This is an accepted manuscript version of the paper: Erickson, T., Finzer, B., Reichsman, F., & Wilkerson, M. (2018). Data moves: One key to data science at the school level. To appear in *Proceedings of the International Conference on Teaching Statistics (ICOTS-10)*. Kyoto, Japan.

DATA MOVES: ONE KEY TO DATA SCIENCE AT THE SCHOOL LEVEL

Tim Erickson[1], Bill Finzer[2], Frieda Reichsman[3], and Michelle Wilkerson[4]
[1]Epistemological Engineering, Oakland, California. USA, eepsmedia@gmail.com;
[2]Concord Consortium, Emeryville, California, USA;
[3]Concord Consortium, Concord, Massachusetts, USA;
[4]University of California at Berkeley, California, USA

Can secondary-school students and their teachers, with limited computational experience, actually do something that seems like data science? In this paper, we propose a set of "data moves" that are characteristic of data science, but are seldom taught or identified explicitly in statistics curricula. These moves are accessible to teachers and students before they deal with the full scope of "professional" data science tasks or the full array of powerful—and sometimes opaque—tools. We suggest that by being aware of these moves, we can create curriculum materials that better prepare students, both for a future in data science and a future as informed citizens.

INTRODUCTION

Data Science. You've heard of it. You know that it's the sexiest new profession. Your online inbox has probably filled with opportunities for continuing education and job openings. For us as statistics educators, obvious questions arise: how will people learn about data science? And what will they learn?

Around the world, some universities and companies—especially their online services—are beginning to offer degrees, certificates, and continuing education in data science. Many of the university programs are collaborations between statistics and computer science departments. These typically require familiar math and stats courses such as regression or linear algebra, but they also tend to require mastery of computational systems such as Python and R, with corresponding scaffolded wrappers such as Jupyter Notebooks and RStudio. As students get beyond the basics, course offerings—depending on the student's concentration—are dominated by courses such as machine learning, business analytics, or bioinformatics. That is, a data science major is computationally heavy.

One wonders how many students are truly prepared for that work. Of course there will be students who are computationally more experienced, and who have spent many hours, often outside of formal classes, learning to code. For them, applying these skills and habits of mind to large data sets will be an interesting challenge, and they will emerge with a Masters in Data Science ready to enjoy a promising career.

But what about the others? Consider students who have a bit less experience—who might be overwhelmed by the combination of math, coding, and understanding data—and need a little help in order to succeed. And consider students who may not seek a data science degree but who, as citizens, will benefit from knowing of what working with data in the data-science age is like. What can we offer them, perhaps at the school level, to help them develop an essential foundation?

One approach would be to see computational skills as the bottleneck, and teach more coding, to make sure, for example, that students know a suitable high-level language earlier in their education.

But we suspect that that there are other things that might be just as valuable for preparing to learn data science. To find them, let's look at data science itself. This is not the place to define something so elusive (Finzer 2013), but it is fair to say that data science—when compared with introductory statistics—typically involves larger, richer data sets, that is, there is more data, in two senses: more cases and more variables. As a consequence, data scientists take specific actions to help them cope with this increased size and complexity. Some of these are also used in intro stats, but they take on new power and urgency when applied in data science contexts. We call these actions data moves.

ABOUT DATA MOVES

We have claimed that a data move alters a dataset's membership, structure, or values (Erickson et al., 2018). This is a dry and tentative definition; it needs elaboration—which

follows—and is open to amendment. But we think that data moves, as we describe them here, constitute an important and usefully-limited subset of what we do when we do data science.

We have found a number of these data moves. We know that the set is not comprehensive—your favorite may not be listed—and we know they overlap. But for now, we are not looking for some set of basis vectors, or a group of distilled postulates; this is not a taxonomy of species. Instead, we are reporting on some animals we see a lot, and how they behave. We've identified six of these as "core" data moves because they appear to be particularly common and powerful:

- Filtering. That is, looking at a subset of the data. Also called scoping or slicing.
- Grouping. Dividing the data into subsets. This is related to filtering, but we think of it and use it differently.
- Summarizing. Computing an aggregate measure for a set of data, often used in conjunction with grouping in order to compare groups.
- Calculating. Making new attributes out of existing data, for example, new computed columns in a data table. Closely related to summarizing.
- Merging. Also joining. Connecting two data sets through concatenation or relation.
- Reorganizing a data set, for example, by making it hierarchical. This is related to joining relationally, but also provides a view of data that connects grouping, summarizing, and calculating.

We will now explore a large multivariate data set in order to illustrate the first four of these core data moves in detail, making commentary about other issues as we go. This dataset contains records of travel on the Bay Area Rapid Transit (BART) system—the metro rail in the San Francisco, California area—for the calendar year 2015. Each case represents one hour of operation between two stations. The variables include the number of `riders` that exited the station called `endAt` during that hour, having entered the BART system at the station called `startAt`. We also include several time variables including a date-time object called `when` and the `day` of the week. There are over 40 stations in the BART system, so over 1600 combinations. With data every hour for a year, there are over 10 million records.

The figures you will see are from CODAP (Finzer 2014), the Common Online Data Analysis Platform, which was designed to help novices and other learners do data analysis without coding. But users of professional data analysis tools will see these data moves as features they are familiar with. For example, inhabitants of the "tidyverse" (Grolemund and Wickham, 2017) will recognize that most of what we call data moves correspond to verbs in the `dplyr` package (Wickham et al., 2017).

For this paper, let's focus on a practical question: How many people took BART to the San Francisco Giants' first game against their arch-rival Los Angeles Dodgers on Tuesday, April 21?

Figure 1: Ridership to Embarcadero station on April 21. Left: Each hour has over 40 points, one for each station in the system. Right: Data from only the Dublin/Pleasanton station to Embarcadero.

You will need some local knowledge: First, the Giants and Dodgers are baseball teams. The Giants' home field, AT&T Park, is in downtown San Francisco, and the two stations you might take to the game are called Embarcadero and Montgomery Street. Most games start at about 7PM.

Let us begin by collecting some relevant data from the dataset—the data for passengers who left the BART system on April 21 at Embarcadero.

Just by choosing this initial set, we are doing the filtering move: we have chosen not to look at all 10 million records, but just these 885 data points, representing 49,347 riders. On the left side of Figure 1, we see data from all the stations at once, which is kind of messy, but you definitely see two peaks. Some additional local knowledge and thinking suggest that the first peak is people coming to downtown San Francisco that morning, to work. The second peak might contain people coming to the game.

To simplify this so we can think about it more clearly, we filter again, and (temporarily) use only the data from a single station, Dublin/Pleasanton. This graph (the right side of Figure 1) also shows two peaks, but notice how much cleaner it is.

We have a fundamental problem we need to address: we don't know how much of that peak is the ball game and how much is something else. That is, we don't know the base level of ridership during that evening period. So we decide to collect the same data for the previous day—when there was not a game—in order to compare game-day data with normal-day data.

This idea, to get a comparison day, is an example of the kind of thinking we want students to do. It's an important data analysis step, but—by our definition—it is not a data move in itself. That good idea, however, prompts a data move, to get data for the other day. (We might categorize this as filtering, in this case filtering in more data, expanding the scope of the investigation.)

In either graph of Figure 2, you can easily see that the second bump appears only on Tuesday. But the right-hand graph is special: we have grouped the data (our second core data move) in order to compare Monday with Tuesday directly. It happens that, in CODAP, we do this in the graph by dragging `day` to the middle of the graph to make a legend; in another system you might actually code it. In the tidyverse, using R, you might need a command like `group_by(day)` in your pipe.

Figure 2: Ridership from Dublin to Embarcadero for April 20 and 21 2015. Left: Both days as a single time series. Right: The two days overlaid. Tuesday, the game day, is dark green.

This "overlay" graph (Figure 2, right) is ingenious. We calculate a new variable (data move number 4) called `hour`, which is based on the date-time `when`, but contains just the time. With `hour` on the axis, both days of data appear superimposed. It also suggests a strategy to find how many people were coming to the game: subtract the two numbers hour by hour, Tuesday minus Monday, so that a positive number indicates more people traveling on game day.

In our view, grouping is a data move but making the graph is not (Erickson et al., 2018). This may seem troubling and pedantic, but the distinction separates data manipulation from

visualization. They are tightly linked, of course; in CODAP the single gesture of dropping in `day` does both. But it's really a sequence: first the drop tells CODAP to organize the data into two groups according to the day; then CODAP colors the points in the graph to match the data. That is, the graph reflects the data. Therefore, to get the graph you want, you have to prepare the data properly. CODAP shortcuts this process to give you what it thinks you want. The tidyverse, in contrast, explicitly makes the distinction by putting data commands in the `dplyr` package and graphics commands in `ggplot2` (Grolemund and Wickham 2017). We could make a similar distinction about modeling—the desire to use a particular model motivates making certain data moves—but let's maintain our focus on data moves.

What about the decision to make an overlay by putting `hour` on the axis? Computing `hour` is a data move—we altered the data set by making a new column—but what about making that graph? There is no question that it's a smart move. It illustrates understanding we want students to have. But making the overlay does not alter membership, structure, or values; it only changes the display. The key takeaway we should stress—whether or not you agree that this is not a data move—is that there is more to good data analysis than data moves.

Now we have to add everything up. So far, we've been looking at only at passengers from Dublin/Pleasanton. We should include riders from other stations. To do that, we should sum all the values at each hour to get the total number of people exiting the Embarcadero station.

For that we need another grouping move—grouping the data by `hour`—and then we need a new data move, summarizing. That produces a new value, the `total`, defined as the sum of the `riders` for each hour. This is a common pattern: a grouping move followed by summarizing, calculating an aggregate measure for each group.

Notice that our first use of grouping—to separate and distinguish data from the two days—is very much what an intro stats student would have to do. But this second use of grouping—grouping by hour in order to separately sum the `riders` for each time, for each day—is not.

Figure 3: Left: Total number of `riders` to Embarcadero on Monday and Tuesday, by hour. Right: The difference between Tuesday and Monday.

One possible next step appears on the right-hand side of Figure 3: for each hour, we subtract the `totals` to get `diff`—the "excess" on Tuesday—which might be attributed to riders going to the game.

But what hours should we add up? Experience and actual statistics knowledge help here: Are the large, 500-person excesses for Tuesday morning just noise because the total ridership is so high? Probably not. The differences are much bigger than root N. Something else is going on. Those big `diffs` suggest that we ought to look at at least one more "control" day in case there was something special about Monday.

Let us decide, though, that it's fair to add up the "excess" between 4 and 8 PM, since that encompasses the peak—and because the game starts at 7. Then we do another filtering move, restricting the cases to those hours, and summarize again to add up the `diffs`. The result: 2710

people, or about 6.5% of the crowd. This still does not include Montgomery station. For simplicity, we stop here, but you can imagine the kinds of additional data moves we would have to make.

DISCUSSION

We have mapped out one way to use data moves to help arrive at an answer to our presenting question. We have left a lot out, for example, what about the other core data moves? What other data moves are there beyond the core? Erickson et al. (2018) address these questions, so we reluctantly abandon them here, except to say that thinking about the data hierarchically (reorganizing) connects the moves we have listed here to calculating (making calculated variables), and provides a new and elegant interpretation of grouping and summarizing. In fact, in CODAP, in order to sum the ridership by hour, it was essential that we reorganize our data table hierarchically.

The key issue, though, is whether data moves, as an idea for education, are useful.

We claim that the intro stats course actually uses data moves seldom, and at a relatively unsophisticated level. This is because most of the data we give our students is pre-digested and sanitized: we give them the data they need and no more. They have no need to alter a dataset's membership, structure, or values, so they have no need to filter, and little need to group or summarize or calculate new table columns.

In fact, students in intro stats compute summaries generally only as the input to some procedure: they compute means and standard deviations in order to perform tests or construct estimates. Here, when we grouped by `hour` and computed sums, we were still exploring. We created so much summary data that it was worth making graphs; the summaries became "case" data in their own right—which we later had to summarize, using additional data moves.

This continued use of data moves is characteristic of data science, essential when we look at larger, more complex data sets. Why is that? Look back at Figure 1 and consider: the second graph is simpler and easier to parse visually because it has fewer points. We got fewer points by filtering, by taking a slice of the data set defined by being from Dublin/Pleasanton. Looking at Figure 3—the result of grouping and summarizing—again we see fewer points, accomplished this time by collapsing the dataset along the "starting stations" dimension, through summation. With fewer points, we were better able to see patterns, focus our attention on the relevant data, and plan our approach.

When we began, we did not know what data moves to make or in what sequence. So we engaged in an iterative and exploratory process, looking at patterns in the data, seeking a "way in" to answer our initial question. The steps in that process alternated between making data moves and creating visualizations, the resulting graphics inspiring the next moves. Perhaps a farsighted student would immediately recognize the need to sum the ridership by hour, over all the source stations, and shortcut this process. But we suspect that is rare indeed, and that it is useful to grapple with the underlying principles of these data moves, and learn their power.

Put another way, data moves are basic tools in a data-science world with larger data sets. In our example, they helped us do "exploratory data science analysis," the opening part of what might be a sophisticated investigation with more advanced data science tools. In that way, they are foundational for real data scientists, and also may be enough for those seeking an introduction.

But do they belong in statistics? To go beyond the limited use of data moves in intro stats, there are (at least) two problems: First, from the point of view of orthodox frequentist inferential statistics instruction, data moves may seem more mechanical than conceptually important. And second, actually doing these moves requires overhead: whether you are using CODAP or R or Python, filtering, grouping, summarizing, and calculating with a large data set means making computation and syntax part of the course.

We suggest, in response, that the ubiquity of powerful computing and large data sets demand that an introductory data course address more data-science-like tasks with richer, messier, and more unruly data. If some traditional stats topics move to a later, more specialized course, so be it. Kaplan (2017), for example, has outlined an introductory course with data science closer to the core.

What about data education in secondary schools? This may be the perfect place to challenge students to cope with rich data sets using exploratory techniques. Before we expose students to formal inferential statistics or to sophisticated, computer-intensive data science methods, they can learn, in tasks like the one outlined above, how to configure interesting data

using these data moves, and through that data, to think well about how to make credible claims and answer interesting questions.

If we succeed in this, students who move on will be better prepared for "real" stats and data science. Perhaps more importantly, with some of the basics of data-with-computation at their fingertips, a more diverse group of students will be able to succeed.

This idea is at the core of why we should care about data moves, so let's dig a little deeper. Suppose a student faces a task like our question of how many people are taking the train to the baseball game. What can go wrong? They might not really understand the structure of the data—for example, they might count people both arriving at and leaving from the station, or not include all possible "source" stations, or not quite understand that one case represents one hour between two stations. They might see that they want to add stacks of points (many stations at the same time), but not really understand that they have to group the points by hour—but keep them separated by day. They might not think about subtracting the data from another day, or if they do, they might not understand how to perform that subtraction. They might not see that, having added all the stations for each hour, they now have yet another time series that's actually useful.

If the student has computational experience—they can code in Python, say—some problems fall away and they can struggle, productively, with the remaining issues, which are largely about the data. But if the student is not already a coder, they have to cope with both data and computation simultaneously. Thinking about data moves is perfect for this problem: as concepts, data moves are not about the computer, but rather the structure and membership of your dataset. But in practice, data moves demand that you know the syntax.

What does it mean to "include" data moves in a curriculum? Surely not a chapter on filters and one on grouping and summarizing. Perhaps it's as simple as pointing moves out to students when we use them, and bearing them in mind as we plan.

CONCLUSION

Understanding data moves is vital when we analyze rich, unruly, data-science-worthy datasets. This was not the case for the smaller, more sanitized datasets we have traditionally used in introductory statistics. But with more data and more computing power, things have to change, and that means more attention to computational skills and computational thinking in data education.

But this idea is young and ill-formed. We need research and additional thinking. What does it look like when students use data moves (see Wilkerson et al., 2018 for a beginning)? Is there a progression? What are the relevant visualization and modeling skills that complement data moves? We look forward to seeing how this discussion develops.

REFERENCES

Erickson, T., Wilkerson, M., Finzer, W. and Reichsman, F. (2018). Data Moves. Submitted to Technology Innovations in Statistics Education.

Finzer, W. (2013). The Data Science Education Dilemma, Technology Innovations in Statistics Education. 7(2). https://escholarship.org/uc/item/7gv0q9dc

Finzer, W. (2014). Common Online Data Analysis Platform [Computer Software] (CODAP). Concord, MA: The Concord Consortium. https://codap.concord.org/releases/latest/static/dg/en/cert/index.html

Grolemund, G. and Wickham, H. (2017). R for Data Science. O'Reilly. http://r4ds.had.co.nz/

Kaplan, D. (2017). "Teaching Stats for Data Science," American Statistician. In press. doi: 10.1080/00031305.2017.1398107

Wickham, H, Francois, R, Henry, L, and Müller, K. (2017). Package dplyr: a Grammar of Data Manipulation. https://cran.r-project.org/web/packages/dplyr/index.html

Wilkerson, M. H., Lanouette, K., Shareff, R. L., Erickson, T., Bulalacao, N., Heller, J., St. Clair, N., Finzer, W., & Reichsman, F. (2018, Under Review). Data moves: Restructuring data for inquiry in a simulation and data analysis environment, submitted to the International Conference for the Learning Sciences: ICLS. London, England: ISLS.

This material is based upon work supported by the National Science Foundation under Grant No. IIS-1530578. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.