# CS 621: Assignment 1

Due: Wednesday, September 14, 2005 (In Lecture or 4130 Upson by 4pm)

Scoring for each problem is on a 0-to-3 scale ( 3 = complete success, 2 = overlooked a small detail, 1 = germ of the right idea, 0 = missed the point of the problem.) One point will be deducted for insufficiently commented code. Unless otherwise stated, you are expected to utilize fully MATLAB'S vectorizing capability subject to the constraint of being flop-efficient. Test drivers and related material are posted on the course website `http://www.cs.cornell.edu/courses/cs621/2005fa/`. For each problem submit output and a listing of all scripts/functions that you had to write in order to produce the output. You are allowed to discuss *background* issues with other students, but the codes you submit must be your own.

### P1. (A Rank-k Correction)

If $A \in \mathbb{R}^{n\times n}$, $u \in \mathbb{R}^n$, and $v \in \mathbb{R}^n$ are given and $k$ is an integer, then there exist $X \in \mathbb{R}^{n\times k}$ and $Y \in \mathbb{R}^{n\times k}$ such that $(A + uv^T)^k = A^k + XY^T$. This means that a rank-1 perturbation in $A$ induces a rank-$k$ perturbation in $A^k$. Complete the following MATLAB function so that it performs as specified.

```
  function [X,Y] = PowerDiff(A,u,v,k)
% A is an n-by-n matrix, u and v are column n-vectors and
% k is a positive integer with k << n.
% X and Y are n-by-k matrices such that (A + uv')^k = A^k + X*Y'
```

Your implementation should involve $ckn^2$ flops where $c$ is a small constant. Benchmark your implementation with the test script `P1` provided on the website. Submit output and listing of `PowerDiff`. Hint: $A + fg^T + de^T = A + [f\ d][g\ e]^T$.

### P2. (Higher Order Kronecker Product)

If $A \in \mathbb{R}^{m\times m}$ and $B \in \mathbb{R}^{n\times n}$ then their *Kronecker Product* $A \otimes B$ is an $m$-by-$m$ block matrix whose $ij$ entry is $a_{ij}B$. For example,

$$\begin{bmatrix} 10 & 20 \\ 30 & 40 \end{bmatrix} \otimes B = \begin{bmatrix} 10B & 20B \\ 30B & 40B \end{bmatrix}$$

Here are some facts:

1. If $F \in \mathbb{R}^{n\times n}$, $x \in \mathbb{R}^{mn}$, and $y = (I_m \otimes F)x$, then $Y = FX$ where $Y = \texttt{reshape}(y, n, m)$ and $X = \texttt{reshape}(x, n, m)$.
2. If $F \in \mathbb{R}^{m\times m}$, $x \in \mathbb{R}^{mn}$, and $y = (F \otimes I_n)x$, then $Y = XF^T$ where $Y = \texttt{reshape}(y, n, m)$ and $X = \texttt{reshape}(x, n, m)$.
3. If $F \in \mathbb{R}^{m\times m}$ and $G \in \mathbb{R}^{n\times n}$, then $F \otimes G = (I_m \otimes G)(F \otimes I_n) = (F \otimes I_n)(I_m \otimes G)$.
4. If $A$, $B$, and $C$ are matrices then $A \otimes B \otimes C = (A \otimes B) \otimes C = A \otimes (B \otimes C)$.

The `reshape` operator is a built-in MATLAB function that (guess what) reshapes the data in a matrix. The following example should suffice to clarify what it does:

$$A \in \mathbb{R}^{2\times 6}, \quad B = \texttt{reshape}(A, 3, 4) \quad \Rightarrow \quad B = \begin{bmatrix} a_{11} & a_{22} & a_{14} & a_{25} \\ a_{21} & a_{13} & a_{24} & a_{16} \\ a_{12} & a_{23} & a_{15} & a_{26} \end{bmatrix}$$

In other words, if $A \in \mathbb{R}^{m_1\times n_1}$ and $m_1n_1 = m_2n_2$, then $B = \texttt{reshape}(A, m_2, n_2)$ is obtained by stacking the columns of $A$ to form a length $m_1n_1$ vector $v$, which is then cut into into $n_2$ pieces which become the columns of $B$.

If $F \in \mathbb{R}^{m\times m}$, $G \in \mathbb{R}^{n\times n}$, and $x \in \mathbb{R}^{mn}$, then $y = (F \otimes G)x$ requires $2mn(m + n)$ flops, which is an order of magnitude less than a general matrix-vector product of the same size.

Develop an efficient implementation of the following function

```
     function y = ThreeWayProd(A,B,C,x)
   % A is m-by-m, B is n-by-n, C is p-by-p, and x is mnp-by-1.
   % y = Mx where M = kron(A,kron(B,C)) = A x B x C
```

Submit a listing and the output when the test script `P2` is run.

### P3. (An Expected Value)

If $A \in \mathbb{R}^{n \times n}$, then it can be shown that $1 \leq f(A) \leq \sqrt{n}$ where $f(A) = \| A \|_F / \| A \|_2$. What can you say about the expected value of $f(A)$ when the entries of $A$ are normally distributed with mean zero and unit standard deviation? Answer by submitting an informative plot that displays estimates of the expected value for values of $n$ in the interval [1,500]. Also submit the script that produced the plot.

### P4. (Fast Complex Matrix Multiplication?)

Show how to compute the complex matrix-matrix multiplication $(C + iD)(E + iF)$ with just three real matrix multiplies. (Hint: Compute $W = (C+D)(E-F)$). Implement a function `Z = FastProd(X,Y)` that incorporates this strategy and compare its efficiency with the one-liner `Z = X*Y` for `n = 50:50:1000`. Use `tic` and `toc` and report results graphically. Submit your test script and implementation of `FastProd`.