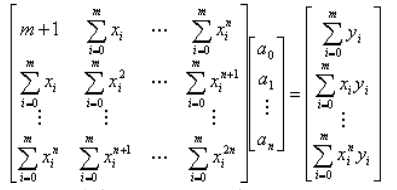
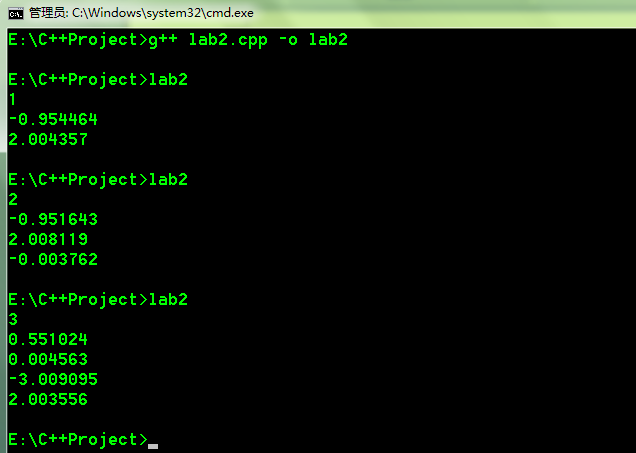
对给定的7个数据点分别用一次，二次和三次多项式进行最小二乘法拟合。

当用n次多项式拟合m + 1个数据点时，通过对误差函数求极值，并选取满足取得极值条件的系数即可完成最小二乘拟合。可证明，n次多项式的系数a[n + 1]的求解可以转换为如下的线性方程组：



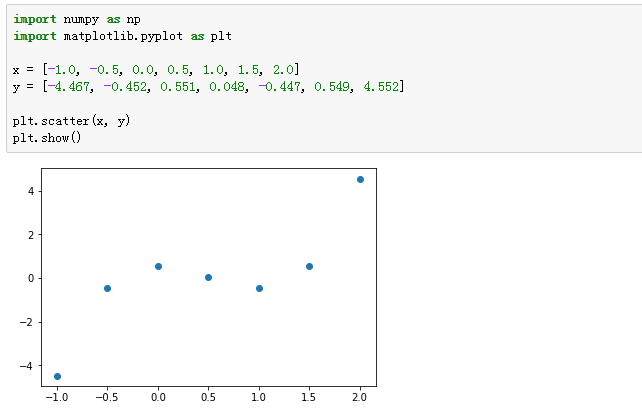
算法只需构造出系数矩阵，采用高斯消元法求解方程组即可得到最小二乘的拟合多项式结果。代码见lab2.cpp，在主函数中构造系数矩阵，利用高斯消元法求解方程组输出结果。运行程序输入n次多项式的系数即可。注意：代码中高斯消元法的求解仅适合与原系数矩阵对角线上不存在0元素的情况，不满足此条件所需要的初等行列变换未予考虑，但就解决本拟合问题能够正常输出结果。

运次过程如下：

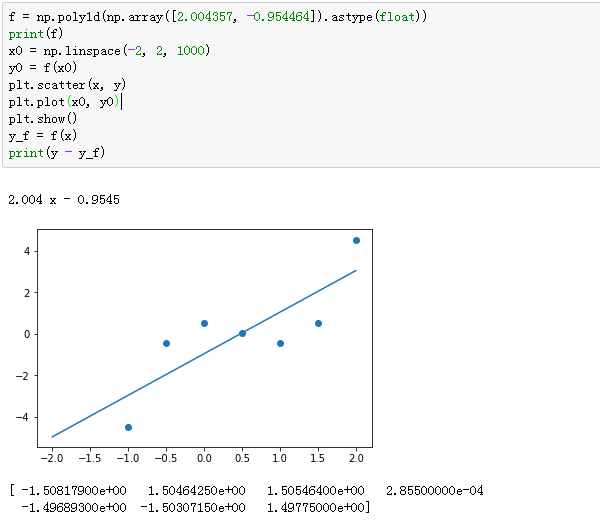


绘图和输出误差采用python实现，运行环境python3.x + jupyter notebook

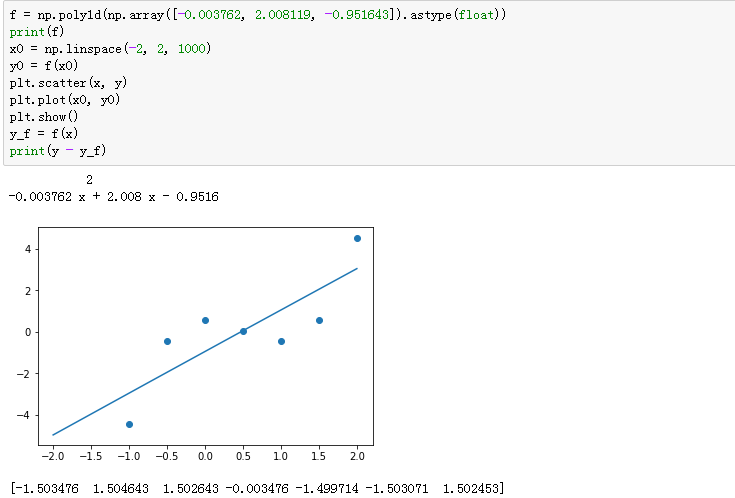
绘制散点图



线性函数拟合及其误差：

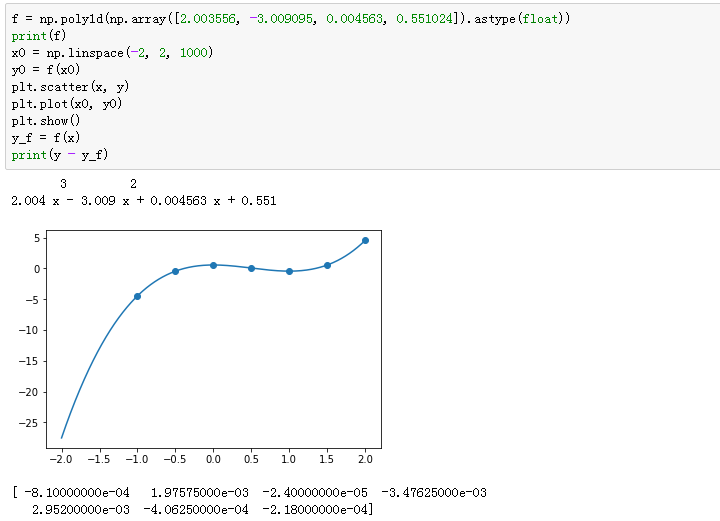


抛物线拟合及其误差：



可以观察到拟合结果和线性拟合及其相近，从数据点的分布也可以看出数据不具有明显的二次关系。

三次函数拟合和误差：



可见拟合效果远好于线性拟合和抛物线拟合。散点图中表现的增减性变化即可看出数据关系适合三次函数拟合。拟合结果十分满足预期。

综合来看：多项式的最小二乘拟合对于多项式的系数选择十分敏感，较好的次数选择有时能够恰到好处的拟合数据。当次数选择得不恰当时，增大次数并不能使结果得到改进，正如实验中采用线性拟合和抛物线拟合几乎没有差别一样，当数据明显不具有某种多项式的性质时，强行拟合的结果往往是徒增计算量。

拟合多项式次数的选择需要通过对散掉图的观察来得到。本次实验中可以明显看出数据的三次函数特性。