5 Models for Assembling Single Tests

This chapter deals with applications of the methodology introduced in the previous chapters to four different classes of test-assembly problems. In each of these problems, the task is to assemble a single test from an item pool, but they differ in the objectives with which the tests are assembled.

The first class of problems departs from Birnbaum's (1968) problem of assembling a single test of discrete items to a target for its information function (Section 1.2.8). An important difference between absolute and relative targets is introduced. We discuss a few methods for specifying both types of targets. In addition, we show how once a target of either type is specified, the test-assembly problem can be modeled as an MIP problem.

Whereas the first class of problems assumes the fit of the item pool to an IRT model, the second class of problems is based on classical test theory only. The objectives are to assemble a test with optimal reliability or predictive validity, respectively. In principle, these objectives lead to problems with nonlinear objective functions. Because we generally want to avoid such functions, an important part of our treatment is to show how these problems can be linearized.

In the third class of problems, the objective is to assemble a test to match a prespecified observed-score distribution for a given population of test takers. The observed-score distribution may be one for a previous test in the same program, but it is also possible to specify a target distribution based on practical considerations only. This objective may seem somewhat unusual. But, as a matter of fact, an identical objective is pursued in the practice of observed-score equating, where, once a test is assembled and administered, the number-correct score is transformed to produce the same score distribution as for a reference test. We show that this form of *post hoc* equating can be avoided by imposing a set of constraints on the test-assembly problem. Surprisingly, the constraints we need have a simple linear form.

The final class consists of item-matching problems. In item matching, the objective is to assemble a new test that matches a reference test item by item. The method we discuss can be used with any combination of item attributes: classical item indices, IRT parameters, or more substantive quantitative or categorical attributes. If a test is assembled to have the same information function or reliability as a reference test, the two tests are weakly parallel. Item matching enables us to assemble a test that is parallel to a reference test in a much stronger sense.

As just noted, the critical difference between these four classes of test-assembly problems resides mainly in their objective functions. These functions may require a few technical constraints. But for any of these models it is possible to add a set of substantive constraints to the model to deal with the remaining content specifications. As numerous examples of such constraints were already presented in Chapter 3, we will omit a further discussion of this option in this chapter.

5.1 IRT-Based Test Assembly

A target for a TIF is a function $\mathcal{T}(\theta)$ that provides goal values for it along the θ scale in use for the item pool. For mainstream response models, such as the 3PL model in (1.16), TIFs are well-behaved, smooth functions. It therefore holds that if we require a TIF to meet a smooth target $\mathcal{T}(\theta)$ at one point on the θ scale, it automatically approximates the target in a neighborhood of this point. Also, target values for fewer points tend to result in much faster solutions. In practice, we therefore specify target values for TIFs at only a few points on the θ scale, which we denote as $\mathcal{T}(\theta_k)$, $k = 1, ..., K$. Extensive simulation studies and ample experience with practical test-assembly problems have shown that this number need not be larger than 3–5 points.

As already discussed in Section 1.2.7, we assume that these points are selected by a test assembler familiar with both the numerical scale and the substantive interpretation of the θ scale in use for the item pool. To give an idea of a typical choice of a set of values θ_k, we consider the case of a target for the TIF that has to provide diagnostic information on a population of persons centered at $\theta = 0$ with standard deviation $\sigma_\theta = 1$. For the 3PL model, target values that can be expected to yield excellent results are typically specified at $(\theta_1, \theta_2, \theta_3) = (-1.0, 0, 1.0)$ or $(\theta_1, \theta_2, \theta_3, \theta_4) = (-1.5, -.5, .5, 1.5)$.

The problem of assembling a test with an information function that has to meet a target is a *multiobjective* test-assembly problem. More specifically, if we intend to minimize the differences between the TIF and its target at values θ_k, $k = 1, ..., K$, we have a problem with K different objectives. General approaches to multiobjective test-assembly problems were discussed in Section 3.3.4. These approaches will be used extensively to solve our current class of problems.

5.1.1 Absolute and Relative Targets

An important distinction exists between absolute and relative targets. A target is *absolute* if it specifies a fixed number of information units at the points θ_k. This type of target was assumed when Birnbaum introduced his approach to test assembly (Section 1.2.8). To specify a meaningful absolute target, we need to be familiar not only with the θ scale but also with the unit of the information measure that the scale implies. If we are unfamiliar with it, unexpected results may occur (for example, unrealistically long or short tests if the test length is left free, or large deviations from the target if it is constrained). For this reason, absolute targets are used almost exclusively when tests are assembled to be parallel to a known reference test. We will use $\mathcal{T}_k$ as shorthand notation for the absolute target values $\mathcal{T}(\theta_k)$, $k = 1, ..., K$, for the TIF.

If an absolute target is specified, we in fact imply that more information than specified by the target is undesirable. From a measurement point of view, this implication seems peculiar, but in practice it often makes sense. An example is admissions testing, with different institutions setting their own admission scores on an observed-score scale for the test. If the information function of a new test overshot the target along a portion of the θ scale, the observed-score distribution would change in the neighborhood of some of the admission scores. As a consequence, without any change in the population of examinees, the proportion of examinees that qualify for admission may go up or down, a result that would certainly embarrass the institutions concerned.

However, in other applications more information is always better, as long as it is distributed along the θ scale in a way that reflects the objectives for the test. Examples are found in broad-range diagnostic testing and testing for licensing with a fixed minimum level of performance required for passing. The only thing we then want to control is the shape of the information function. But if we are interested only in the shape of the target but not in its height, we in fact have a relative target for the TIF. Formally, a *relative target* can be defined as a set of numbers $\mathcal{R}_k > 0$ that represent the required amount of information at θ_k relative to the other points in the set $k = 1, ..., K$. For instance, if we want the test to have twice as much information at θ_k as at θ_{k+1}, the numbers $\mathcal{R}_k$ and $\mathcal{R}_{k+1}$ need to be chosen such that $\mathcal{R}_k/\mathcal{R}_{k+1} = 2$. Because we have to specify

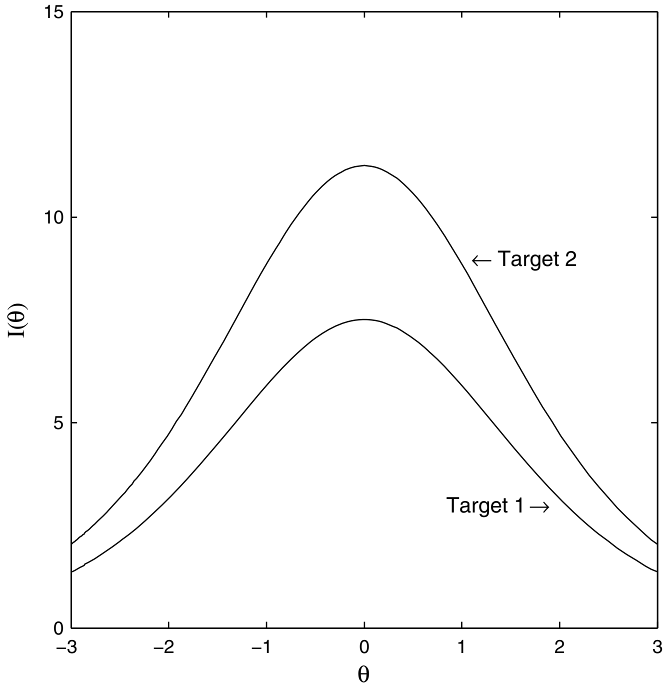

FIGURE 5.1. Example of two targets for a test information function with the same shape but heights differing by a factor of 1.5.

ratios of numbers only, fortunately to select them we need not be familiar with the unit of the information measure. Hence, as will become clear below, the choice of a relative target is less likely to result in test assembly with unexpected results.

An example of two target information functions with the same relative shape is given in Figure 5.1. The two targets have identical ratios for their values at each pair of θ values. As a consequence, their height differs only by a common factor, which is 1.5 in this example. When we model test-assembly problems with a relative target below, we introduce a new decision variable to represent this factor.

5.1.2 Methods for Specifying Targets for Information Functions

To specify an absolute target for a new testing program, we could simply follow a trial-and-error method, alternately selecting a set of values for the target function and checking the actual TIFs for the tests assembled from the pool. In principle, this method works fine because solutions are quickly obtained. The following alternative methods are more direct, however:

1. The first alternative is based on descriptive statistics of the distribution of the item parameters in the pool. From these statistics, we can choose a set of n combinations of values that are typical of the pool, where n is the intended test length. We can then calculate the information function for this set and edit it to improve its representation of the primary objectives for the testing program. For instance, if the test has to be diagnostic over a θ interval, we could replace the information function by one with a uniform shape at average height. Or if the test is for decision making at one or more cutoff scores θ_c, we may feel obliged to increase the information at these scores, taking information away from other areas along the scale.

2. A second alternative is to ask test specialists to assemble manually a specimen of the tests that should be used in the program. This specimen should meet all other specifications, except those involving the information functions of the items, and consists of items that the specialists consider typical. The actual information function of the specimen provides us with a good estimate of the general level of information possible for the test. The function can then be edited for better representation of the goals for the testing program. An advantage of the second method is that it takes all nonstatistical item attributes into account and therefore is based on a feasible test from the item pool. A disadvantage is that it is not based on the distribution of the item parameter values in the pool.

3. An interesting method is the Kelderman method, which is based on the equality in (1.18) between the information function and the inverse of the (asymptotic) variance of the ML estimator of θ

   $$[I(\theta)]^{-1} = \mathrm{Var}(\widehat{\theta} \mid \theta), \tag{5.1}$$

   where our notation shows the dependence of the variance on the true value of θ. For two persons with true abilities θ_1 and θ_2, it holds for the variance of the difference between their estimators that

   $$\begin{aligned}\mathrm{Var}(\widehat{\theta}_1 - \widehat{\theta}_2 \mid \theta_1, \theta_2) &= \mathrm{Var}(\widehat{\theta}_1 \mid \theta_1) + \mathrm{Var}(\widehat{\theta}_2 \mid \theta_2) \\ &= [I(\theta_1)]^{-1} + [I(\theta_2)]^{-1}. \end{aligned} \tag{5.2}$$

   Because both estimators have an (asymptotic) normal distribution, it follows that

   $$\Pr\{\widehat{\theta}_1 > \widehat{\theta}_2 \mid \theta_1, \theta_2\} = \Phi\left(\frac{\theta_1 - \theta_2}{\{[I(\theta_1)]^{-1} + [I(\theta_2)]^{-1}\}^{1/2}}\right), \tag{5.3}$$

where $\Phi(.)$ is the distribution function of the standard normal distribution. If $\theta_1 < \theta_2$, $\Pr\{\widehat{\theta}_1 > \widehat{\theta}_2 \mid \theta_1, \theta_2\}$ is the probability that the scores on the tests for persons at θ_1 and θ_2 are ordered erroneously. Making the information values explicit, we obtain

$$[I(\theta_1)]^{-1} + [I(\theta_2)]^{-1} = \left\{(\theta_1 - \theta_2)[\Phi^{-1}(\Pr\{\widehat{\theta}_1 > \widehat{\theta}_2 \mid \theta_1, \theta_2\})]^{-1}\right\}^2. \tag{5.4}$$

In the Kelderman method, a set of pairs of points $(\theta_1, \theta_2\ )$ is presented to a panel of test specialists, who are asked to specify the probabilities with which they are willing to accept test scores for persons at these points who order them erroneously. If these probabilities are known, so is the right-hand side of (5.4), and we have a set of equations, one for each pair, that can be solved for the unknown information values on their left-hand sides, which serve as target values. Although this method also runs the risk of resulting in target values leading to unexpected test lengths, it has the advantage of translating the costs due to measurement errors directly into target values for the information function.

To specify a relative target for a TIF, the following two approaches are available:

1. Each of the three methods above can be used, but we ignore the absolute nature of the resulting numbers, using them only as relative target values in the test-assembly models we introduce below.

2. A simpler alternative is to offer test specialists an arbitrary number of chips (100, say) and ask them to distribute them over the points θ_k, $k = 1, ..., K$, in an item map (Figure 1.5) such that their distribution reflects the relative accuracy needed in the test scores for persons at these points. The number of chips at θ_k is then the relative target value $\mathcal{R}_k$. The total number of chips is arbitrary because the numbers $\mathcal{R}_k$ are unitless.

5.1.3 Assembling Tests for Absolute Targets

Before discussing several examples of objective functions that can be used to select a TIF to meet a set of target values, we discuss a set of constraints that also does the job. Let $\delta_k \geq 0$ and $\varepsilon_k \geq 0$ be small tolerances with which the TIF is allowed to be larger or smaller than the target values $\mathcal{T}_k$. Adding the following set of constraints to the model forces the TIF to be close to the target:

$$\sum_{i=1}^{I} I_i(\theta_k)x_i \leq \mathcal{T}_k + \delta_k, \quad \text{for all } k, \tag{5.5}$$

$$\sum_{i=1}^{I} I_i(\theta_k)x_i \geq \mathcal{T}_k - \varepsilon_k, \quad \text{for all } k. \tag{5.6}$$

The tolerances in (5.5.) and (5.6) are indexed by k to allow them to be dependent on the value of $\mathcal{T}_k$; for example, somewhat larger for the middle values of $\mathcal{T}_k$ or at values θ_k where the item pool has been relatively depleted.

An advantage of using a set of constraints to realize a target is that we still have the opportunity to formulate an objective function for another attribute. On the other hand, to avoid infeasibility, the tolerances δ_k and ε_k have to be chosen realistically for the item pool. If we follow the alternatives below, for a well-designed item pool infeasibility is no problem.

Our first approach to the multiobjective problem of matching a target at K points is the weighted-objectives approach in Section 3.3.4. Let w_k be the weight for the objective of minimizing the positive deviation of the TIF from target value $\mathcal{T}_k$. The following combination of objective function and constraints allows us to minimize a weighted sum of positive deviations from the K target values,

$$\text{minimize } \sum_{k=1}^{K} w_k \sum_{i=1}^{I} I_i(\theta_k)x_i, \tag{5.7}$$

subject to

$$\sum_{i=1}^{I} I_i(\theta_k)x_i \geq \mathcal{T}_k, \quad \text{for all } k \tag{5.8}$$

(Exercise 5.1).

If we had omitted the set of constraints in (5.8), the objective function would minimize the total weighted sum of the TIF values at the values θ_k. Because of the presence of the constraints, the objective function minimizes only positive deviations from the target values.

The objective in (5.7) and (5.8) permits compensation between individual values of the TIF, and the result may therefore show an undesirably large local deviation. This element of unpredictability is absent in the following application of the minimax principle introduced in Section 3.3.4:

$$\text{minimize } y \tag{5.9}$$

subject to

$$\sum_{i=1}^{I} I_i(\theta_k)x_i \leq \mathcal{T}_k + y, \quad \text{for all } k, \tag{5.10}$$

$$\sum_{i=1}^{I} I_i(\theta_k)x_i \geq \mathcal{T}_k, \quad \text{for all } k, \tag{5.11}$$

$$y \geq 0, \tag{5.12}$$

where y is a real-valued decision variable. The constraints in (5.11) require the TIF to be larger than the target values, while the constraints in (5.10) define decision variable y as an upper bound to all positive deviations from these values. The upper bound is minimized in (5.9). Ideally, the best possible result is obtained for $y = 0$, but typically a slightly larger value is obtained because it is hard for a sum of item-information functions from a pool to meet a set of target values exactly (Exercise 5.2).

Thus far, one of our assumptions has been that small positive deviations of the TIF from the target values are permitted but negative deviations are forbidden. If both types of deviations are considered equally undesirable, the target values become goal values for the TIF, and the following alternative to the model for the weighted-objective approach in (5.7) and (5.8) can be useful:

$$\text{minimize} \sum_{k=1}^{K} w_k (y_k^{\text{pos}} + y_k^{\text{neg}}) \tag{5.13}$$

subject to

$$\sum_{i=1}^{I} I_i(\theta_k) x_i = \mathcal{T}_k - y_k^{\text{pos}} + y_k^{\text{neg}}, \quad \text{for all } k, \tag{5.14}$$

$$y_k^{\text{pos}} \geq 0, \quad \text{for all } k, \tag{5.15}$$

$$y_k^{\text{neg}} \geq 0, \quad \text{for all } k, \tag{5.16}$$

with $w_k \geq 0$ for all k.

The constraints in (5.14)–(5.16) define the new decision variables y_k^{pos} and y_k^{neg} as possible positive and negative deviations from the target values $\mathcal{T}_k$. If the objective function takes a minimal value, at each θ_k only one of the two variables can be positive and the other is equal to zero. For example, substitution of $y_k^{\text{pos}} = 0$ in (5.14) shows that y_k^{neg} is a possible negative deviation at θ_k in the solution. Likewise, y_k^{pos} is a possible positive deviation (Exercise 5.3).

The attribute in (5.13) is the weighted sum of absolute deviations of the TIF values from $\mathcal{T}_k$; (5.13)–(5.16) is thus a linear equivalent of the following objective function, which minimizes the sum of the absolute deviations from the target values:

$$\text{minimize} \sum_{k=1}^{K} w_k \left| \sum_{i=1}^{I} I_i(\theta_k) x_i - \mathcal{T}_k \right|. \tag{5.17}$$

It is for this reason that (5.13)–(5.16) can be used as a two-sided alternative to (5.7) and (5.8).

Likewise, (5.9)–(5.12) can be replaced by a minimax approach in which the largest absolute deviation from $\mathcal{T}_k$ is minimized. The optimization problem then becomes

$$\text{minimize } y \tag{5.18}$$

subject to

$$\sum_{i=1}^{I} I_i(\theta_k)x_i \leq \mathcal{T}_k + y, \quad \text{for all } k, \tag{5.19}$$

$$\sum_{i=1}^{I} I_i(\theta_k)x_i \geq \mathcal{T}_k - y, \quad \text{for all } k, \tag{5.20}$$

$$y \geq 0. \tag{5.21}$$

The constraints in (5.19) and (5.20) enclose the differences between the TIF and target values in an interval about zero, $[-y, y]$, and the size of the interval is minimized by (5.18) (Exercise 5.4).

Our favorite approach is the minimax model in (5.18)–(5.21), particularly if the test is assembled in a program where a fixed target has to be maintained over time. In such applications, positive and negative deviations from the target values are equally undesirable. By minimizing the largest deviation from the target, the model presses the TIF as closely as possible against the target, avoiding surprises in the form of large local deviations.

5.1.4 Assembling Tests for Relative Targets

If the target for the TIF is relative, we maximize its height at each θ_k, $k = 1, ..., K$, but at the same time want to maintain its relative shape. Intuitively, the problem seems to be one with K objectives and a set of additional constraints to maintain the shape of the TIF.

We begin with the formulation of the constraints. Because the target values $\mathcal{R}_k$ have no fixed unit, one of them can be set equal to one, provided we adjust all other values correspondingly. Suppose we choose to set $\mathcal{R}_1 = 1$. The following $K - 1$ constraints require the TIF at θ_k to be $\mathcal{R}_k$ times as large as at θ_1 and therefore guarantee the desired shape of the TIF:

$$\sum_{i=1}^{I} I_i(\theta_k)x_i = \mathcal{R}_k \sum_{i=1}^{I} I_i(\theta_1)x_i, \text{ for } k \geq 2 \tag{5.22}$$

(Exercise 5.5).

By imposing these constraints, we automatically reduce the number of K objectives to one. Therefore, to maximize the TIF at the K values θ_k simultaneously, we only need to maximize the TIF value at one of these values. Suppose we choose to maximize test information at θ_1. In principle, the following objective function then seems to complete our formalization of the problem:

$$\text{maximize} \sum_{i=1}^{I} I_i(\theta_1)x_i. \tag{5.23}$$

An annoying complication, however, is that (5.22) contains equality constraints on a quantitative test attribute. Such constraints should always be avoided because of possible infeasibility; see our discussion of (3.50). A simple remedy may seem to replace them by the following inequalities:

$$\sum_{i=1}^{I} I_i(\theta_k)x_i \geq \mathcal{R}_k \sum_{i=1}^{I} I_i(\theta_1)x_i, \quad \text{for } k \geq 2. \tag{5.24}$$

A consequence of this step is that, though the solution can be expected to realize these inequalities close to equality, the information at θ_1 tends to stay somewhat behind. This effect can be remedied by lowering the target values at θ_2,...,θ_K somewhat relative to the value at θ_1. But a more satisfactory solution is to substitute a new variable y for the common factor $\sum_{i=1}^{I} I_i(\theta_1)x_i$ in the lower bounds in (5.24) and formulate the model as

$$\text{maximize } y \tag{5.25}$$

subject to

$$\sum_{i=1}^{I} I_i(\theta_k)x_i \geq \mathcal{R}_k y, \quad \text{for all } k, \tag{5.26}$$

$$y \geq 0. \tag{5.27}$$

This argument has resulted in another application of the maximin principle. This claim becomes clear if both sides of (5.26) are divided by $\mathcal{R}_k$. Variable y then becomes an explicit common lower bound to the relative information $\mathcal{R}_k^{-1} \sum_{i=1}^{I} I_i(\theta_1)x_i$ at the points θ_k, which is maximized in (5.25).

5.1.5 Cutoff Scores

If a test is used for decisions with a cutoff score θ_c, often all we need is informative estimates $\widehat{\theta}$ in the neighborhood of θ_c. If these estimates have to meet a prespecified level of information, the results in Section 5.1.3, which are for simultaneous optimization at θ_k, $k = 1, ..., K$, specialize to optimization only at θ_c.

For the same case of decision making, a relative target for the TIF boils down to simple maximization at θ_c, ignoring the information at all other values of θ; that is, to the objective function

$$\text{maximize } \sum_{i=1}^{I} I_i(\theta_c)x_i. \tag{5.28}$$

5.1.6 Empirical Examples

An empirical example of a single test assembled from a previous pool of 753 items from the *Law School Admission Test* (LSAT) is given. The total length of the test was 101 items. The items were in three different sections and measured analytic reasoning, logical reasoning, and reading comprehension. As in the LSAT, one of these sections was doubled in our example. The specifications of the LSAT were modeled as constraints dealing with such attributes as test and section length, item type and content, answer key, gender and minority orientation of the items, and word counts. The objective function was that for the maximin model in (5.18)–(5.21). All specifications were modeled except those for the stimuli and item sets in two of the sections of the test that have an item-set structure. Examples for the LSAT that do include these specifications are given in Chapter 7, which is devoted entirely to the assembly of tests with item sets.

The LSAT has an information function between two functions over the interval between $\theta = -3.0$ and $\theta = 3.0$ that serve as its lower and upper bound. To show an example with an absolute target function, we chose the function that represented the midpoints between these bounds. To show the impact of the number of θ values at which the TIF is controlled, three different tests were assembled. The TIF of the first test was controlled only at θ_k=0, the TIF of the second test was controlled at θ_k=−1.2, 0, and 1.2, and the TIF of the third test was controlled at θ_k=−2.1, −1.2, 0, 1.2 , and 2.1. The three models had a total of 754 variables (the number of items in the pool plus minimax variable y), 114 content constraints, and 2–10 constraints used to control the TIFs.

As shown in the top panel of Figure 5.2, the TIF for the first test met the target at θ_k=0, exactly as required. But it was too low for the smaller θ values and too high for the larger values. The addition of the constraints on the TIF at θ_k=−1.2 and 1.2 was already sufficient to meet the target function over the entire range. In fact, the TIFs in the middle panel of Figure 5.2 obtained for this case and the one with an additional control at θ_k=−2.1 and 2.1 in the bottom panel are indistinguishable for all practical purposes. These results confirm what we have found in numerous IRT-based test-assembly problems and had already formulated as a recommendation in the introduction to Section 5.1: In practice, it is sufficient to control the TIF only at 3–5 well-chosen values.

Although the results in Figure 5.2 may look impressive, the graphs do not reveal the most important result in these examples—which is the fact that each of these three tests met the entire set of the content specifications for the LSAT.

5.2 Classical Test Assembly

A classical objective in test assembly is to maximize the reliability coefficient of the test. If the test is used for prediction of an external criterion (e.g., success in a program or a job), tests are assembled to have maximum predictive validity. Both objectives are nonlinear. To apply the

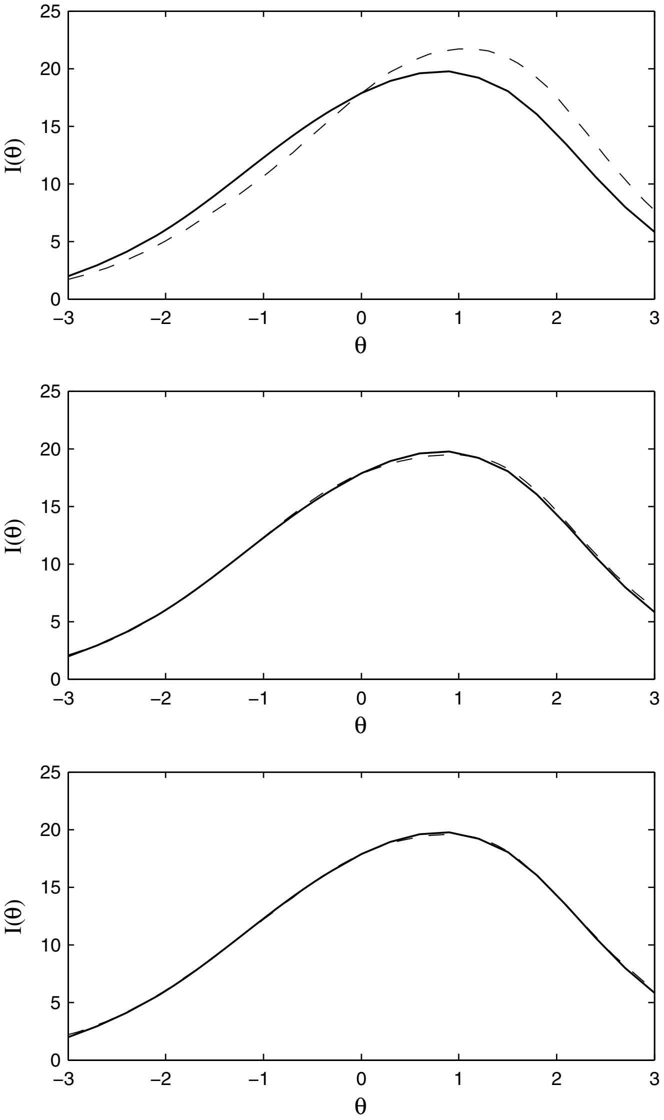

FIGURE 5.2. Information functions of three LSAT forms assembled for a common target (bold line) at $\theta_k = 0$ (top), $\theta_k = -1.2, 0, 1.2$ (middle), and $\theta_k = -2.1$, -1.2, 0, 1.2, 2.1 (bottom).

methodology in this book, we thus have to decide how to linearize these objectives.

5.2.1 Maximizing Test Reliability

Generally, it is difficult to estimate the reliability coefficient of a test. But a well-known lower bound to the reliability coefficient is Cronbach's coefficient α, which was written in (1.14) as

$$\alpha = \frac{n}{n-1}\left[1 - \frac{\sum\limits_{i=1}^{n}\sigma_i^2}{\left(\sum\limits_{i=1}^{n}\sigma_i\rho_{iX}\right)^2}\right], \tag{5.29}$$

where n is the length of the test, σ_i^2 and ρ_{iX} are the variance and discriminating power (item-test correlation) of item i, respectively, and X is the observed score on the test.

Suppose we have a pool of items, $i = 1, ..., I$, with estimates of the item parameters σ_i^2 and ρ_{iX} and want to assemble a test of n items with a maximum value for (5.29). We postpone a discussion of a problem involved in the definition of the scale of X in ρ_{iX} to Section 5.2.4.

If we use 0-1 decision variables for the selection of the items, the value of α for an arbitrary test with n items from the pool can be written as

$$\alpha = \frac{n}{n-1}\left[1 - \frac{\sum\limits_{i=1}^{I}\sigma_i^2 x_i}{\left(\sum\limits_{i=1}^{I}\sigma_i\rho_{iX}x_i\right)^2}\right]. \tag{5.30}$$

Hence, selecting a test with α as the objective function involves an optimization problem that is nonlinear in the variables.

However, as n is fixed, maximization of α is equivalent to the minimization problem

$$\text{minimize } \frac{\sum\limits_{i=1}^{I}\sigma_i^2 x_i}{\left(\sum\limits_{i=1}^{I}\sigma_i\rho_{iX}x_i\right)^2} \tag{5.31}$$

subject to

$$\sum_{i=1}^{I} x_i = n, \tag{5.32}$$

$$x_i \in \{0,1\}, \quad \text{for all } i. \tag{5.33}$$

Although the objective function is still nonlinear, both its denominator and numerator contain expressions that are linear in the variables x_i. The problem is therefore equivalent to an optimization problem with two linear objectives: (1) minimization of $\sum_{i=1}^{I} \sigma_i^2 x_i$ and (2) maximization of $\sum_{i=1}^{I} \sigma_i \rho_{iX} x_i$.

A standard approach in multiobjective optimization is to formulate one of the objectives as the objective function and reformulate the other as a constraint (Section 3.3.4). We choose to formulate the first objective as the constraint and the second as the objective function, and we replace (5.31)–(5.33) by the problem

$$\text{maximize} \sum_{i=1}^{I} \sigma_i \rho_{iX} x_i \tag{5.34}$$

subject to

$$\sum_{i=1}^{I} \sigma_i^2 x_i \leq \kappa, \tag{5.35}$$

$$\sum_{i=1}^{I} x_i = n, \tag{5.36}$$

$$x_i \in \{0, 1\}, \quad \text{for all } i, \tag{5.37}$$

where $\kappa > 0$ is a constant.

Our choice to formulate the constraint on $\sum_{i=1}^{I} \sigma_i^2 x_i$ can be motivated by the fact that further analysis of (5.31) shows that α is more sensitive to its denominator than its numerator. In addition, for dichotomous items, (5.35) constrains the sum of the item variances $\sigma_i^2 = \pi_i(1-\pi_i)$, which has a known range of possible values: Its minimum is equal to zero and its maximum equal to $.25n$. It can be shown that larger values for α tend to be obtained for κ closer to $.25n$ than to zero. Empirical results substantiating this claim are reported in Section 5.2.4. If the applications are for a new item pool and we have no idea what value to choose for κ, the best approach is to solve the model in (5.34)–(5.37) for a sequence of values of κ approaching the maximum and choose the solution with the largest values of α.

5.2.2 Maximizing Predictive Validity

A similar approach is possible for the problem of maximizing the predictive validity of a test. Let Y be the external criterion that the test has to predict; the validity coefficient is the product-moment correlation between test scores X and Y, ρ_{XY}.

As shown in (1.15), for a test of n items, the validity coefficient can be written as

$$\rho_{XY} = \frac{\sum\limits_{i=1}^{I} \sigma_i \rho_{iY}}{\sum\limits_{i=1}^{I} \sigma_i \rho_{iX}}, \tag{5.38}$$

where ρ_{iY} is the item-criterion correlation or item validity. Both the numerator and denominator of the validity coefficient have expressions that are linear in the items, and the same type of linear decomposition as for coefficient α is possible.

In this case, the decision of which expression to optimize is based on the following argument: Both expressions depend on σ_i, but we can expect the item discriminations, ρ_{iX}, to show a somewhat larger variation than the item validities, ρ_{iY}. It therefore makes sense to choose the expression in the denominator for the objective function.

The following model results:

$$\text{minimize} \sum_{i=1}^{I} \sigma_i \rho_{iX} x_i \tag{5.39}$$

subject to

$$\sum_{i=1}^{I} \sigma_i \rho_{iY} x_i \leq \kappa, \tag{5.40}$$

$$\sum_{i=1}^{I} x_i = n, \tag{5.41}$$

$$x_i \in \{0, 1\}, \quad \text{for all } i. \tag{5.42}$$

The minimum and maximum values possible for $\sigma_i \rho_{iY} x_i$ are equal to zero and .50, respectively. The maximum is reached if $\pi_i = .50$ and ρ_{iY} is 1.0, but it is unlikely to have values of ρ_{iY} larger than .40 in practice. For a new problem, again it is recommended to run the model with κ varying the between the minimum and maximum possible values of the sum in (5.40) for the item pool and choose the solution test with the largest value for ρ_{XY} as the solution.

5.2.3 Constraining Test Reliability

Applications of classical test theory can be met in which the intention is not to maximize the test reliability but to keep it as close as possible to a target value. This treatment of reliability is standard in a testing problem for which each next test form has to be parallel to a reference test.

A simple set of constraints to maintain the value of α in a testing program is

$$\sum_{i=1}^{I} \sigma_i \rho_{iX} x_i \leq \kappa_1 + \delta, \tag{5.43}$$

$$\sum_{i=1}^{I} \sigma_i \rho_{iX} x_i \geq \kappa_1 - \delta, \tag{5.44}$$

$$\sum_{i=1}^{I} \sigma_i^2 x_i \leq \kappa_2 + \varepsilon, \tag{5.45}$$

$$\sum_{i=1}^{I} \sigma_i^2 x_i \geq \kappa_2 - \varepsilon, \tag{5.46}$$

where κ_1 is the empirical value of the reference test for $\sum_{i=1}^{n} \sigma_i \rho_{iX}$, κ_2 the value for $\sum_{i=1}^{n} \sigma_i^2$, and δ and ε are small tolerances.

In principle, the same type of constraints are possible for the predictive validity coefficient in (5.38), but we are not aware of any problems in the practice of testing for which this solution would make sense.

Observe that in (5.43)–(5.46) we constrain two sums of item attributes across tests. Tests can be made parallel in a stronger sense if we constrain attributes on an item-by-item basis. This problem belongs to the topic of item matching, which will be addressed in Section 5.4.

5.2.4 Empirical Example

The model in (5.34)–(5.37) was used in a simulation study with a pool of 500 items and values for the item variances and discriminations generated for a population of test takers with a standard normal distribution of θ. The test length was set at $n = 20$. The maximum value of $\sum_{i=1}^{I} \sigma_i^2 x_i$ was 5, and the bound κ in (5.35) was varied between 3 and 5, with step size .5.

One of the problems with banking large numbers of items on empirical values for their classical indices is the definition of the item-discrimination index ρ_{iX}. Using this index makes sense only if the total scores X are comparable across items. In practice, this requirement can be met if the values of the index have been collected using tests that are (approximately) parallel. In this simulation study, we were able to calculate the values of the index using the observed scores of simulated test takers for the entire item pool, B. We first assembled the test using the item-bank correlations, ρ_{iB}. Once the test was assembled, we used simulated observed scores on it to calculate the actual item-test correlation, ρ_{iX}, and recalculated α. Because all responses were generated under the unidimensional 3PL model, the two correlations were monotonically related, and optimization using ρ_{iB} and ρ_{iX} resulted in the same test.

The results are presented in Table 5.1. The second and third columns report the values of coefficient alpha for ρ_{iB} and ρ_{iX}, denoted as α^* and α, respectively. These columns show that the values of α were always higher than those of α^*. This inequality holds because α was calculated for a total score on the best items in the (unidimensional) pool. Table 5.1 also shows better results for larger values of κ. In fact, the best results were obtained for $\kappa = 5$, which is the maximum value of $\sum_{i=1}^{I} \sigma_i^2 x_i$ possible for a test with $n = 20$ items. For this value of κ, the constraint in (5.35) was thus

κ	α^*	α
5.0	.8395	.8712
4.5	.8388	.8678
4.0	.8288	.8559
3.5	.8008	.8401
3.0	.7696	.8205

TABLE 5.1. Values of coefficient alpha for tests assembled for κ=3.0 (.5) 5.0.

redundant; the same results would have been obtained if we had selected the n items with the largest values for $\sigma_i \rho_{iX}$.

This conclusion does not generalize to test-assembly problems with empirical values for the item indices, however: If the pool is not purely unidimensional, the optimum value of α is obtained for κ somewhat lower than the maximum. More importantly, if the test has to be selected to meet a set of content specifications, we cannot pick the n items with the largest values for $\sigma_i \rho_{iX}$ but need (5.34)–(5.37) as the core of a full-fledged test-assembly model to select the best test from the set of feasible solutions.

5.3 Matching Observed-Score Distributions

Most long-running testing programs report their scores on a scale introduced before they began to use IRT for analyzing the test items and assembling their tests. These scales are typically observed-score scales; for example, number-correct scales with an additional (monotonic) transformation to give the scores a standard range. In this section, we ignore this additional transformation without any loss of generality.

The use of observed-score scales entails the necessity of score equating, and the method of equipercentile equating has been the standard of the testing industry for a long time. In an equipercentile equating study, the new test is administered along with a reference test; for example, using a sampling design with randomly equivalent groups. The data from the study are used to find the transformation that maps the new number-correct scores to the scale of the reference test.

It is possible to replace this form of *post hoc* observed score equating by a few simple constraints in the test-assembly model that guarantee the number-correct scores on the new test to be on the same scale as the number-correct scores on the reference test. The test-assembly model then automatically performs what can be called *observed-score pre-equating.*

Test assembly with these constraints has several practical advantages:

1. No resources are spent on separate equating studies.
2. Scores on the new test can be reported immediately after the test is administered.

3. The scale for the scores on the new test is not distorted by a nonlinear transformation but keeps its interpretation as a number-correct scale. In principle, a simple count of the correct answers on the new test is all that is needed to report scores.

4. Uncertainties inherent in traditional equating studies, such as those due to imperfect implementation of an equating design, an arbitrary population definition, and smoothing of the observed-score distributions, are avoided.

5. The observed scores on the new test are equitable; for any person, they have the same error distribution as the scores on the reference test.

Of course, these advantages are only realized if the items fit the response model used in the testing program. This condition is stringent. But if it is not met, the quality of the item pool may be doubtful. If the cause of misfit is a violation of the unidimensionality assumption for θ, observed-score equating becomes a meaningless operation at all.

5.3.1 Conditions on the Response Functions

Suppose we have two tests, each consisting of n items. The response functions of the items in the two tests are denoted as $p_i(\theta)$ and $p_j(\theta)$, where both i and j run over $1, ..., n$. We use X and Y to denote the number-correct scores on the two tests. The rth power of the response probabilities of item i is denoted as $p_i^r(\theta)$. For example, if $r = 2$, it thus holds that $p_i^2(\theta)$ is the square of the probability of a correct response on item i by a person with ability θ.

A general property of the distributions of the observed scores X and Y on these two tests is the following:

Proposition 5.1. For any population of persons, the distributions of the observed scores X and Y are identical if and only if

$$\sum_{i=1}^{n} p_i^r(\theta) = \sum_{j=1}^{n} p_j^r(\theta), \quad -\infty < \theta < \infty, \tag{5.47}$$

for $r = 1, ..., n$.

For $r = 1$, the sum of the response functions in (5.47) is known as the *test characteristic function,* or TCF, in (1.22). The proposition thus shows that for two observed-score distributions to be identical, not only should the characteristic functions of the two tests match, but the same should hold for the sums of the higher-order powers of their response functions. In addition, it is known that the importance of these conditions strongly decreases with the order of the power. In fact, if the test length increases, eventually all conditions for $r \geq 2$ become superfluous. For the test lengths met in practice, the conditions have to be satisfied only for, say, the first 2 or 3 powers.

The conditions in (5.47) are on expressions that are well-behaved, smooth functions of θ. For this reason, just as for the TIFs in Section 5.1, if the conditions are approximated for a few well-chosen values of θ, they are in fact approximated over the entire portion of the scale covered by these values. Also, note that, though they contain powers of response probabilities, the conditions in (5.47) are linear in the items. Both features suggest incorporating constraints into the test-assembly model that realize the conditions with respect to a reference test. The model then automatically produces a test that has the same observed-score scale as the reference test for any population of test takers.

5.3.2 Constraints in the Test-Assembly Model

Let $j = 1, ..., n$ denote the items in the reference test. The sums of the powers of the response probabilities in (5.47) for this test at a given set of values θ_k, $k = 1, ..., K$, are known constants,

$$\mathcal{T}_{rk} = \sum_{i=1}^{n} p_i^r(\theta_k), \tag{5.48}$$

which can be calculated directly from the response functions of the reference test. We use $\mathcal{T}_{rk}$ as target values for the sums of the rth powers of the response probabilities of the test that is assembled from the pool, that is, for the sums

$$\sum_{i=1}^{I} p_i^r(\theta_k) x_i \tag{5.49}$$

for all k.

The problem is another example of a multiobjective test-assembly problem. We therefore propose the following weighted minimax approach:

$$\text{minimize } y \tag{5.50}$$

subject to

$$\sum_{i=1}^{I} p_i^r(\theta_k) x_i \leq \mathcal{T}_{rk} + w_r y, \quad \text{for all } k \text{ and } r \leq R, \tag{5.51}$$

$$\sum_{i=1}^{I} p_i^r(\theta_k) x_i \geq \mathcal{T}_{rk} - w_r y, \quad \text{for all } k \text{ and } r \leq R, \tag{5.52}$$

$$\sum_{i=1}^{I} x_i = n, \tag{5.53}$$

$$x_i \in \{0, 1\}, \quad \text{for all } i, \tag{5.54}$$

$$y \geq 0, \tag{5.55}$$

where $w_r > 0$ is the weight for the rth power and R is the condition with the highest order used. The constraints in (5.51) and (5.52) enclose the differences between the sums of powers of the probabilities for the test and the target values in intervals about zero, $[-w_r y, w_r y]$, and the common factor y in the size of these intervals is minimized in (5.50). Generally, because $p_i^r(\theta_k)$ is smaller for a larger value of r, we should choose smaller values of w_r for lower values of r. However, in the empirical examples discussed in Section 5.3.4, we already got excellent results using $w_r = 1$ for all values of r.

5.3.3 Discussion

Unlike the target values $\mathcal{T}_{rk}$ in (5.48) suggest, these values need not be calculated for an actual test. They can also be derived from a typical set of item-parameter values in the item pool and then be maintained during the program. This setup guarantees the maintenaince of a fixed observed-score scale. Changes in the actual observed-score distributions are then entirely due to changes in the ability distribution of the persons, and we can directly use the former to monitor the latter.

The attentive reader may have noted that the conditions in (5.47) are in fact on the conditional distributions of X and Y given θ. If it holds that the two conditional distributions are identical over the whole range of values of θ, the marginal distributions are identical for *any* population of test takers. The method in the preceding section is thus population-independent. Another advantage of the current method of *local observed-score equating* over the practice of using marginal observed-score distributions is that the equated scores are equitable; that is, they have identical error distributions for each test taker (see point 5 in the introduction to Section 5.3). A discussion of all key differences between local and global equating is, however, beyond the scope of this book.

5.3.4 Empirical Examples

The same pool of 753 items from the LSAT and the same set of content constraints as in the examples in Section 5.1.6 was used.

A reference test was assembled from the pool to meet all the constraints using the same target information function as in Section 5.1.6. The target values $\mathcal{T}_{rk}$ in (5.48) were calculated from the response functions of this test. The remaining part of the item pool was used to assemble a test to meet these target values. The model used to assemble the test was exactly the same as for the reference test, except for the objective function and constraints on the TIF in (5.18)–(5.20), which were replaced by those on the sums of powers of the response probabilities in (5.50)–(5.55).

The tests were assembled under three different conditions: (1) constraints on the target values for the first-order sums of powers ($r = 1$), (2) on the first two orders ($r = 1, 2$), and (3) on the first three orders ($r = 1, 2, 3$). For each condition, two different tests were assembled, one with the constraints only at $\theta_k = 0$ and the other at $\theta_k = -1.2$, 0, and 1.2. For each case, the weights in the constraints were put equal to w_r=1. The observed-score distributions for the solutions and the target test were calculated for a population of test takers with a standard normal distribution for θ using the well-known Lord-Wingersky algorithm. (For this algorithm, see the literature section at the end of this chapter.)

The observed-score distributions for the solutions and their targets are displayed in Figures 5.3–5.5. For each of the three conditions, the best results were always obtained for the constraints at three θ values. Nevertheless, the results for one θ value in the condition with $r = 1$ were already surprisingly good. The best results in these six examples were obtained for the constraints for $r = 1, 2$ at three θ values (lower panel in Figure 5.4), with those for $r = 1$ at three θ values (lower panel in Figure 5.4) being virtually identical.

5.4 Item Matching

Several of the test-assembly problems addressed so far can be considered as problems in which we tried to optimize a match between a test attribute and a target. The test attributes were generally sums of item attributes (item-information functions and powers of item response functions). As will be demonstrated in this section, it is also possible to assemble a test with the objective of matching the attributes with those of a reference test item by item. This type of matching is much stronger: If two tests are matched at the item level, they are also matched with respect to the sums of their attributes. But the reverse is not necessarily true; if we match at the test level only, large compensation between the attributes of the individual items is possible.

Two main versions of this type of test assembly are discussed. We first show how to model the problem of matching the items in a new test to those in a reference test. Thereafter, we show how the same approach can be used to split a given test into two halves that are as parallel as possible; for example, in terms of their classical item parameters. This problem arises if we try to optimize a split-half estimate of the test reliability; split-half coefficients are estimates of a lower bound to the test reliability, and the closer the two halves are to being parallel, the sharper the bound. Our

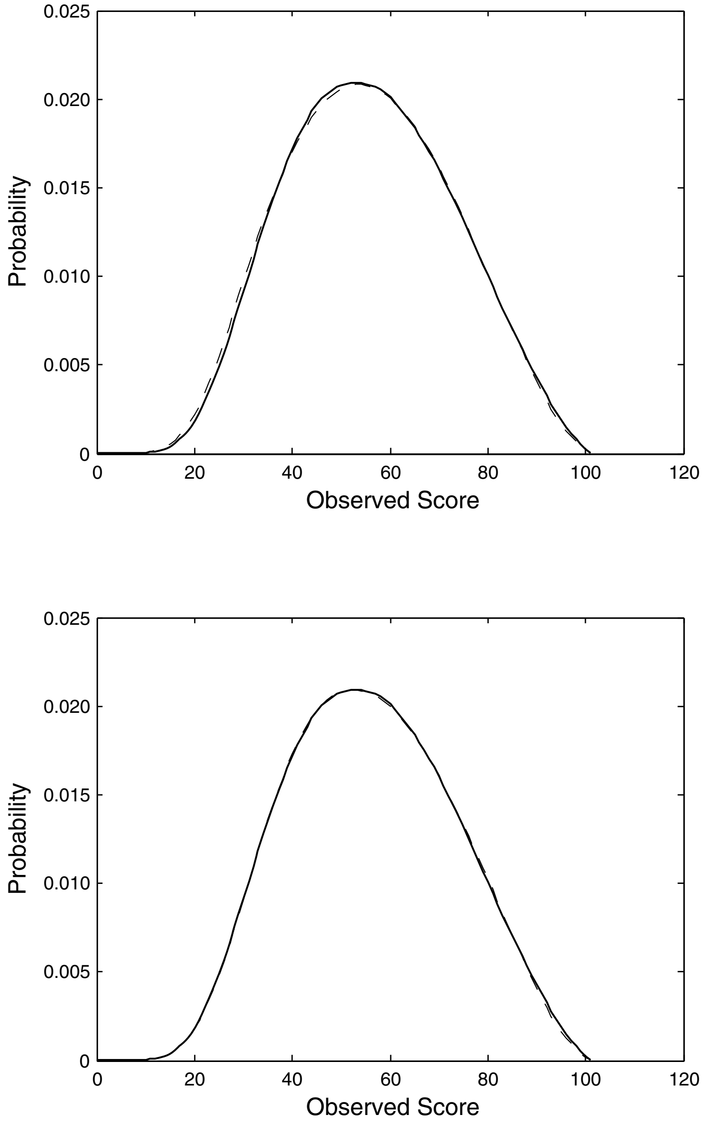

FIGURE 5.3. Observed-score distribution (dashed line) for a test assembled to a target (bold line) at $\theta_k = 0$ (top) and $\theta_k = -1.2, 0, 1.2$ (bottom) ($r = 1$).

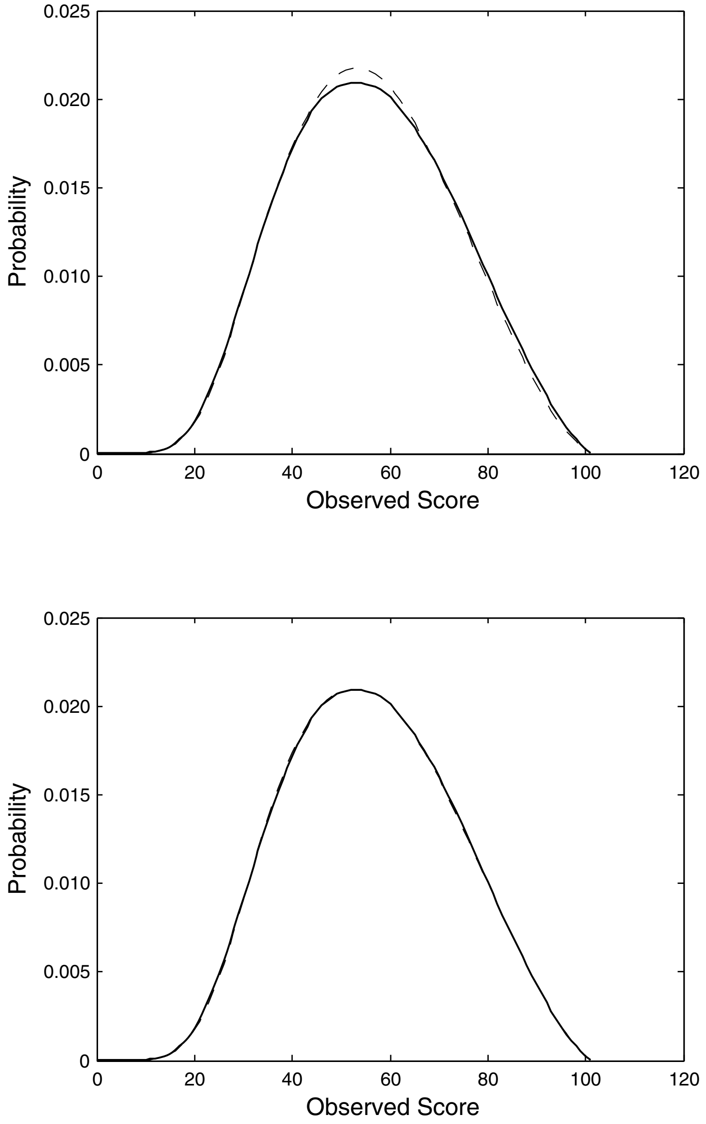

FIGURE 5.4. Observed-score distribution (dashed line) for a test assembled to a target (bold line) at $\theta_k = 0$ (top) and $\theta_k = -1.2$, 0, 1.2 (bottom) $(r = 1, 2)$.

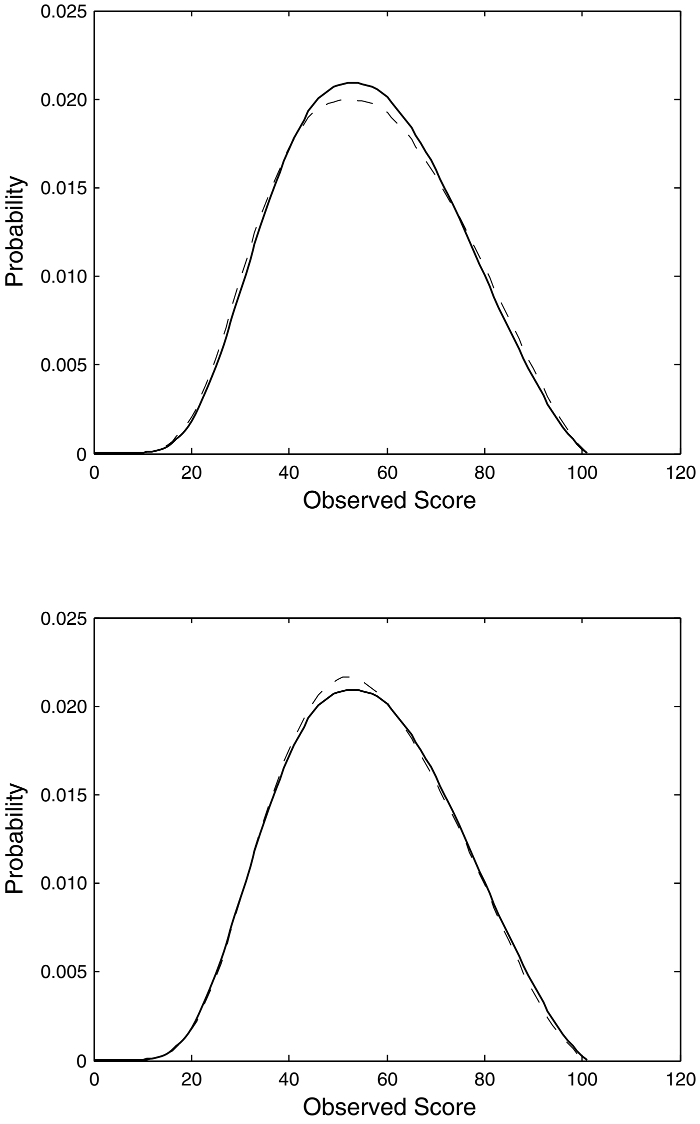

FIGURE 5.5. Observed-score distribution (dashed line) for a test assembled to a target (bold line) at $\theta_k = 0$ (top) and $\theta_k = -1.2$, 0, 1.2 (bottom) $(r = 1, 2, 3)$.

method is a generalization of a graphical method for splitting a test into two halves known as Gulliksen's matched random subsets method.

In item-matching problems, the objective typically is with respect to a few quantitative item attributes, mostly statistical attributes such as the classical item parameters π_i and ρ_{iX}, the IRT parameters a_i , b_i, c_i, and the values of the item-response or item-information functions, $p_i(\theta)$ and $I_i(\theta)$, at a few well-chosen θ values. Categorical item attributes, which are usually needed to deal with the content specifications of the test, can be dealt with by imposing additional categorical constraints on the test. So far, we have used q_i as a generic symbol for a quantitative attribute of item i. Because we will now be dealing with multiple quantitative attributes, we adopt a second index and use q_{jl}, $l = 1, ..., L$, to denote the L attributes considered.

Like most of the earlier problems in this chapter, the problem of matching items on L attributes is another multiobjective problem. In fact, due to the type of decision variables needed to formalize the problem, the number of objectives is much larger than L. We deal with these objectives in two different fashions, which, by now, have become our standard treatment of such problems: by combining them into a single objective function using a weighting procedure or applying the minimax principle.

5.4.1 Matching Items in a Reference Test

Suppose we have a reference test with items to which the new test has to be matched. The items in the reference test are denoted as $j = 1, ..., n$. Their attributes are the targets for a new test assembled from a pool of items, $i = 1, ..., I$. In Section 3.1, we indicated that a fruitful way of identifying the decision variables for a new type of test-assembly problem is to view the selection of the items from the pool as an assignment problem. The current problem is one in which we need to assign n items from the pool to the n items in the reference test such that together they form a set of n pairs with optimally matching attributes. This formulation suggests the use of a separate decision variable for each possible pair of items, (i, j); that is, decision variables

$$x_{ij} = \begin{cases} 1 & \text{if item } i \text{ is matched with item } j \\ 0 & \text{otherwise.} \end{cases} \tag{5.56}$$

The total number of variables is $n \times I$. Item matching thus involves a much larger optimization problem than the ones discussed earlier in this chapter. The choice of these variables also involves a new problem. We now have to keep the values of these variables consistent across pairs; if an item is assigned to a pair, it cannot be assigned to another pair. Finally, the choice of variables makes clear that we have a problem with $n \times L$ objectives; for each of the n pairs, the items have to be matched on L different attributes.

In each of our approaches to the current problem, we combine the item attributes into a measure for the distance between two items. A useful measure is the following weighted version of the Euclidean measure, δ_{ij}, between items i and j:

$$\delta_{ij} = \left[\sum_{l=1}^{L} w_l (q_{il} - q_{jl})^2 \right]^{-1/2}, \tag{5.57}$$

with $w_l > 0$.

If all weights are set equal to $w_l = 1$, the measure is the length of the line between items i and j in a multivariate plot of their attributes. The possibility of choosing different weights w_l for each attribute q_l can be used to allow for possible differences in scale and/or importance between the item attributes.

The first model is

$$\text{minimize} \sum_{j=1}^{n} \sum_{i=1}^{I} \delta_{ij} x_{ij} \tag{5.58}$$

subject to

$$\sum_{i=1}^{I} x_{ij} = 1, \quad \text{for all } j, \tag{5.59}$$

$$\sum_{j=1}^{n} x_{ij} \leq 1, \quad \text{for all } i, \tag{5.60}$$

$$x_{ij} \in \{0, 1\}, \quad \text{for all } i \text{ and } j. \tag{5.61}$$

The objective function in (5.58) minimizes the sum of the distances between the items in the pairs. In principle, we could have chosen a weighted sum. But because all pairs are equally important, no further weighting seems necessary. The constraints in (5.59) and (5.60) are to keep the values of the variables consistent; each of the items in the reference test has exactly one item assigned to it, and each item in the pool can be assigned at most once.

For the objective function in (5.58), results are possible in which an unexpected large term is compensated by a set of smaller terms. The minimax principle deals directly with such cases. Two different applications of the principle are possible. In the first application, we replace (5.58) by

$$\text{minimize } y \tag{5.62}$$

subject to

$$\delta_{ij} x_{ij} \leq y, \quad \text{for all } i \text{ and } j, \tag{5.63}$$

$$\delta_{ij} x_{ij} \geq -y, \quad \text{for all } i \text{ and } j. \tag{5.64}$$

This combination of objective function and constraints minimizes the largest distance between the items over all pairs (Exercise 5.8).

It is also possible to apply the principle at the level of the individual attribute values of the items. We then replace the constraints in (5.63) and (5.64) by

$$(q_{il} - q_{jl})x_{ij} \leq w_l y, \quad \text{for all } i, \ j, \text{ and } l, \tag{5.65}$$

$$(q_{il} - q_{jl})x_{ij} \geq -w_l y, \quad \text{for all } i, \ j, \text{ and } l. \tag{5.66}$$

In this version of the problem, the distance measure in (5.57) is no longer needed. The weights w_l in this measure now figure in the definitions of the intervals $[-w_l y, w_l y]$ about zero, in which the differences between the attribute values $q_{il} - q_{jl}$ for the items in the pairs are enclosed. Just as in the somewhat less stringent preceding version of the problem, the objective function minimizes the common factor y in the size of these intervals.

5.4.2 Test Splitting

The previous problems can be solved for any set of quantitative item attributes. The next problem is typically formulated for the classical item indices π_i and ρ_{iX}. We now have a test consisting of the items $i = 1, ..., n$, and we want to split the test into two halves with an optimal match between their items. Upon correcting for test length, the correlation between the scores on the two test halves, known as the "split-half reliability coefficient," is a lower bound to the reliability coefficient of the test. If the two test halves are chosen to be as parallel as possible, the lower bound approximates the reliability coefficient.

A traditional graphical method for finding an optimal split is *Gulliksen's matched random subsets method*. The method is based on a bivariate scatter plot of the n items with the values π_i and ρ_{iX} as coordinates. An example of a Gulliksen plot is given in Figure 5.6 later in this chapter. Using this plot, pairs of items are formed that minimize the distances between the items in the pairs. The two test halves are formed by randomly assigning to them the two items in each pair.

To formulate this method as an MIP problem, we need the same decision variables as in (5.56) but now with both i and j running over the same items $1, ..., n$ in the test. As the measure of the distance between i and j, we use (5.57) with π_i and ρ_{iX} as attributes. Just as in the Gulliksen method, the problem is solved in two stages.

First-Stage Model

In the first stage, we form pairs of items solving the model

$$\text{minimize} \sum_{i=1}^{n} \sum_{j=1}^{n} \delta_{ij} x_{ij} \tag{5.67}$$

subject to

$$\sum_{j|j\neq i} x_{ij} = 1, \quad \text{for all } i, \tag{5.68}$$

$$x_{ij} = x_{ji}, \quad \text{for all } i \neq j, \tag{5.69}$$

$$\sum_{i,j|i=j} x_{ij} = 0, \tag{5.70}$$

$$x_{ij} \in \{0,1\}, \quad \text{for all } i \text{ and } j \tag{5.71}$$

(Exercise 5.9).

The objective function is defined as the sum of the distances over all possible pairs of items in the test, whereas the constraints in (5.68) require that each item be assigned precisely to one pair. Because i and j run over the same set of items, it holds that pair (i,j) is identical to (j,i) and that pairs can only be formed between items $i \neq j$. These conditions are imposed by the constraints in (5.69) and (5.70). A more parsimonious formulation of this model is given in Section 5.4.3 below.

Second-Stage Model

In the second stage, we have $n/2$ pairs of items, and our task is to assign the items in these pairs to the test halves $h = 1,2$. We use $i_p = 1,2$ to identify the items in pair p. Rather than assigning the items randomly, as in the Gulliksen method, we use this stage for further optimization. To do so, we need the variables

$$x_{i_ph} = \begin{cases} 1 & \text{if item } i_p \text{ is assigned to test half } h \\ 0 & \text{otherwise.} \end{cases} \tag{5.72}$$

The model for classical test assembly in Section 5.2.1, as well as the fact that optimal results were obtained in the empirical example in Section 5.2.4 with the constraint in (5.40) redundant, suggests to assign items to test halves such that the difference between the sums $\sum \sigma_i \rho_{iX}$ for the two halves is minimal. The following minimax model realizes this objective:

$$\text{minimize } y \tag{5.73}$$

subject to

$$\sum_{i=1}^{2}\sum_{p=1}^{n/2} \sigma_{i_p}\rho_{i_pX}(x_{i_p1} - x_{i_p2}) \leq y, \tag{5.74}$$

$$\sum_{i=1}^{2}\sum_{p=1}^{n/2} \sigma_{i_p}\rho_{i_pX}(x_{i_p1} - x_{i_p2}) \geq -y, \tag{5.75}$$

$$\sum_{i=1}^{2} x_{i_ph} = 1, \quad \text{for all } p \text{ and } h, \tag{5.76}$$

$$\sum_{p=1}^{n/2} \sum_{h=1}^{2} x_{i_p h} = 1, \quad \text{for all } i, \tag{5.77}$$

$$x_{i_p h} \in \{0, 1\}, \quad \text{for all } i,\ p, \text{ and } h. \tag{5.78}$$

The constraints in (5.76) enforce the assignment of one item from each pair p to each test half h, whereas those in (5.77) are necessary to guarantee that each item is assigned to a test half.

5.4.3 Discussion

The formulation in (5.67)–(5.71) was chosen for didactic reasons only. A more parsimonious formulation is possible if we use the variables in (5.56) only for $i < j$; that is, in the upper off-diagonal triangle of the matrix of all possible values of (i, j). The previous model can then be replaced by

$$\text{minimize} \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \delta_{ij} x_{ij} \tag{5.79}$$

subject to

$$\sum_{i=1}^{j-1} x_{ij} + \sum_{i=j+1}^{n} x_{ji} = 1, \quad \text{for all } j, \tag{5.80}$$

$$x_{ij} \in \{0, 1\}, \quad \text{for all } i < j. \tag{5.81}$$

The sum in the objective function is now only over the variables in the upper triangle, and the constraints use the same set of variables to force each item to be assigned to exactly one pair. The sets of constraints in (5.69) and (5.70) are thus no longer needed (Exercise 5.10).

Item-matching problems are instructive in that they illustrate several new definitions of decision variables. Also, the problem of splitting a test into halves that are item-by-item parallel shows that some test-assembly problems can only be solved in more than one stage.

It is easy to generalize the problem of test splitting above to the problem of splitting a set of items into three or more parallel parts. This problem arises, for example, when we assemble a set of rotating item pools for use in adaptive testing (Section 11.5.5).

5.4.4 Empirical Example

The method of splitting a test into two parallel halves was applied to a 20-item version of an achievement test from the *IEA Second Mathematics Study*. The values of the items for π_i and ρ_{iX} were estimated from a sample

Item	π_i	ρ_{iX}	Item	π_i	ρ_{iX}
1	.85	.39	11	.83	.52
2	.50	.41	12	.68	.54
3	.60	.40	13	.80	.43
4	.66	.60	14	.84	.45
5	.87	.25	15	.86	.34
6	.28	.37	16	.52	.47
7	.87	.40	17	.62	.58
8	.48	.48	18	.61	.40
9	.74	.47	19	.51	.48
10	.65	.60	20	.66	.58

TABLE 5.2. Estimated values for item difficulty and discrimination index.

Item Pair	$\lvert\pi_i - \pi_j\rvert$	$\lvert\rho_{iX} - \rho_{jX}\rvert$
1,7	.02	.01
2,6	.22	.04
3,18	.01	.00
10,4	.01	.00
5,15	.01	.09
8,19	.03	.00
9,13	.06	.04
14,11	.01	.07
20, 12	.02	.04
16, 17	.10	.11

TABLE 5.3. Optimal item pairs and test halves (first items in the same half).

of 5,418 students in the Dutch part of this study. The estimates are given in Table 5.2; the Gulliksen plot of these estimates is given in Figure 5.6. For some of the items in the plot, it is obvious how to pair them; for others, several alternatives are possible.

Table 5.3 shows the results obtained for the models in stages 1 and 2 above, with all weights w_l in (5.57) set equal to one. The item pairs in Table 5.3 are reported such that the first items in each pair were those assigned to the first test half and the second items to the second half. The table also shows the differences between the π_i and ρ_{iX} values of the items in the pairs. All differences were small except the one for π_i for the pair with items 2 and 6. Figure 5.5 shows that item 6 was an outlier in the distribution of π_i values on the horizontal axis; a solution with a small difference in π_i values was thus impossible.

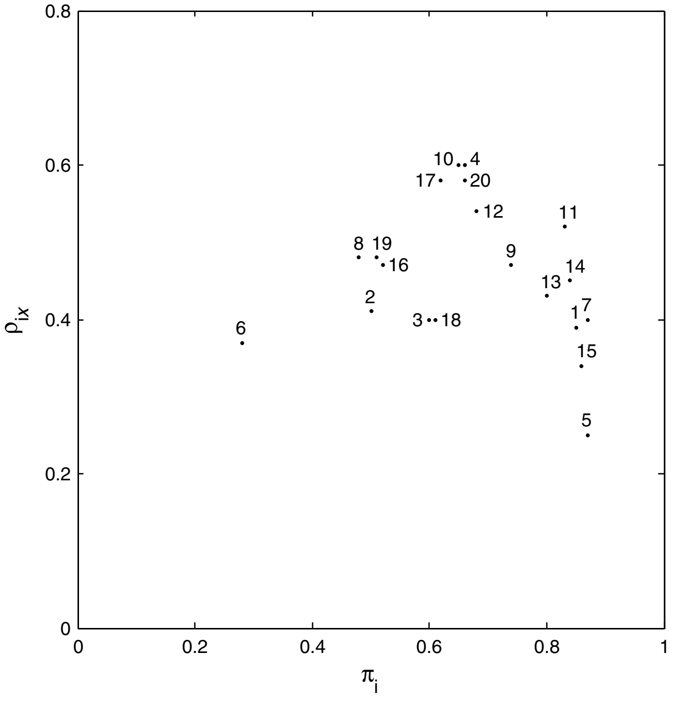

FIGURE 5.6. Gulliksen plot with estimates of π_i and ρ_{iX} for a test from the IEA Second Mathematics Study.

5.5 Literature

Kelderman's method for specifying target information functions was formulated in his 1987 paper. The maximin approaches to test assembly with absolute or relative targets for a TIF were introduced in van der Linden and Boekkooi-Timminga (1989). For a more comprehensive discussion of the differences between absolute and relative targets, see van der Linden and Adema (1998). Using extensive simulation studies, Timminga (1985) showed that controlling the test-information function at a smaller number of θ values generally resulted in faster solutions and that problems with 3–5 values already gave excellent results. Both Baker, Cohen, and Barmish (1988) and de Gruijter (1990) addressed the case of a uniform absolute target for the test-information function. They pointed to the fact that the results for this type of target can be sensitive to the composition of the item pool, particularly to the number of items available near the endpoints of the interval over which the target is specified. Test assembly with target values for the TIF at a cutoff score is discussed in Glas (1988). Reviews of IRT-based test assembly are given in Adema, Boekkooi-Timminga, and Gademan (1992), Adema, Boekkooi-Timminga, and van der Linden (1991), Armstrong, Jones, and Wang (1995), Timminga and Adema (1995), van der Linden (1994b, 1998a, 2001b, 2002), and Veldkamp (2005).

The models for classical test assembly in Sections 5.2.1 and 5.2.2 were formulated in Adema and van der Linden (1989) and van der Linden (submitted), respectively. An alternative approach to the problem of maximizing the reliability of a test was offered in Armstrong, Jones, and Wang (1998), who used network-flow programming with Lagrangian relaxation (Section 4.3) to formulate a search procedure for a solution to the nonlinear objective in (5.31).

The idea of assembling a test to meet a target for its observed-score distribution was formulated in van der Linden and Luecht (1996). These authors tried to achieve this goal by matching both the characteristic function and the information function of the test to a target. Their intuition was that the former would control the true scores and the latter the errors on the test; together they would therefore control its observed-score distribution. The fact that this job can only be done by controlling sums of (lower-order) powers of the response functions was derived in van der Linden and Luecht (1998). The differences between global and local observed-score equating discussed in Section 5.3.3 were further explored in van der Linden (2000c, 2005e). The Lord-Wingersky algorithm for calculating observed-score distributions on tests was presented in their 1984 paper.

Gulliksen introduced his matched random subsets method in his 1950 monograph on test theory. The formalization of the method in Section 5.4.2 and 5.4.3 was given in van der Linden and Boekkooi-Timminga (1988). Armstrong and Jones (1992) suggested extending the model with a set of constraints that enable solution of the model in polynomial time (see Section 4.2.3).

5.6 Summary

1. Tests can be assembled both to an absolute and a relative target for their information function. An absolute target requires the test-information function to be at a fixed height. If a relative target is used, only the shape of the test-information function is specified, but its height is optimized.

2. If the test-assembly model has a constraint that fixes the length of the test, an absolute target can easily lead to infeasibility. For a relative target, this is impossible.

3. Because information functions are smooth, the number of points at which target values for a TIF should be specified need not be larger

than 3–5 well-chosen points. If the objective is to maximize the test information at a cutoff score, the model has only one point.

4. An absolute target for a TIF can be derived from descriptive statistics of the distribution of the item parameters in the pool, a specimen of the test assembled manually, or from (asymptotic) probabilities of test scores ordering the abilities of persons erroneously specified by a panel of test specialists (Kelderman method).

5. A simple way of specifying a relative target for a TIF is by asking test specialists to distribute an arbitrary number of chips over the points θ_k in an item map such that their distribution reflects the relative accuracy of the test scores needed at these points.

6. Test-assembly problems with targets for the test-information functions are problems with multiple objectives, one for each target value. The models presented in this chapter solve them by combining the objectives as a weighted sum or applying the minimax principle.

7. In classical test assembly, we maximize the reliability or the predictive validity of the test. Both objectives involve a nonlinear function consisting of two linear expressions. We can maximize both the reliability and the validity by using one of these expressions as the objective function and constraining the other by a well-chosen upper bound.

8. A test can be assembled to have an observed-score distribution matching the distribution of a reference test for the same population of test takers. The only things required are a few constraints on the sums of the lower-order powers of the response probabilities at a few points θ_k in the test-assembly model.

9. In item-matching problems, a new test is assembled with attributes that match those in a reference test item by item. The same type of problem arises when we want to split a given test into two halves with an optimal match between their item attributes.

10. Item-matching problems lead to the definition of decision variables at the level of pairs of items or combinations of items and test halves.

5.7 Exercises

5.1 The model in (5.7) and (5.8) yields a test with a TIF approaching the target values $\mathcal{T}_k$ from above. Reformulate the model for a test with a TIF approaching the target values from below.

5.2 A run with the test-assembly model in (5.9)–(5.12) for $\mathcal{T}_1 = 10$, $\mathcal{T}_2 = 15$, and $\mathcal{T}_3 = 10$ and the test length fixed at $n = 40$ yields a solution with a value for the objective function equal to $y = .8$. Interpret the result. How reasonable is the result for a pool with an average item-discrimination parameter a_i equal to 1.3?

5.3 A run with the test-assembly model in (5.13)–(5.16) for the same target values as in Exercise 5.2 and $w_k = 1$, $k = 1, 2, 3$, gives the following results for the variables in the objective function: $y_1^{\text{pos}} = .4$ $y_2^{\text{pos}} = 0$, $y_3^{\text{pos}} = .7$, $y_1^{\text{neg}} = 0$, $y_2^{\text{neg}} = .5$, and $y_3^{\text{neg}} = 0$. Calculate the values of the TIF at $k = 1, 2, 3$.

5.4 A run with the test-assembly model in (5.18)–(5.21) yields a solution with a value for the objective function equal to $y = .8$. How do we know if this value represents a positive or a negative deviation from a target value for the TIF?

5.5 The model in (5.25)–(5.27) maximizes the TIF subject to a set of constraints on its shape. Formulate a model for maximizing the height of a TIF that accepts small positive and negative deviations from the intended shape.

5.6 Formulate a model for minimizing the difference between the reliability of a test and a reference test.

5.7 The results in Figure 5.5 show a minor deterioration of the observed-score distribution for the example with the extra constraint for $r = 3$, whereas (5.47) predicts a better approximation if this constraint is added. Explain the result.

5.8 The constraints in (5.63) and (5.64) are quantitative constraints at the item level. Why does (5.64) not need the form with the inverse inequality sign in (3.14)?

5.9 Generalize the model in (5.67)–(5.71) to the case of splitting a test into three parallel parts.

5.10 Show that (5.79)–(5.81) is identical to (5.67)–(5.71).

5.11 A five-item reference test has the following values for the difficulty parameter: $b_1 = -1$, $b_2 = -1$, $b_3 = 0$, $b_4 = .5$, and $b_5 = 1$. Formulate a model for the selection of a new test of five items from a pool of I items with values for the difficulty parameter matching those in the reference test as closely as possible. Why does the model not need a constraint on the length of the test?

http://www.springer.com/978-0-387-20272-3

Linear Models for Optimal Test Design
van der Linden, W.J.
2005, XXIV, 408 p. 44 illus., Hardcover
ISBN: 978-0-387-20272-3