A Primer on Student Growth Percentiles

Damian W. Betebenner
The Center for Assessment
DBetebenner@nciea.org

April 6, 2011

For Çiğdem

Contents

Introduction

Why Student Growth?

Accountability systems constructed according to federal adequate yearly progress (AYP) requirements currently rely upon annual "snap-shots" of student achievement to make judgments about school quality. Since their adoption, such *status measures* have been the focus of persistent criticism (Linn, 2003; Linn, Baker, & Betebenner, 2002). Though appropriate for making judgments about the achievement level of students at a school for a given year, they are inappropriate for judgments about educational *effectiveness*. In this regard, status measures are blind to the possibility of low achieving students attending effective schools. It is this possibility that has led some critics of No Child Left Behind (NCLB) to label its accountability provisions as unfair and misguided and to demand the use of growth analyses as a better means of auditing school quality.

A fundamental premise associated with using student growth for school/teacher accountability is that "good" schools/teachers bring about student growth in excess of that of "bad" schools/teachers. Students associated with such schools/teachers—commonly referred to as highly effective/ineffective schools/teachers—tend to demonstrate extraordinary growth that is causally attributed to the school or teachers instructing the students. The inherent believability of this premise is at the heart of current enthusiasm to incorporate growth into accountability systems.

Recent legislation at the federal level have seized upon this enthusiasm. In November 2005, Secretary of Education Spellings announced the Growth Model Pilot Program (GMPP) permitting states to use growth model results as a means for compliance with NCLB achievement mandates was met with great enthusiasm by states. (Spellings, 2005). More recently, the Race to the Top initiative extended the use of growth measure to the teacher level, propelling states to use student growth on large scale assessment as part of teacher evaluations (U.S. Department of Education, 2010).

Consistent with current accountability systems that hold schools and educators responsible for the assessment outcomes of their students, the primary thrust of growth analyses over the last decade has been to determine the amount of student progress/growth attributable to the school or teacher (Braun, 2005; Rubin, Stuart, & Zanutto, 2004; Ballou, Sanders, & Wright, 2004; Raudenbush, 2004). Such analyses, often called *value-added* analyses, attempt to estimate the teacher or school contribution to student achievement. This contribution, called the *school* or *teacher effect*, purports to quantify the impact on achievement that this school or teacher would have, on average, upon similar students assigned to them for instruction. Clearly, such analyses lend themselves to accountability systems that hold schools or teachers responsible for student achievement.

Despite their utility in high stakes accountability decisions, the causal claims of teacher or school effectiveness addressed by value-added models (VAM) often fail to address questions of primary interest to education stakeholders. For example, VAM analyses generally ignore a fundamental interest of stakeholders regarding student growth: How much growth did a student make? The disconnect reflects a mismatch between the questions of interest and the statistical model employed. In this direction, Harris (2007) distinguishes value-added for program evaluation (VAM-P) versus value-added for accountability (VAM-A). More broadly, the current climate of high-stakes test-based accountability combined with the emphasis of value-added toward school and teacher effects has skewed discussions about growth models toward causal claims at the expense of description. Research (Yen, 2007) and personal experience suggest stakeholders appear more interested in the reverse: description first that can be used secondarily as part of causal fact finding.

In a survey conducted by Yen (2007), supported by the author's own experience working with state departments of education to implement growth models, parents, teacher, and administrators were asked what "growth" questions were most of interest to them.

Parent Questions:

- Did my child make a year's worth of progress in a year?
- Is my child growing appropriately toward meeting state standards?
- Is my child growing as much in Math as Reading?
- Did my child grow as much this year as last year?

Teacher Questions:

- Did my students make a year's worth of progress in a year?
- Did my students grow appropriately toward meeting state standards?
- How close are my students to becoming Proficient?
- Are there students with unusually low growth who need special attention?

Administrator Questions:

- Did the students in our district/school make a year's worth of progress in all content areas?
- Are our students growing appropriately toward meeting state standards?
- Does this school/program show as much growth as that one?
- Can I measure student growth even for students who do not change proficiency categories?
- Can I pool together results from different grades to draw summary conclusions?

As Yen remarks, all these questions rest upon a desire to understand whether observed student progress is "reasonable or appropriate" (Yen, 2007, p. 281). More broadly, the questions seek a description rather than a parsing of responsibility for student growth. Ultimately, questions may turn to who/what is responsible. However, as indicated by this list of questions, they are not stakeholders' foremost concern.

In what follows, student growth percentiles and percentile growth projections/trajectories are introduced as a means of understanding student growth in both a normative and a criterion referenced fashion. The calculation of these quantities for large scale state assessment data using the student growth percentile (SGP) package (Betebenner, 2011) within the R statistical software environment is then described (R Development Core Team, 2010; Betebenner, 2011). With these values calculated we show how growth data can be utilized both normatively and in a criterion referenced manner to inform discussion about education quality. We assert that the establishment of a normative basis for student growth eliminates a number of the problems of incorporating growth into accountability systems providing needed insight to various stakeholders by addressing the basic question of how much a student has progressed.

Student Growth Percentiles

It is a common misconception that to measure student growth in education, the subject matter and grades over which growth is examined must be on the same scale—referred to as a vertical scale. Not only is a vertical scale not necessary, but its existence obscures concepts necessary to fully understand growth. Growth, fundamentally, requires change to be examined for a single construct like reading achievement across time—*growth in what?* A single scale on which the construct is measured is not necessary.

Consider the familiar situation from pediatrics where the interest is on measuring the height and weight of children over time. The scales on which height and weight are measured possess properties that educational assessment scales aspire towards but can never meet.[1]

> An infant male toddler is measured at 2 and 3 years of age and is shown to have grown 4 inches. The magnitude of increase—4 inches—is a well understood quantity that any parent can grasp and measure at home using a simple yardstick. However, parents leaving their pediatrician's office knowing only how much their child has grown would likely be wanting for more information. In this situation, parents are not interested in an absolute criterion of growth, but instead in a normative criterion locating that 4 inch increase alongside the height increases of similar children. Examining this height increase relative to the increases of similar children permits one to diagnose how (ab)normal such an increase is.

Given this reality in the examination of change where scales of measurement are perfect,

[1] Height and weight scales are interval (actually, ratio scales) where a unit increase reflects an equivalent increase in the underlying quality being measured no matter where on the scale the increase occurs.

it is absurd to think that in education, where scales are quasi-interval, one can/should examine growth differently.[2]

Suppose that scales did exist in education similar to height/weight scales that permitted the calculation of absolute measures of annual academic growth for students. A parent's query about, "How much did my child progress?", would be answered with some quantity of scale score points—an answer that would leave most parents bewildered wondering whether the number of points is good or bad. As in pediatrics, the search for a description regarding change in achievement over time (i.e., growth) is best served by considering a normative quantification of student growth—*a student growth percentile*.

A student's growth percentile describes how (ab)normal a student's growth is by examining their current achievement relative to their *academic peers*—those students beginning at the same place. That is, a student growth percentile examines the current achievement of a student relative to other students who have, in the past, "walked the same achievement path". Heuristically, if the state assessment data set were extremely large (in fact, infinite) in size, one could open the infinite data set and select out those students with the exact same prior scores and compare how the selected student's current year score compares to the current year scores of those students with the same prior year's scores—their academic peers. If the student's current year score exceeded the scores of most of their academic peers, in a normative sense they have done well. If the student's current year score was less than the scores of their academic peers, in a normative sense they have not done well.

The four panels of Figure 1.1 depict what a student growth percentile represents in a situation considering students having only two consecutive achievement test scores.

Upper Left Panel Considering all pairs of 2005 and 2006 scores for all students in the state yields a bivariate (two variable) distribution. The higher the distribution, the more frequent the pair of scores.

Upper Right Panel Taking account of prior achievement (i.e., conditioning upon prior achievement) fixes a the value of the 2005 scale score (in this case at 600) and is represented by the red slice taken out of the bivariate distribution.

Lower Left Panel Conditioning upon prior achievement defines a *conditional distribution* which represents the distribution of outcomes on the 2006 test assuming a 2005 score of 600. This distribution is indicating as the solid red curve.

Lower Right Panel The conditional distribution provides the context against which a student's 2006 achievement can be examined and understood normatively. Students with achievement in the upper tail of the conditional distribution have demonstrated high rates of growth relative to their academic peers whereas those students with achievement in the lower tail of the distribution have demonstrated low rates of

[2] The scales on which students are measured are often assumed to possess properties similar to height and weight but they don't. Specifically, scales are assumed to be interval where it is assumed that a difference of 100 points at the lower end of the scale refers to the same difference in ability/achievement as 100 points at the upper end of the scale. This assumption, however, fails to hold.

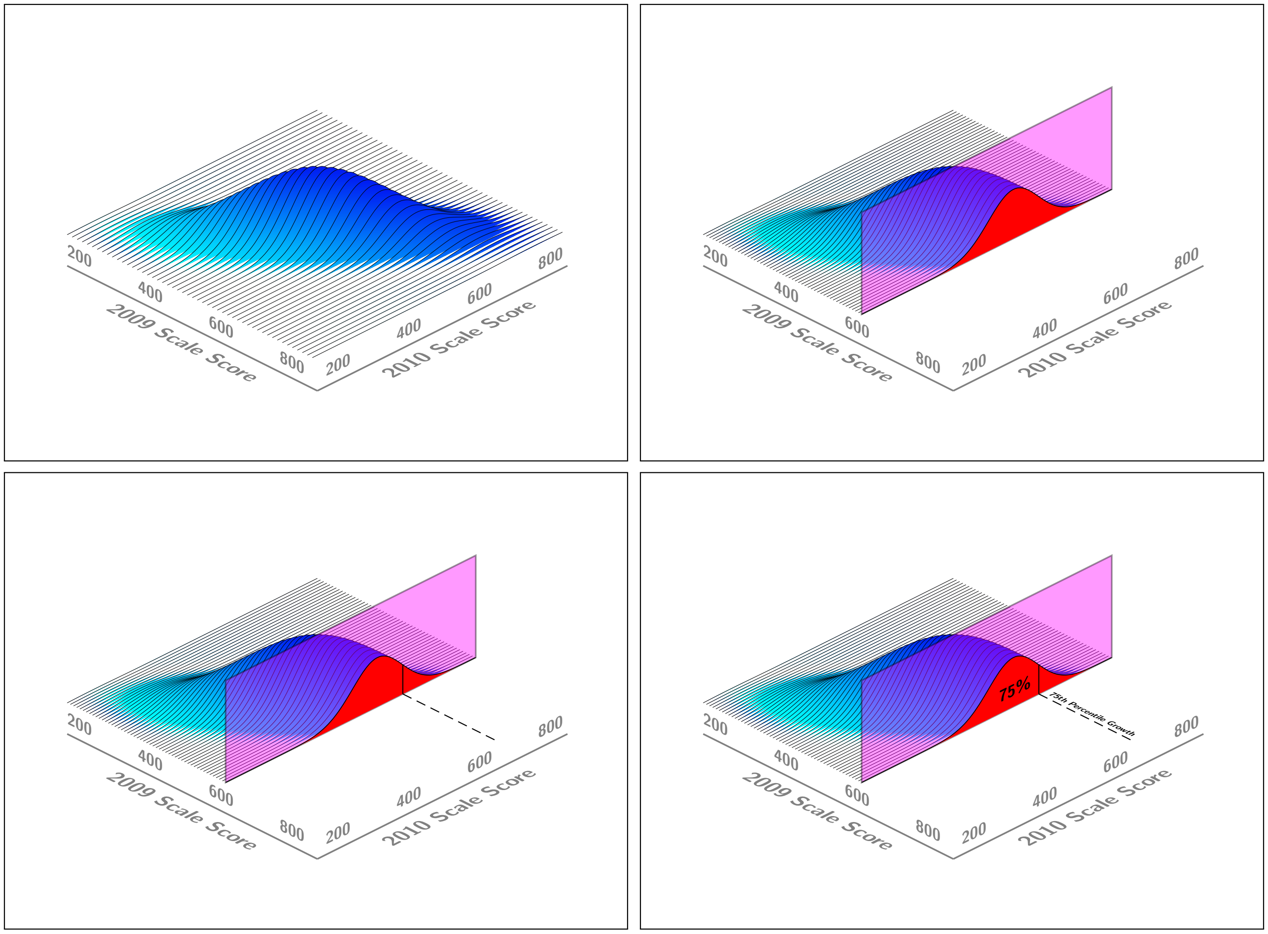

Figure 1.1: Figures depicting the distribution associated with 2005 and 2006 student scale scores together with the conditional distribution and associated growth percentile

growth. Students with current achievement in the middle of the distribution could be described as demonstrating "average" or "typical" growth. In the figure provided the student scores approximately 650 on the 2006 test. Within the conditional distribution, the value of 650 lies at approximately the 70th percentile. Thus the student's growth from 600 in 2005 to 650 in 2006 met or exceeded that of approximately 70 percent of students starting from the same place. This 50 point increase is above average. It is important to note that qualifying a student growth percentile as "adequate", "good", or "enough" is a standard setting procedure that requires stakeholders to examine a student's growth *vis-à-vis* external criteria such as performance standards/levels.

Figure 1.1 also serves to illustrate the relationship between a vertical scale and student growth percentiles. Using the vertical scale implied by Figure 1.1, the student grew 50 points (from 600 to 650) between 2005 and 2006. This 50 points represents the absolute magnitude of change. Quantifying the magnitude of change is scale dependent. For example, different vertical achievement scales in 2005 and 2006 would yield different annual scale score increases: A scale score increase of 50 could be changed to a scale score increase of 10 using a simple transformation of the vertical scale on which all the students are measured. However, relative to other students, their growth has not changed—their growth percentile is invariant to scale transformations common in educational assessment. Student growth percentiles normatively situate achievement change bypassing questions associated with the magnitude of change, and directing attention toward relative standing which, we would assert, is what stakeholders are most interested in.

School Level Results

An advantage of quantifying growth at the student level is that it is a simple task to combine the individual level growth results to quantify group performance. For example, after growth percentiles are calculated for each of 500 students at a school, the distribution of growth percentiles for those 500 students represents how much the students at that school grew in the previous year. Summarizing this distribution's "average" would supply a single number describing the growth of a "typical" student at a given school. Because it is inappropriate to calculate the arithmetic average of a set of percentiles, the median is used as the single number which best describes where the middle of the distribution of student growth percentiles lies.

It is important to note, however, that the median is a *summary* measure of the growth of *many* students at the school. In reality at almost all schools one can find students who grow slowly—students with low growth percentiles—as well as students who grow quickly—students with high growth percentiles. There is room for improvement in all schools.

If students were randomly assigned to schools, then the median growth percentile for a school is expected to be 50. Schools with median student growth percentiles above 50 have students demonstrating, on average, greater than expected growth. And schools with median student growth percentiles below 50 have students demonstrating, on average, less than expected growth. Figure 1.2 shows one way in which school performance can be illustrated using achievement and growth measures. In this way, student growth percentiles can be used to identify schools where student growth is extraordinarily good and bad. Like with student achievement, causal attribution of responsibility for growth should be made only after consideration of ample evidence extending beyond just the large scale assessment results.

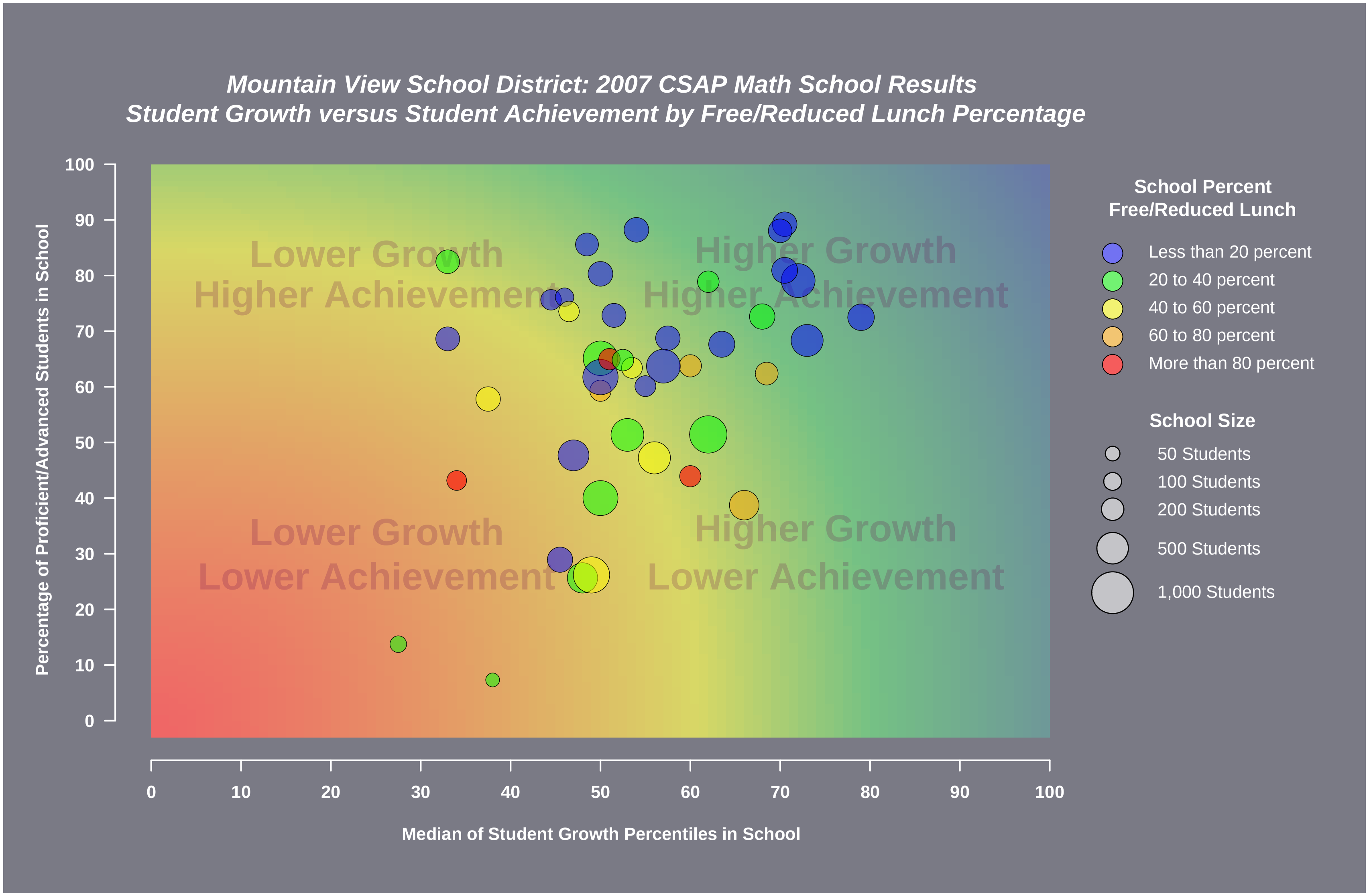

Figure 1.2: Depiction of median student growth percentile against percentage at/above proficient for schools in a school district

Measurement of student growth and assignment of responsibility for that growth involves answering two distinct but related questions:

- How much the the students at this school grow/progress?
- How much did this school contribute to student growth?

The median student growth percentile is descriptive and makes no assertion about the cause of student growth. This differs from current value-added models where the purpose is to specify the contribution to student achievement provided by a given school or teacher. It is likely, that schools and teachers have a significant impact upon student learning: That their efforts are reflected in the academic progress of students. The median student growth percentile is *one of many* indicators that stakeholders can use to judge the quality of the education students receive. It is hoped that as growth percentiles become more widely available, stakeholders will use this piece of data in combination with other data assess the progress of the student as well as the factors contributing to their progress.

The SGP Package

Overview

This chapter provides a *brief* and *complete* set of instructions on how to calculate **student growth percentiles** using the open source and freely available R software combined with the SGP package (R Development Core Team, 2010; Betebenner, 2011). R is available from the Comprehensive R Archive Network (CRAN, http://www.cran.r-project.org/) for Windows, OSX, and Unix/Linux. After installing R, installation of SGP and other packages is performed from the menu via downloads from CRAN. Begin by installing R and the SGP package. Comprehensive R and SGP package help documentation is available locally following the installation as well as online from CRAN.

Hardware requirements for R are modest allowing one to use R even in places with older computer hardware in use. For those thinking of undertaking serious large scale or operational data analysis using, for example, the SGP package, modest hardware is likely to be insufficient to carry out these analyses. R loads all data into memory requiring a good deal of memory to perform analyses. Most users utilize 32 bit operating systems (e.g., most Windows implementations are 32 bit). These operating systems allow for access to only about 4 gb of memory. If one is planning on undertaking large analyses, we suggest using a 64 bit operating system (e.g., Linux) with 8 gb or more of memory available. Such systems have served us well in our own analyses.

In addition to memory, speed is also an issue in running operational analyses. For large state data sets, it can take more than a day to conduct the range of analyses available via the SGP package. Having a high end processor enables the calculations to take place much more rapidly. In addition, beginning with version 0.0-8.0 of the SGP package, parallel analyses will be available allowing for analyses to be distributed across hardware with multi-core processors. This has yielded an order of magnitude increase in the processing times for large scale analyses. Parallel processing has allowed us to analyze student growth percentiles and percentile growth trajectories for a state with 300,000 students per grade across two subject and three years in less than 3 hours. As these parallel implementations are refined further, we expect another order of magnitude increase in speed.

As of SGP 0.0-7.0, the package contains three data sets, `sgpData`, `sgpData_LONG` and `stateData`, and two low level functions: `studentGrowthPercentiles` and `studentGrowthProjections`. The package also contains higher level functions to simplify use of the lower level functions. These functions include `abcSGP`, `prepareSGP`, `analyzeSGP`, `combineSGP`, and `summarizeSGP`.

In addition, the package includes two graphics functions: `bubblePlot` and `studentGrowthPlot`. Please note the two licenses associated with the use of the SGP package. For functions used for the analysis of data are licensed under the Creative Commons BY-SA 3.0 US license. The license allows for use of these functions, even for commercial use, with attribution and with modifications being shared back to the community. The graphics functions (currently, `bubblePlot` and `studentGrowthPlot`) are licensed under the Creative Commons BY-NC-SA 3.0 US license. In particular, these functions and their output cannot be used for any commercial purpose. Please see the package license for further details.

Data requirements

It is certainly an axiom that the most onerous task of doing data analysis in general is getting data in a form amenable for analysis. Use the SGP package is no different. The data used by the SGP package must fulfill certain requirements in order that the analyses are conducted correctly. In the most basic use of the SGP package, the user uses the `studentGrowthPercentiles` and `studentGrowthProjections` to calculate student growth percentiles and percent growth projections/trajectories, respectively. These functions require data in what is typically called a *wide* format where each case/row represents the results for a unique student in the dataset. In later sections where more advanced operational analyses are presented, data is supplied to functions in what is typically called a *long* format that enables much more efficient storage and use of the student growth percentile methodology and functions over time. Higher level functions in the SGP package reshape the data to the wide format for submission to the SGP function.

We begin with a description of the wide format file. The columns of the data file must conform to the following:

- The first column/variable indicates the unique student `ID`.
- The next set of columns/variables indicates the tested `GRADE` of the student for the years in the data.
- The final set of columns/variables indicates the `SCALE SCORES` for the student for the respective grades.

For example, given a data set spanning five years from 2007 to 2011, the data would be formatted as:

ID	GRADE_2007	GRADE_2008	GRADE_2009	GRADE_2010	GRADE_2011	SS_2007	SS_2008	SS_2009	SS_2010	SS_2011

Because the function internally renames variables, *variable names are **NOT** important, just their position within the data file.* Additionally, through the use of the foreign package one can easily read data into R prepared from a variety of statistical software packages. R also has can access directly from relational databases for more thorough integration into data warehouse structures.

Growth Percentile Calculation: Part 1

At the core of the SGP package are two functions (studentGrowthPercentiles and studentGrowthProjections) and one sample data set (sgpData) that can be used with both functions to illustrate how they work. All the other functions in the SGP package build on these two core functions. The installed help pages for the SGP package give thorough details about all the function options together with examples that calculate student growth percentiles and percentile growth projections/trajectories using the data. Use the help pages for these functions to supplement the details provided in this primer.

Using studentGrowthPercentiles

The simplest way to perform analyses for an entire state data set is to construct a "master file" in your favorite text editor (e.g., Wordpad, Notepad, emacs, vi) containing R syntax specific for the user's state specific needs. With an appropriately constructed master file, growth analyses for an entire state can be performed using a single R file. The help pages for studentGrowthPercentiles and studentGrowthProjections contain examples of master files that are subsequently used in this primer. For a state that annually tests students in three subjects, three master files, one for each subject, could be used to conduct the analyses. To "run" a master file one typically uses the source command at the R prompt:

```
> source("master_READING_2011.R")
```

Here the master file in this case is for 2011 reading. The help documentation associated with both the studentGrowthPercentiles and studentGrowthProjections functions contains example files that can be easily modified to accommodate the requirements of different large scale data sets. To help the user understand the R syntax contained within a master file, we consider the text provided in the studentGrowthPercentiles example.

```
require(SGP)
```

the require command in R loads packages that provide functions and data sets necessary to perform analyses. The SGP package, in turn, loads other packages it requires automatically. Note that the # symbol is the comment symbol in R.

```
# require(foreign)
# sgpData <- read.spss(file="/path/to/my/data/my_spss_file.sav",
   to.data.frame=T) ## SPSS
# sgpData <- read.xport(file="/path/to/my/data/my_sas_file.xport")  ## SAS
   XPORT
```

By installing the SGP package, three datasets, sgpData, sgpData_LONG and stateData, are available for use from the command line. Some users not familiar with R will likely use other data analysis/preparation software to manipulate their data. R has the ability to import data in numerous format. The examples above show how functions in the foreign package can be used to import SPSS or SAS files. Specifically, if one wishes to load an SPSS or SAS Xport file named `my_spss_file.sav` or texttmy_sas_file.xport, respectively, then one can uncomment the syntax above to import that file, replacing the `load` statement.

With the packages and data loaded, one can begin growth percentile calculations. Growth percentiles are generally calculated by a student's grade cohort. As will become apparent, there are repeated pieces of R syntax used. The syntax used to calculate growth percentiles using the `sgpData` provided in the SGP package in grades 4 through 8 using 2007 to 2011 data is:

```
## GRADE 4

g4_sgp <- studentGrowthPercentiles(
             panel.data = sgpData,
             sgp.labels=list(my.year=2011, my.subject="READING"),
             grade.progression=c(3,4))
)

save(g4_sgp, file="sgp_READING_2011_g4.Rdata")

## GRADE 5

g5_sgp <- studentGrowthPercentiles(
             panel.data = sgpData,
             sgp.labels=list(my.year=2011, my.subject="READING"),
             grade.progression=3:5
)

save(g5_sgp, file="sgp_READING_2011_g5.Rdata")
```

The first command uses the `studentGrowthPercentiles` function and assigns the result (assigning a result in R is done using `<-` or `=`). Note that the result of running these commands are the creation of 2 objects, `g4_sgp and g5_sgp`. If one doesn't wish to save each grade's results just comment out the `save` commands.

Growth percentile calculations produce a number of objects in addition to the student growth percentiles. Many of these objects are required for subsequent analyses including student growth projections using the `studentGrowthProjections` function. These additional objects are packaged within the output of the function which is an R `list` object. In R, a `list` object is a very general type of object allowing the user to put together many other types of objects within a single object. To view the names of the pieces within the list, just type `names(g5_sgp)` at the prompt. Previously, in versions of SGP prior to 0.0-6.0, the function constructed numerous auxiliary files with the output. Embedding the objects within a single object allows much greater control of data elements that are used in subsequent analyses, making it easier for the user to use the SGP software.

The two objects `g4_sgp` and `g5_sgp` contain results that will likely be combined into results reflect growth for all student in 2011 Reading. Given this common use case, the `SGP` package accommodates the combines automatically when performing repeated analyses on the same content area and year. However, it is instructive to see how we would merge results from these two different objects and save the output both as an R data file as well as a comma separated variable (csv) file:

```
sgp_READING_2011_gall <- rbind(g4_sgp$SGPercentiles$READING.2011,
   g5_sgp$SGPercentiles$READING.2011)
save(sgp_READING_2011_gall, file="sgp_READING_2011_gall.Rdata")
write.csv(sgp_READING_2011_gall, file="sgp_READING_2011_gall.csv",
   row.names=FALSE, quote=FALSE, na="")
```

Like all modern programming languages,R provides a number of powerful control structures that allow loops to be employed to simplify commands associated with repeated tasks. For example, if we wish to run analyses for all grades from 4 to 8, the following loop will calculate student growth percentiles for all grades:

```
my.grade.sequences <- list(3:4, 3:5, 3:6, 3:7, 4:8)
my.sgpData <- list(Panel_Data=sgpData)   ### Put sgpData into Panel_Data
   slot

for (i in seq_along(my.grade.sequences)) {
     my.sgpData <- studentGrowthPercentiles(panel.data=my.sgpData,
                     sgp.labels=list(my.year=2011, my.subject="Reading"),
                     percentile.cuts=c(1,35,65,99),
                     grade.progression=my.grade.sequences[[i]])
}
```

This example demonstrates a number of ways in which data in R can be stored and accessed. The variable `my.grade.sequences` is a list of grade sequences that will be looped over to run the grade level analyses associated with the `sgpData` data set. The use of the `studentGrowthPercentiles` function also differs from the previous uses where instead of a `data.frame` being supplied as an argument, a list is supplied with the `data.frame` embedded in a list. This, as we'll see, allows for us to recycle the object and cumulatively added results to the object. Examining the `Coefficient_Matrices`, `Knots_Boundaries`, `Goodness_of_Fit`

The object `my.sgpData` contains all of the results associated with the analysis of the 2011 Reading data. One can use the same object to store SGP reading analyses from other years as well as results from other subjects. The function uses the `sgp.labels` argument to index the different results so there is no limit on how many results one can store in a single R object. Saving an object in R is a simple task:

```
save(my.sgpData, file="my.sgpData.Rdata")
```

It is, however, often desirable to export results, particularly if they need to be imported into another analysis program. R can export data in multiple formats. To export just the student growth percentile results from the Reading 2011 analyses as a `csv` file, one could use the commands:

```
write.csv(my.sgpData$SGPercentiles$READING.2011, file="sgp_all.csv",
   row.names=FALSE, quote=FALSE, na="")
```

To explore other ways of saving and manipulating data, R includes a number of useful manuals with the package.

Objects Derived from studentGrowthPercentiles calculations

Use of the `studentGrowthPercentiles` function results in the creation of a number of derivative objects (with the default function settings). The objects (which are actually `list` objects) include: Coefficient Matrices (stored in the `Coefficient_Matrices` slot, cutscores associated with state assessments (stored in the `Cutscores` slot, Goodness of fit results (stored in the `Goodness_of_Fit` slot, Knots and Boundaries (stored in the `Knots_Boundaries` slot, panel data used for the analyses (stored in the Panel_Data slot), student growth percentile results (stored in the `SGPercentiles` slot), percentile growth trajectory/projection results (stored in the `SGProjections` slot, and simulated SGP results (stored in the `Simulated_SGP` slot). Depending upon the analyses run by the user, these slots might or might not be used.

Examining the `Coefficient_Matrices`, `Knots_Boundaries`, `Goodness_of_Fit` slots from the previous analyses show that there are slots within slots. The argument to the function `sgp.labels` is used to construct the slots within each of the larger `list` objects. structure in R is analogous to a directory structure within a computer where different data objects are stored.

Knots and Boundaries

When calculating and making comparisons of growth percentile results across years, it is preferable to remove any ambiguities from the data analysis process that might impact year-to-year comparisons. The knots and boundaries used in the growth percentile analyses are two such ambiguities that if allowed to vary might subtly change results. To fix the knots and boundaries, after performing growth percentile analyses in a base year, the user is encouraged to use the knots and boundaries calculated in subsequent analyses in following years.

The SGP package is designed to allow the user to anchor the analyses to a specific set of knots in two ways. The `studentGrowthPercentiles` function accepts the argument use.my.knots.and.boundaries which is a list with slot `my.subject` and `my.year` allowing the user the ability to use the pre-calculated knots and boundaries from a base year. For example, a user might have conducted analyses for Reading in 2008. The knots and boundaries for that analyses are stored in `Knots_Boundaries$READING.2008`. If the user want to use those same knots and boundaries with the 2009, 2010 and 2011 analyses, then the additional argument `use.my.knot.and.boundaries=list(my.subject=''Reading``, my.year=2008)` will allow the function to access these previously calculated knots and boundaries for subsequent analysis.

Additionally, to facilitate state level use of the package, as of version 0.0-6.0 of the package, there is an embedded `stateData` set that contains relevant data for states currently using or investigating student growth percentiles. In particular, `stateData` contains established knots and boundaries used for the analyses in a number of state. Both the sgpData and sgpData_LONG data sets have an embedded set of data in stateData for the state DEMO. To calculate SGPs using these embedded knots and boundaries as well as report the growth level associated with each student growth percentile (e.g., low, SGP from 1 to 34, typical, SGP from 35 to 65, and high, SGP from 66 to 99), the follow syntax could be used.

```
my.grade.sequences <- list(3:4, 3:5, 3:6, 3:7, 4:8)
my.sgpData <- list(Panel_Data=sgpData)

for (i in seq_along(my.grade.sequences)) {
     my.sgpData <- studentGrowthPercentiles(panel.data=my.sgpData,
                      sgp.labels=list(my.year=2011, my.subject="Reading"),
                      growth.level="DEMO",
                      use.my.knots.boundaries="DEMO",
                      percentile.cuts=c(1,35,65,99),
                      grade.progression=my.grade.sequences[[i]])
}
```

In addition, the stateData contains cutscores associated with state assessments, achievement level names and many other data labels to be used in the graphics functions associated with the package. See the stateData help file for further information.

Coefficient Matrices

The heart and most time consuming part of the SGP analysis is the production of coefficient matrices from the b-spline quantile regression procedure that provides the 100 functional relationships relating prior student scores to current scores across the percentile distribution. These matrices are stored in the Coefficient_Matrices slot according to the sgp.labels that are used with the function call. Once calculated, it is possible to use the studentGrowthPercentiles function with these already calculated coefficient matrices to calculate student growth percentiles. This is done by using the use.my.coefficient.matrices argument and supplying a list containing my.subject and my.year that will be used locate the coefficient matrices within the Coefficient_Matrices slot.

Pre-made coefficient matrices can serve many functions. Scheduled for implementation in version 0.0-9.0 will be the ability to calculate base-line referenced student growth percentiles using coefficient matrices derived from a baseline analysis. States wanting such analyses and their coefficient matrices available, these matrices will be embedded in the stateData data set so that users can calculate student growth percentiles based upon these matrices using just the two letter state acronym. Updates to the SGP package are indicated in the NEWS section of the SGP page on the CRAN website (http://cran.r-project.org/web/packages/SGP/NEWS).

Goodness of Fit

As part of the default settings for the studentGrowthPercentiles function, figures are produced that allow the user to see how well the data is being fit by the b-spline based quantile regression procedure. Because percentile are, by definition, a uniform distribution, across all prior score levels, we expect to see a uniform distribution of percentiles. That is, regardless of whether the student enter the year as a low achiever or high, their chances of having a high or low growth percentile are based upon the uniform distribution.

The goodness of fit plots are grid graphical objects stored in the Goodness_of_Fit slot. To view the grade 4 plot from the previous analyses, the following code could be used:

```
grid.draw(sgp_g4$Goodness_of_Fit$READING.2011$GRADE_4)
```

It is also possible to export these figures into numerous format using graphical output drivers available in R. For example, to export as PDF files, the following code could be used:

```
for (i in names(my.sgpData$Goodness_of_Fit$READING.2011)) {
    pdf(file=paste(i, "_Reading_2011_GOF.pdf", sep=""), width=8.5,
       height=4.5)
       grid.draw(my.sgpData[["Goodness_of_Fit"]][["READING.2011"]][[i]])
    dev.off()
}
```

Simulated SGPs

For states that have CSEMs (conditional standard errors of measurement) embedded in the `stateData` set, the `studentGrowthPercentiles` function can calculate simulated SGPs that can be used for the construction of both individual and group level confidence intervals. This functionality is controlled using the `calculate.confidence.intervals` argument to the function. The argument is a list of the form `list(state= , confidence.quantiles= , simulation.iterations= , distribution= , round= )` specifying the state from which to use the CSEMs (note, CSEM data must be embedded in `stateData` set. To have your state CSEMs embedded in the `stateData` set, please contact the package administrator), confidence.quantiles to report from the simulated student growth percentiles calculated for each student, simulation.iterations indicating the number of simulated student growth percentiles to calculate, distribution indicating whether to the the Normal or Skew-Normal to calculate SGPs, and round (defaults to 1, which is an integer—see `round_any` from plyr package for details) giving the level to round to. If requested, simulations are calculated and simulated SGPs are stored in the `Simulated_SGPs` slot.

Percentile Growth Trajectory/Projection Calculation

Following the calculation of student growth percentiles, it is often desirable to use the results from these analyses to investigate what level of growth is necessary for the student to reach desirable levels of achievement in the future. These "growth projections" or "percentile growth trajectories" allow stakeholders to quantify and ultimately discuss what it will take for a student to reach desired levels of achievement. These analyses, sometimes referred to as growth-to-standard analyses, can form the basis for a criterion referenced growth model suitable for use with AYP determinations (Betebenner, 2009).[1]

[1] The Colorado Growth Model which uses percentile growth projections/trajectories to determine whether a student is track to reach/maintain proficiency was approved in January, 2009, by the U.S. DOE for use in determining AYP as part of the Growth Model Pilot Program.

Using `studentGrowthProjections`

Percentile growth projections/trajectories use the coefficient matrices, saved in the `Coefficient_Matr` slot, derived in the calculation of student growth percentiles to calculate what future rates of growth, expressed in the growth percentile metric, lead to. The user can specify a set of cutpoints that the function uses to determine what (consecutive) percentile growth is required to reach these cutpoints. For example, users can specify cutpoints associated with state achievement levels. To facilitate use of the function, users can supply a two letter acronym and access embedded cutscores for the state or they can supply a list of cutscores of their own.[2] Based upon these supplied cutscores, the function then calculates the individual (consecutive) growth percentiles necessary to reach these levels in coming years. The user can either specify the time frame to reach the achievement levels or the function will, by default, calculate target until it reaches the maximum available grade in the data.

In addition to getting cutscores necessary to reach or maintain predefined achievement level targets, the user can specify a set of percentiles and the function will output the scale scores associated with consecutive percentile growth trajectory associate with those percentiles for any number of years available in the future.

Like in calculation of student growth percentiles using the `studentGrowthPercentiles` function, the simplest way to calculate percentile growth projections/trajectories for an entire state data set is to construct a "master file" containing the necessary R syntax. The help pages for `studentGrowthProjections` contain an example of a master file that will be used in this section. Like with student growth percentiles, for a state that annually tests students in three subjects, three master files, one for each subject, could be used to conduct the projection analyses. The master files are "run" using the `source` command at the R prompt:

```
> source("master_sgp_proj_reading_2011.r")
```

Here the percentile growth projections master is for 2011 reading.

Because percentile growth trajectories/projections rely upon previously calculated coefficient matrices in order to extrapolate into the future, use of the `studentGrowthProjections` function follows after using the `studentGrowthPercentiles` function.

R syntax used to calculate percentile growth trajectories/projections has been simplified considerably from earlier versions of the SGP package. Arguments are available as part of the function that allow the fine tuning of the previous function. Use of these arguments should only be attempted by those with a thorough understanding of the methodology and the use of the SGP package.

The calculation of the `my.sgpData` object done previously, the R syntax required to calculate student growth percentiles is as follows.

[2] If your state's cutscores are not embedded in the `stateData` data set and you'd like them to be, please contact the package administrator.

```
my.grade.progressions <- list(3, 3:4, 3:5, 3:6, 4:7)

for (i in seq_along(my.grade.progressions)) {
     my.sgpData <- studentGrowthProjections(panel.data=my.sgpData,
                        sgp.labels=list(my.year=2011, my.subject="Reading"),
                        performance.level.cutscores="DEMO",
                        percentile.trajectory.values=c(25, 50, 75),
                        grade.progression=my.grade.progressions[[i]])
}
```

Growth Percentile Calculation: Part 2

Based upon feedback from state data analysts over the past two years, the SGP package has matured to simplify many of the analyses and include numerous features based upon requests from users. Over the last 9 months, the package has growth significantly in terms of its sophistication and feature set. Besides just calculate student growth percentiles and percentile growth trajectories/projections, the function calculates a number of summary quantities associated with school level reporting and will include the ability to create high quality graphical representations of the results.

Data requirements

Based upon work with more than a dozen states, the most difficult part of doing these analyses is getting the data in the appropriate format so that the analyses become "simple". The goal of the SGP package has always been to turn well formatted data into growth results, quickly and faithfully. In line with this goal, in working with states, it is now (almost as of version 0.0-7.0) possible to go from well formatted data to a growth model and reporting system completely within the package. Namely, a state growth model includes not only individual level norm and criterion-referenced growth results for students but, in addition, summary level results at the state, district, school, and teacher level. To perform all of these computation on the results of the student growth percentile analyses themselves requires meticulous preparation of the data so that all subsequent analyses follow seamlessly.

Within the SGP package is a prototype data set (sgpData_LONG) that analysts can use to model their own data on. The data represents 5 years of annual assessment data from a fictitious state (consisting of 3 districts) with students tested from grade 3 to 10 in reading and mathematics.

with the following variables:

```
> names(sgpData_LONG)
 [1] "ID"                                "LAST_NAME"
 [3] "FIRST_NAME"                        "CONTENT_AREA"
 [5] "YEAR"                              "GRADE"
 [7] "SCALE_SCORE"                       "ACHIEVEMENT_LEVEL"
 [9] "GENDER"                            "ETHNICITY"
[11] "FREE_REDUCED_LUNCH_STATUS"         "ELL_STATUS"
```

```
[13] "IEP_STATUS"
   "GIFTED_AND_TALENTED_PROGRAM_STATUS"
[15] "SCHOOL_NUMBER"                      "SCHOOL_NAME"
[17] "EMH_LEVEL"                          "DISTRICT_NUMBER"
[19] "DISTRICT_NAME"                      "SCHOOL_ENROLLMENT_STATUS"
[21] "DISTRICT_ENROLLMENT_STATUS"         "STATE_ENROLLMENT_STATUS"
[23] "VALID_CASE"
```

Once appropriately prepared, the data can then be summarily analyzed using set of higher level functions implemented in the SGP package:

`prepareSGP` This function prepares the long data and embeds it appropriately in a larger list in preparation for student growth percentile and percentile growth trajectory/projection analysis.

`analyzeSGP` This function conducts the student growth percentile and percentile growth trajectory/projection analyses, across multiple years, content areas and grades if requested including calculating simulated SGPs for student and group level confidence intervals. The function calculates two kinds of projections: straight projections and lagged projections. The latter are used in growth to standard analyses that can be used to judge the adequacy of student growth.

`combineSGP` This function takes the results of the analyzeSGP function and combines them back with the long file that was embedded in the `prepareSGP` step.

`summarizeSGP` This function creates summary level aggregates (e.g., school level medians) based upon grouping variables that exist in the long data file. For example, aggregations by state, district and school as well as demographic aggregations within these units are provided.

`reportSGP` To be implemented in version 0.0-8 or 0.0-9. This function will take the results derived from the previous steps and provide

`abcSGP` This function includes all of the previous functions into a simple function that can be both general and specific.

abcSGP

To motivate the power of some of the higher level functions currently implemented and under development, consider abcSGP. When used with the embedded data, `sgpData_LONG`, it is extremely simple to calculate all relevant quantities:

```
DEMO_Data <- abcSGP(sgp_object=sgpData_LONG, state="DEMO")
```

Running this code takes the long data file and runs it through all of the analyses laid out thus far. On a high speed work station not taking advantage of any parallelizing, the above analyses takes 50 minutes to perform. Using parallel processing, we anticipate the above analyses taking less than 5 minutes.

bubblePlot and studentGrowthPlot

As of SGP version 0.0-7.0 there includes to functions that allow for visualization of the growth percentile analyses at the summary level as well as the individual level. The function `bubblePlot` has been used in multiple states to help communicate the results of the growth analyses. The function `studentGrowthPlot` has been used by three states to produce individual level student reports for ALL student in their testing programs. Results from their use can be seen in Figures 2.1 and 2.2, respectively. The function `reportSGP` will utilize these plotting functions extensively and help states wanting to produce student or group level representations of data. Stay tuned to the SGP CRAN web page for updates.

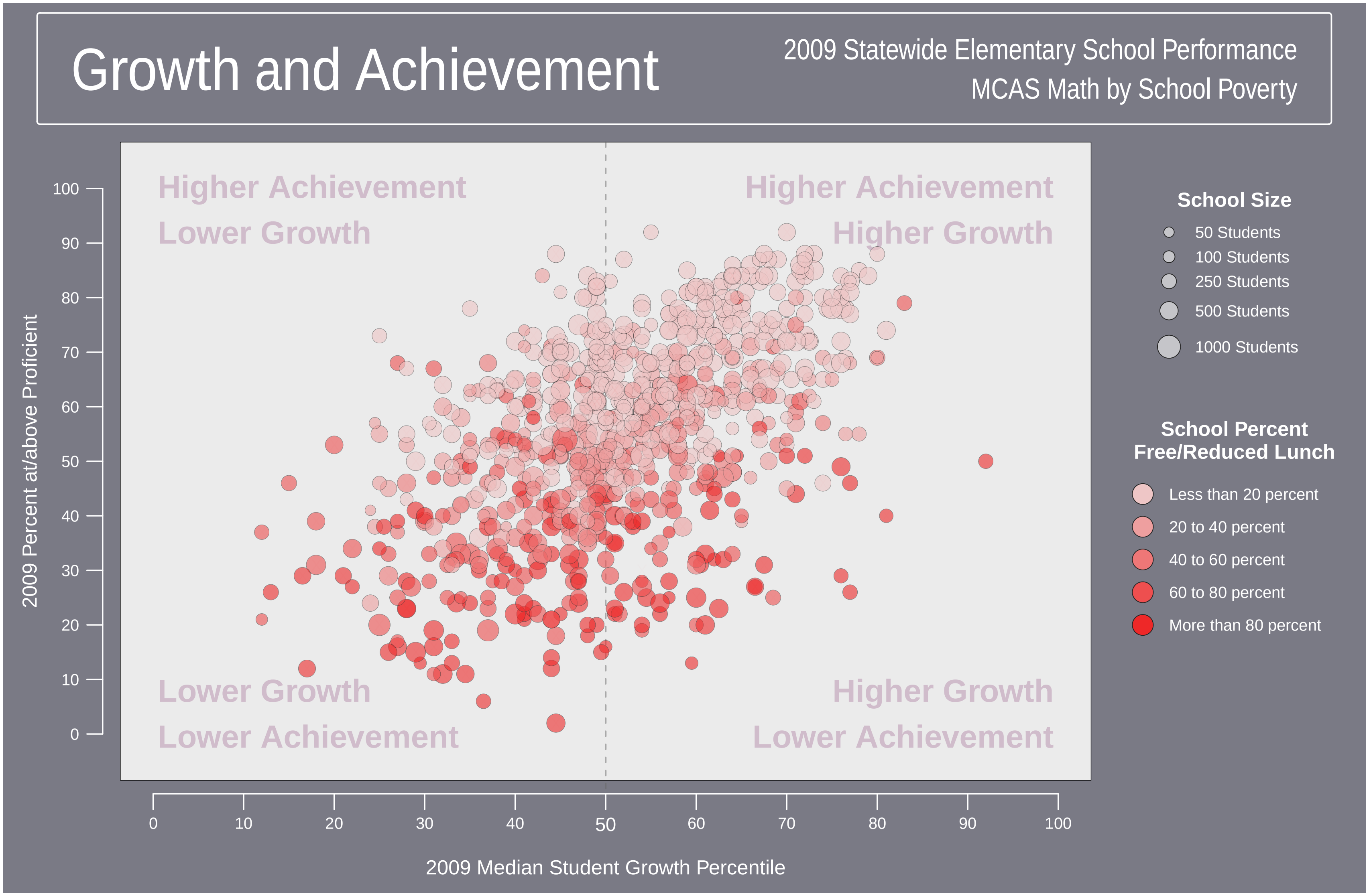

Figure 2.1: 2009 Massachusetts state elementary school growth and achievement results for high poverty schools

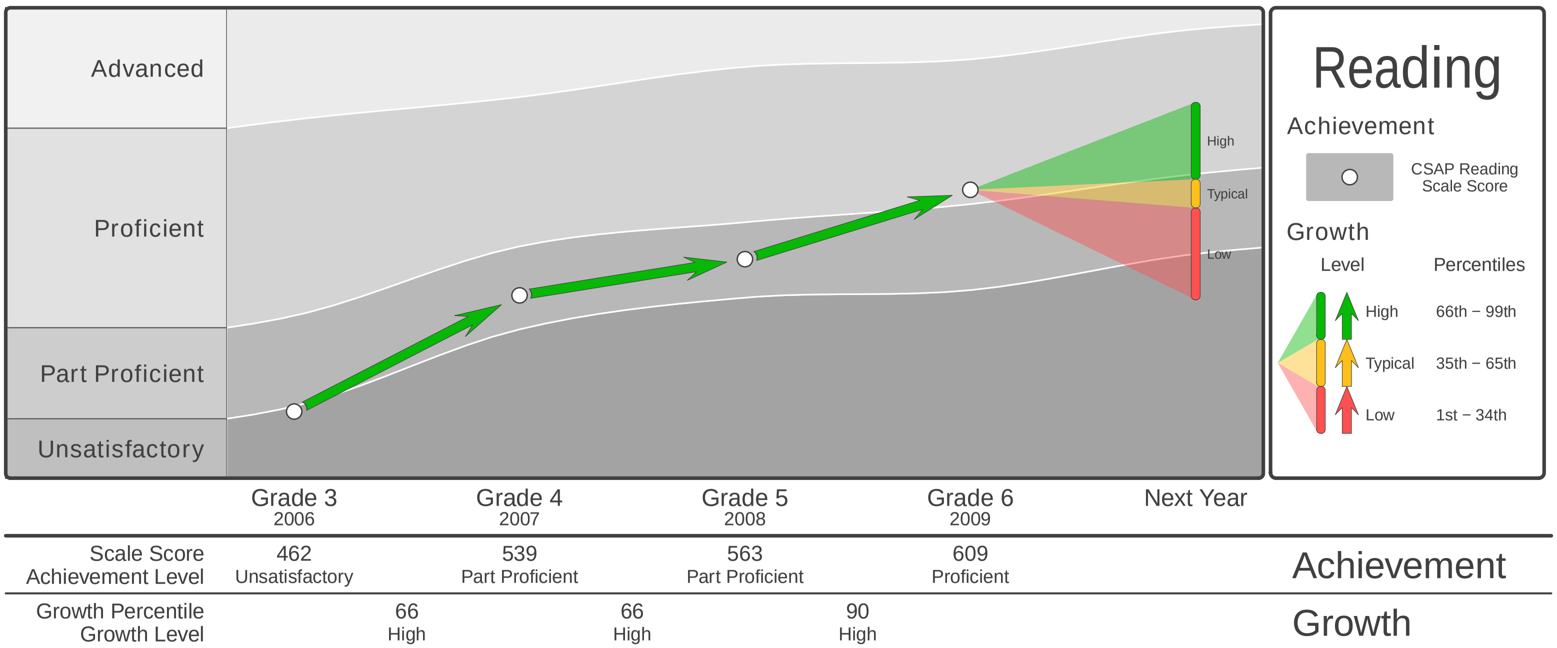

Figure 2.2: Individual student growth chart in reading depicting growth and achievement over time

References

Ballou, D., Sanders, W., & Wright, P. (2004). Controlling for student background in value-added assessment for teachers. *Journal of Educational and Behavioral Statistics*, *29*(1), 37–65.

Betebenner, D. W. (2009). Norm- and criterion-referenced student growth. *Educational Measurement: Issues and Practice*, *28*(4), 42–51.

Betebenner, D. W. (2011). SGP: Student growth percentile and percentile growth projection/trajectory functions [Computer software manual]. (R package version 0.0-6.7)

Braun, H. I. (2005). *Using student progress to evaluate teachers: A primer on value-added models* (Tech. Rep.). Princeton, New Jersey: Educational Testing Service.

Harris, D. N. (2007). *The policy uses and "policy validity" of value-added and other teacher quality measures* (Tech. Rep.). Princeton, NJ: Educational Testing Service.

Linn, R. L. (2003, July). *Accountability: Responsibility and reasonable expectations* (Tech. Rep.). Los Angeles, CA: Center for the Study of Evaluation, CRESST.

Linn, R. L., Baker, E. L., & Betebenner, D. W. (2002). Accountability systems: Implications of requirements of the No Child Left Behind Act of 2001. *Educational Researcher*, *31*(6), 3–16.

R Development Core Team. (2010). R: A language and environment for statistical computing [Computer software manual]. Vienna, Austria. Available from `http://www.R-project.org` (3-900051-07-0)

Raudenbush, S. W. (2004). What are value-added models estimating and what does this imply for statistical practice? *Journal of Educational and Behavioral Statistics*, *29*(1), 121–129.

Rubin, D. B., Stuart, E. A., & Zanutto, E. L. (2004). A potential outcomes view of value-added assessment in education. *Journal of Educational and Behavioral Statistics*, *29*(1), 103–116.

Spellings, M. (2005, Nov). *Secretary Spellings announces growth model pilot* [Press Release]. U.S. Department of Education. (Retrieved August 7, 2006 from `http://www.ed.gov/news/pressreleases/2005/11/1182005.html`)

U.S. Department of Education. (2010, June 24). *Race to the Top Assessment Program* [Announcement of Race to the Top Program by U.S. Department of Education.]. Announcement of Race to the Top Program by U.S. Department of Education. (Retrieved June 24, 2010 from `http://www2.ed.gov/programs/racetothetop-assessment/index.html`)

Yen, W. M. (2007). Vertical scaling and No Child Left Behind. In N. J. Dorans, M. Pommerich, & P. W. Holland (Eds.), *Linking and aligning scores and scales* (pp. 273–283). New York: Springer.