Homework 1

1 Note that

$$\|x-y\|^2 =< x-y, x-y >=< x,x > -2 < y,x > + < y,y >$$

Similary,

$$\int_0^1 (x(t)-y(t))^2 dt = \int_0^1 x(t)^2 dt - 2\int_0^1 x(t)y(t)dt + \int_0^1 y(t)^2 dt$$

Hence, we can deduce that inner product that induces the above norm is

$$< x,y >= \int_0^1 x(t)y(t)dt$$

Thus to get the coefficients for the basis $B = \{1, t, t^2\}$ we need to solve the system

$$Ga = x$$

for a, where $G_{ij} = \int_0^1 B_i B_j dt$ and $x_i = \int_0^1 e^t B_i dt$.

In **R**, the following lines of code solves this system for us:

```
A[1,1] = integrate(function(t){rep(1,length(t))}, 0, 1)$value
A[2,1] = A[1,2] = integrate(function(t){t}, 0, 1)$value
A[3,1] = A[1,3] = integrate(function(t){(t^2)}, 0, 1)$value
A[3,2] = A[2,3] = integrate(function(t){(t^3)}, 0, 1)$value
A[2,2] = integrate(function(t){(t^2)}, 0, 1)$value
A[3,3] = integrate(function(t){(t^4)}, 0, 1)$value

b <- matrix(0,nrow = 3, ncol = 1)
b[1] = integrate(function(t){(exp(t))}, 0, 1)$value
b[2] = integrate(function(t){(exp(t))*t}, 0, 1)$value
b[3] = integrate(function(t){(exp(t))*(t^2)}, 0, 1)$value

a = solve(A,b)
```

Figure 5 shows us a comparison of how the method described above is superior to the Taylor expansion in estimating e^t for $t \in [0,1]$. To further support this, we also find that $\|x-\hat{x}\| = 2.783544e^{-05}$ whereas $\|x-x_{Taylor}\| = 0.006349231$.

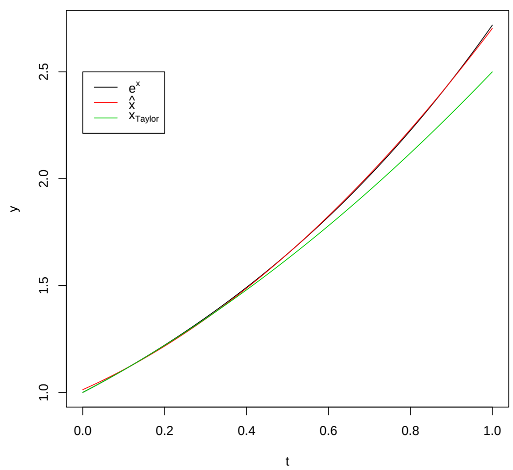

Figure 1: Plot of $e^t, \hat{x}$ and x_{Taylor}. It is clear that $\hat{x}$ is superior to x_{Taylor}.

Finally, we repeat the exercise with $\|x - y\| = \int_0^1 16(t-1/2)^2(x(t) - y(t))^2 dt$. Note that

$$\int_0^1 16(t-1/2)^2(x(t) - y(t))^2 dt$$
$$= \int_0^1 16(t-1/2)^2(x(t)^2 - 2x(t)y(t) - y(t)^2) dt$$
$$= \int_0^1 16(t-1/2)^2 x(t)^2 dt - 2\int_0^1 16(t-1/2)^2 x(t)y(t) dt - \int_0^1 16(t-1/2)^2 y(t)^2 dt$$

Thus we deduce $< x, y >= \int_0^1 16(t-1/2)^2 x(t)y(t)dt$. In this case, the following **R** code solves the required linear system:

```
A1 <- matrix(0,nrow = 3, ncol = 3)
A1[1,1] = integrate(function(t){16*(t-1/2)^2*1}, 0, 1)$value
A1[2,1] = A1[1,2] = integrate(function(t){16*(t-1/2)^2*t}, 0, 1)$value
A1[3,1] = A1[1,3] = integrate(function(t){16*(t-1/2)^2*(t^2)}, 0, 1)$value
A1[3,2] = A1[2,3] = integrate(function(t){16*(t-1/2)^2*(t^3)}, 0, 1)$value
A1[2,2] = integrate(function(t){16*(t-1/2)^2*(t^2)}, 0, 1)$value
A1[3,3] = integrate(function(t){16*(t-1/2)^2*(t^4)}, 0, 1)$value
```

```
b1 <- matrix(0,nrow = 3, ncol = 1)
b1[1] = integrate(function(t){16*(t-1/2)^2*(exp(t))}, 0, 1)$value
b1[2] = integrate(function(t){16*(t-1/2)^2*(exp(t))*t}, 0, 1)$value
b1[3] = integrate(function(t){16*(t-1/2)^2*(exp(t))*(t^2)}, 0, 1)$value

a1 = solve(A1,b1)
```

Figure 2 shows how close these two estimates are in predicting e^t. In addition, in this case $\|x - \hat{x}_1\| = 4.42977e^{-05}$, which is virtually the same as the error from $\hat{x}$. Hence, changing the inner product in this case doesn't affect the best linear approximation too much.

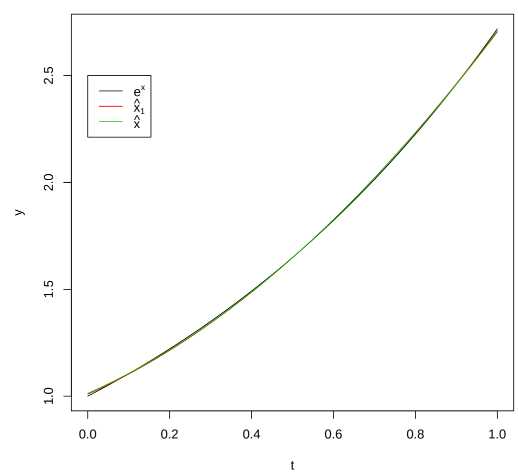

Figure 2: Plot of $e^t, \hat{x}$ and $\hat{x}_1$. Both of them perform about the same.

2 First note that

$$\begin{aligned}
&2\int_0^1 \cos(\pi n t)\cos(\pi n' t)dt \\
&= \int_0^1 \cos(\pi(n-n')t) + \cos(\pi(n+n')t)dt \\
&= [\frac{\sin(\pi(n-n')t)}{\pi(n-n')} + \frac{\sin(\pi(n+n')t)}{\pi(n+n')}]_0^1 \\
&= \begin{cases} 1 \text{ if } n = n' \\ 0 \text{ if } n \neq n' \end{cases}
\end{aligned}$$

We can also show that

$$\sum_{i=1}^{N} v_n(i)v_{n'}(i) = \begin{cases} 1 \text{ if } n = n' \\ 0 \text{ else} \end{cases}$$

Note that $v_0(i)v_0(i) = 1/N \implies \sum_{i=1}^{N} v_0(i)v_0(i) = 1$. Now note that

$$\begin{aligned}
\sum_{i=1}^{N} v_n(i)v_n(i) &= 2/N \sum_{i=1}^{N} \cos^2(\pi n \frac{i-1/2}{N}) \\
&= 2/N(1/4)(\sin(2\pi n)\csc(\pi n/N) + 2N) \\
&= 1
\end{aligned}$$

Similarly, $\sum_{i=1}^{N} v_n(i)v_{n'}(i) = 0$ when $n \neq n'$.

Basically the procedure we would use is to find the orthogonal projection onto the $k-$dimensional subspace spanned by $\mathcal{T}$ where k is a much smaller number than N. Doing this in a linear fashion involves finding the best fit coefficients for the first k basis vectors of the space we are interested in and discarding the rest.

The pictures shown in Figures 3a, 3b and 3c are used for this exercise. The angle between the images in 3a and 3b 0.546625, while the angle between 3b and 3c is 0.7093961 and the angle between 3a and 3c is 0.6004843. As expected the smallest angle is between the images of the flowers even though it is quite clear that they are very different pictures.

(a) (b) (c)

Figure 3: Pictures used for analysis with DCT basis

It is simple to show that

$$< \phi_{m,n}, \phi_{m,n} >= \sum_{i=1}^{N} \sum_{j=1}^{N} v_n(i)^2 v_m(j)^2 = \sum_{i=1}^{N} v_n(i)^2 \sum_{j=1}^{N} v_m(j)^2 = 1$$

by the previous part. Using the same part and the orthogonality of $v_n(i)$ and $v_{n'}(i)$ when $n \neq n'$, we can show

$$< \phi_{m,n}, \phi_{m,n} >= \sum_{i=1}^{N} v_n(i) v_{n'}(i) \sum_{j=1}^{N} v_m(j) v_{m'}(j) = 0$$

, i.e. the 2D DCT basis is orthogonal.

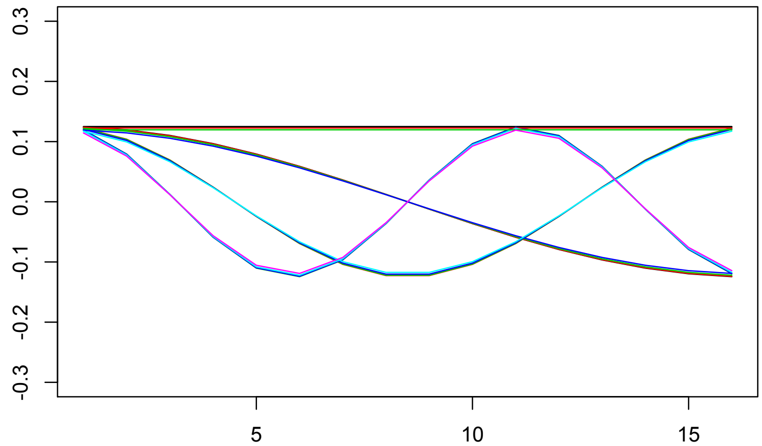

Figure 4: Basis Plots

Finally, to construct $\hat{x}_{25\%}$ and for comparing the norm errors versus k, the number of basis elements we keep, we use the following **R** code as instructed:

```
numberk = 10
norm = rep(0,numberk)
k = floor(seq(20,100,length=numberk))
for (kk in 1:numberk){
    Aflower = matrix(0,nrow=k[kk],ncol=k[kk])
    for(n in 1:k[kk]){
        for(m in 1:k[kk]){
            Aflower[m,n] =
sum(outer(vni(m,1:size,size),vni(n,1:size,size))*flower)
        }
    }

    xhatflower = matrix(0,nrow=size,ncol=size)
    for(n in 1:k[kk]){
        for(m in 1:k[kk]){
```

```
            xhatflower = xhatflower +
Aflower[m,n]*outer(vni(m,1:size,size),vni(n,1:size,size))
        }
    }
    display(channel(xhatflower,"gray"))
    norm[kk] = sqrt(sum((flower - xhatflower)^2))
}
```

The original images were rediced to 200×200. As $N = 200$, to get 25% we only need to go up to $N/2 = 100$. Here we start at very sparse solutions and only go up to 25% as the most densely packed image. It is clear from the plots that $\|x - \hat{x}_k\|$ decrease as k increases. The image quality also improves. Also as once notices, for some reason, the images became a lot darker upon reconstruction.

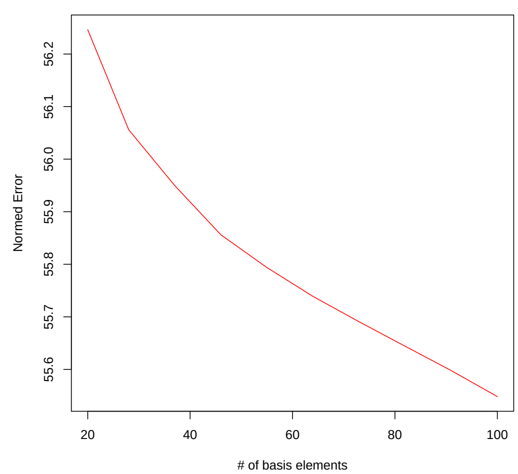

Figure 5: Plot of Normed Error versus the number of basis elements retained using the DCT

(a) (b) (c)

Figure 6: Pictures reconstucted using DCT basis with k = 10, 55 and 100