

7.5 实例解析

——国家财政数据趋势演算

- 国家统计局每年会公开许多数据，比如国民经济核算指标等。国家统计局公布的大部分数据都以二维表格形式展现，然而，藏在这些数据背后的价值要比表格所展现的更多。

- 以国家财政收支的公开数据为例，这里展示如何利用Python挖掘数据变化的规律。将从网上获取的公开信息存为finance.csv文件

指标	2000年	2015年	2016年
全部收入	13395. 2	152269. 2	159605
中央收入	6989. 2	69267. 2	72365. 6
地方收入	6406. 1	83002	87239. 4
全部支出	15886. 5	175877. 8	187755. 2
中央支出	5519. 9	25542. 2	27403. 9
地方支出	10366. 7	150335. 6	160351. 4

- 是否可以根据这些数据，估算出所缺少年份的大致收支情况甚至是预测未来呢？

国家财政数据趋势演算

- 由个别数据预测规律属于数值分析的内容，可以通过线性回归方程建立简单模型，线性回归方程的公式为：

$$\hat{y} = b\hat{x} + a$$

年份

- 其中， $b = \frac{\sum_{i=1}^n x_i y_i - n\bar{x}\bar{y}}{\sum_{i=1}^n x_i^2 - n\bar{x}^2}$ ， $a = \bar{y} - b\bar{x}$
- 年份对应的数值

■ 题意：

■ 我们要做的事情就是，通过现有数值

指标	2000年	2015年	2016年
全部收入	13395.2	152269.2	159605
中央收入	6989.2	69267.2	72365.6
地方收入	6406.1	83002	87239.4
全部支出	15886.5	175877.8	187755.2
中央支出	5519.9	25542.2	27403.9
地方支出	10366.7	150335.6	160351.4

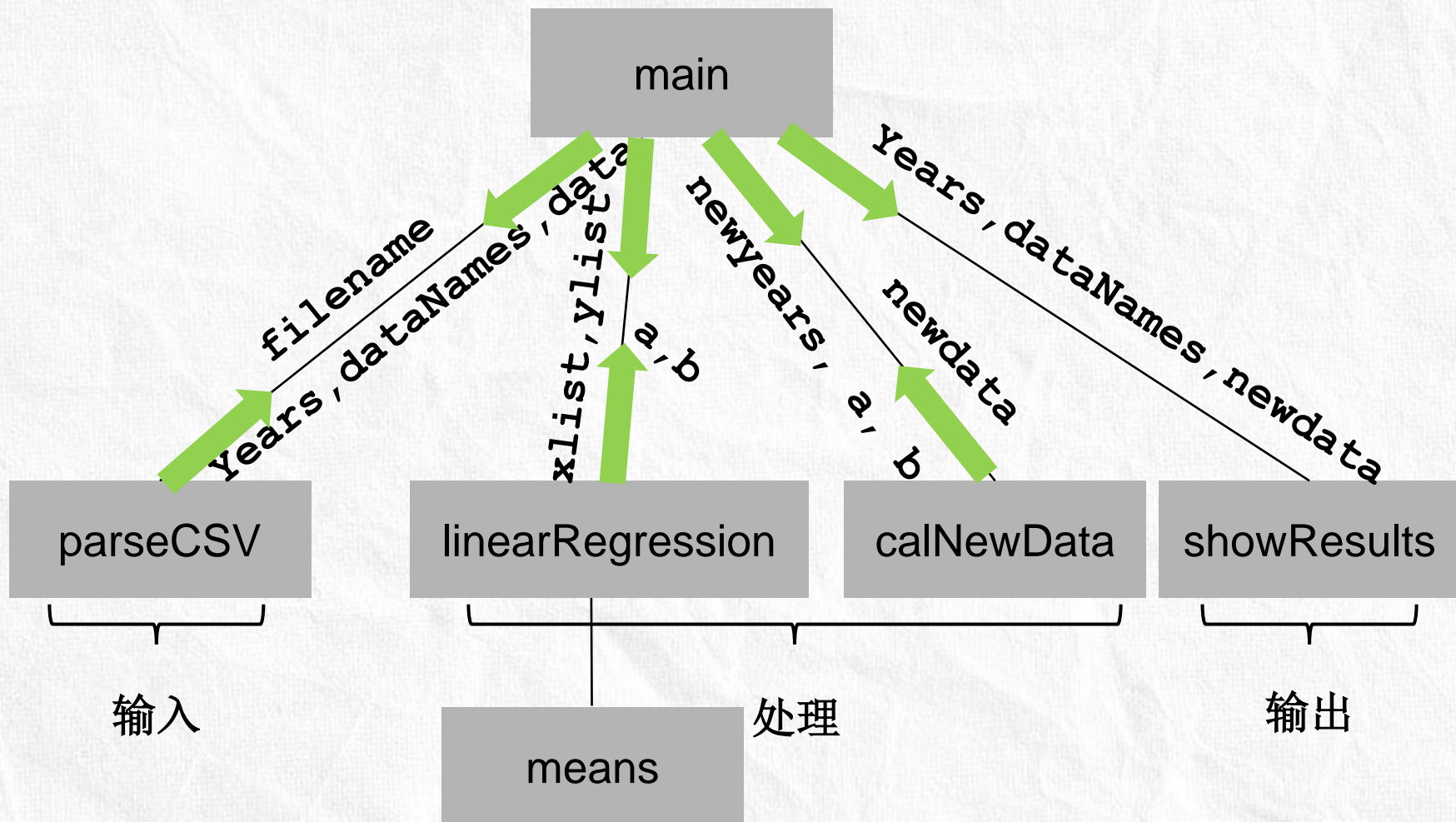
计算出

a, b

推测出

x

y



国家财政数据趋势演算

```
1  #现有数据列表
2  ls=[
3      ["指标","2000年","2015年","2016年"],
4      ["全部收入","13395.2","152269.2","159605"],
5      ["中央收入","6989.2","69267.2","72365.6"],
6      ["地方收入","6406.1","83002","87239.4"],
7      ["全部支出","15886.5","175877.8","187755.2"],
8      ["中央支出","5519.9","25542.2","27403.9"],
9      ["地方支出","10366.7","150335.6","160351.4"]
10 ]
```



```
11  #保存现有数据
12  f=open("C:\\Users\\li\\python\\finance.csv","w")
13  for row in ls:
14      f.write(",".join(row)+ "\n")
15  f.close()
```

```
16 # 提取原表中的年份、数据、数据名称分别放在三个数组里
17 def parseCSV(filename):
18     dataNames, data = [], []
19     f = open(filename, 'r', encoding='utf-8')
20     for line in f:
21         splitedLine = line.strip().split(',')
22         if '指标' in splitedLine[0]:
23             years=[int(x[:-1]) for x in splitedLine[1:]]
24         else:
25             dataNames.append('{:10}'.format(splitedLine[0]))
26             data.append([float(x) for x in splitedLine[1:]])
27     f.close()
28     return years, dataNames, data
29 def means(data): #求数据的均值
30     return sum(data) / len(data)
```

```
31  #线性回归, 求a, b
32  def linearRegression(xlist, ylist):
33      xmeans, ymeans = means(xlist), means(ylist)
34      bNumerator = - len(xlist) * xmeans * ymeans
35      bDenominator = - len(xlist) * xmeans ** 2
36      for x, y in zip(xlist, ylist):
37          bNumerator += x * y
38          bDenominator += x ** 2
39      b = bNumerator / bDenominator
40      a = ymeans - b * xmeans
41      return a, b
```



```
42 #求其他年份的数据
43 def calNewData(newyears, a, b):
44     return [(a + b * x) for x in newyears]
45 #显示结果，主要是设置外观
46 def showResults(years, dataNames, newDatas):
47     print('{:^60}'.format('国家财政收支线性估计'))
48     header = '指标'
49     for year in years:
50         header += '{:10}'.format(year)
51     print(header)
52     for name, lineData in zip(dataNames, newDatas):
53         line = name
54         for data in lineData:
55             line += '{:>10.1f}'.format(data)
56         print(line)
```

```
57 #主函数调用
58 def main():
59     newyears = [x+2010 for x in range(7)]
60     newDats = []
61     years, dataNames, datas = parseCSV('finance.csv')
62     for data in datas:
63         a, b = linearRegression(years, data)
64         newDats.append(calNewData(newyears, a, b))
65     showResults(newyears, dataNames, newDats)
66
67 main()
```

>>>

国家财政收支线性估计

指标	2010	2011	2012	2013	2014	2015	2016
全部收入	105359.6	114550.1	123740.6	132931.0	142121.5	151312.0	160502.4
中央收入	48169.1	52283.8	56398.5	60513.2	64627.9	68742.7	72857.4
地方收入	57190.6	62266.3	67342.1	72417.8	77493.6	82569.3	87645.1
全部支出	122936.9	133645.7	144354.5	155063.3	165772.1	176480.9	187189.8
中央支出	19037.5	20390.9	21744.3	23097.7	24451.1	25804.5	27157.9
地方支出	103899.4	113254.8	122610.2	131965.6	141321.0	150676.4	160031.9



THANKS