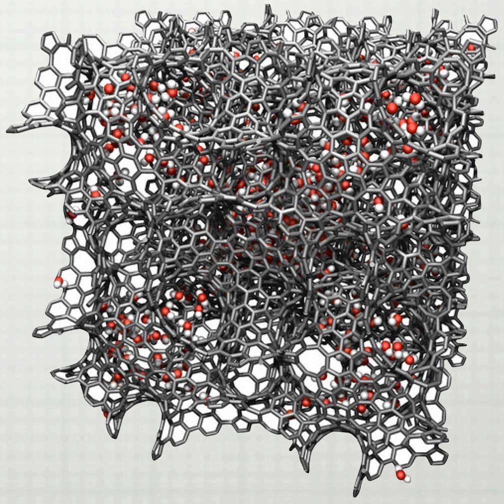
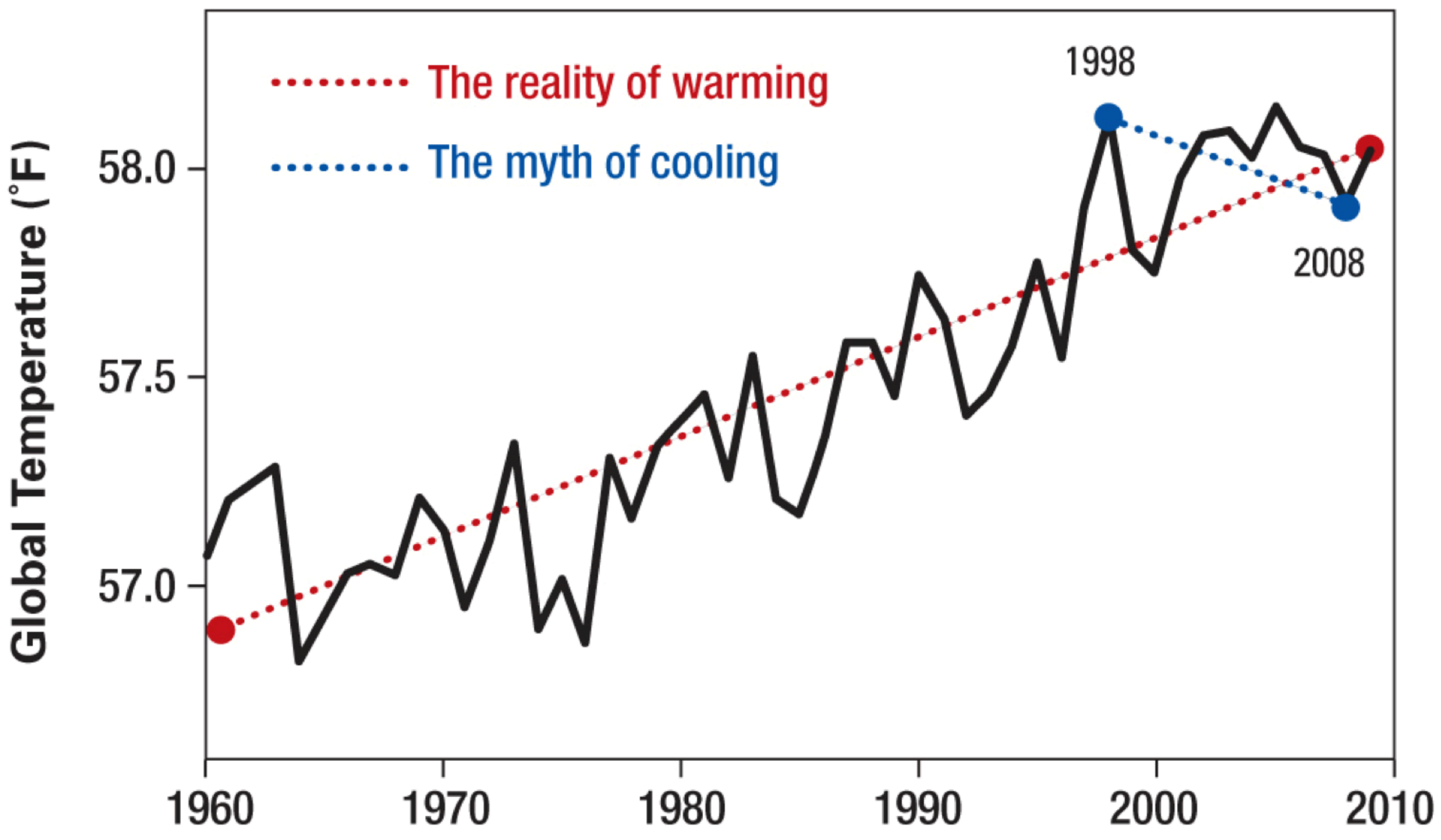
PHY-4810 COMPUTATIONAL PHYSICS

LECTURE 7: DATA FITTING

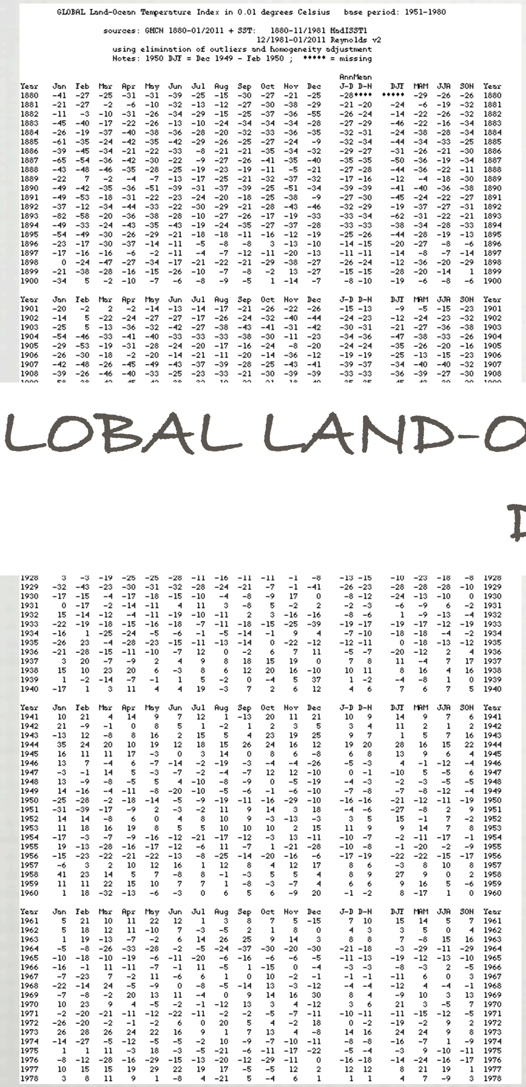




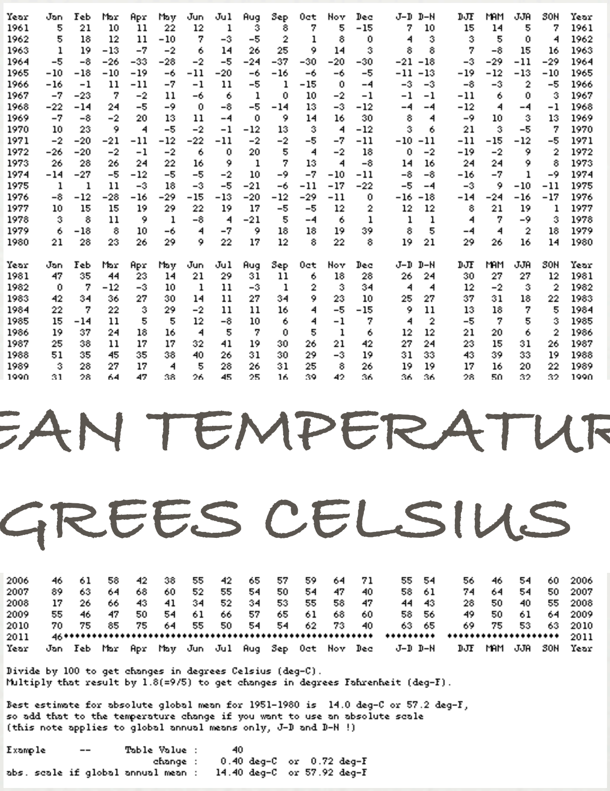
Global Surface Temperature Over the Last Fifty Years

HTTP://DATA.GISS.NASA.GOV/GISTEMP/TABLEDATA/GLB.TS+DSST.TXT

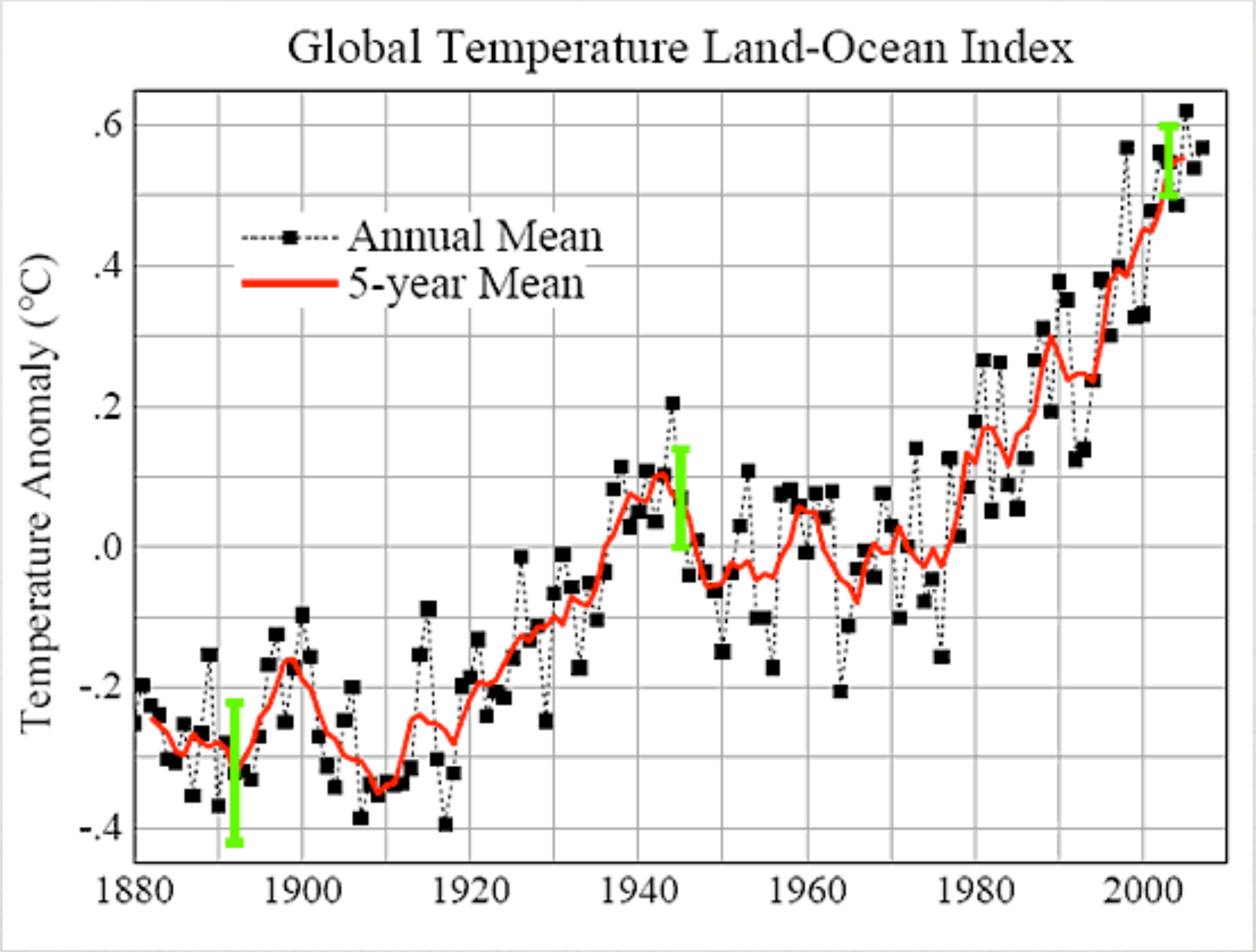
GLOBAL LAND-OCEAN TEMPERATURE INDEX IN 0.01 DEGREES CELSIUS



```
        GLOBAL Land-Ocean Temperature Index in 0.01 degrees Celsius   base period: 1951-1980

                  sources: GHCN 1880-01/2011 + SST:   1880-11/1981 HadISST1
                                                      12/1981-01/2011 Reynolds v2
                    using elimination of outliers and homogeneity adjustment
                    Notes: 1950 DJF = Dec 1949 - Feb 1950 ;  ***** = missing

                                                                     AnnMean
Year   Jan  Feb  Mar  Apr  May  Jun  Jul  Aug  Sep  Oct  Nov  Dec     J-D D-N    DJF  MAM  JJA  SON  Year
1880   -41  -27  -25  -31  -31  -39  -25  -15  -30  -27  -21  -25     -28****  *****  -29  -26  -26  1880
1881   -21  -27   -2   -6  -10  -32  -13  -12  -27  -30  -38  -29     -21 -20    -24   -6  -19  -32  1881
1882   -11   -3  -10  -31  -26  -34  -29  -15  -25  -37  -36  -55     -26 -24    -14  -22  -26  -32  1882
1883   -45  -40  -17  -22  -26  -13  -10  -24  -34  -34  -34  -28     -27 -29    -46  -22  -16  -34  1883
1884   -26  -19  -37  -40  -38  -36  -28  -20  -32  -33  -36  -35     -32 -31    -24  -38  -28  -34  1884
1885   -61  -35  -24  -42  -35  -42  -29  -26  -25  -27  -24   -9     -32 -34    -44  -34  -33  -25  1885
1886   -39  -45  -34  -21  -22  -33   -8  -21  -21  -35  -34  -32     -29 -27    -31  -26  -21  -30  1886
1887   -65  -54  -36  -42  -30  -22   -9  -27  -26  -41  -35  -40     -35 -35    -50  -36  -19  -34  1887
1888   -43  -48  -46  -35  -28  -25  -19  -23  -19  -11   -5  -21     -27 -28    -44  -36  -22  -11  1888
1889   -22    7   -2   -4   -7  -13  -17  -25  -21  -32  -37  -32     -17 -16    -12   -4  -18  -30  1889
1890   -49  -42  -35  -36  -51  -39  -31  -37  -39  -25  -51  -34     -39 -39    -41  -40  -36  -38  1890
1891   -49  -53  -18  -31  -22  -23  -24  -20  -18  -25  -38   -9     -27 -30    -45  -24  -22  -27  1891
1892   -37  -12  -34  -44  -33  -22  -30  -29  -21  -28  -43  -46     -32 -29    -19  -37  -27  -31  1892
1893   -82  -58  -20  -36  -38  -28  -10  -27  -26  -17  -19  -33     -33 -34    -62  -31  -22  -21  1893
1894   -49  -33  -24  -43  -35  -43  -19  -24  -35  -27  -37  -28     -33 -33    -38  -34  -28  -33  1894
1895   -54  -49  -30  -26  -29  -21  -18  -18  -11  -16  -12  -19     -25 -26    -44  -28  -19  -13  1895
1896   -23  -17  -30  -37  -14  -11   -5   -8   -8    3  -13  -10     -14 -15    -20  -27   -8   -6  1896
1897   -17  -16  -16   -6   -2  -11   -4   -7  -12  -11  -20  -13     -11 -11    -14   -8   -7  -14  1897
1898     0  -24  -47  -27  -34  -17  -21  -22  -21  -29  -38  -27     -26 -24    -12  -36  -20  -29  1898
1899   -21  -38  -28  -16  -15  -26  -10   -7   -8   -2   13  -27     -15 -15    -28  -20  -14    1  1899
1900   -34    5   -2  -10   -7   -6   -8   -9   -5    1  -14   -7      -8 -10    -19   -6   -8   -6  1900

Year   Jan  Feb  Mar  Apr  May  Jun  Jul  Aug  Sep  Oct  Nov  Dec     J-D D-N    DJF  MAM  JJA  SON  Year
1901   -20   -2    2   -2  -14  -13  -14  -17  -21  -26  -22  -26     -15 -13     -9   -5  -15  -23  1901
1902   -14    5  -22  -24  -27  -27  -17  -26  -24  -32  -40  -44     -24 -23    -12  -24  -23  -32  1902
1903   -25    5  -13  -36  -32  -42  -27  -38  -43  -41  -31  -42     -30 -31    -21  -27  -36  -38  1903
1904   -54  -46  -33  -41  -40  -33  -33  -33  -38  -30  -11  -23     -34 -36    -47  -38  -33  -26  1904
1905   -29  -53  -19  -31  -28  -24  -20  -17  -16  -24   -8  -20     -24 -24    -35  -26  -20  -16  1905
1906   -26  -30  -18   -2  -20  -14  -21  -11  -20  -14  -36  -12     -19 -19    -25  -13  -15  -23  1906
1907   -42  -48  -26  -45  -49  -43  -37  -39  -28  -25  -43  -41     -39 -37    -34  -40  -40  -32  1907
1908   -39  -26  -46  -40  -33  -25  -23  -33  -21  -30  -39  -39     -33 -33    -36  -39  -27  -30  1908

1928     3   -3  -19  -25  -25  -28  -11  -16  -11  -11   -1   -8     -13 -15    -10  -23  -18   -8  1928
1929   -32  -43  -23  -30  -31  -32  -28  -24  -21   -7   -1  -41     -26 -23    -28  -28  -28  -10  1929
1930   -17  -15   -4  -17  -18  -15  -10   -4   -8   -9   17    0      -8 -12    -24  -13  -10    0  1930
1931     0  -17   -2  -14  -11    4   11    3   -8    5   -2    2      -2  -3     -6   -9    6   -2  1931
1932    15  -14  -12   -4  -11  -19  -10  -11    2    3  -16  -16      -8  -6      1   -9  -13   -4  1932
1933   -22  -19  -18  -15  -16  -18   -7  -11  -18  -15  -25  -39     -19 -17    -19  -17  -12  -19  1933
1934   -16    1  -25  -24   -5   -6   -1   -5  -14   -1    9    4      -7 -10    -18  -18   -4   -2  1934
1935   -26   23   -4  -28  -23  -15  -11  -13  -14    0  -22  -12     -12 -11      0  -18  -13  -12  1935
1936   -21  -28  -15  -11  -10   -7   12    0   -2    6    7   11      -5  -7    -20  -12    2    4  1936
1937     3   20   -7   -9    2    4    9    8   18   15   19    0       7   8     11   -4    7   17  1937
1938    15   10   23   20    6   -3    8    6   12   20   16  -10      10  11      8   16    4   16  1938
1939     1   -2  -14   -7   -1    1    5   -2    0   -4    5   37       1  -2     -4   -8    1    0  1939
1940   -17    1    3   11    4    4   19   -3    7    2    6   12       4   6      7    6    7    5  1940

Year   Jan  Feb  Mar  Apr  May  Jun  Jul  Aug  Sep  Oct  Nov  Dec     J-D D-N    DJF  MAM  JJA  SON  Year
1941    10   21    4   14    9    7   12    1  -13   20   11   21      10   9     14    9    7    6  1941
1942    21   -9   -1    0    8    5    1   -2    1    2    3    5       3   4     11    2    1    2  1942
1943   -13   12   -8    8   16    2   15    5    4   23   19   25       9   7      1    5    7   16  1943
1944    35   24   20   10   19   12   18   15   26   24   16   12      19  20     28   16   15   22  1944
1945    16   11   11   17   -3    0    3   14    0    8    6   -8       6   8     13    9    6    4  1945
1946    13    7   -4    6   -7  -14   -2  -19   -3   -4   -4  -26      -5  -3      4   -1  -12   -4  1946
1947    -3   -1   14    5   -3   -7   -2   -4   -7   12   12  -10       0  -1    -10    5   -5    6  1947
1948    13   -9   -8   -5    5    4  -10   -8   -9    0   -5  -19      -4  -3     -2   -3   -5   -5  1948
1949    14  -16   -4  -11   -8  -20  -10   -5   -6   -1   -6  -10      -7  -8     -7   -8  -12   -4  1949
1950   -25  -28   -2  -18  -14   -5   -9  -19  -11  -16  -29  -10     -16 -16    -21  -12  -11  -19  1950
1951   -31  -39  -17   -9    2   -3   -2   11    9   14    3   18      -4  -6    -27   -8    2    9  1951
1952    14   14   -8    6    0    4    8   10    9   -3  -13   -3       3   5     15   -1    7   -2  1952
1953    11   18   16   19    8    5    5   10   10   10    2   15      11   9      9   14    7    8  1953
1954   -17   -3   -7   -9  -16  -12  -21  -17  -12   -3   13  -11     -10  -7     -2  -11  -17   -1  1954
1955    19  -13  -28  -16  -17  -12   -6   11   -7    1  -21  -28     -10  -8     -1  -20   -2   -9  1955
1956   -15  -23  -22  -21  -22  -13   -8  -25  -14  -20  -16   -6     -17 -19    -22  -22  -15  -17  1956
1957    -6    3    2   10   12   16    1   12    8    4   12   17       8   6     -3    8   10    8  1957
1958    41   23   14    5    7   -8    8   -1   -3    5    5    4       8   9     27    9    0    2  1958
1959    11   11   22   15   10    7    7    1   -8   -3   -7    4       6   6      9   16    5   -6  1959
1960     1   18  -32  -13   -6   -3    0    6    5    6   -9   20      -1  -2      8  -17    1    0  1960

Year   Jan  Feb  Mar  Apr  May  Jun  Jul  Aug  Sep  Oct  Nov  Dec     J-D D-N    DJF  MAM  JJA  SON  Year
1961     5   21   10   11   22   12    1    3    8    7    5  -15       7  10     15   14    5    7  1961
1962     5   18   12   11  -10    7   -3   -5    2    1    8    0       4   3      3    5    0    4  1962
1963     1   19  -13   -7   -2    6   14   26   25    9   14    3       8   8      7   -8   15   16  1963
1964    -5   -8  -26  -33  -28   -2   -5  -24  -37  -30  -20  -30     -21 -18     -3  -29  -11  -29  1964
1965   -10  -18  -10  -19   -6  -11  -20   -6  -16   -6   -6   -5     -11 -13    -19  -12  -13  -10  1965
1966   -16   -1   11  -11   -7   -1   11   -5    1  -15    0   -4      -3  -3     -8   -3    2   -5  1966
1967    -7  -23    7   -2   11   -6    6    1    0   10   -2   -1      -1  -1    -11    6    0    3  1967
1968   -22  -14   24   -5   -9    0   -8   -5  -14   13   -3  -12      -4  -4    -12    4   -4   -1  1968
1969    -7   -8   -2   20   13   11   -4    0    9   14   16   30       8   4     -9   10    3   13  1969
1970    10   23    9    4   -5   -2   -1  -12   13    3    4  -12       3   6     21    3   -5    7  1970
1971    -2  -20  -21  -11  -12  -22  -11   -2   -2   -5   -7  -11     -10 -11    -11  -15  -12   -5  1971
1972   -26  -20   -2   -1   -2    6    0   20    5    4   -2   18       0  -2    -19   -2    9    2  1972
1973    26   28   26   24   22   16    9    1    7   13    4   -8      14  16     24   24    9    8  1973
1974   -14  -27   -5  -12   -5   -5   -2   10   -9   -7  -10  -11      -8  -8    -16   -7    1   -9  1974
1975     1    1   11   -3   18   -3   -5  -21   -6  -11  -17  -22      -5  -4     -3    9  -10  -11  1975
1976    -8  -12  -28  -16  -29  -15  -13  -20  -12  -29  -11    0     -16 -18    -14  -24  -16  -17  1976
1977    10   15   15   19   29   22   19   17   -5   -5   12    2      12  12      8   21   19    1  1977
1978     3    8   11    9    1   -8    4  -21    5   -4    6    1       1   1      4    7   -9    3  1978
```



```
Year   Jan  Feb  Mar  Apr  May  Jun  Jul  Aug  Sep  Oct  Nov  Dec     J-D D-N    DJF  MAM  JJA  SON  Year
1961     5   21   10   11   22   12    1    3    8    7    5  -15       7  10     15   14    5    7  1961
1962     5   18   12   11  -10    7   -3   -5    2    1    8    0       4   3      3    5    0    4  1962
1963     1   19  -13   -7   -2    6   14   26   25    9   14    3       8   8      7   -8   15   16  1963
1964    -5   -8  -26  -33  -28   -2   -5  -24  -37  -30  -20  -30     -21 -18     -3  -29  -11  -29  1964
1965   -10  -18  -10  -19   -6  -11  -20   -6  -16   -6   -6   -5     -11 -13    -19  -12  -13  -10  1965
1966   -16   -1   11  -11   -7   -1   11   -5    1  -15    0   -4      -3  -3     -8   -3    2   -5  1966
1967    -7  -23    7   -2   11   -6    6    1    0   10   -2   -1      -1  -1    -11    6    0    3  1967
1968   -22  -14   24   -5   -9    0   -8   -5  -14   13   -3  -12      -4  -4    -12    4   -4   -1  1968
1969    -7   -8   -2   20   13   11   -4    0    9   14   16   30       8   4     -9   10    3   13  1969
1970    10   23    9    4   -5   -2   -1  -12   13    3    4  -12       3   6     21    3   -5    7  1970
1971    -2  -20  -21  -11  -12  -22  -11   -2   -2   -5   -7  -11     -10 -11    -11  -15  -12   -5  1971
1972   -26  -20   -2   -1   -2    6    0   20    5    4   -2   18       0  -2    -19   -2    9    2  1972
1973    26   28   26   24   22   16    9    1    7   13    4   -8      14  16     24   24    9    8  1973
1974   -14  -27   -5  -12   -5   -5   -2   10   -9   -7  -10  -11      -8  -8    -16   -7    1   -9  1974
1975     1    1   11   -3   18   -3   -5  -21   -6  -11  -17  -22      -5  -4     -3    9  -10  -11  1975
1976    -8  -12  -28  -16  -29  -15  -13  -20  -12  -29  -11    0     -16 -18    -14  -24  -16  -17  1976
1977    10   15   15   19   29   22   19   17   -5   -5   12    2      12  12      8   21   19    1  1977
1978     3    8   11    9    1   -8    4  -21    5   -4    6    1       1   1      4    7   -9    3  1978
1979     6  -18    8   10   -6    4   -7    9   18   18   19   39       8   5     -4    4    2   18  1979
1980    21   28   23   26   29    9   22   17   12    8   22    8      19  21     29   26   16   14  1980

Year   Jan  Feb  Mar  Apr  May  Jun  Jul  Aug  Sep  Oct  Nov  Dec     J-D D-N    DJF  MAM  JJA  SON  Year
1981    47   35   44   23   14   21   29   31   11    6   18   28      26  24     30   27   27   12  1981
1982     0    7  -12   -3   10    1   11   -3    1    2    3   34       4   4     12   -2    3    2  1982
1983    42   34   36   27   30   14   11   27   34    9   23   10      25  27     37   31   18   22  1983
1984    22    7   22    3   29   -2   11   11   16    4   -5  -15       9  11     13   18    7    5  1984
1985    15  -14   11    5    5   12   -8   10    6    4   -1    7       4   2     -5    7    5    3  1985
1986    19   37   24   18   16    4    5    7    0    5    1    6      12  12     21   20    6    2  1986
1987    25   38   11   17   17   32   41   19   30   26   21   42      27  24     23   15   31   26  1987
1988    51   35   45   35   38   40   26   31   30   29   -3   19      31  33     43   39   33   19  1988
1989     3   28   27   17    4    5   28   26   31   25    8   26      19  19     17   16   20   22  1989
1990    31   28   64   47   38   26   45   25   16   39   42   36      36  36     28   50   32   32  1990

2006    46   61   58   42   38   55   42   65   57   59   64   71      55  54     56   46   54   60  2006
2007    89   63   64   68   60   52   55   54   50   54   47   40      58  61     74   64   54   50  2007
2008    17   26   66   43   41   34   52   34   53   55   58   47      44  43     28   50   40   55  2008
2009    55   46   47   50   54   61   66   57   65   61   68   60      58  56     49   50   61   64  2009
2010    70   75   85   75   64   55   50   54   54   62   73   40      63  65     69   75   53   63  2010
2011    46*******************************************************   ********  ******************** 2011
Year   Jan  Feb  Mar  Apr  May  Jun  Jul  Aug  Sep  Oct  Nov  Dec     J-D D-N    DJF  MAM  JJA  SON  Year

Divide by 100 to get changes in degrees Celsius (deg-C).
Multiply that result by 1.8(=9/5) to get changes in degrees Fahrenheit (deg-F).

Best estimate for absolute global mean for 1951-1980 is  14.0 deg-C or 57.2 deg-F,
so add that to the temperature change if you want to use an absolute scale
(this note applies to global annual means only, J-D and D-N !)

Example      --      Table Value :      40
                          change :    0.40 deg-C  or  0.72 deg-F
abs. scale if global annual mean :   14.40 deg-C  or 57.92 deg-F
```



HTTP://WWW.ECY.WA.GOV/CLIMATECHANGE/WHATIS.HTM

ON THE MENU

- ☐ **Lagrange multipliers**
- ☐ **Spline**
- ☐ **Fit to theory**
- ☐ **How to use Numerical Recipes**

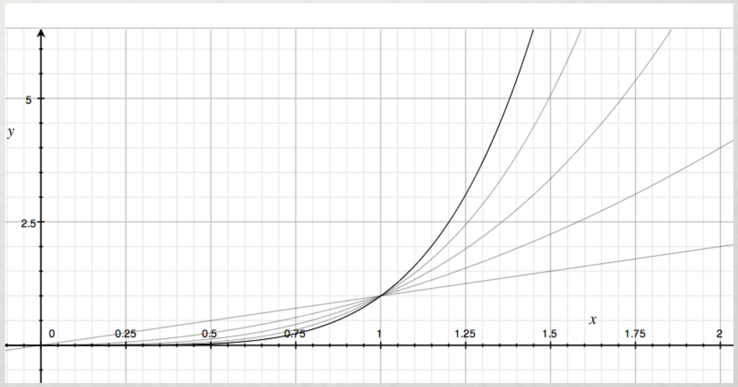
PART 1: LAGRANGE INTERPOLATION

METHOD AND ALGORITHM

- **Any function g(x) can be approximated as a polynomial of degree (n – 1) in each interval i for a sufficiently high *n*:**

$$\boxed{g_i(x) \simeq a_0 + a_1 x + a_2 x^2 + \cdots + a_{n-1} x^{n-1}}$$

- **Because our fit is <u>local</u>, we do not assume that one g(x) can fit all the data in a list, but instead will use a different polynomial, that is, a different set of $\underline{a_i}$ values, for each region of the list.**



LAGRANGE INTERPOLATION FORMULA

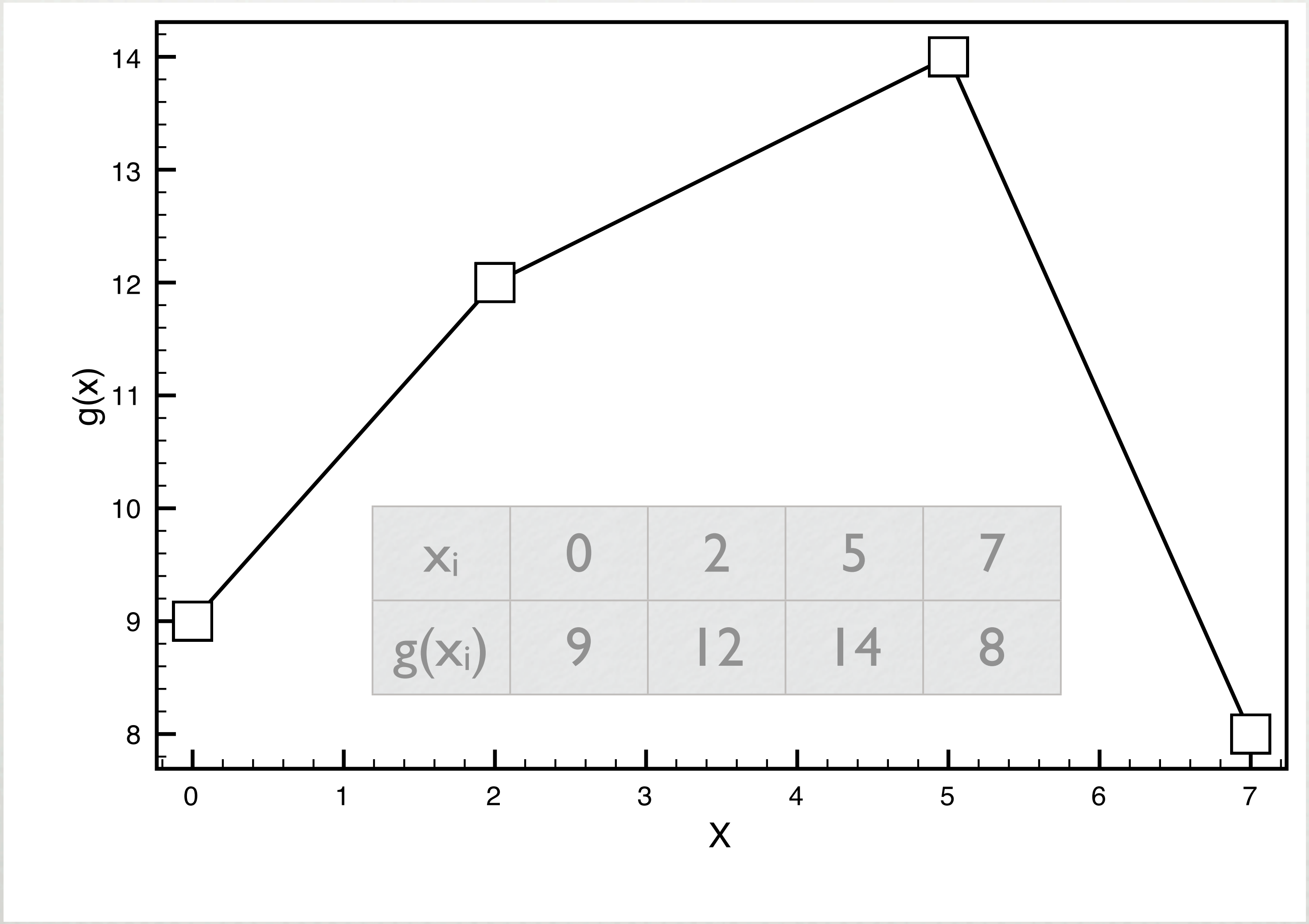
- **Lagrange figured out a closed-form approach that directly fits the *(n – 1)*-order polynomial to *n* values of the function *g(x)* evaluated at the points x_i. The formula is written as the sum of polynomials:**

$$g(x) \simeq g_1\lambda_1(x) + g_2\lambda_2(x) + \cdots + g_n\lambda_n(x)$$

$$\lambda_i(x) = \prod_{j(\neq i)=1}^{n} \frac{x - x_j}{x_i - x_j} = \frac{x - x_1}{x_i - x_1}\frac{x - x_2}{x_i - x_2} \cdots \frac{x - x_n}{x_i - x_n}$$

- **For <u>three</u> points, this formula provides a <u>second-degree polynomial</u>, while for <u>eight</u> points it gives a <u>seventh-degree</u> polynomial.**

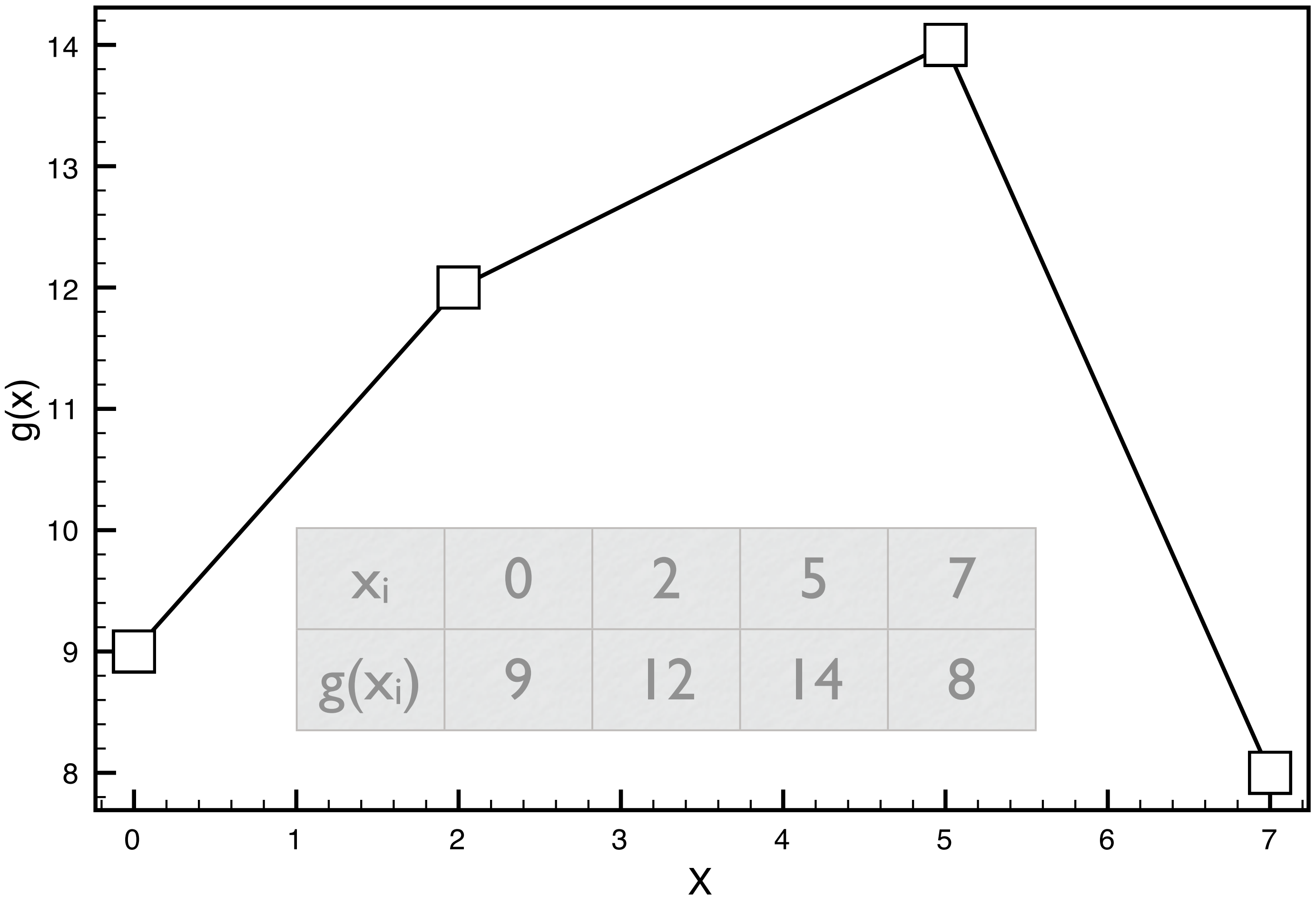
EXAMPLE: POLYNOMIAL OF ORDER 3



x_i	0	2	5	7
$g(x_i)$	9	12	14	8

$$g(x) \simeq g_1\lambda_1(x) + g_2\lambda_2(x) + \cdots + g_n\lambda_n(x)$$

$$\lambda_i(x) = \prod_{j(\neq i)=1}^{n} \frac{x - x_j}{x_i - x_j} = \frac{x - x_1}{x_i - x_1}\frac{x - x_2}{x_i - x_2}\cdots\frac{x - x_n}{x_i - x_n}$$



x_i	0	2	5	7
$g(x_i)$	9	12	14	8

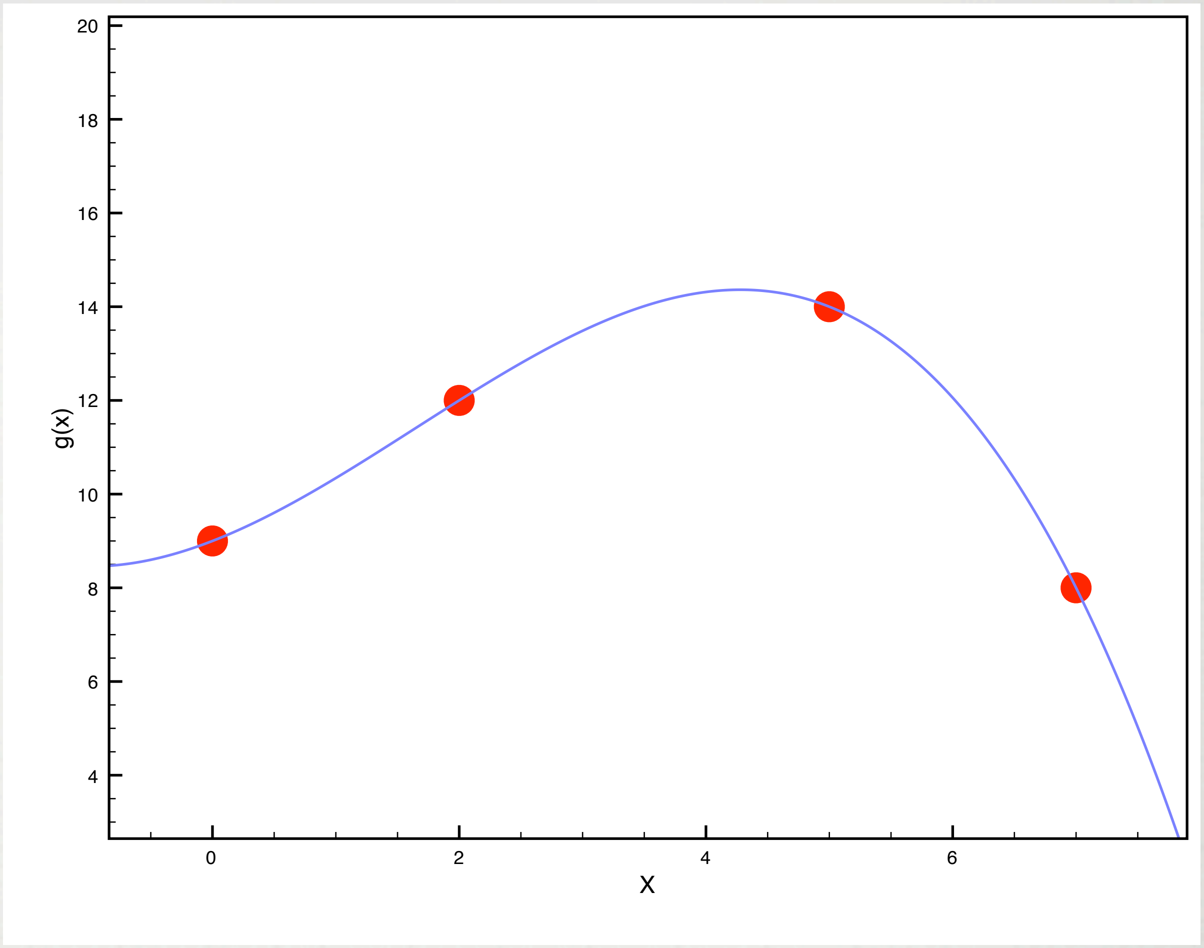
$$g(x) = g_1\lambda_1(x) + g_1\lambda_2(x) + g_1\lambda_3(x)$$

$$\lambda_1(x) = \frac{(x-2)(x-5)(x-7)}{-2\times -5\times -7}$$

$$\lambda_2(x) = \frac{x(x-5)(x-7)}{2\times -3\times -5}$$

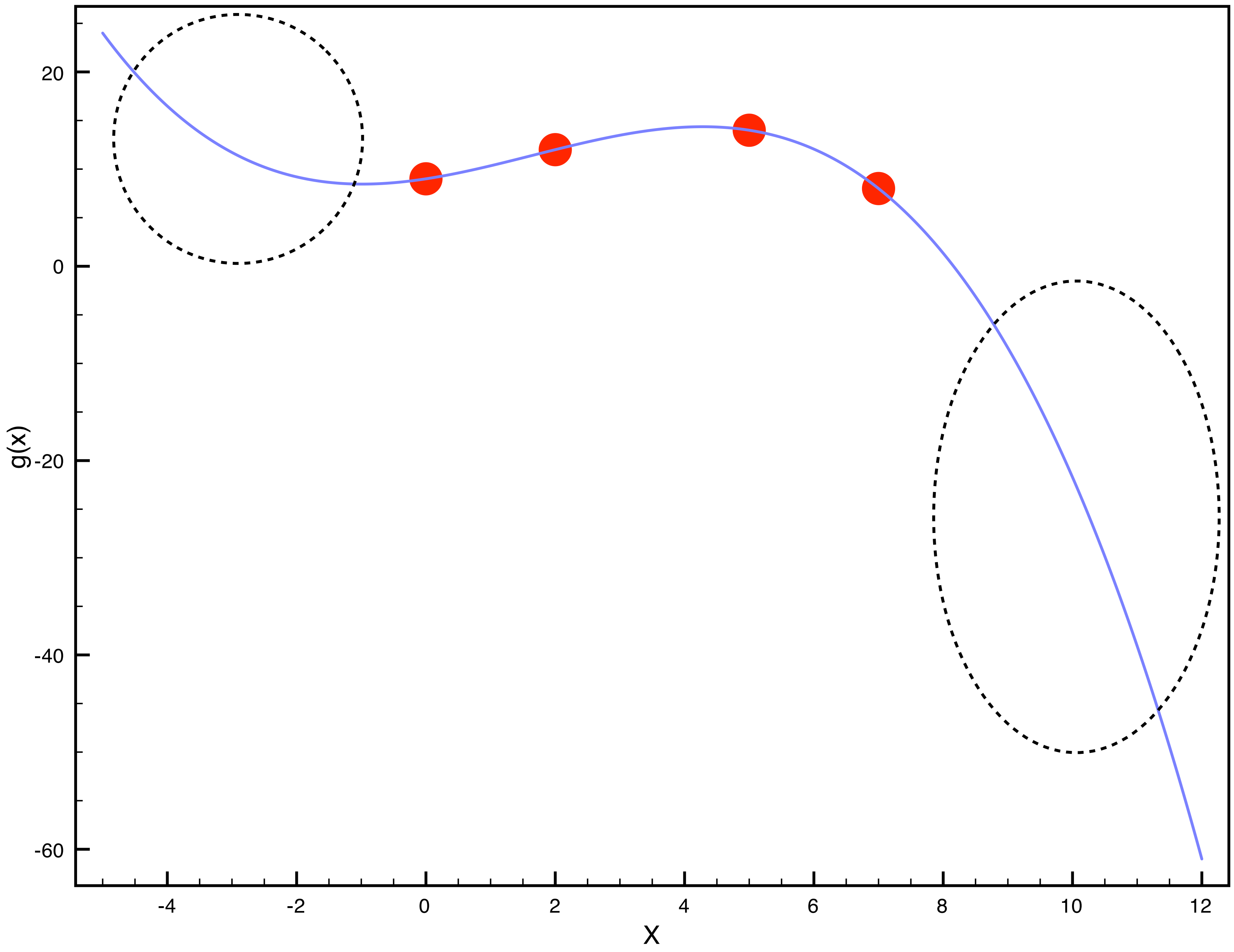
$$\lambda_3(x) = \frac{x(x-2)(x-7)}{5\times 3\times -2}$$

$$\lambda_4(x) = \frac{x(x-2)(x-5)}{7\times 5\times 2}$$



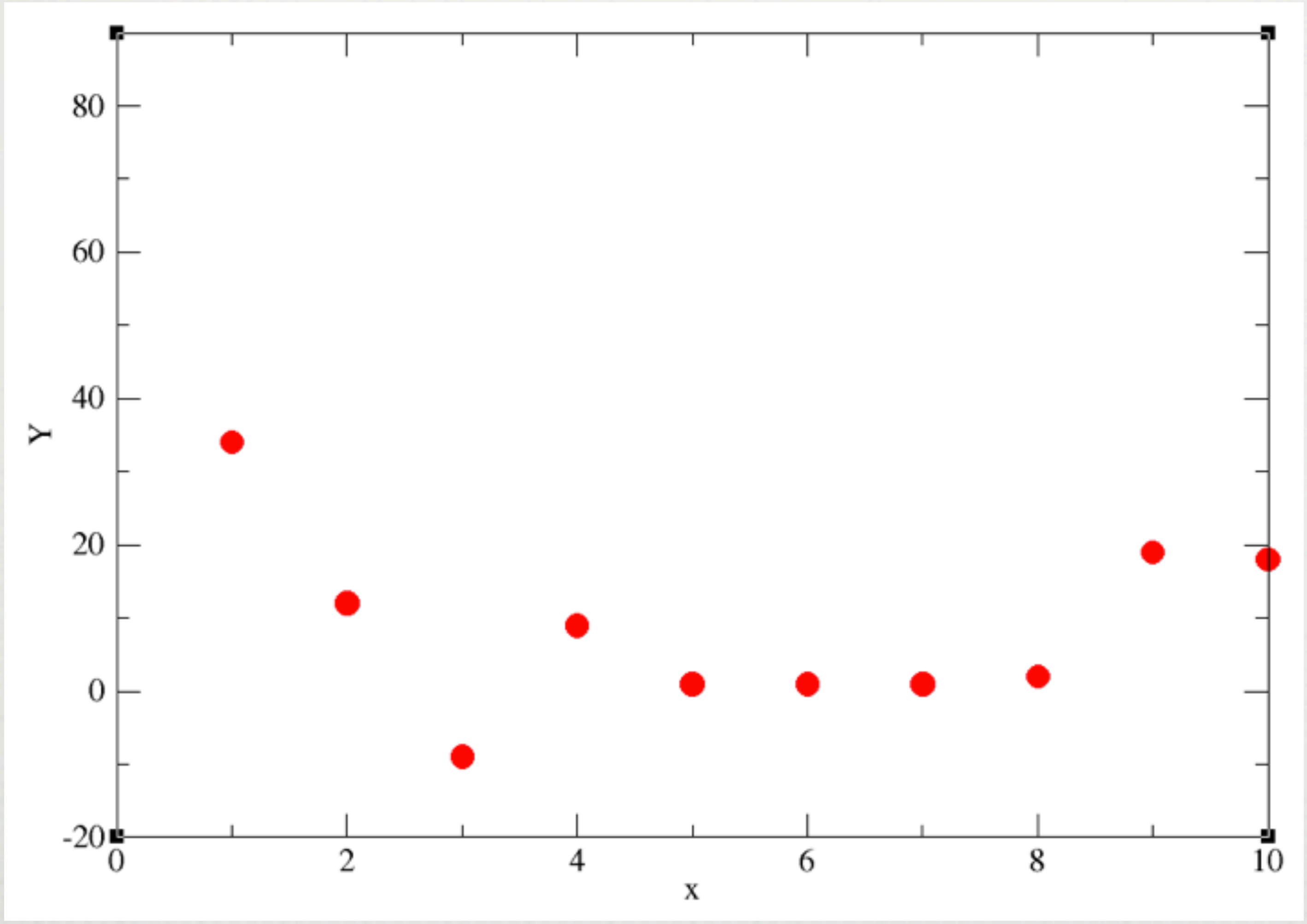
$$g(x) = \frac{-9}{70}(x-2)(x-5)(x-7)+\frac{2}{5}x(x-5)(x-7)+\frac{-7}{15}x(x-2)(x-7)+\frac{4}{35}x(x-2)(x-5)$$

…HOWEVER…RESIST THE TEMPTATION… (INTERPOLATION VS. EXTRAPOLATION)

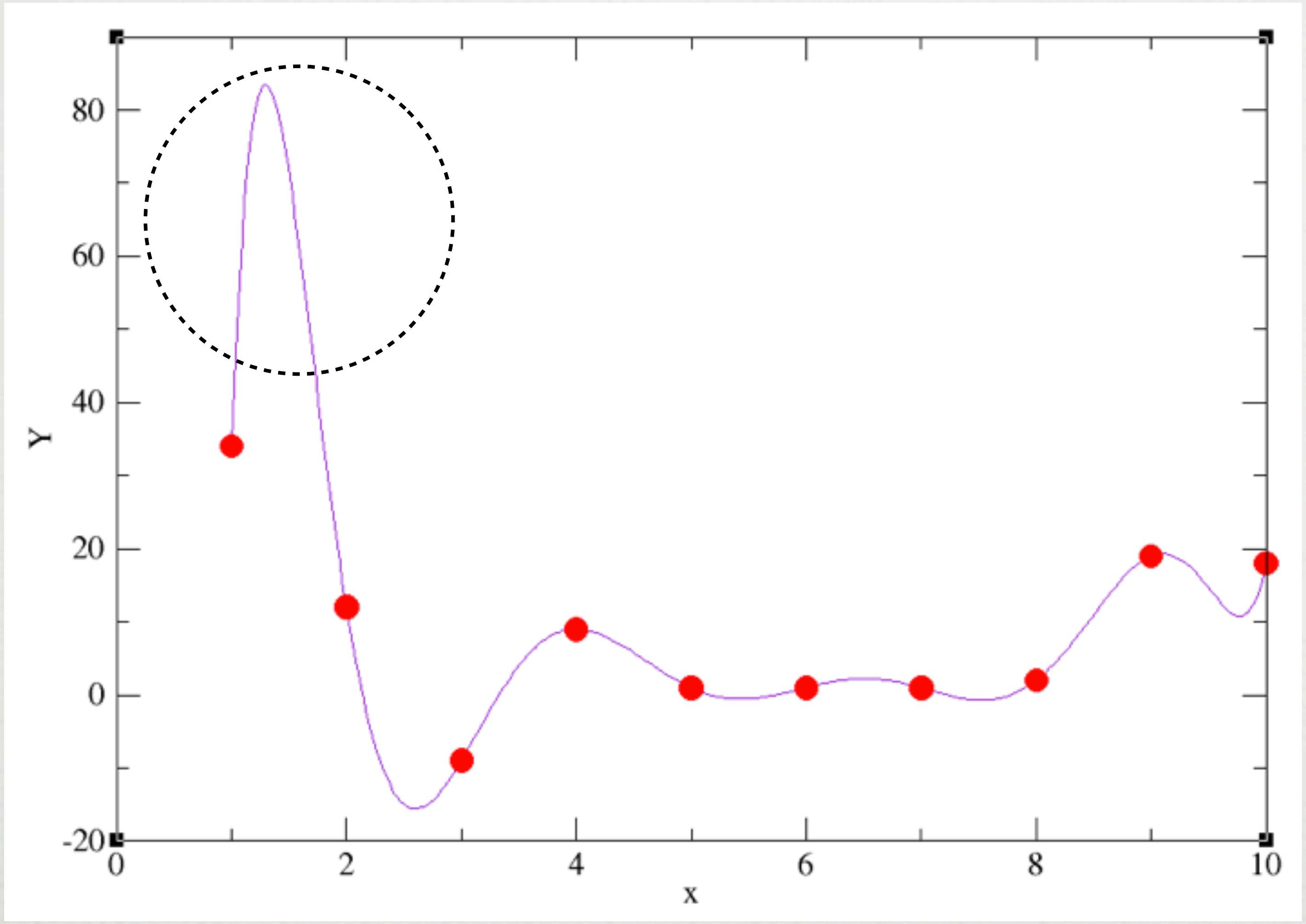


20
0
-20
-40
-60
g(x)
-4
-2
0
2
4
6
8
10
12
X

10 DATA POINTS: POLYNOMIAL OF ORDER 9



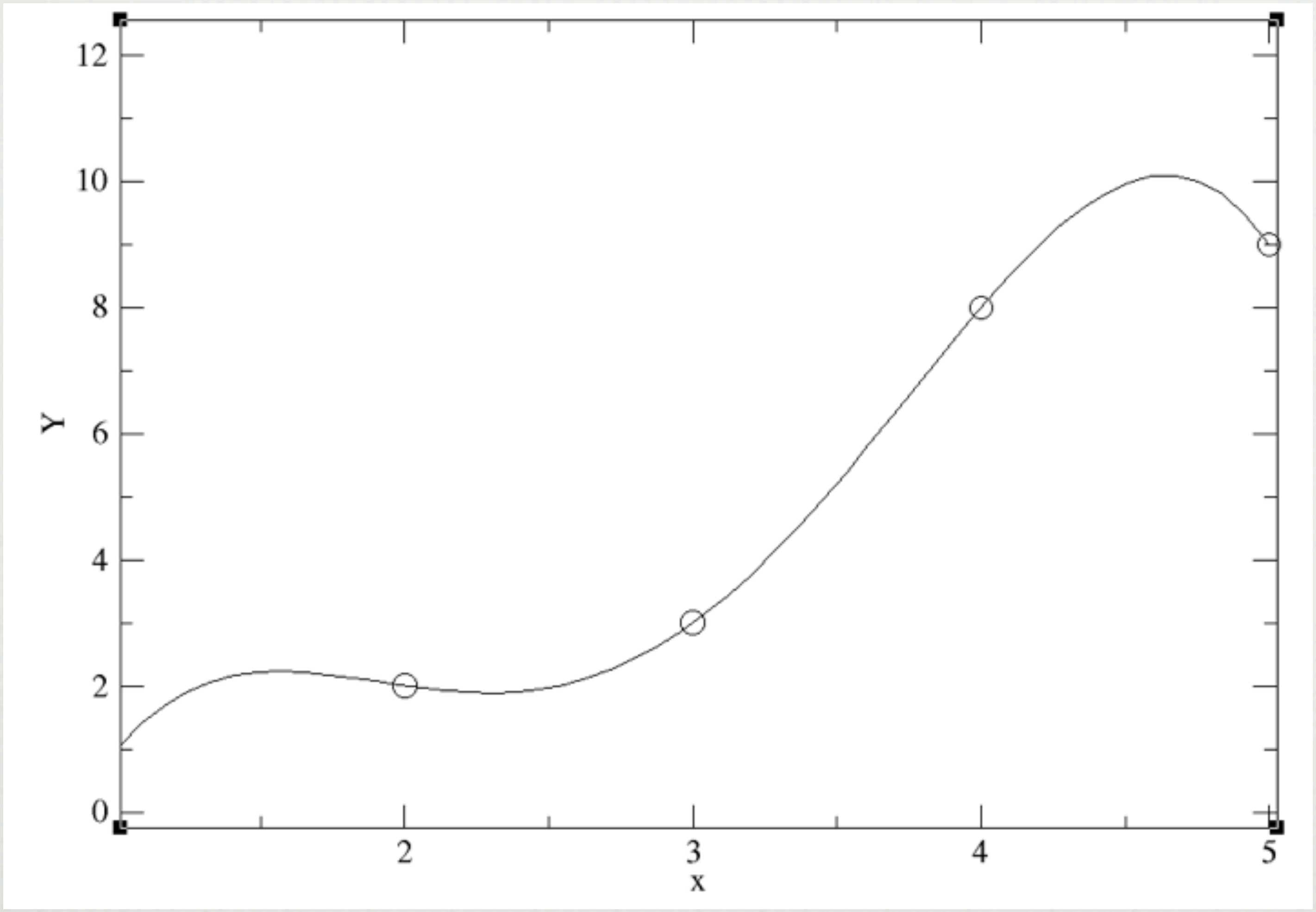
POLYNOMIAL FIT

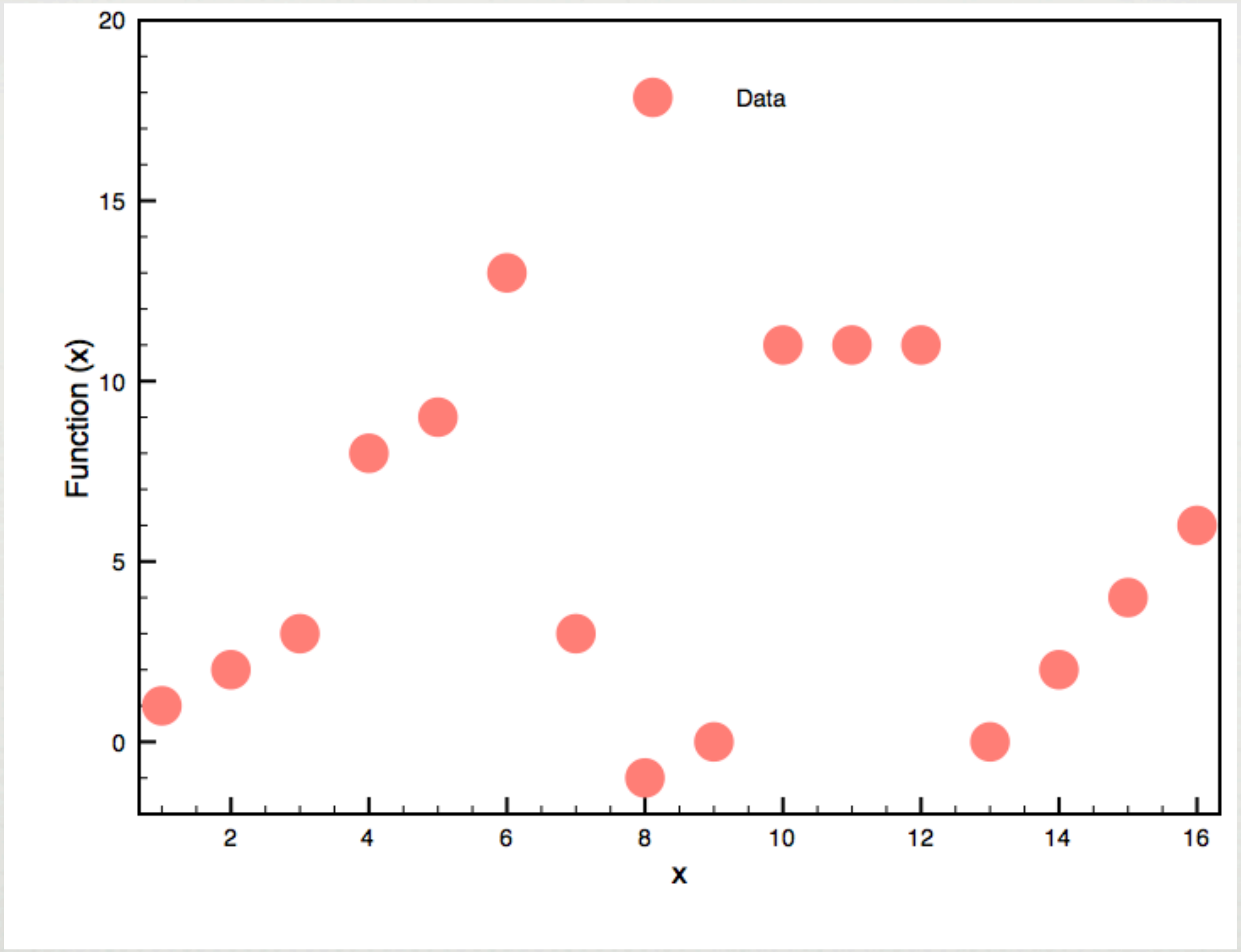


LET'S HAVE A LOOK AT THE C++ CODE

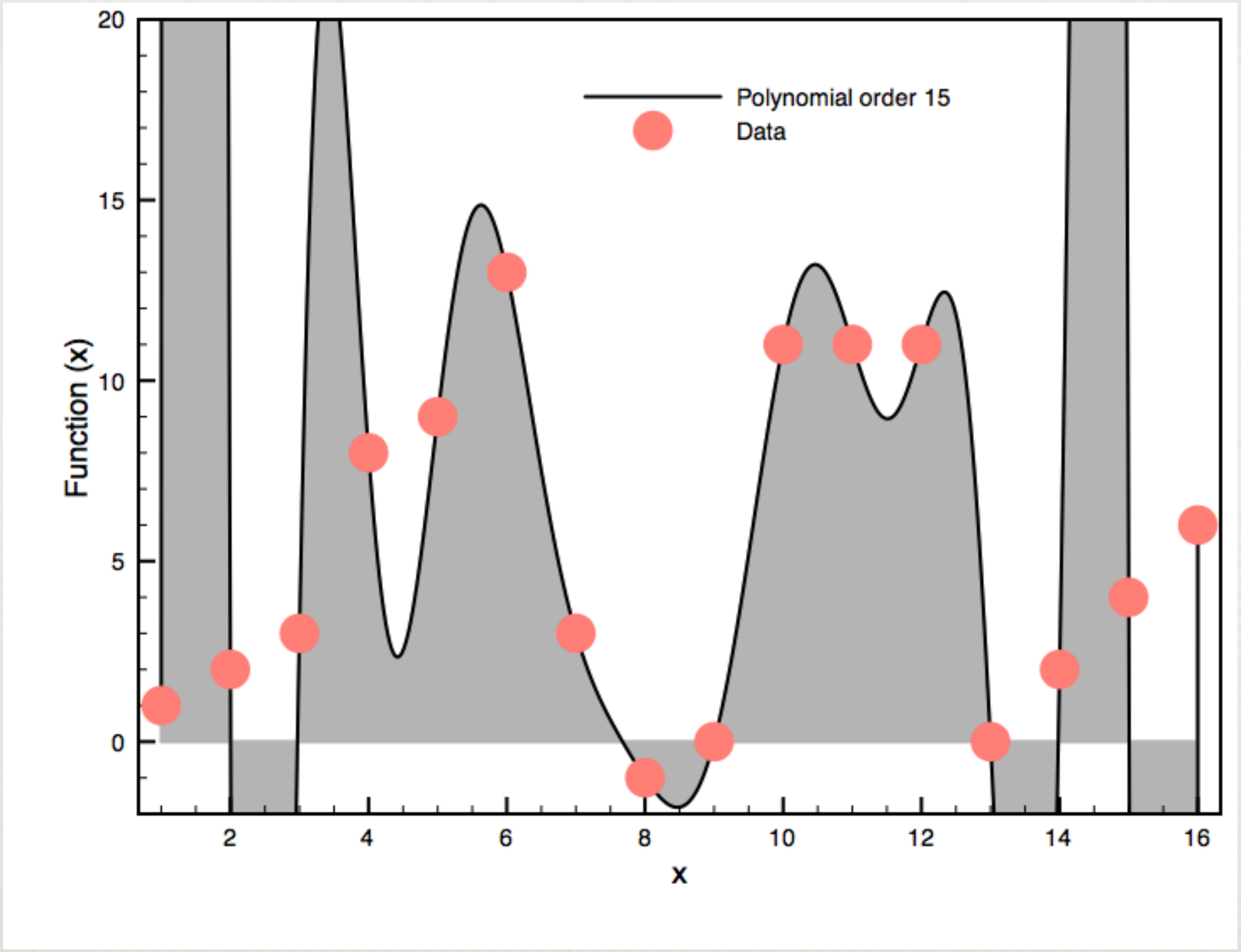
```
#INCLUDE <IOSTREAM>
USING NAMESPACE STD;

INT MAIN () {
    // INSERT CODE HERE...
    INT I, J, K;
    INT KMAX=393;
    DOUBLE XX=0, GXX=0;
    DOUBLE XXMIN, XXMAX;
    //
    INT NPOINTS;
    CIN >> NPOINTS;
    //
    DOUBLE X[NPOINTS], G[NPOINTS];
    DOUBLE LAMBDA[NPOINTS];

    //
    FOR (I=0;I<NPOINTS;I++){
           CIN >> X[I] >> G[I];
    }
    //
    XXMIN=X[0];
    XXMAX=X[NPOINTS-1];
    //
    FOR(K=0;K<KMAX;K++){
           XX=XXMIN+(DOUBLE)K*(XXMAX-XXMIN)/((DOUBLE)KMAX-1);
           GXX=0.0;
           FOR (I=0;I<NPOINTS;I++){
                 LAMBDA[I]=1.;
                 FOR (J=0;J<NPOINTS;J++){
                       IF(J==I) CONTINUE;
                       LAMBDA[I]*=(XX-X[J])/(X[I]-X[J]);
                 }
                 GXX+=LAMBDA[I]*G[I];
           }
           COUT << XX <<  " " << GXX << ENDL;
    }
    RETURN 0;
}
```

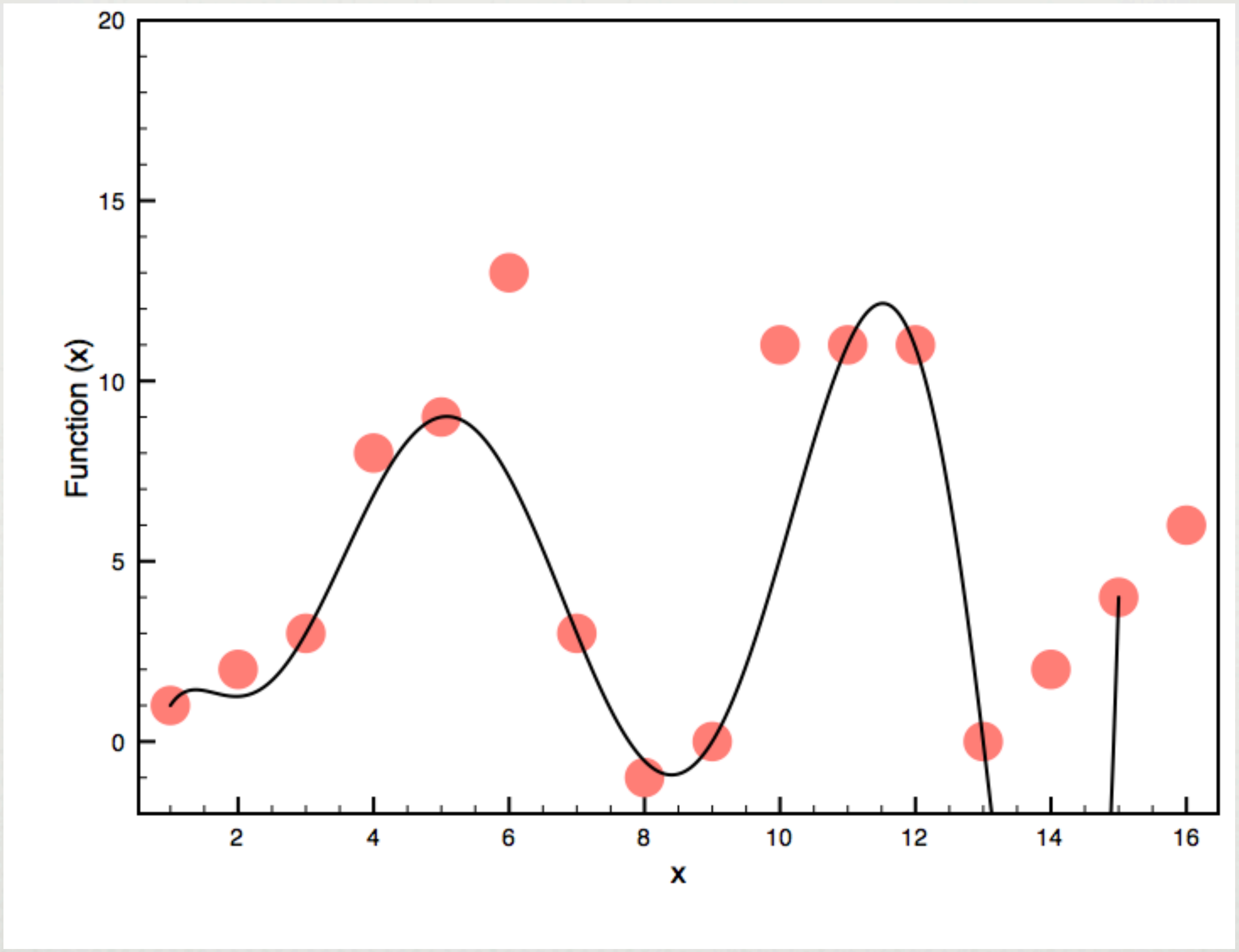
EXAMPLE OF POLYNOMIAL OBTAINED USING NEVILLE'S ALGORITHM





Data
Function (x)
x
20
15
10
5
0
2
4
6
8
10
12
14
16


Polynomial order 15
Data
Function (x)
x
20
15
10
5
0
2
4
6
8
10
12
14
16


20
15
10
5
0
Function (x)
2
4
6
8
10
12
14
16
x

NEVILLE'S ALGORITHM

- **A much better algorithm to construct the (unique) polynomial is called Neville's algorithm**

```
//NEVILLE's ALGORITHM
#include <iostream>
using namespace std;

int main () {
    int i,k,m;
    int kmax=50;
    double xx=0, gxx=0;
    double xxmin, xxmax;
    double sum;
    //
    int npoints;
    cin >> npoints;
    //
    double x[npoints], g[npoints];
    double P0[npoints], P[npoints];

    //
    for (i=0;i<npoints;i++){
          cin >> x[i] >> g[i];
    }
    //
    xxmin=x[0];
    xxmax=x[npoints-1];
    //
    for(k=0;k<kmax;k++){
          xx=xxmin+(double)k*(xxmax-xxmin)/((double)kmax-1);
          gxx=0.0;
          sum=0;

          for(m=0;m<npoints;m++){

                if(m==0){
                      for (i=0;i<npoints;i++){
                            P[i]=g[i];
                      }
                }
                else {
                      for (i=0;i<npoints-m;i++){
                            P[i]=((xx-x[i+m])*P0[i]+(x[i]-xx)*P0[i+1])/(x[i]-x[i+m]);
                      }
                }
                for (i=0;i<npoints-m;i++){
                      P0[i]=P[i];
                }

          }
          cout << xx <<  " " << P[0] << endl;
    }

    return 0;
}
```

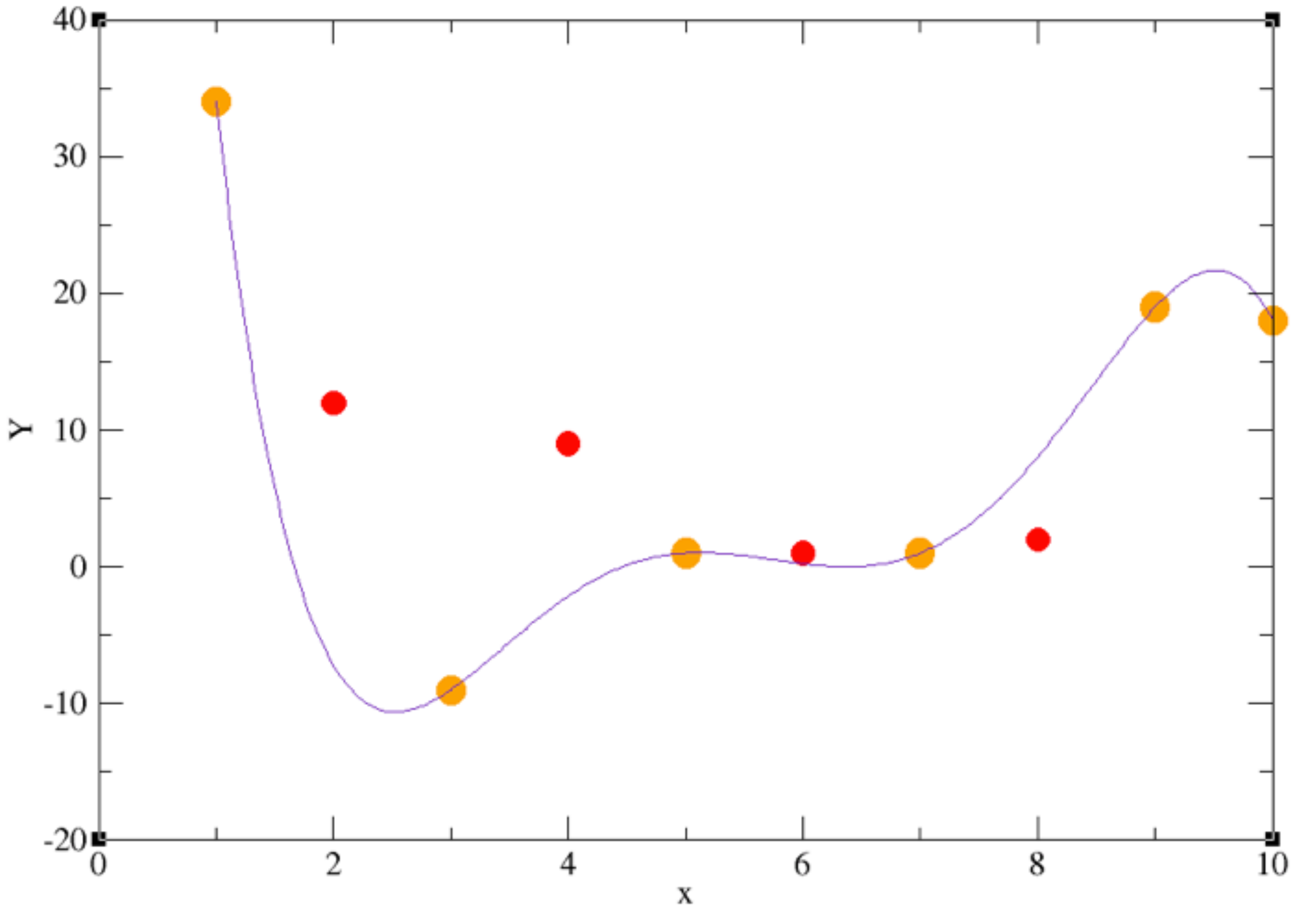
PRACTICAL CONSIDERATIONS

- If the data contain little noise, this polynomial can be used with some confidence within the range of data, but with risk beyond the range of data.

- Notice that Lagrange interpolation makes no restriction that the points in the table be evenly spaced.

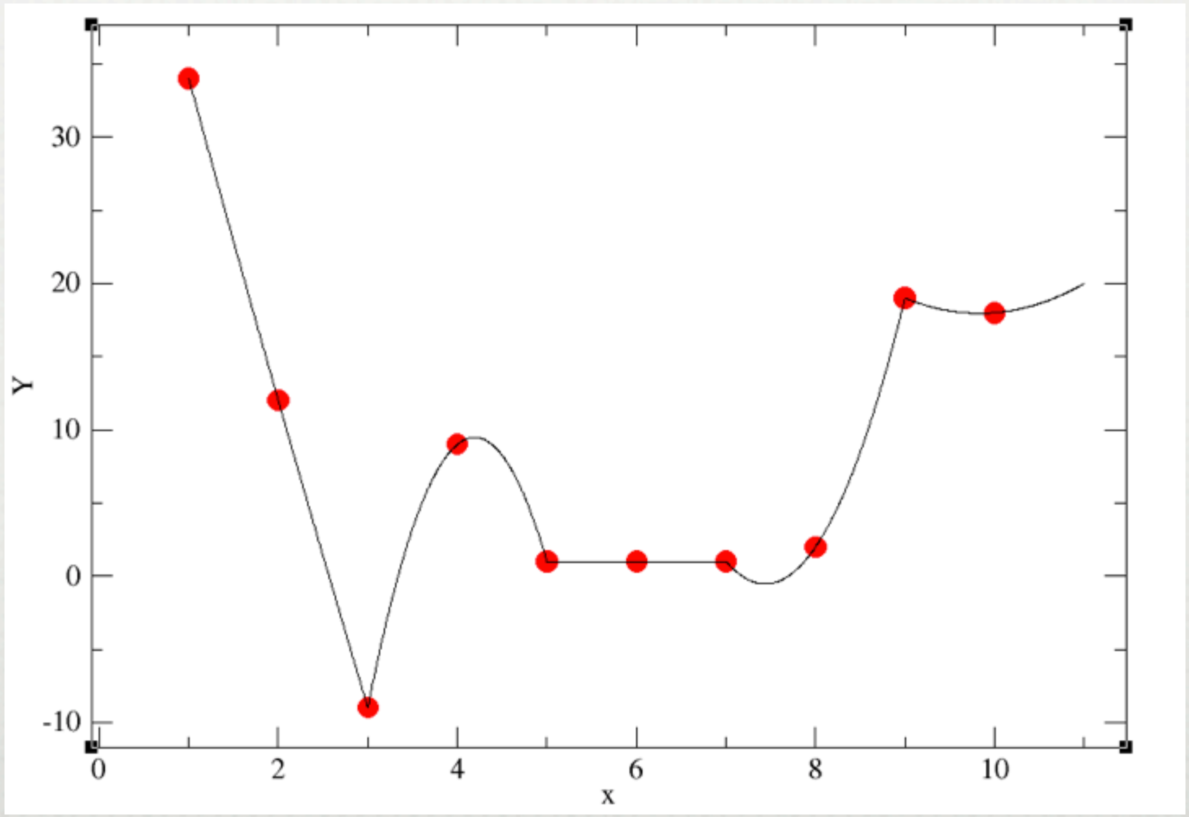
- As a check, it is also worth noting that the sum of the Lagrange multipliers equals one, $\sum \lambda_i = 1$.

- Usually the Lagrange fit is made to only a small region of the table with a small value of n, even though the formula works perfectly well for fitting a high-degree polynomial to the entire table.

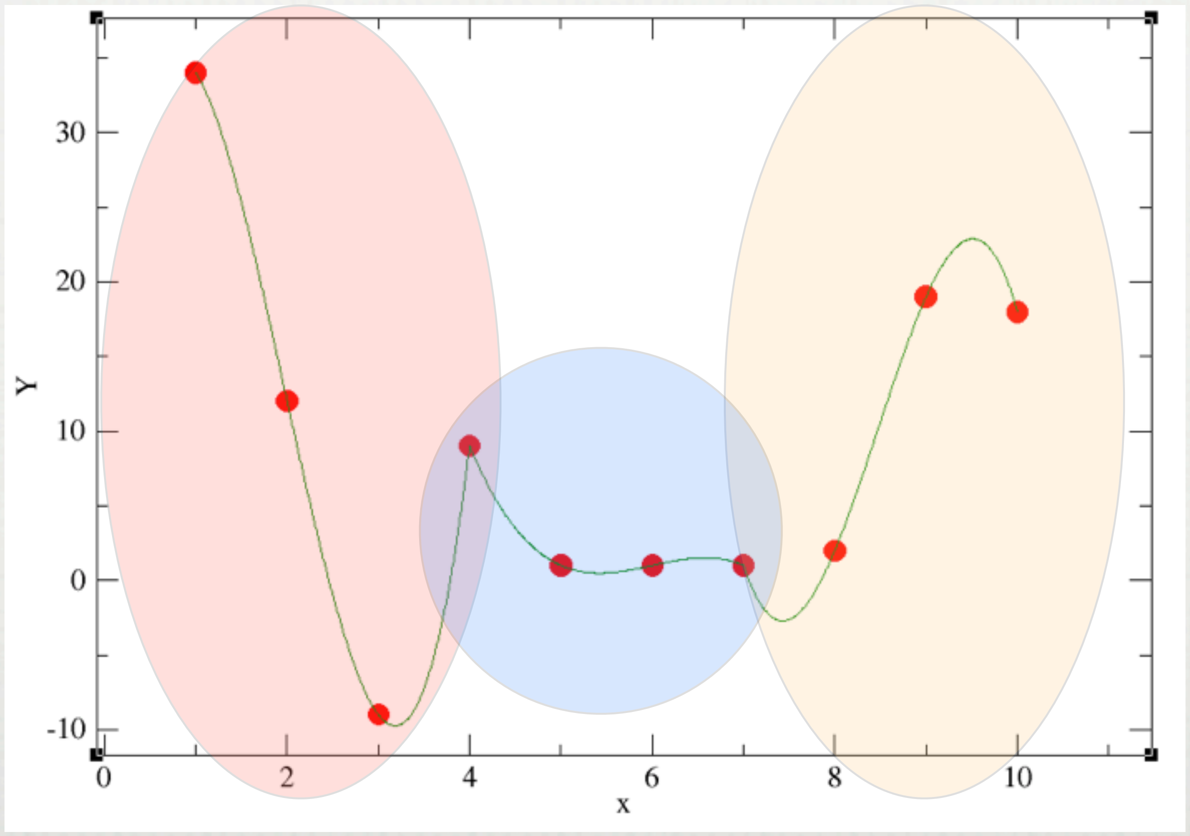

40
30
20
10
0
-10
-20
Y
0
2
4
6
8
10
x

LOCAL INTERPOLATION

- Here, we interpolate the function, small interval by small interval, with a low order polynomial (here: 3-points or parabola)

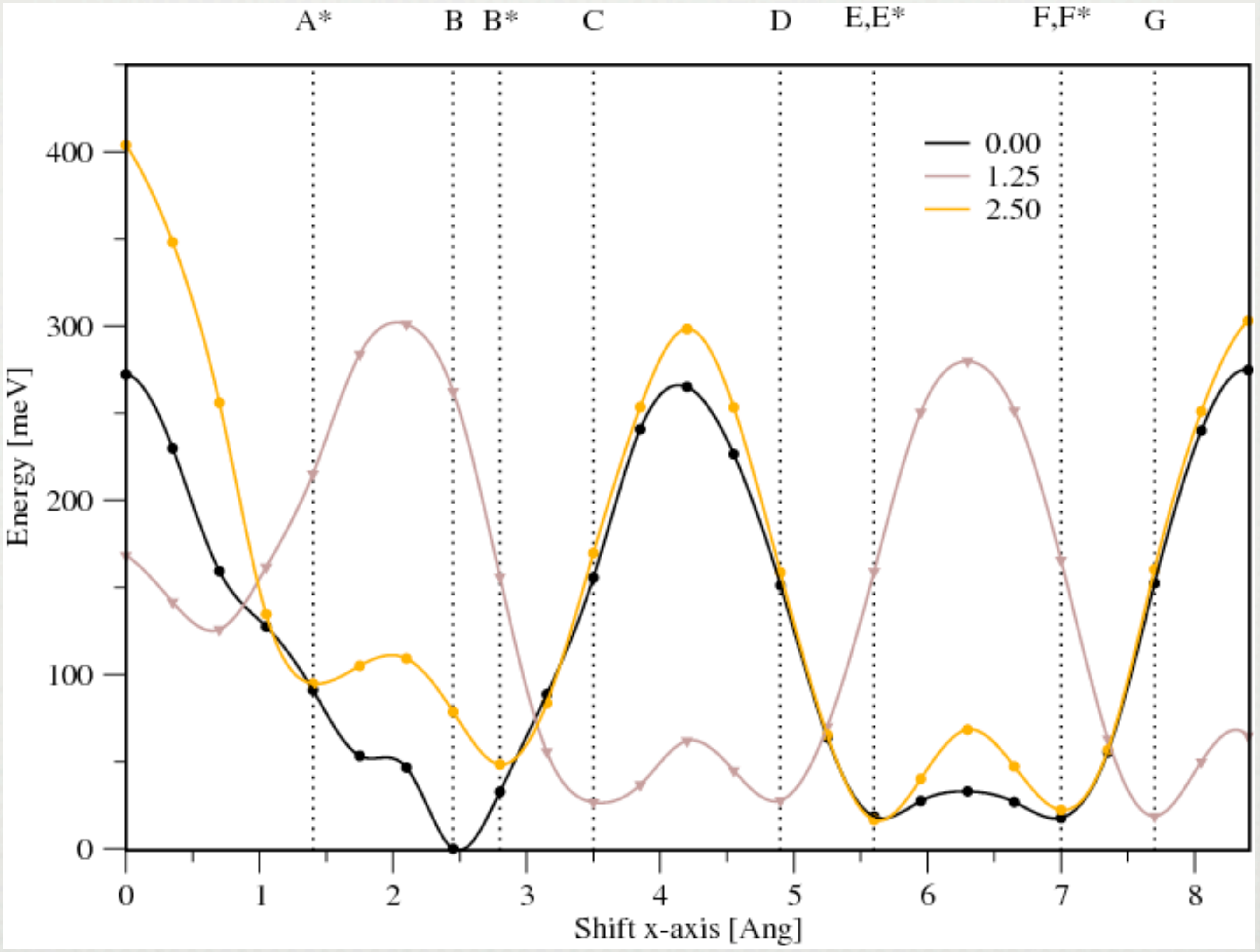


4-POINT SEGMENTATION (THIRD ORDER POLYNOMIAL)



PROBLEMS: DISCONTINUITY IN DERIVATIVE!

PART2:
SPLINE


A*
B
B*
C
D
E,E*
F,F*
G
0.00
1.25
2.50
Energy [meV]
Shift x-axis [Ang]
0
100
200
300
400
0
1
2
3
4
5
6
7
8

SPLINE: INTRODUCTION

- ☐ **Fitting parabolas (three-point interpolation) within a table may avoid the erroneous and possibly catastrophic deviations of a high-order formula.**
- ☐ **(Two-point interpolation, which connects the points with straight lines, may not lead you far astray, but it is rarely pleasing to the eye or precise.)**
- ☐ **A sophisticated variation of n = 4 interpolation, known as cubic splines, often leads to surprisingly eye-pleasing fits.**

SPLINE: INTRODUCTION (II)

- In this approach, cubic polynomials are fit to the function in each interval, with the additional constraint that the first and second derivatives of the polynomials must be continuous from one interval to the next. This continuity of slope and curvature is what makes the spline fit particularly eye-pleasing.

SPLINE: C’TD

- **The series of cubic polynomials obtained by spline-fitting a table can be integrated and differentiated, and is guaranteed to have well-behaved derivatives.**
- **The complexity of simultaneously matching polynomials and their derivatives over all the interpolation points leads to many simultaneous, linear equations to be solved.**
- **This makes splines unattractive for hand calculations, yet easy for computers.**

- The basic approximation of splines is the representation of the function g(x) in the subinterval [x_i, x_{i+1}] with a cubic polynomial:

$$
\begin{aligned}
g(x) &\simeq g_i(x) \qquad \text{for } x_i \le x \le x_{i+1} \\
g_i(x) &= g_i + g_i'(x-x_i) + \tfrac{1}{2}g_i''(x-x_i)^2 + \tfrac{1}{6}g_i'''(x-x_i)^3
\end{aligned}
$$

- This representation makes it clear that the coefficients in the polynomial equal the values of g(x) and to its first, second, and third derivatives at the tabulated points x_i. Derivatives beyond the third vanish.
  - The computational chore is to determine these derivatives in terms of the N tabulated values g_i.
  - The matching of g_i from one interval to the next (at the nodes) provides the equations

$$g_i(x_{i+1}) = g_{i+1}(x_{i+1}) \qquad i = 1, N-1$$

  - The matching of the first and second derivatives at each subinterval's boundary provides the equations

$$g_{i-1}'(x_i) = g_i'(x_i) \qquad g_{i-1}''(x_i) = g_i''(x_i)$$

To provide the additional equations needed to determine all constants, the third derivatives at adjacent nodes are matched. Values for the third derivatives are found by approximating them in terms of the second derivatives (our old FD approximation):

$$g_i''' \simeq \frac{g_{i+1}'' - g_i''}{x_{i+1} - x_i}$$

It is straightforward though complicated to solve for all the parameters.

BOUNDARIES

Matching at the boundaries of the intervals results in only N – 2 linear equations for N unknowns. Further input is required.

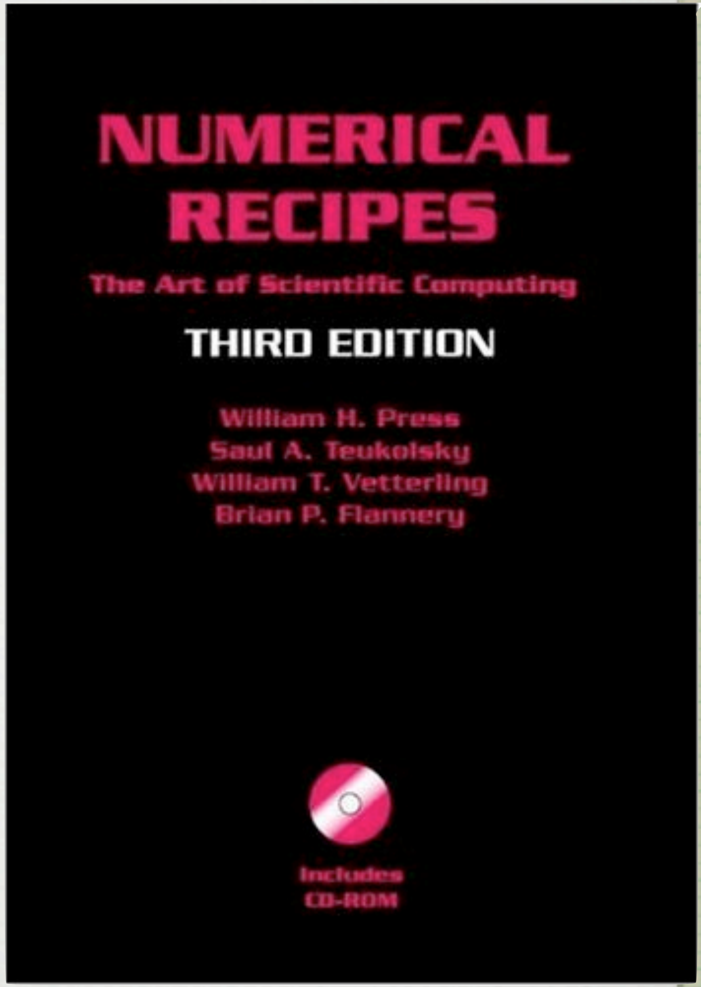
It usually is taken to be the boundary conditions at the endpoints $a = x_1$ and $b = x_N$, specifically, the second derivatives $g''(a)$, and $g''(b)$.

● Natural spline: Set $g''(a) = g''(b) = 0$, (This is “natural” because the derivative vanishes for the flexible spline drafting tool (its ends being free).)

● Input values for g′ at boundaries: The computer uses $g'(a)$ to approximate $g''(a)$. If you do not know the first derivatives, you can calculate them numerically from the table of gi values.

● Input values for g″ at boundaries: Knowing values is of course better than assuming values, but it requires more input. If the values of g″ are not known, they can be approximated by applying a forward-difference approximation to the tabulated values

HARD WORK



NUMERICAL RECIPES

The Art of Scientific Computing

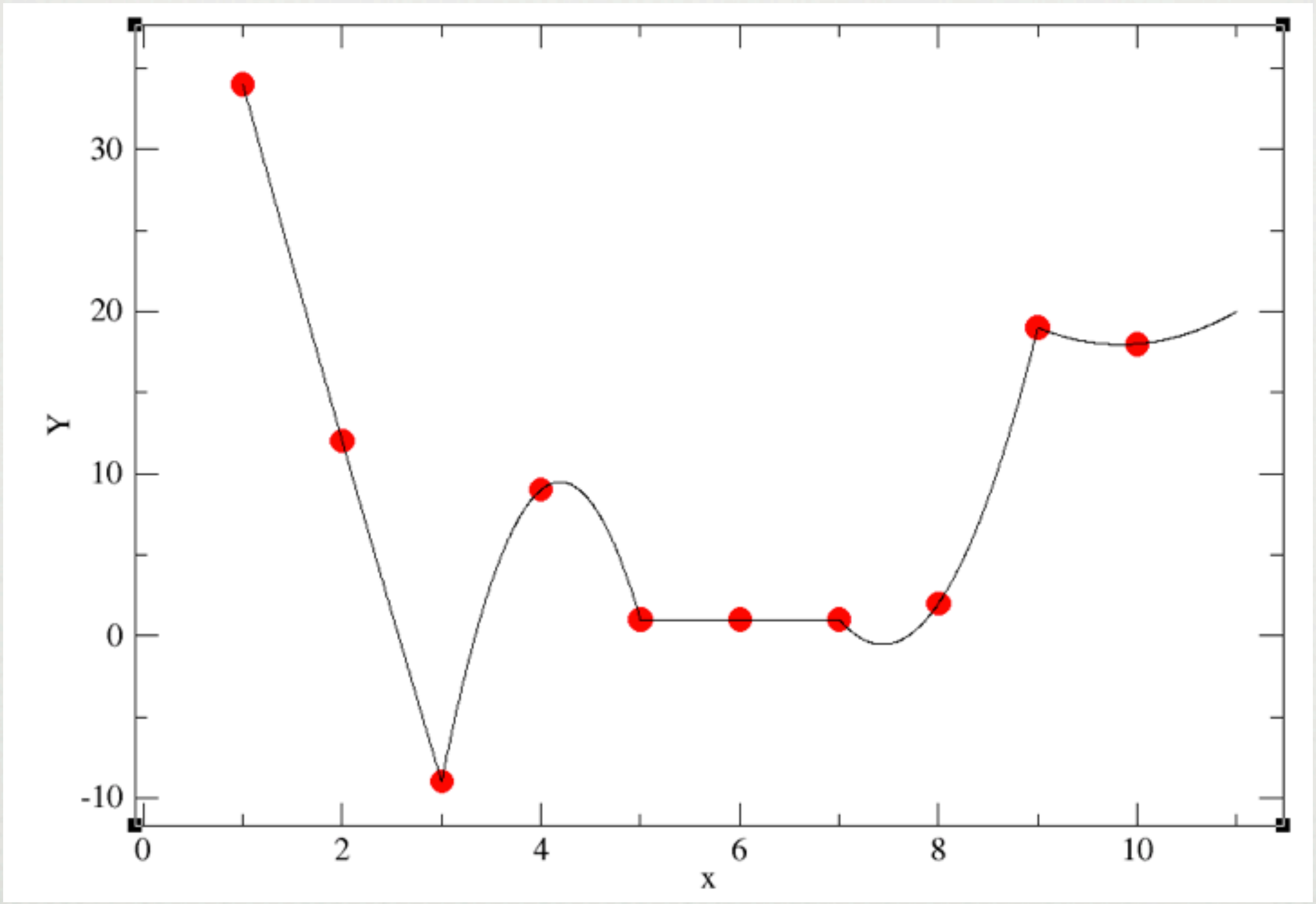
THIRD EDITION

William H. Press
Saul A. Teukolsky
William T. Vetterling
Brian P. Flannery

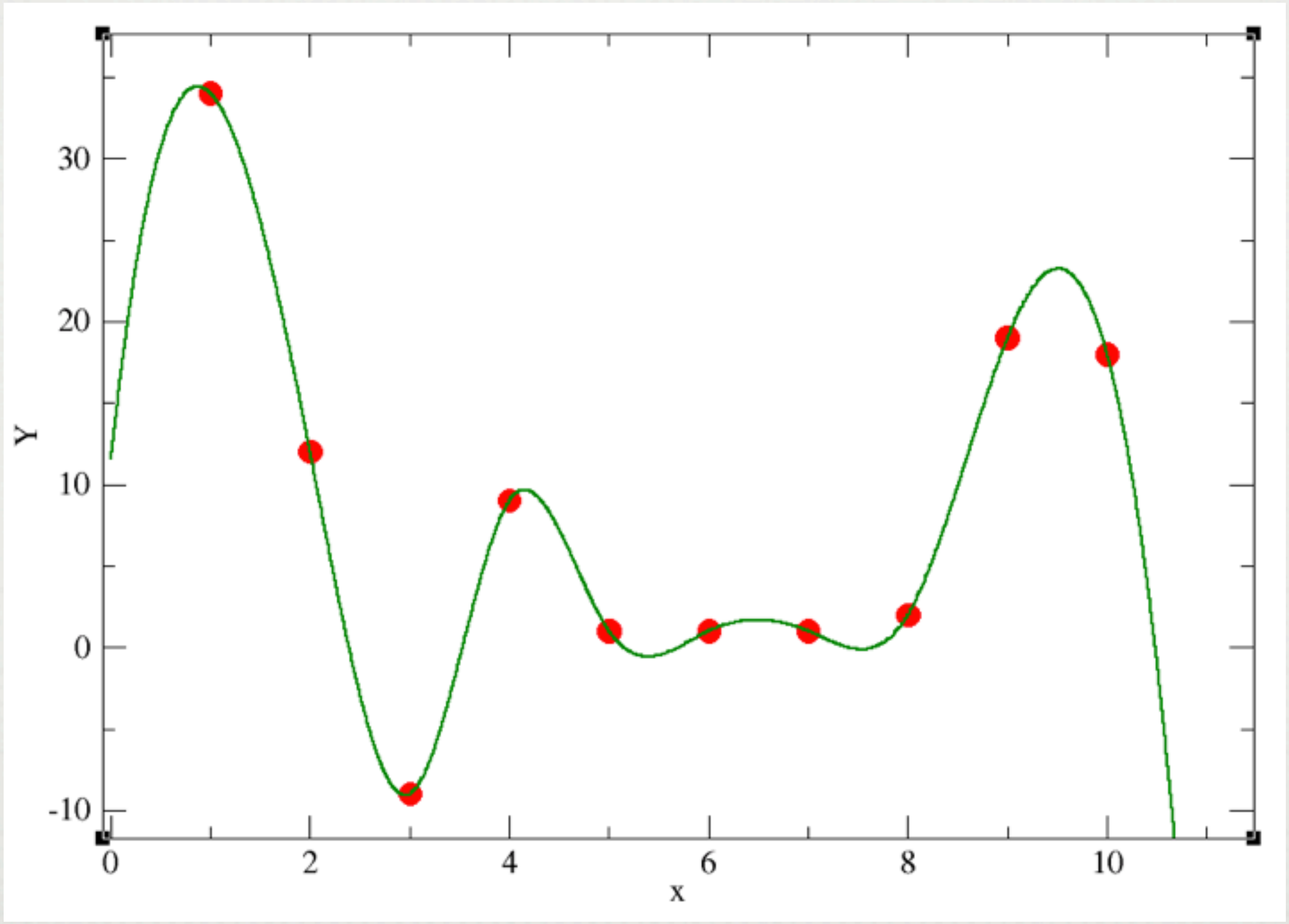
Includes CD-ROM

SPLINE WITH NUMREC

```
#include <iostream>
#include "nr3.h"
#include "interp_1d.h"
using namespace std;
int main () {
   int i,k;
   int kmax=50;
   double xx=0, gxx=0;
   double xxmin, xxmax;
   //
   int npoints;

//GET INPUT
    cin >> npoints;
//allocate memory
   VecDoub x(npoints), g(npoints);
//read in data points
   for (i=0;i<npoints;i++){
       cin >> x[i] >> g[i];
   }
//CREATE SPLINE OBJECT
   Spline_interp myspline (x,g);

//CREATE INTERPOLATION
   xxmin=x[0];
   xxmax=x[npoints-1];
     //
   for(k=0;k<kmax;k++){
       xx=xxmin+(double)k*(xxmax-xxmin)/((double)kmax-1);
       gxx= myspline.interp(xx);
       cout << xx << " " << gxx <<endl;
   }
}
```


30
20
10
0
-10
Y
0
2
4
6
8
10
x

SPLINE!



USING SPLINES FOR INTEGRATION

- **A powerful integration scheme is to fit an integrand with splines, and then integrate the cubic polynomials analytically.**
- **If you have the ability to actually calculate the function for arbitrary x, Gaussian quadrature may be preferable. We know that the spline fit to g in each interval is the cubic (9.8),**

$$g(x) \simeq g_i + g_i'(x-x_i) + \tfrac{1}{2}g_i''(x-x_i)^2 + \tfrac{1}{6}g_i'''(x-x_i)^3$$

- **It is easy to integrate this to obtain the integral of g for this interval, and then to sum over all intervals:**

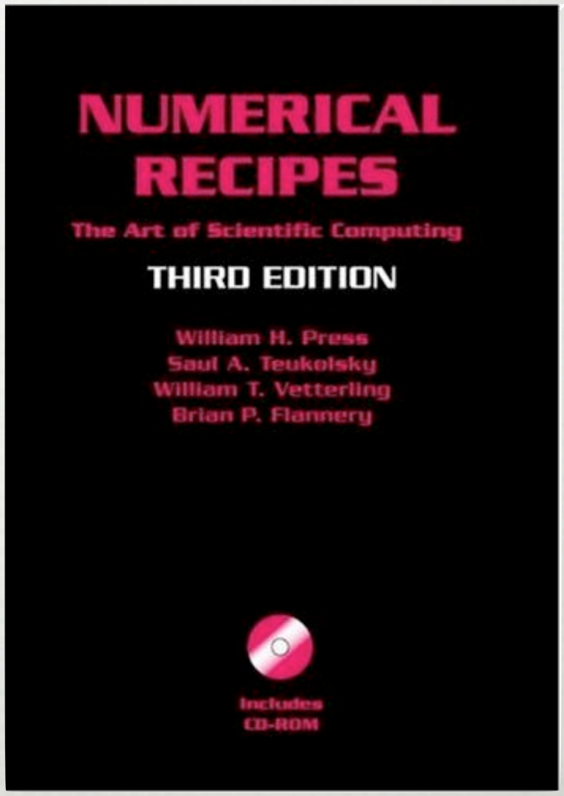
$$\int_{x_i}^{x_{i+1}} g(x)dx \;\simeq\; \left(g_i x + \tfrac{1}{2}g_i' x_i^2 + \tfrac{1}{6}g_i'' x^3 + \tfrac{1}{24}g_i''' x^4\right)\Big|_{x_i}^{x_{i+1}}$$

$$\int_{x_j}^{x_k} g(x)dx = \sum_{i=j}^{k}\int_{x_i}^{x_{i+1}} g(x)dx$$

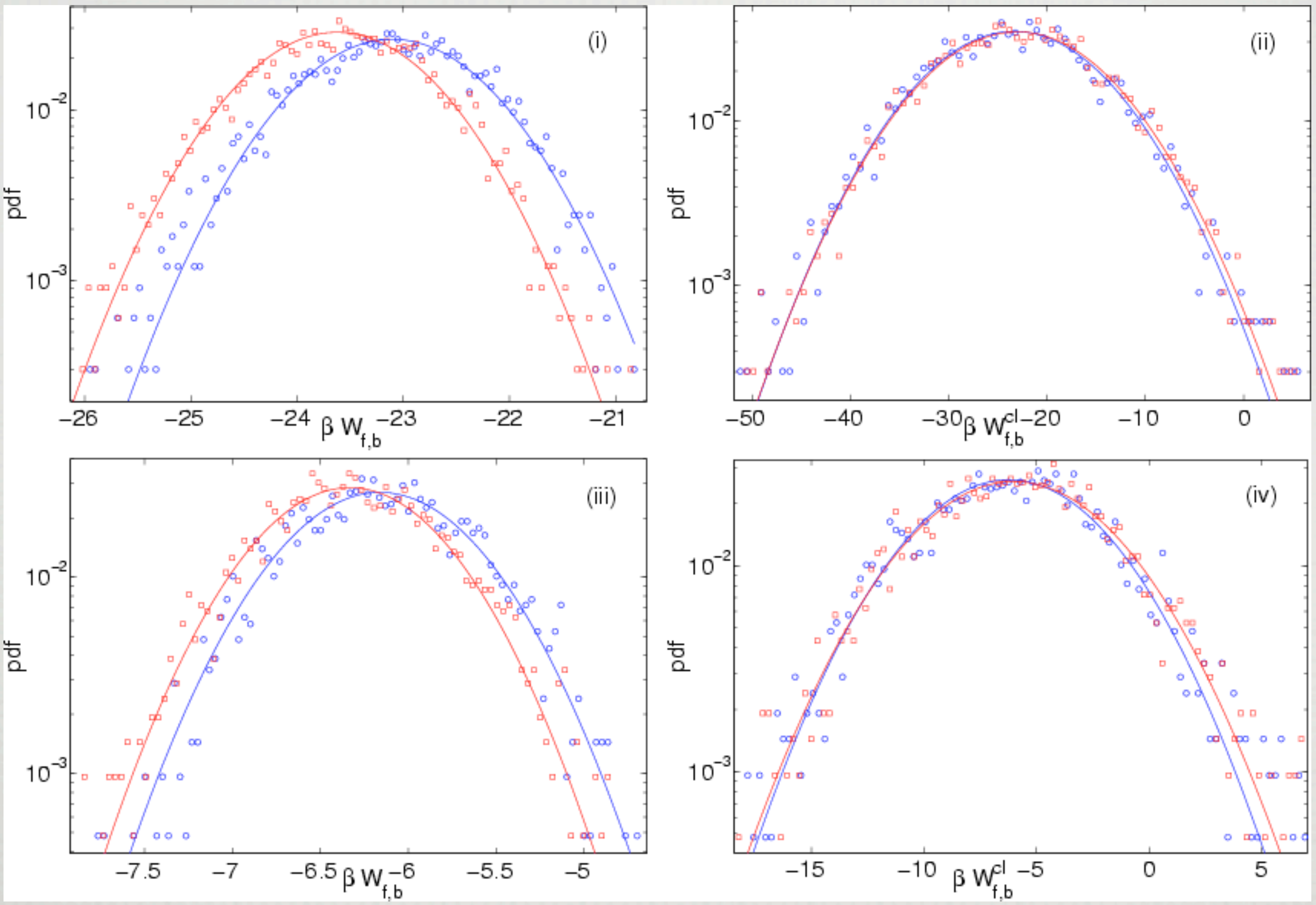
- **Making the intervals smaller does not necessarily increase precision as subtractive cancellations may get large.**

HOW DO WE USE THESE METHODS?

- Preferred method: using numerical libraries
- Go online to nr.com or look in a book...or download the routines...
- Example: spline
- http://www.nr.com/dependencies/index.php



PART 3: FIT TO A FUNCTION (“FIT TO THEORY”)


(i)
(ii)
(iii)
(iv)
pdf
10^{-2}
10^{-3}
$\beta W_{f,b}$
$\beta W^{cl}_{f,b}$
−26
−25
−24
−23
−22
−21
−50
−40
−30
−20
−10
0
−7.5
−7
−6.5
−6
−5.5
−5
−15
−10
−5
5

LEAST SQUARE FITTING: "GOODNESS OF A FIT"

- Imagine that you have measured N_D data values of the independent variable *y* as a function of the dependent variable *x*:

$$(x_i, y_i \pm \sigma_i) \qquad\qquad i = 1, N_D$$

- where $\pm\sigma_i$ is the uncertainty in the ith value of *y*.

- Our goal is to determine how well a mathematical function *y = g(x)* (also called theory or model) can describe these data.

$$\boxed{g(x) = g(x; \{a_1, a_2, \ldots, a_{M_P}\}) = g(x; \{a_m\})}$$

- **We assume that the model function g(x) contains, in addition to the functional dependence on x, an additional dependence upon MP parameters $\{a_1, a_2, \ldots, a_{MP}\}$.**
- **The parameters $\{a_m\}$ are not variables, in the sense of numbers read from a meter, but rather are parts of the theoretical model such as the size of a box, the mass of a particle, or the depth of a potential well.**

MINIMIZATION "CHI-SQUARE"

$$\boxed{\chi^2 \quad \overset{\text{def}}{=} \quad \sum_{i=1}^{N_D} \left(\frac{y_i - g(x_i; \{a_m\})}{\sigma_i} \right)^2}$$

- **We use the chi-squared (χ^2) measure as a gauge of how well a theoretical function g reproduces data**

$$\chi^2 \overset{\text{def}}{=} \sum_{i=1}^{N_D} \left(\frac{y_i - g(x_i; \{a_m\})}{\sigma_i} \right)^2$$

- **The sum is over the ND experimental points (x_i, $y_i \pm \sigma_i$).**
- **The definition is such that smaller values of χ^2 are better fits, with $\chi^2 = 0$ occurring if the theoretical curve went through the center of every data point.**
- **Notice also that the $1/\sigma_i^2$ weighting means that measurements with larger errors contribute less to χ^2.**

- **Least-squares fitting refers to adjusting the theory until a minimum in χ^2 is found**
- **The goal is to find a curve that produces the least value for the summed squares of the deviations of the data from the function *g(x)*.**
- **In general, this is the best fit possible or the best way to determine the parameters in a theory.**
- **The MP parameters {a_m, m = 1, MP} that make χ^2 an extremum are found by solving the MP equations:**

$$\boxed{\frac{\partial \chi^2}{\partial a_m} = 0 \quad \Rightarrow \quad \sum_{i=1}^{N_D} \frac{[y_i - g(x_i)]}{\sigma_i^2} \frac{\partial g(x_i)}{\partial a_m} = 0 \quad (m = 1, M_P)}$$

GOODNESS OF A FIT

- When the deviations from theory are due to random errors and when these errors are described by a Gaussian distribution, there are some useful rules of thumb to remember.

- You know that your fit is good if the value of χ^2 calculated via the definition is approximately equal to the number of degrees of freedom $\chi^2 \approx ND - MP$, where ND is the number of data points and MP the number of parameters in the theoretical function.

- If your χ^2 is much less than $ND - MP$, it does not mean that you have a "great" theory or a really precise measurement; instead, you probably have too many parameters or have assigned errors (σ_i values) that are too large. In fact, too small a χ^2 may indicate that you are fitting the random scatter in the data rather than missing ~ 1 of the error bars, as expected for Gaussian statistics.

FITTING TO A STRAIGHT LINE: LINEAR REGRESSSION

- ☐ The MP simultaneous equations simplify considerably if the functions g(x; {am}) depend linearly on the parameter values a_i:

$$g\left(x;\{a_1,a_2\}\right)=a_1+a_2x$$

- ☐ In this case there are MP = 2 parameters, the slope a_2 and the y intercept a_1.

- ☐ A unique solution is not possible unless the number of data points is equal to or greater than the number of parameters.

- ☐ For this linear case, there are just two derivatives,

$$\frac{\partial g(x_i)}{\partial a_1}=1 \qquad \frac{\partial g(x_i)}{\partial a_2}=x_i$$

- ☐ and after substitution, the χ^2 minimization equations can be solved

LINEAR REGRESSION (2)

$$a_1 = \frac{S_{xx}S_y - S_xS_{xy}}{\Delta} \qquad a_2 = \frac{SS_{xy} - S_xS_y}{\Delta}$$

$$S = \sum_{i=1}^{N_D} \frac{1}{\sigma_i^2} \qquad S_x = \sum_{i=1}^{N_D} \frac{x_i}{\sigma_i^2} \qquad S_y = \sum_{i=1}^{N_D} \frac{y_i}{\sigma_i^2}$$

$$S_{xx} = \sum_{i=1}^{N_D} \frac{x_i^2}{\sigma_i^2} \qquad S_{xy} = \sum_{i=1}^{N_D} \frac{x_iy_i}{\sigma_i^2} \qquad \Delta = SS_{xx} - S_x^2$$

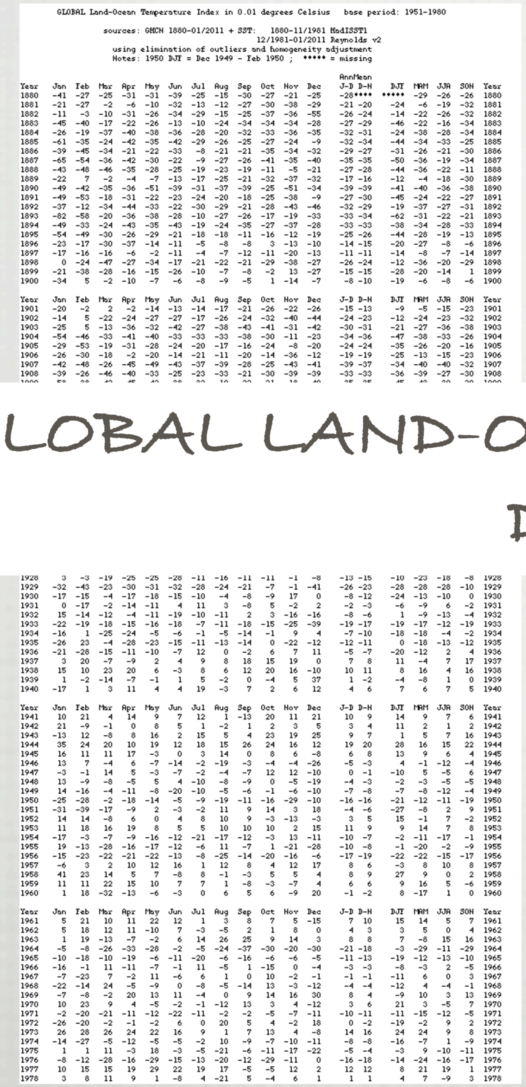
- Statistics also gives you an expression for the variance or uncertainty in the deduced parameters:

$$\sigma_{a_1}^2 = \frac{S_{xx}}{\Delta} \qquad \sigma_{a_2}^2 = \frac{S}{\Delta}$$

LINEAR REGRESSION

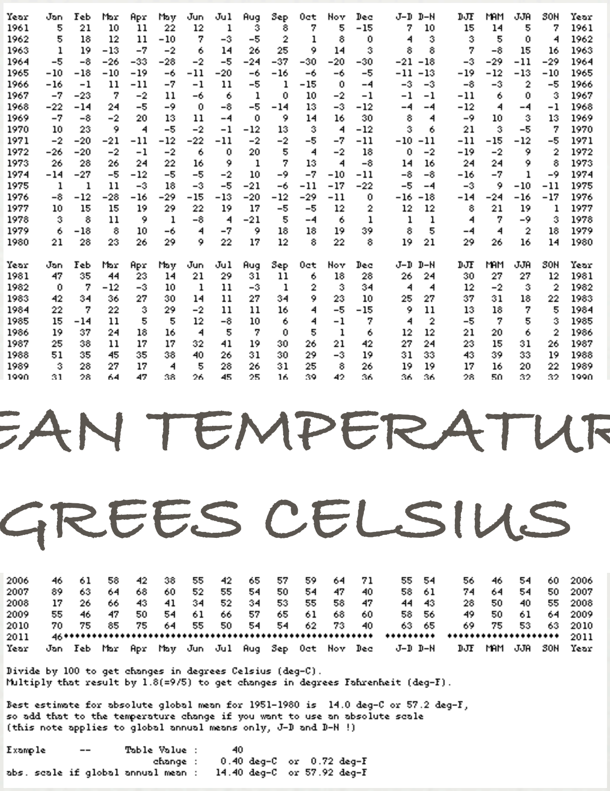
```
#include <iostream>
#include "nr3.h"
#include "gamma.h"
#include "incgammabeta.h"
#include "fitab.h"

using namespace std;

int main () {
   int i,k;
   int kmax=500;
   double xx=0, gxx=0;
   double aa, bb, xxmin, xxmax;
   //
   int npoints;
    cin >> npoints;
   //
   VecDoub x(npoints), g(npoints);
   //
   for (i=0;i<npoints;i++){
        cin >> x[i] >> g[i];
   }
   Fitab myreg(x,g);
//
   xxmin=x[0];
   xxmax=x[npoints-1];
    //
   aa=myreg.a;
   bb=myreg.b;
   cout << aa << " " << bb << " " << myreg.chi2<< endl;

   for(k=0;k<kmax;k++){
        xx=xxmin+(double)k*(xxmax-xxmin)/((double)kmax-1);
        gxx=bb*xx+aa;
        cout << xx << " " << gxx <<endl;
   }
}
```

GLOBAL LAND-OCEAN TEMPERATURE INDEX IN 0.01 DEGREES CELSIUS



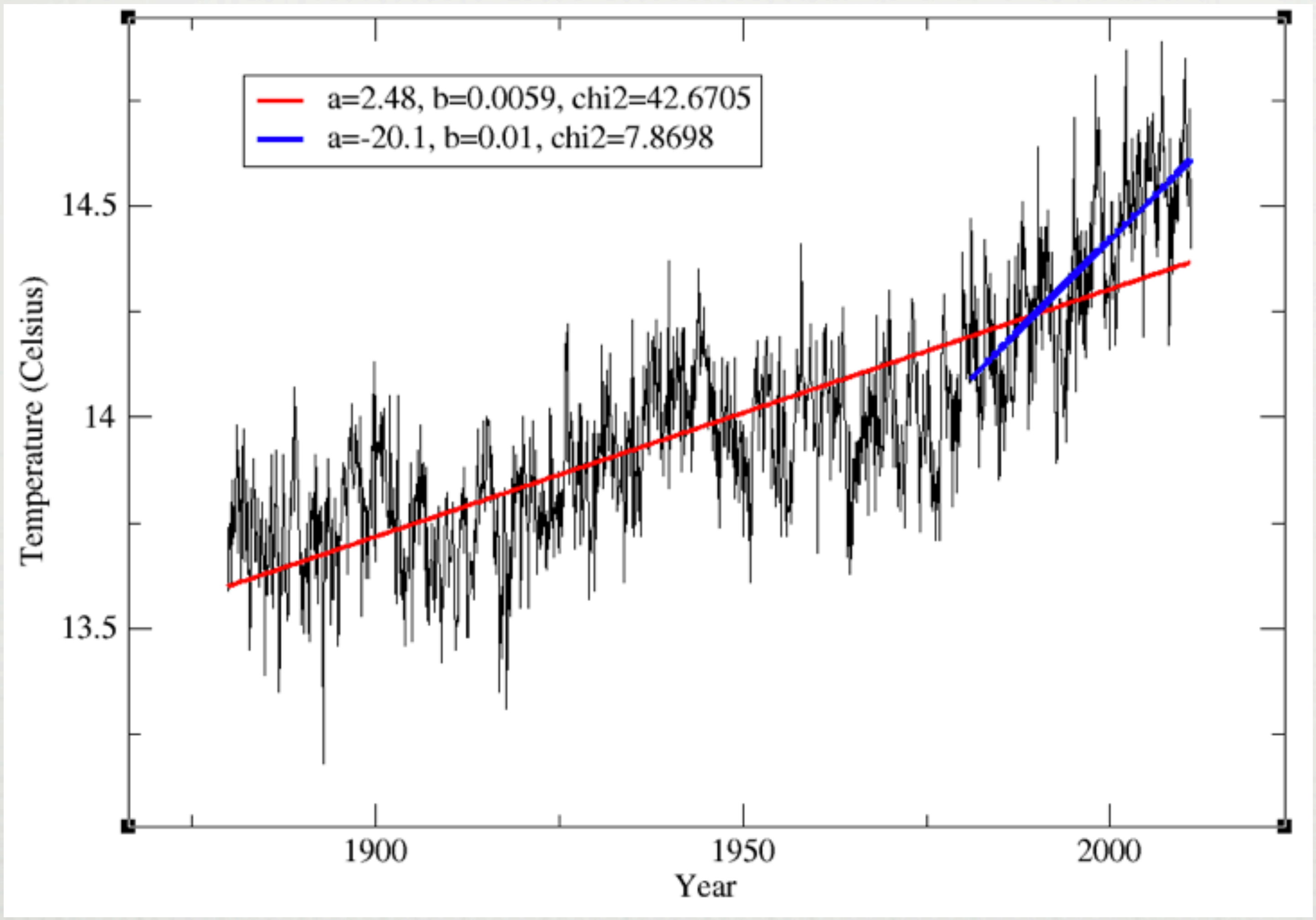
```
        GLOBAL Land-Ocean Temperature Index in 0.01 degrees Celsius   base period: 1951-1980

                 sources: GHCN 1880-01/2011 + SST:   1880-11/1981 HadISST1
                                                     12/1981-01/2011 Reynolds v2
                   using elimination of outliers and homogeneity adjustment
                   Notes: 1950 DJF = Dec 1949 - Feb 1950 ;  ***** = missing

                                                                   AnnMean
Year  Jan  Feb  Mar  Apr  May  Jun  Jul  Aug  Sep  Oct  Nov  Dec    J-D  D-N    DJF  MAM  JJA  SON  Year
1880  -41  -27  -25  -31  -31  -39  -25  -15  -30  -27  -21  -25    -28****  *****  -29  -26  -26  1880
1881  -21  -27   -2   -6  -10  -32  -13  -12  -27  -30  -38  -29    -21  -20    -24   -6  -19  -32  1881
1882  -11   -3  -10  -31  -26  -34  -29  -15  -25  -37  -36  -55    -26  -24    -14  -22  -26  -32  1882
1883  -45  -40  -17  -22  -26  -13  -10  -24  -34  -34  -34  -28    -27  -29    -46  -22  -16  -34  1883
1884  -26  -19  -37  -40  -38  -36  -28  -20  -32  -33  -36  -35    -32  -31    -24  -38  -28  -34  1884
1885  -61  -35  -24  -42  -35  -42  -29  -26  -25  -27  -24   -9    -32  -34    -44  -34  -33  -25  1885
1886  -39  -45  -34  -21  -22  -33   -8  -21  -21  -35  -34  -32    -29  -27    -31  -26  -21  -30  1886
1887  -65  -54  -36  -42  -30  -22   -9  -27  -26  -41  -35  -40    -35  -35    -50  -36  -19  -34  1887
1888  -43  -48  -46  -35  -28  -25  -19  -23  -19  -11   -5  -21    -27  -28    -44  -36  -22  -11  1888
1889  -22    7   -2   -4   -7  -13  -17  -25  -21  -32  -37  -32    -17  -16    -12   -4  -18  -30  1889
1890  -49  -42  -35  -36  -51  -39  -31  -37  -39  -25  -51  -34    -39  -39    -41  -40  -36  -38  1890
1891  -49  -53  -18  -31  -22  -23  -24  -20  -18  -25  -38   -9    -27  -30    -45  -24  -22  -27  1891
1892  -37  -12  -34  -44  -33  -22  -30  -29  -21  -28  -43  -46    -32  -29    -19  -37  -27  -31  1892
1893  -82  -58  -20  -36  -38  -28  -10  -27  -26  -17  -19  -33    -33  -34    -62  -31  -22  -21  1893
1894  -49  -33  -24  -43  -35  -43  -19  -24  -35  -27  -37  -28    -33  -33    -38  -34  -28  -33  1894
1895  -54  -49  -30  -26  -29  -21  -18  -18  -11  -16  -12  -19    -25  -26    -44  -28  -19  -13  1895
1896  -23  -17  -30  -37  -14  -11   -5   -8   -8    3  -13  -10    -14  -15    -20  -27   -8   -6  1896
1897  -17  -16  -16   -6   -2  -11   -4   -7  -12  -11  -20  -13    -11  -11    -14   -8   -7  -14  1897
1898    0  -24  -47  -27  -34  -17  -21  -22  -21  -29  -38  -27    -26  -24    -12  -36  -20  -29  1898
1899  -21  -38  -28  -16  -15  -26  -10   -7   -8   -2   13  -27    -15  -15    -28  -20  -14    1  1899
1900  -34    5   -2  -10   -7   -6   -8   -9   -5    1  -14   -7     -8  -10    -19   -6   -8   -6  1900

Year  Jan  Feb  Mar  Apr  May  Jun  Jul  Aug  Sep  Oct  Nov  Dec    J-D  D-N    DJF  MAM  JJA  SON  Year
1901  -20   -2    2   -2  -14  -13  -14  -17  -21  -26  -22  -26    -15  -13     -9   -5  -15  -23  1901
1902  -14    5  -22  -24  -27  -27  -17  -26  -24  -32  -40  -44    -24  -23    -12  -24  -23  -32  1902
1903  -25    5  -13  -36  -32  -42  -27  -38  -43  -41  -31  -42    -30  -31    -21  -27  -36  -38  1903
1904  -54  -46  -33  -41  -40  -33  -33  -33  -38  -30  -11  -23    -34  -36    -47  -38  -33  -26  1904
1905  -29  -53  -19  -31  -28  -24  -20  -17  -16  -24   -8  -20    -24  -24    -35  -26  -20  -16  1905
1906  -26  -30  -18   -2  -20  -14  -21  -11  -20  -14  -36  -12    -19  -19    -25  -13  -15  -23  1906
1907  -42  -48  -26  -45  -49  -43  -37  -39  -28  -25  -43  -41    -39  -37    -34  -40  -40  -32  1907
1908  -39  -26  -46  -40  -33  -25  -23  -33  -21  -30  -39  -39    -33  -33    -36  -39  -27  -30  1908
```

```
1928    3   -3  -19  -25  -25  -28  -11  -16  -11  -11   -1   -8    -13  -15    -10  -23  -18   -8  1928
1929  -32  -43  -23  -30  -31  -32  -28  -24  -21   -7   -1  -41    -26  -23    -28  -28  -28  -10  1929
1930  -17  -15   -4  -17  -18  -15  -10   -4   -8   -9   17    0     -8  -12    -24  -13  -10    0  1930
1931    0  -17   -2  -14  -11    4   11    3   -8    5   -2    2     -2   -3     -6   -9    6   -2  1931
1932   15  -14  -12   -4  -11  -19  -10  -11    2    3  -16  -16     -8   -6      1   -9  -13   -4  1932
1933  -22  -19  -18  -15  -16  -18   -7  -11  -18  -15  -25  -39    -19  -17    -19  -17  -12  -19  1933
1934  -16    1  -25  -24   -5   -6   -1   -5  -14   -1    9    4     -7  -10    -18  -18   -4   -2  1934
1935  -26   23   -4  -28  -23  -15  -11  -13  -14    0  -22  -12    -12  -11      0  -18  -13  -12  1935
1936  -21  -28  -15  -11  -10   -7   12    0   -2    6    7   11     -5   -7    -20  -12    2    4  1936
1937    3   20   -7   -9    2    4    9    8   18   15   19    0      7    8     11   -4    7   17  1937
1938   15   10   23   20    6   -3    8    6   12   20   16  -10     10   11      8   16    4   16  1938
1939    1   -2  -14   -7   -1    1    5   -2    0   -4    5   37      1   -2     -4   -8    1    0  1939
1940  -17    1    3   11    4    4   19   -3    7    2    6   12      4    6      7    6    7    5  1940

Year  Jan  Feb  Mar  Apr  May  Jun  Jul  Aug  Sep  Oct  Nov  Dec    J-D  D-N    DJF  MAM  JJA  SON  Year
1941   10   21    4   14    9    7   12    1  -13   20   11   21     10    9     14    9    7    6  1941
1942   21   -9   -1    0    8    5    1   -2    1    2    3    5      3    4     11    2    1    2  1942
1943  -13   12   -8    8   16    2   15    5    4   23   19   25      9    7      1    5    7   16  1943
1944   35   24   20   10   19   12   18   15   26   24   16   12     19   20     28   16   15   22  1944
1945   16   11   11   17   -3    0    3   14    0    8    6   -8      6    8     13    9    6    4  1945
1946   13    7   -4    6   -7  -14   -2  -19   -3   -4   -4  -26     -5   -3      4   -1  -12   -4  1946
1947   -3   -1   14    5   -3   -7   -2   -4   -7   12   12  -10      0   -1    -10    5   -5    6  1947
1948   13   -9   -8   -5    5    4  -10   -8   -9    0   -5  -19     -4   -3     -2   -3   -5   -5  1948
1949   14  -16   -4  -11   -8  -20  -10   -5   -6   -1   -6  -10     -7   -8     -7   -8  -12   -4  1949
1950  -25  -28   -2  -18  -14   -5   -9  -19  -11  -16  -29  -10    -16  -16    -21  -12  -11  -19  1950
1951  -31  -39  -17   -9    2   -3   -2   11    9   14    3   18     -4   -6    -27   -8    2    9  1951
1952   14   14   -8    6    0    4    8   10    9   -3  -13   -3      3    5     15   -1    7   -2  1952
1953   11   18   16   19    8    5    5   10   10   10    2   15     11    9      9   14    7    8  1953
1954  -17   -3   -7   -9  -16  -12  -21  -17  -12   -3   13  -11    -10   -7     -2  -11  -17   -1  1954
1955   19  -13  -28  -16  -17  -12   -6   11   -7    1  -21  -28    -10   -8     -1  -20   -2   -9  1955
1956  -15  -23  -22  -21  -22  -13   -8  -25  -14  -20  -16   -6    -17  -19    -22  -22  -15  -17  1956
1957   -6    3    2   10   12   16    1   12    8    4   12   17      8    6     -3    8   10    8  1957
1958   41   23   14    5    7   -8    8   -1   -3    5    5    4      8    9     27    9    0    2  1958
1959   11   11   22   15   10    7    7    1   -8   -3   -7    4      6    6      9   16    5   -6  1959
1960    1   18  -32  -13   -6   -3    0    6    5    6   -9   20     -1   -2      8  -17    1    0  1960

Year  Jan  Feb  Mar  Apr  May  Jun  Jul  Aug  Sep  Oct  Nov  Dec    J-D  D-N    DJF  MAM  JJA  SON  Year
1961    5   21   10   11   22   12    1    3    8    7    5  -15      7   10     15   14    5    7  1961
1962    5   18   12   11  -10    7   -3   -5    2    1    8    0      4    3      3    5    0    4  1962
1963    1   19  -13   -7   -2    6   14   26   25    9   14    3      8    8      7   -8   15   16  1963
1964   -5   -8  -26  -33  -28   -2   -5  -24  -37  -30  -20  -30    -21  -18     -3  -29  -11  -29  1964
1965  -10  -18  -10  -19   -6  -11  -20   -6  -16   -6   -6   -5    -11  -13    -19  -12  -13  -10  1965
1966  -16   -1   11  -11   -7   -1   11   -5    1  -15    0   -4     -3   -3     -8   -3    2   -5  1966
1967   -7  -23    7   -2   11   -6    6    1    0   10   -2   -1     -1   -1    -11    6    0    3  1967
1968  -22  -14   24   -5   -9    0   -8   -5  -14   13   -3  -12     -4   -4    -12    4   -4   -1  1968
1969   -7   -8   -2   20   13   11   -4    0    9   14   16   30      8    4     -9   10    3   13  1969
1970   10   23    9    4   -5   -2   -1  -12   13    3    4  -12      3    6     21    3   -5    7  1970
1971   -2  -20  -21  -11  -12  -22  -11   -2   -2   -5   -7  -11    -10  -11    -11  -15  -12   -5  1971
1972  -26  -20   -2   -1   -2    6    0   20    5    4   -2   18      0   -2    -19   -2    9    2  1972
1973   26   28   26   24   22   16    9    1    7   13    4   -8     14   16     24   24    9    8  1973
1974  -14  -27   -5  -12   -5   -5   -2   10   -9   -7  -10  -11     -8   -8    -16   -7    1   -9  1974
1975    1    1   11   -3   18   -3   -5  -21   -6  -11  -17  -22     -5   -4     -3    9  -10  -11  1975
1976   -8  -12  -28  -16  -29  -15  -13  -20  -12  -29  -11    0    -16  -18    -14  -24  -16  -17  1976
1977   10   15   15   19   29   22   19   17   -5   -5   12    2     12   12      8   21   19    1  1977
1978    3    8   11    9    1   -8    4  -21    5   -4    6    1      1    1      4    7   -9    3  1978
```



```
Year  Jan  Feb  Mar  Apr  May  Jun  Jul  Aug  Sep  Oct  Nov  Dec    J-D  D-N    DJF  MAM  JJA  SON  Year
1961    5   21   10   11   22   12    1    3    8    7    5  -15      7   10     15   14    5    7  1961
1962    5   18   12   11  -10    7   -3   -5    2    1    8    0      4    3      3    5    0    4  1962
1963    1   19  -13   -7   -2    6   14   26   25    9   14    3      8    8      7   -8   15   16  1963
1964   -5   -8  -26  -33  -28   -2   -5  -24  -37  -30  -20  -30    -21  -18     -3  -29  -11  -29  1964
1965  -10  -18  -10  -19   -6  -11  -20   -6  -16   -6   -6   -5    -11  -13    -19  -12  -13  -10  1965
1966  -16   -1   11  -11   -7   -1   11   -5    1  -15    0   -4     -3   -3     -8   -3    2   -5  1966
1967   -7  -23    7   -2   11   -6    6    1    0   10   -2   -1     -1   -1    -11    6    0    3  1967
1968  -22  -14   24   -5   -9    0   -8   -5  -14   13   -3  -12     -4   -4    -12    4   -4   -1  1968
1969   -7   -8   -2   20   13   11   -4    0    9   14   16   30      8    4     -9   10    3   13  1969
1970   10   23    9    4   -5   -2   -1  -12   13    3    4  -12      3    6     21    3   -5    7  1970
1971   -2  -20  -21  -11  -12  -22  -11   -2   -2   -5   -7  -11    -10  -11    -11  -15  -12   -5  1971
1972  -26  -20   -2   -1   -2    6    0   20    5    4   -2   18      0   -2    -19   -2    9    2  1972
1973   26   28   26   24   22   16    9    1    7   13    4   -8     14   16     24   24    9    8  1973
1974  -14  -27   -5  -12   -5   -5   -2   10   -9   -7  -10  -11     -8   -8    -16   -7    1   -9  1974
1975    1    1   11   -3   18   -3   -5  -21   -6  -11  -17  -22     -5   -4     -3    9  -10  -11  1975
1976   -8  -12  -28  -16  -29  -15  -13  -20  -12  -29  -11    0    -16  -18    -14  -24  -16  -17  1976
1977   10   15   15   19   29   22   19   17   -5   -5   12    2     12   12      8   21   19    1  1977
1978    3    8   11    9    1   -8    4  -21    5   -4    6    1      1    1      4    7   -9    3  1978
1979    6  -18    8   10   -6    4   -7    9   18   18   19   39      8    5     -4    4    2   18  1979
1980   21   28   23   26   29    9   22   17   12    8   22    8     19   21     29   26   16   14  1980

Year  Jan  Feb  Mar  Apr  May  Jun  Jul  Aug  Sep  Oct  Nov  Dec    J-D  D-N    DJF  MAM  JJA  SON  Year
1981   47   35   44   23   14   21   29   31   11    6   18   28     26   24     30   27   27   12  1981
1982    0    7  -12   -3   10    1   11   -3    1    2    3   34      4    4     12   -2    3    2  1982
1983   42   34   36   27   30   14   11   27   34    9   23   10     25   27     37   31   18   22  1983
1984   22    7   22    3   29   -2   11   11   16    4   -5  -15      9   11     13   18    7    5  1984
1985   15  -14   11    5    5   12   -8   10    6    4   -1    7      4    2     -5    7    5    3  1985
1986   19   37   24   18   16    4    5    7    0    5    1    6     12   12     21   20    6    2  1986
1987   25   38   11   17   17   32   41   19   30   26   21   42     27   24     23   15   31   26  1987
1988   51   35   45   35   38   40   26   31   30   29   -3   19     31   33     43   39   33   19  1988
1989    3   28   27   17    4    5   28   26   31   25    8   26     19   19     17   16   20   22  1989
1990   31   28   64   47   38   26   45   25   16   39   42   36     36   36     28   50   32   32  1990
```

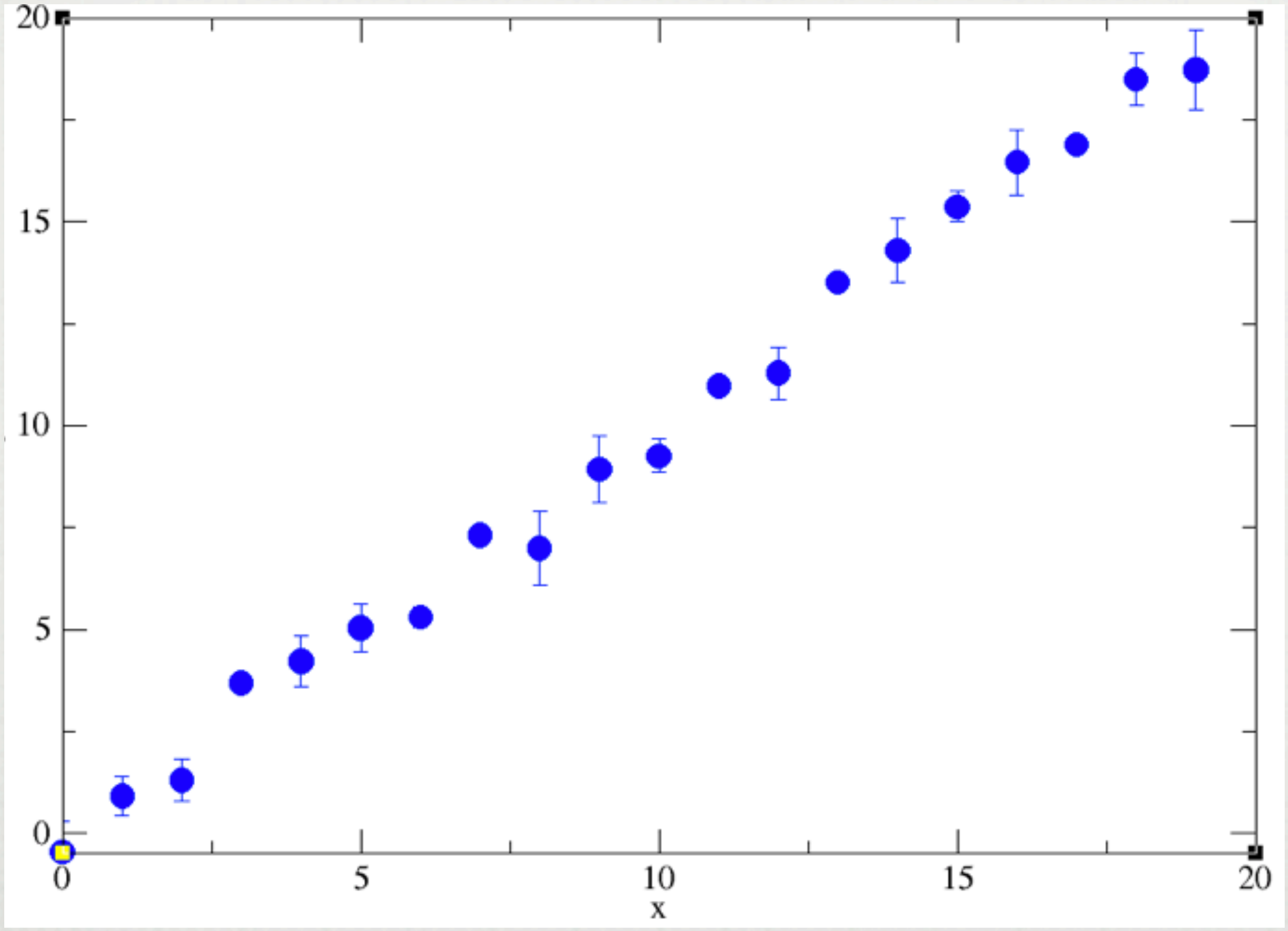
```
2006   46   61   58   42   38   55   42   65   57   59   64   71     55   54     56   46   54   60  2006
2007   89   63   64   68   60   52   55   54   50   54   47   40     58   61     74   64   54   50  2007
2008   17   26   66   43   41   34   52   34   53   55   58   47     44   43     28   50   40   55  2008
2009   55   46   47   50   54   61   66   57   65   61   68   60     58   56     49   50   61   64  2009
2010   70   75   85   75   64   55   50   54   54   62   73   40     63   65     69   75   53   63  2010
2011   46*******************************************************  **********  ********************  2011
Year  Jan  Feb  Mar  Apr  May  Jun  Jul  Aug  Sep  Oct  Nov  Dec    J-D  D-N    DJF  MAM  JJA  SON  Year

Divide by 100 to get changes in degrees Celsius (deg-C).
Multiply that result by 1.8(=9/5) to get changes in degrees Fahrenheit (deg-F).

Best estimate for absolute global mean for 1951-1980 is  14.0 deg-C or 57.2 deg-F,
so add that to the temperature change if you want to use an absolute scale
(this note applies to global annual means only, J-D and D-N !)

Example      --      Table Value :      40
                          change :    0.40 deg-C  or  0.72 deg-F
abs. scale if global annual mean :   14.40 deg-C  or 57.92 deg-F
```


a=2.48, b=0.0059, chi2=42.6705
a=-20.1, b=0.01, chi2=7.8698
14.5
14
13.5
Temperature (Celsius)
1900
1950
2000
Year

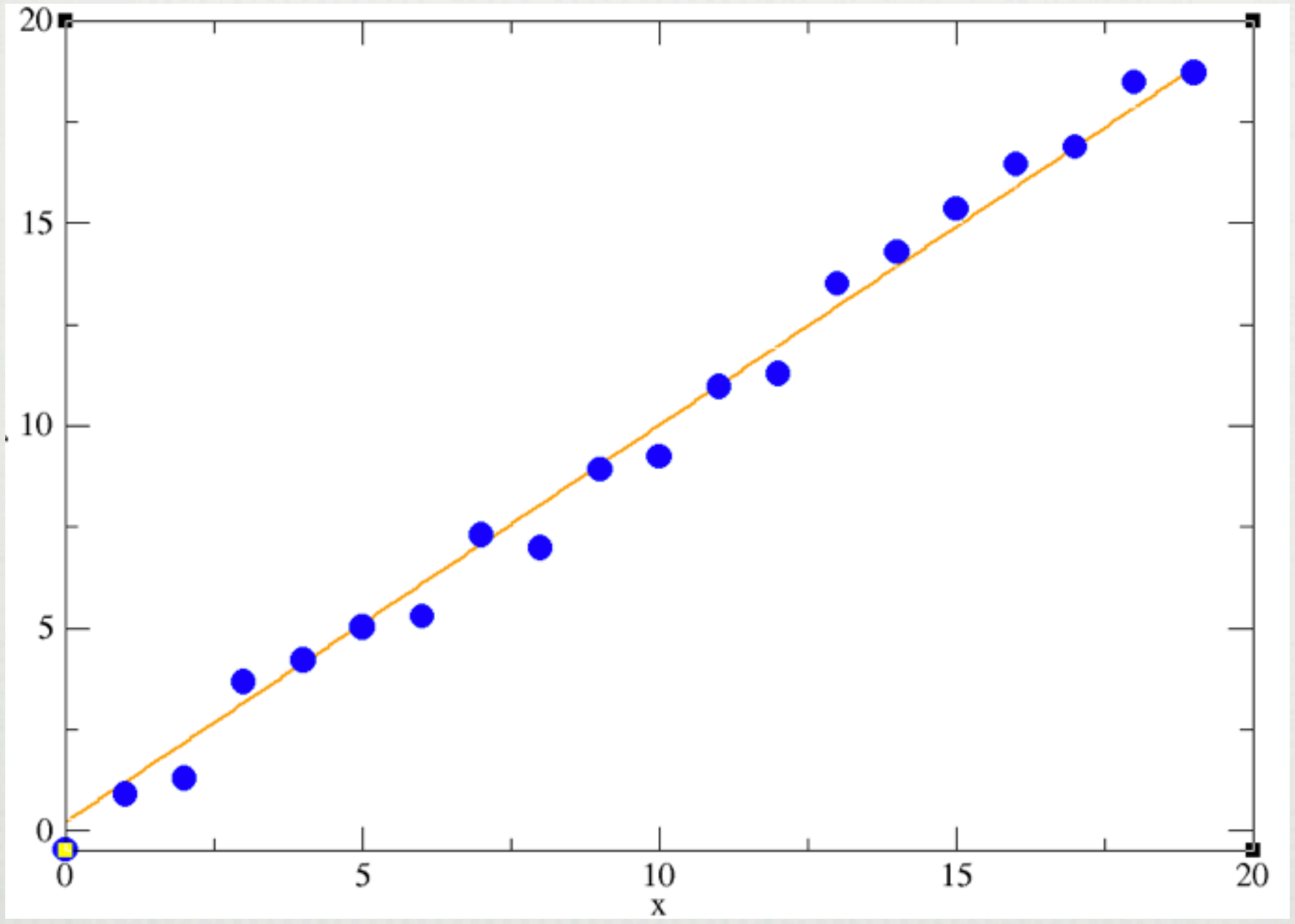
WITH EXPERIMENTAL ERROR!

```
#include <iostream>
#include "nr3.h"
#include "gamma.h"
#include "incgammabeta.h"
#include "fitab.h"

using namespace std;
double randDouble(double low, double high)
{
    double temp;
    temp = ((double) rand() / (static_cast<double>(RAND_MAX) + 1.0))* (high - low) + low;
    return temp;
}
int main () {
    int i,k;
    int kmax=500;
    double xx=0, gxx=0;
    double aa, bb, xxmin, xxmax;
    //
    int npoints=20;
    srand((unsigned) time(0));
    VecDoub x(npoints), g(npoints), sigma(npoints);

    for (i=0;i<npoints;i++){
         x[i]=double(i);
         g[i]=randDouble(-1.0,1.0)+x[i];
         sigma[i]=randDouble(-1.,1.0);
         cout << x[i] << " " << g[i] << " " << sigma[i]<< endl;
    }
    Fitab myreg(x,g,sigma);

     //
    xxmin=x[0];
    xxmax=x[npoints-1];
     //
    aa=myreg.a;
    bb=myreg.b;
    cout << aa << " " << bb << " " << myreg.chi2<< endl;
    for(k=0;k<kmax;k++){
         xx=xxmin+(double)k*(xxmax-xxmin)/((double)kmax-1);
         gxx=bb*xx+aa;
         cout << xx << " " << gxx <<endl;
    }
}
```

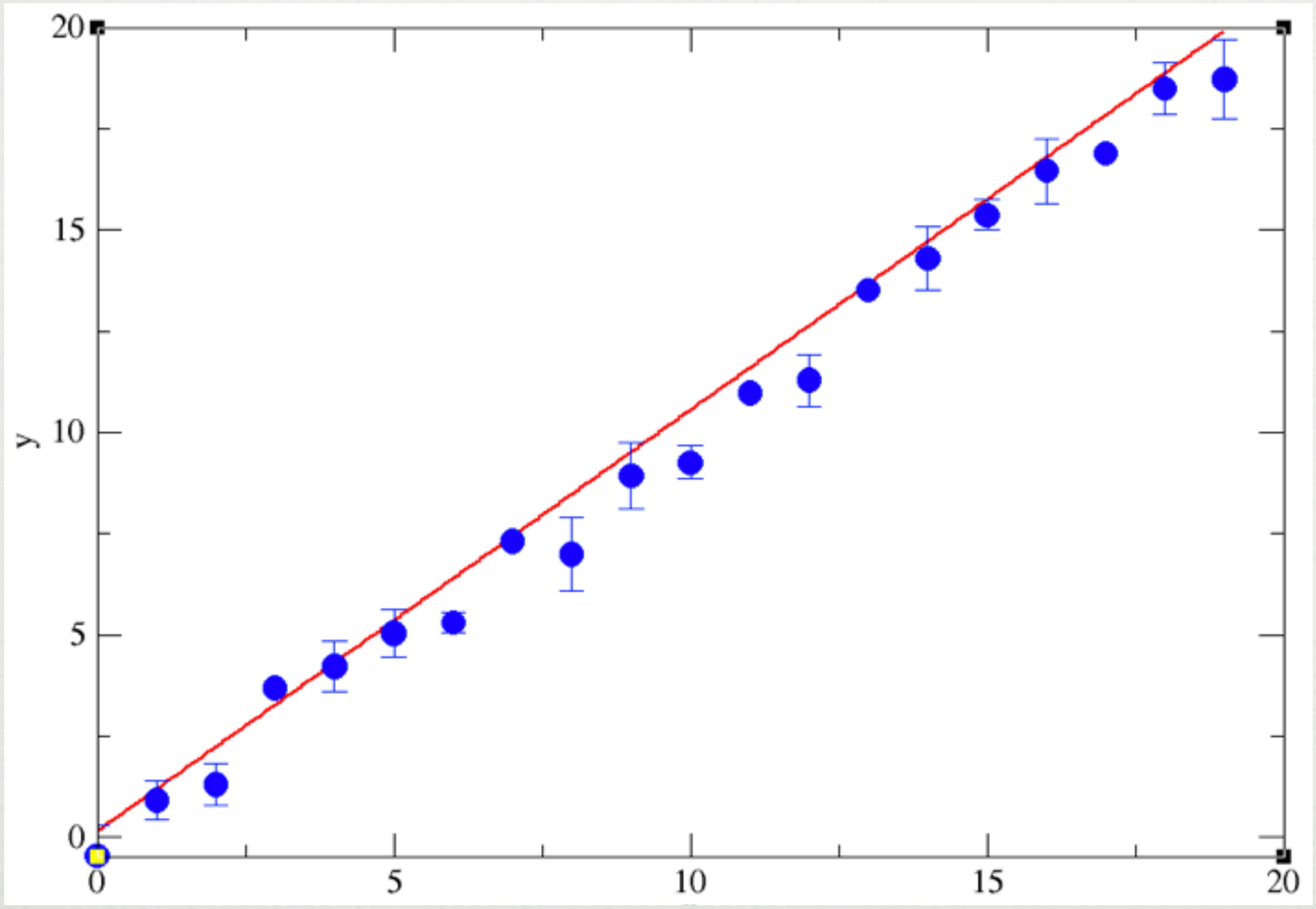
DATA SET



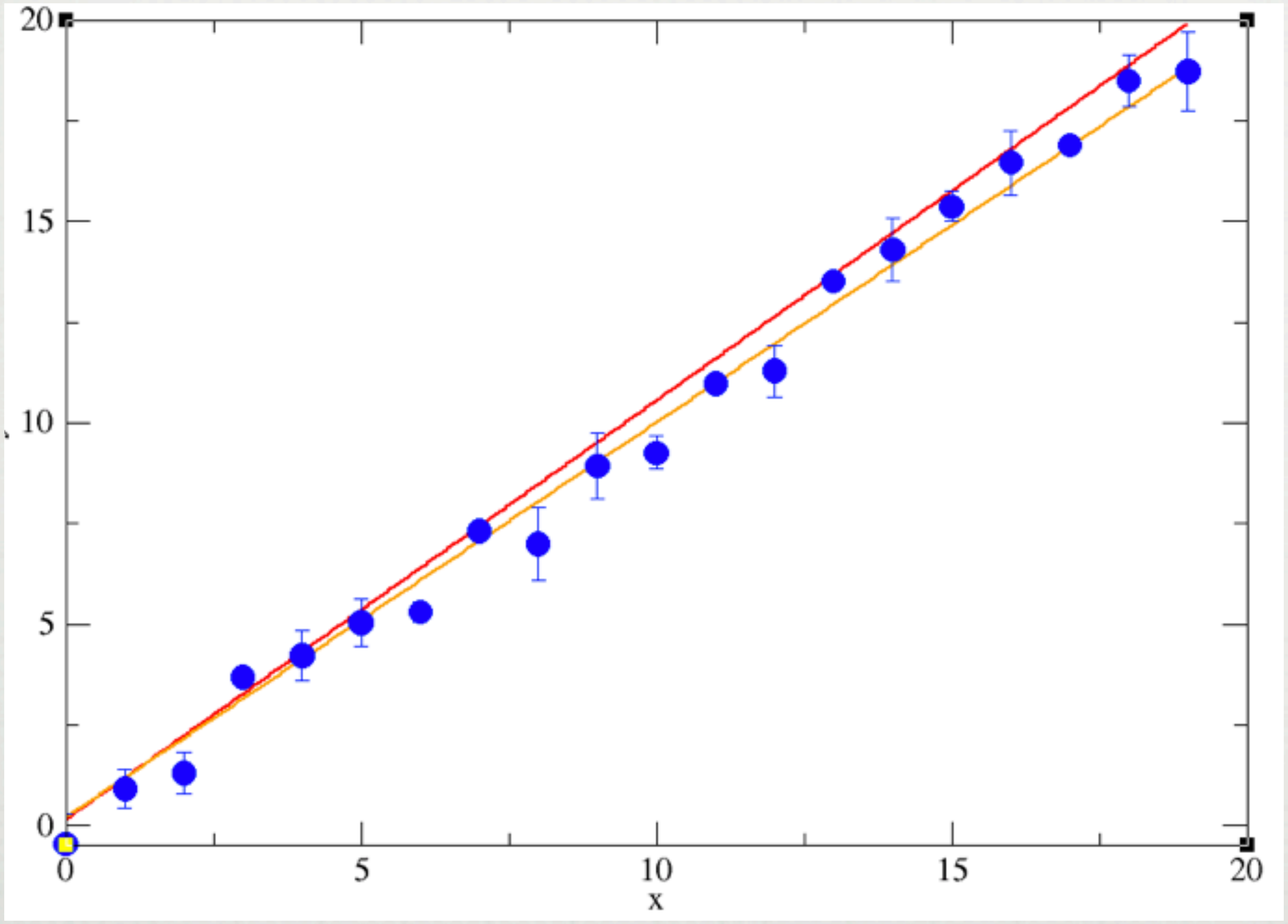
REGRESSION



REGRESSION WITH ERRORS



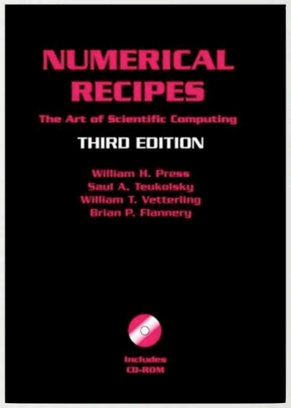
COMPARISON



SUMMARY

- Polynomial fits
  - Lagrange formula and Neville's algorithm
  - Can fit polynomial of order N-1 passing exactly through N points
- Spline fits
  - More pleasing for the eye
  - 3rd order polynomial with continuity of slope and curvature
- Least Xi^2 fits
  - best fit does not necessarily goes though all the points
  - very powerful when we know the analytical form of the solution

IN PRACTICE



- ☐ **We use “numerical recipes”**
- ☐ **Today: we will work on computing splines using numrec.**
- ☐ **See LMS**

SPLINE WITH NUMREC

```
#include <iostream>
#include "nr3.h"
#include "interp_1d.h"
using namespace std;
int main () {
    int i,k;
    int kmax=50;
    double xx=0, gxx=0;
    double xxmin, xxmax;
    //
    int npoints;

//GET INPUT
    cin >> npoints;
//allocate memory
    VecDoub x(npoints), g(npoints);
//read in data points
    for (i=0;i<npoints;i++){
        cin >> x[i] >> g[i];
    }
//CREATE SPLINE OBJECT
    Spline_interp myspline (x,g);

//CREATE INTERPOLATION
    xxmin=x[0];
    xxmax=x[npoints-1];
     //
    for(k=0;k<kmax;k++){
        xx=xxmin+(double)k*(xxmax-xxmin)/((double)kmax-1);
        gxx= myspline.interp(xx);
        cout << xx << " " << gxx <<endl;
    }
}
```