AN INFORMAL APPROACH TO LINEAR LEAST SQUARES

James Quinlan

University of New England

Abstract: Modeling data using the least squares method is used extensively in practice, and is therefore an essential topic for students of data science. This paper describes a GeoGebra applet used to facilitate understanding of the objective and the underlying mathematics of the least squares method. Additionally, lists, one of the most robust and valuable GeoGebra features is highlighted through the use of the `Sequence` command. The discussion includes ideas for enhancing the applet.

Keywords: least squares, approximation, GeoGebra: lists, `Sequence`

Introduction

Students often struggle, when using software, to understand concepts that underlie mathematics because the intermediate steps are hidden from them in a decidedly “black-box" fashion. Buchberger [1] recommended that the black-box software be replaced with a “glass-box" in which steps that promote student understanding are included. The current topic is a prime example. While students have probably encountered the Least Squares Method (LSM) on several occasions, possibly even on a graphing calculator as far back as 7th grade, the underlying mathematics has been hidden from them.

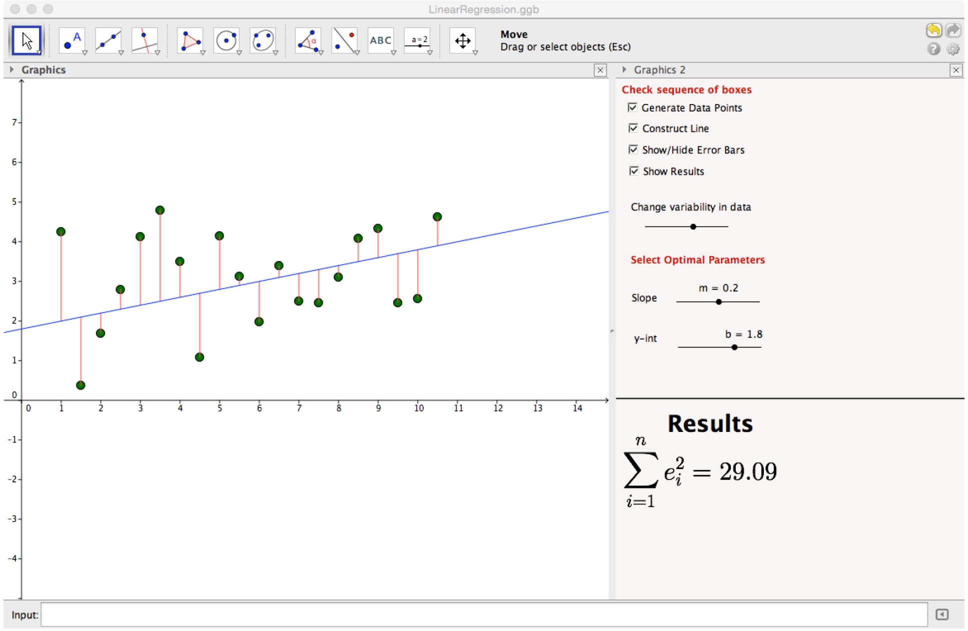
Case in point, a project assigned in an undergraduate computer programming course asked approximately 15 to 20 students to code the LSM. The project requirements included reading the data set and solving the normal equations, which were to be derived offline using calculus, a prerequisite of the course, to produce the best linear model for the data. In the first semester the project was assigned, the concepts that underlie the method were described in a handout as well as covered and discussed during class. Additionally, students were given several references (for example, [6]) in the handout that provided in-depth coverage of the technique. However, less than one quarter of the students were able to successfully complete the assignment without direct assistance from the instructor or other students.

In response to these difficulties, a GeoGebra [3] applet was created for subsequent semesters affording students the opportunity to interact with most of the underlying concepts and experience the method dynamically. After using the applet in conjunction with the other resources, all but a few students were able to successfully code the project without much instructor assistance.

This article demonstrates how GeoGebra can be used to investigate and understand the method of least squares to fitting data. First, we describe the applet including the interface objects and commands needed. Additionally we describe its use in understanding the LSM. Next, we discuss ways to improve the applet, provide some in-depth ideas for students to explore, and highlight calls by prominent organizations to include its study in school mathematics. We point out the incredible opportunity this topic affords teachers to improve their technological skills, sharpen their mathematics, and prove their pedagogical knowledge.

Least Squares Method and GeoGebra Applet

The GeoGebra applet used to demonstrate the underlying processes of the LSM is composed of two windows, *Graphics* and *Graphics 2*. See Figure 4.1. *Graphics* contains the scatterplot of the data (X_i, Y_i) in green, the linear predictive model ($\hat{Y}_i = aX_i + b$) in blue, and the residuals in red. *Graphics 2* contains standard user interface controls like checkboxes and sliders as well as a text object showing the computational results of the sum of the squared errors (SSE).



Check sequence of boxes

☑ Generate Data Points

☑ Construct Line

☑ Show/Hide Error Bars

☑ Show Results

Change variability in data

Select Optimal Parameters

Slope m = 0.2

y-int b = 1.8

Results

$$\sum_{i=1}^{n} e_i^2 = 29.09$$

Figure 4.1 GeoGebra applet for understanding the Least Squares Method

Checkboxes are used to show/hide the model, the residuals, and the SSE. A slider is used to adjust the variability in the data set. The order in which the checkbox controls are placed is important to the learning process. That is, first we have to generate a data set, then construct the linear model, then reveal the error in the model.

With or without the SSE shown, students can dynamically explore different models by moving the sliders associated with varying the two parameters, slope and y-intercept. If the SSE is shown, students observe that better fitting lines are associated with smaller numbers.

The crux of the matter in terms of developing the applet is knowledge of lists in GeoGebra. First, two lists are generated using the `Sequence` command, one for the X's and one for the Y's. The X values are selected at equally spaced points while the Y values incorporate normally distributed random errors. These values form a list of ordered pairs called, in GeoGebra, `data`. The residuals are a list of segments created with the `Sequence` command. See Appendix 1 for complete construction protocol.

There are several potential improvements to the GeoGebra applet that can further develop advanced students' conceptual understanding of LSM. The applet could have options to append an outlier to the data set as the method is sensitive to outliers, to include heteroskedasic data, allow for non-normality in residuals, and have options for alternative data models such as quadratic, logarithmic, and exponential.

Discussion

Because the objective of LSM is to minimize total error, while experimenting, it is natural for students to question why compute the SSE instead of just the sum of the residuals. Upon further examination and continued discussion among their peers, the applet allows them to observe that the model overestimates some values while underestimating others (in other words, some of the residuals are positive and some negative), thus realizing the potential for catastrophic cancellation. Moreover, the use of GeoGebra in this project can aid students in a number of other ways.

First, it may help reinforce previous material. For example, the use of a dynamic sliders controlling the model parameters serve as a reminder of the effects slope and intercept have on a line. Second, it may be a platform to introduce new topics such as summation notation. Third, it provides an opportunity to quickly and efficiently view some concepts such as variability in data with a dynamic graphical representation which would be difficult if not impossible in a static environment. There are many other notions to explore and discover with the help of GeoGebra, too. In particular,

1. the sum of the residuals is zero (this can be easily proven for students that derive the normal equations),
2. the sum of the observed values Y_i equals the sum of the fitted values $\hat{Y}_i = mX_i + b$,
3. the weighted sums, $\sum_{i=1}^{n} X_i e_i = 0$, and $\sum_{i=1}^{n} \hat{Y}_i e_i = 0$ hold, and
4. the regression line always goes through the point $(\bar{X}, \bar{Y})$.

Although originally developed to demonstrate the LSM process to mathematics students in an undergraduate programing course, the method and applet are appropriate for students in middle and high school. Both the National Governors Association and the National Council of Teachers of Mathematics call for students to have experience fitting data and deciding among many possible linear models. In particular, the *Common Core State Standards* recommend fitting data to a linear model and assessing its fit informally even at the middle school level [2]. The National Council of Teachers of Mathematics calls for students to understand the least squares regression line with a visual model and measure error in this model as well as determine the “goodness of fit" [5].

Lastly, developing and implementing technology into the curriculum can be a powerful means of teaching but requires significant combination of mathematical, technological, and pedagogical knowledge. In particular, teachers must have a deep understanding of the underlying mathematics, know multiple ways to represent mathematical objects, be aware of potential student difficulties, and be able to create helpful illustrations [4].

Conclusion

This interactive GeoGebra applet provides students at multiple levels with an opportunity to understand, at least informally, the LSM using a dynamic visual model. It quickly and effectively conveys the overall objective of the method of finding parameters (slope and y-intercept) that minimize the sum of the squared errors. Additionally, LSM provides a context for improving Technological, Pedagogical, and Content Knowledge (TPACK). The most important addition to any future versions of the applet is the idea of “goodness of fit".

REFERENCES

1. Buchberger, B. (1990). Should students learn integration rules? *ACM SIGSAM Bulletin, 24*(1), 10-17.
2. Common Core State Standards Initiative. (2010). Common Core State Standards for Mathematics. Washington, DC: National Governors Association Center for Best Practices and the Council of Chief State School Officers.
3. Hohenwarter, M. (2002). *GeoGebra*. Available online at http://www.geogebra.org/cms/en/

4. Milner-Bolotin, M., Fisher, H. & MacDonald A. (2013) Modeling Active Engagement Pedagogy through Classroom Response Systems in a Physics Teacher Education Course. *LUMAT, 1*(5), 523-542.
5. National Council of Teachers of Mathematics. (2015). Principles and standards for school mathematics. Retrieved from http://www.nctm.org/Standards-and-Positions/Principles-and-Standards/Data-Analysis-and-Probability
6. Nachtsheim, C. J., Neter, J., Kutner, M. H., & Wasserman, W. (2004). *Applied linear regression models*. McGraw-Hill Irwin.

Appendix 1: Construction protocol

1. Define a number (the standard deviation of the data points), `std = 1.2`
2. Define a number (the slope), `m = 0.2`
3. Define a number (the y-intercept), `b = 0.2`
4. Define a linear function, `f(x) = m*x+b`
5. Define a number (of data points), `n = 20`
6. Define a list of x's, `xs = Sequence[1 + 0.5k, k, 0, n]`
7. Define a list of y's, `ys = Sequence[k / k RandomNormal[3, std], k, 1, n]`
8. Define a list of data points,
   ```
   data = Sequence[(Element[xs, i], Element[ys, i]), i, 1, n]
   ```
9. Define sequence of error bar segments,
   ```
   errors = Sequence[Segment[Element[data, i],
   (x(Element[data, i]), f(x(Element[data, i])))], i, 1, n]
   ```
10. Define a list of squared errors, `sqerror = errors^2`
11. Define a number, the sum of square errors (SSE), `SSE = Sum[sqerror]`
12. Define text to display results,
    ```
    Results = FormulaText["$ \sum_i=1^n e_i^2 = $"] + FormulaText[SSE]
    ```