CS 3430: S19: SciComp with Py
Lecture 17

The Method of Least Squares

Vladimir Kulyukin
Department of Computer Science
Utah State University

Review

Variance and Mean

Let X be a random variable. Let x_1, x_2, ... x_n be its possible values.

The variance of a set of n equally likely values is defined as

$$Var(X) = \frac{1}{n}\sum_1^n (x_i - \mu)^2,$$

where

$$\mu = \frac{1}{n}\sum_{i=1}^n x_i.$$

The standard deviation σ is $\sqrt{Var(X)}$.

Gaussian Functions

A Gaussian function (named after Carl Friedrich Gauss) is any function of the form

$$f(x) = a \cdot e^{-\frac{(x-b)^2}{2c^2}},$$

where a, b, and c are arbitrary real constants.

Gaussian Probability Density Functions

Gaussian functions are used to represent the probability density function (PDF) of a normally distributed variable with the expected value μ and the standard deviation σ

$$g(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}.$$

The Bell Curve IQ Model

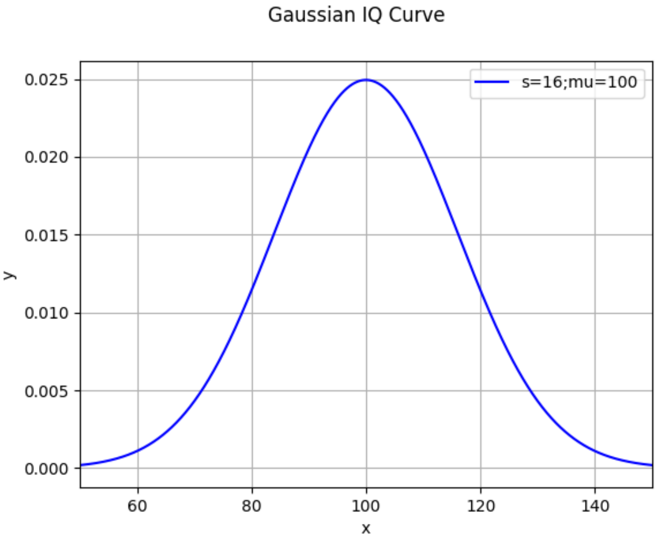
In the Bell Curve IQ model, the median IQ is arbitrarily set at 100, so half the population has an IQ less than 100 and half greater than 100.

IQs are assumed to distribute according to a bell-shaped curve called a **normal curve**.

The proportion of all people having IQs between A and B is given by

$$\int_A^B \frac{1}{16\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-100}{16}\right)^2}.$$

Gaussian IQ Curve (aka Bell Curve)



Least Squares

Motivation

Data scientists compile graphs of thousands/millions of different quantities.

Examples: the purchasing value of a currency unit as a function of time; the pressure of fixed volume of air as a function of temperature; the average income of people as a function of their years of formal education; the incidence of strokes as a function of blood pressure, etc.

Real data points tend to be irregularly distributed due to the complicated nature of the underlying phenomena, sensor errors, observation errors, data curation errors, etc.

Motivation

In spite of the imperfect nature of the data, we are still faced with the problem of making assessments and predictions based on them; the problem amounts to filtering the sources of errors in the data and isolating the **basic underlying trend.**

Frequently, on the basis of some suspicion or a working hypothesis, we may hypothesize that the underlying trend is linear, i.e., the main trend in the data is a straight line.

The next question is, which straight line? There are, in theory, infinitely many lines that can model the underlying data.

The method of least squares is a computational procedure that allows us to find the straight line that best fits the underlying data.

Straight Line Data Fitting Problem

Given observed data points $(x_1, y_1), (x_2, y_2), ..., (x_N, y_N)$, find the straight line that best fits these points.

What is the Best Fit?

If (x_i, y_i) is one of the observed points, we can measure how far it is from a given line $y = Ax + B$ by the vertical distance from the point to the line.

Since the point on the line with x-coordinate x_i is $Ax_i + B$, this vertical distance is the distance between the y-coordinates $Ax_i + B$ and y_i.

Let the error E_i be $E_i = (Ax_i + B) - y_i$, then $E_i = 0$, $E_i < 0$, $E_i > 0$. To avoid ambiguity, it is more convenient to square the error, i.e.

$$E_i^2 = (Ax_i + B - y_i)^2$$

Total Error

The total error in approximating the data points $(x_1, y_1), (x_2, y_2), ..., (x_N, y_N)$ by the line $y = Ax + B$ is

$$E = E_1^2 + E_2^2 + ... + E_N^2 = \sum_{i=1}^{N} E_i^2.$$

E is called the **least-squares error** of the observed points with respect to the line.

If all the observed points line on the line $y = Ax + B$, $E = 0$. In general, we cannot expect to find a line $y = Ax + B$ that fits the observed data perfectly, i.e., $E = 0$.

Fitting Regression Line to Data

Given the observed data points $(x_1, y_1), (x_2, y_2), ..., (x_N, y_N)$, find a straight line for which the error E is as small as possible. This line is called the **least-squares line** or **regression line**.

Fitting Regression Line to Data

For a general set of observed data points, the minimization process can be generalized through the following algebraic formulas for A and B:

$$A = \frac{N \cdot \sum xy - \sum x \cdot \sum y}{N \cdot \sum x^2 - (\sum x)^2},$$

$$B = \frac{\sum y - A \cdot \sum x}{N},$$

where

1. $\sum x =$ sum of the x-coordinates of the data points;
2. $\sum y =$ sum of the y-coordinates of the data points;
3. $\sum xy =$ sum of the products of the coordinates of the data points;
4. $\sum x^2 =$ sum of the squares of the x-coordinates of the data points;
5. $N =$ number of data points.

Problem: Car-Accident-Related Deaths in the U.S.

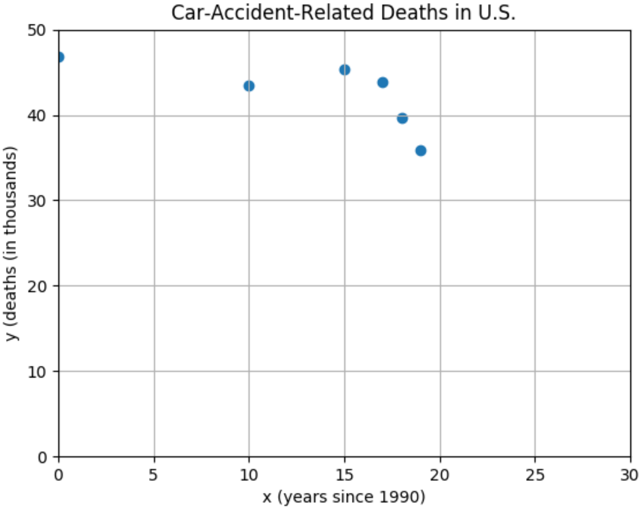
The following table gives the number (in thousands) of car-accident-related deaths in the U.S. for certain years.

Year	Number (in thousands)
1990	46.8
2000	43.4
2005	45.3
2007	43.9
2008	39.7
2009	35.9

- Use the preceding formulas to obtain the straight line that best fits these data;
- Use the straight line to estimate the number of car-accident-related deaths in 2012.

Solution

Here is plot of the observed data points.



Solution

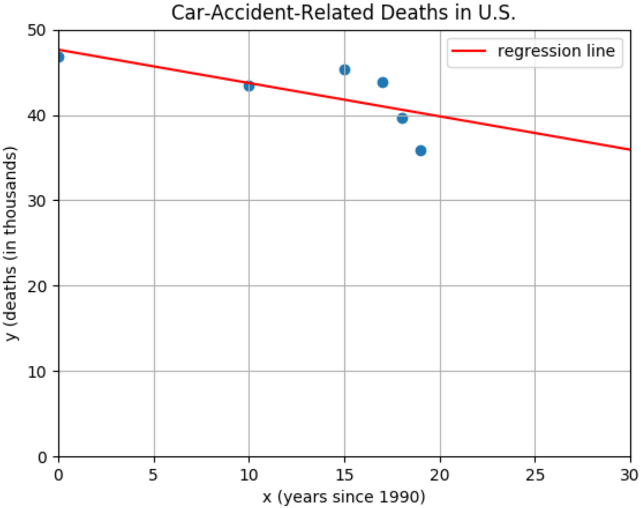
```
x = np.array([0, 10, 15, 17, 18, 19])
y = np.array([46.8, 43.4, 45.3, 43.9, 39.7, 35.9])

def fit_regression_line(x, y):
  N = len(x)
  assert len(y) == N
  sum_xy = sum(xy[0] * xy[1] for xy in zip(x, y))
  sum_x = sum(xi for xi in x)
  sum_y = sum(yi for yi in y)
  sum_x_sqr = sum(xi**2 for xi in x)
  A = (1.0*(N*sum_xy - sum_x*sum_y))/(N*sum_x_sqr - sum_x**2)
  B = (sum_y - A*sum_x)/(1.0*N)
  rlf = lambda x: A*x + B
  return A, B, rlf
```

Solution

```
def plot_regression_line(x, y):
  plt.title(’Car-Accident-Related Deaths in U.S.’)
  plt.xlabel(’x (years since 1990)’)
  plt.ylabel(’y (deaths (in thousands)’)
  plt.autoscale(tight=True)
  plt.xlim([0, 30])
  plt.ylim([0, 50])
  A, B, rlf = fit_regression_line(x, y)
  xvals = np.linspace(x[0], 30)
  yvals = np.array([rlf(xv) for xv in xvals])
  plt.plot(xvals, yvals, label=’regression line’, c=’r’)
  assert len(x) == len(y)
  plt.scatter(x, y)
  plt.legend(loc=’best’)
  plt.grid()
  plt.show()
```

Solution

Here is plot of the observed data points and the regression line.



References

1. L. Goldstein, D. Lay, D. Schneider, N. Asmar. *Calculus and its Applications*, Chapters 7. Pearson.
2. www.python.org.