Matlab Functions for Final Exam

polyfit

P = POLYFIT(X,Y,N) finds the coefficients of a polynomial P(X) of degree N that fits the
data Y best in a least-squares sense. P is a row vector of length N+1 containing the
polynomial coefficients in descending powers, P(1)*X^N + P(2)*X^(N-1)+...+P(N)*X + P(N+1).

polyval

Y = POLYVAL(P,X) returns the value of a polynomial P evaluated at X. P is a vector of
length N+1 whose elements are the coefficients of the polynomial in descending powers,
Y = P(1)*X^N + P(2)*X^(N-1) + ... + P(N)*X + P(N+1)

spline

PP = SPLINE(X,Y) provides the piecewise polynomial form of the cubic spline interpolant to
the data values Y at the data sites X, for use with the evaluator PPVAL and the spline
utility UNMKPP. X must be a vector. If Y is a vector, then Y(j) is taken as the value to
be matched at X(j).

ppval

V = PPVAL(PP,XX) returns the value, at the entries of XX, of the piecewise polynomial f
contained in PP.

mkpp

PP = MKPP(BREAKS,COEFS) puts together a piecewise polynomial PP from its breaks and
coefficients. BREAKS must be a vector of length L+1 with strictly increasing elements
representing the start and end of each of L intervals. The matrix COEFS must be L-by-K,
with the i-th row, COEFS(i,:), representing the local coefficients of the order K
polynomial on the interval [BREAKS(i) ... BREAKS(i+1)], i.e., the polynomial
COEFS(i,1)*(X-BREAKS(i))^(K-1) + COEFS(i,2)*(X-BREAKS(i))^(K-2) + ... +
COEFS(i,K-1)*(X-BREAKS(i)) + COEFS(i,K).

unmkpp

[BREAKS,COEFS,L,K] = UNMKPP(PP) extracts from the piecewise polynomial PP its breaks,
coefficients, number of pieces, and order.
fzero

Suppose \( f(x,P_1,P_2,...) \) is a continuous function of the scalar \( x \) with parameters \( P_1, P_2,... \). If \( f \) changes sign on the interval \([L,R]\), then 
\[
z = \text{fzero}(@f,[L,R],[],P_1,P_2,...)
\]
assigns to the variable \( z \) a zero of the function \( f \). The parameters \( P_1, P_2,... \) must be assigned values before the call.

fminbnd

Suppose \( f(x,P_1,P_2,...) \) is a continuous function of the scalar \( x \) with parameters \( P_1, P_2,... \). If \( f \) is defined on the interval \([L,R]\), then 
\[
xstar = \text{fminbnd}(\@x) f(x,P_1,P_2,...),L,R)
\]
assigns to the variable \( xstar \) an approximate minimizer of \( f \) on the interval \([L,R]\). The parameters \( P_1, P_2,... \) must be assigned values before the call.

fminsearch

Suppose \( f(x,P_1,P_2,...) \) is a continuous function of the vector \( x \) with parameters \( P_1, P_2,... \). If \( f \) is defined in a neighborhood of \( x_0 \), then 
\[
xstar = \text{fminbnd}(\@x) f(x,P_1,P_2,...),x_0)
\]
assigns to the variable \( xstar \) an approximate minimizer of \( f \) in the vicinity of \( x_0 \). The parameters \( P_1, P_2,... \) must be assigned values before the call.

quad

Suppose \( f(x,P_1,P_2,...) \) is a continuous function of the scalar \( x \) with parameters \( P_1, P_2,... \). If \( f \) is defined on the interval \([a,b]\), then 
\[
Q = \text{quad}(\@x) f(x,P_1,P_2,...),a,b,tol)
\]
assigns to the variable \( Q \) an approximation of the integral of \( f \) from \( a \) to \( b \). The absolute error of is usually in the vicinity of \( tol \). The parameters \( P_1, P_2,... \) must be assigned values before the call.

Backslash “\"\"

If \( A \) is square and nonsingular, then 
\[
x = A\backslash b
\]
 solves the linear system \( Ax = b \).
If \( A \) is square, nonsingular, and triangular, then 
\[
x = A\backslash b
\]
 solves \( Ax = b \) in \( O(n^2) \) flops.
If \( A \) has more rows than columns, then 
\[
x = A\backslash b
\]
 solves the least squares problem 
\[
\min \| Ax - b \|_2.
\]

LU

If \( A \) is an \( n \)-by-\( n \) matrix, then 
\[
[L,U,P] = \text{LU}(A)
\]
produces a unit lower triangular matrix \( L \), upper triangular matrix \( U \), and permutation matrix \( P \) so that \( P*A = L*U \).

CHOL

If \( X \) is positive definite, then 
\[
R = \text{CHOL}(X)
\]
produces an upper triangular \( R \) so that \( R'*R = X \).

QR

\( A \) is \( m \)-by-\( n \), then 
\[
[Q,R] = \text{QR}(A)
\]
produces an \( m \)-by-\( n \) upper triangular matrix \( R \) and an \( m \)-by-\( m \) orthogonal matrix \( Q \) so that \( A = Q*R \).

SVD

Singular value decomposition. \( [U,S,V] = \text{SVD}(X) \) produces a diagonal matrix \( S \), of the same dimension as \( X \) and with nonnegative diagonal elements in decreasing order, and orthogonal matrices \( U \) and \( V \) so that \( X = U*S*V' \).