated and partially solved in classroom work.

1. Generate data. Choose p, $q \in \{2, 3, 4, 5, 6\}$, $q \leqslant p$, and coefficients $c_{l m}(t) = a_{l m} \sin(\omega_{l m} t)$, $a_{l m}$ random numbers uniformly distributed in [0,1], $\omega_{l m} = 2\pi(l + m)$, $l = 0, \ldots, p$, $m = 0, \ldots, q$. Compute the positions of points on the surface at times $t_l = l\delta t$, $\delta t = 1/N$, $N = 32$, $l = 0, 1, \ldots, N$, and angles $\theta_j = 2\pi j / P$, $\varphi_k = \pi k / P$, $P = 18$.

   Solution. Generate the random mode amplitudes

```
Python] from pylab import *;
Python] p=3; q=3; c=random((p+1,q+1)); print c
   [[ 0.00201693  0.86704318  0.00954874  0.93731186]
    [ 0.52890472  0.69028659  0.43355757  0.42248209]
    [ 0.14867015  0.43640441  0.70319709  0.38934012]
    [ 0.68919168  0.87405977  0.26983005  0.57376622]]
Python] omega=zeros((p+1,q+1));
        for l in range(p+1):
          for m in range(q+1):
            omega[l,m]=2*pi*(l+m)
Python]
```

   Load 3D plotting routines

```
Python] from mpl_toolkits.mplot3d import Axes3D
Python] from matplotlib import cm, colors
Python]
```

   Load the spherical harmonics library (1 bonus point awarded for coding of spherical harmonics instead of loading a library) , and evaluate the spherical harmonics at requested θ, φ values.

```
Python] from scipy.special import sph_harm
Python] N=32; t=linspace(0.,1.,N+1)
Python] P=18; theta=linspace(0.,pi,P+1); phi=linspace(0.,2*pi,P+1)
Python] phi,theta=meshgrid(phi,theta)
Python] B=zeros((p+1,q+1,P+1,P+1))
Python] for l in range(p+1):
          for m in range(l+1):
            B[l,m] = sph_harm(m,l,theta,phi).real
Python] any(isnan(B))
    False
Python]
```

   Plot some of the spherical harmonics

```
Python] fig=plt.figure(figsize=plt.figaspect(1.))
Python] xS=sin(phi)*cos(theta);
Python] yS=sin(phi)*sin(theta);
Python] zS=cos(phi);
Python] ax=fig.add_subplot(111,projection='3d')
        ax.plot_surface(xS,yS,zS,rstride=1,cstride=1)
        show();
```

```
Python] Bcolors=zeros(B.shape);
        for l in range(p+1):
          for m in range(l+1):
            mxB=B[l,m].max(); mnB=B[l,m].min();
            Bcolors[l,m] = (B[l,m]-mnB)/max((mxB-mnB),B[l,m,0,0])
Python] fig=plt.figure(figsize=plt.figaspect(1.))
        ax=fig.add_subplot(111,projection='3d')
        ax.plot_surface(xS,yS,zS,rstride=1,cstride=1,
        facecolors=cm.rainbow(Bcolors[1,1]))
        show();
Python] fig=plt.figure(figsize=plt.figaspect(1.))
        ax=fig.add_subplot(111,projection='3d')
        ax.plot_surface(xS,yS,zS,rstride=1,cstride=1,
        facecolors=cm.rainbow(Bcolors[2,1]))
        show();
Python]
```

The plots were save to disk and are show in Fig. 1.

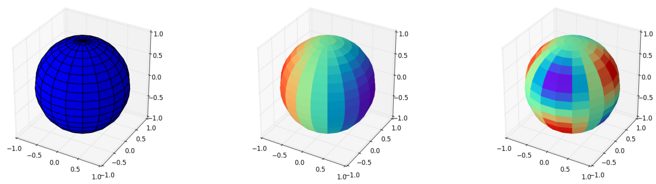

Figure 5.

Some of the spherical harmonics

Generate the surface at the requested times

```
Python] r=zeros((N+1,P+1,P+1));
Python] for n in range(N+1):
          for l in range(p+1):
            for m in range(l+1):
              r[n] = r[n] + c[l,m]*B[l,m]
Python]
```

Transform the spherical coordinates to Cartesian coordinates for plotting

```
Python] x=zeros(r.shape); y=zeros(r.shape); z=zeros(r.shape)
Python] for n in range(N+1):
          x[n] = r[n]*sin(phi)*cos(theta)
          y[n] = r[n]*sin(phi)*sin(theta)
          z[n] = r[n]*cos(phi)
Python] fig=figure(figsize=plt.figaspect(1.))
        ax=fig.add_subplot(111,projection='3d')
        ax.plot_surface(x[0],y[0],z[0],rstride=1,cstride=1)
        show();
Python]
```

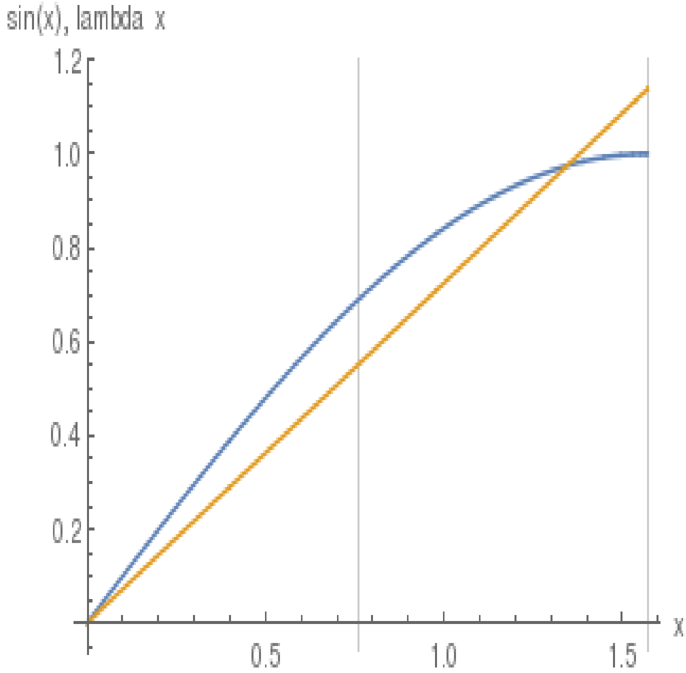

Figure 6. Surface at $t = 0$.

2. Investigate recovery of time-varying coefficients by projection onto the spherical harmonic basis.

3. Construct a piecewise linear approximation of the spherical harmonics.

4. Investigate approximation of the time-varying coefficients by projection of the data onto the piecewise linear basis. Compare to exact coefficients.