Sensitivity analysis and Input Validation

1. Sensitivity analysis approach
   a. In general, we study how changes in FURS inputs affect FURS outputs. We deploy a one-at-a-time approach, incrementally affecting a single input variable.
      i. Certainly there is something to be said for using copula-simulated data and in general covariance approaches could be more elegant. However testing FURS must address the period-over-period idiosyncrasies of category constituents and correlation to broader market behavior. These are not easily generalized with simulation. There is a place for simulated covarying data in FURS research however for this use case we opt for historical data and the one-at-a-time approach.
   b. Input variables
      i. Input variables in FURS nomenclature are "subcomponents." The selection and weights of each subcomponent is prescribed by one of 24 FURS sub-models[1]. For any given FURS rating there exists a reference table that parametrizes categories to a sub-model and a sub-model to a set of weights for a set of subcomponents. A reference table example is embedded in Appendix X.
      ii. Some FURS subcomponents undergo preprocessing before being used in a dot product calculation of subcomponent percentile rank and subcomponent weights. These processes are thoroughly detailed in sections XXX and XXX of the FURS whitepaper. Table 1 below shows the flow of data from its intake state to its ultimate state for use in FURS. In the "Preprocessing" column we informally label rows indirect or direct, the formulations of which are shown in Equations 1 and 2, respectively.

$$subcomponenet\ value = f(intake\ state) \quad [1]$$

$$subcomponenet\ value = intake\ state \quad [2]$$

Intake State	Preprocessing	Subcomponent Values	Ultimate State
[redacted]	[redacted]	[redacted]	[redacted]
[redacted]	[redacted]	[redacted]	
[redacted]	[redacted]	[redacted]	
[redacted]	[redacted]	[redacted]	
[redacted]	[redacted]	[redacted]	
[redacted]	[redacted]	[redacted]	

[1] Note on forthcoming cohorts … also how the weights are determined (by calibration)

[redacted]	[redacted]	[redacted]	
[redacted]	[redacted]	[redacted]	
[redacted]	[redacted]	[redacted]	
[redacted]	[redacted]	[redacted]	

Table 1

iii. Additionally, we consider the statistical distribution of all values in the "Intake State" and "Subcomponent Value" columns in Table 1. As an example, Figure 1 and Table 2 below show charts and summary statistics for 30 periods of the intake state of [redacted]. The PDF and CDF charts show all categories over all periods. The Bar plots shows all period broken out by category, where the markers designate boxplot stats. Analysis of all distributions is embedded in Appendix X.

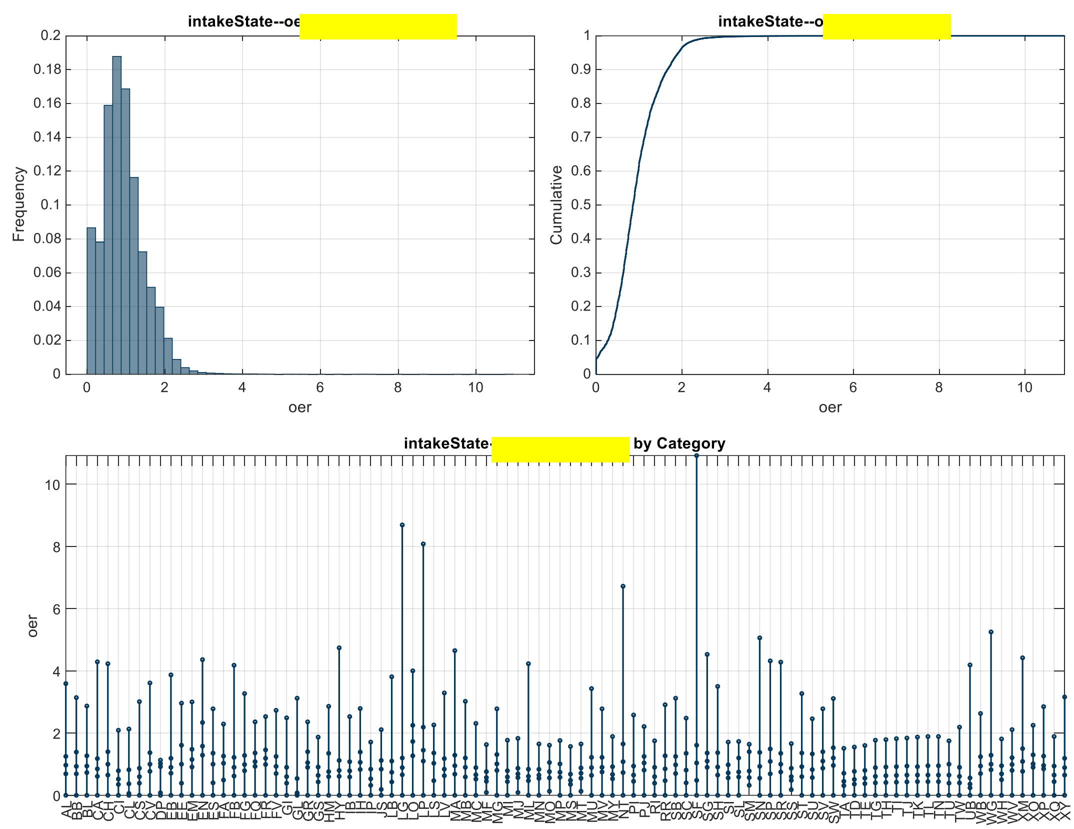

Figure 1

intakeState--	
Mean	0.92
Median	0.86
Min	0.00
Max	10.92
Range	10.92
Std.Dev.	0.55
Count (all cats, all periods)	644,410

Table 2

c. Perturbing Input Data

i. We perturb the values listed in the “Subcomponent Values” column in Table 1. This data comes from a historical values file. It is first partitioned by category. Then category subcomponent values are expressed at the fund level per period. We incrementally perturb one subcomponent at a time. The procedure is as follows:

1. Create a “scope” vector of scalars e.g. [0,3] in increments of 0.15. These are the incremental scalars. Create a “breadth” vector [0,1] in increments of 0.05. These are the incremental percentage of rows we will perturb per subcomponent column. We now have a set of scope/breadth permutations. In this case the permutation set is 21 times 21 equals 441 scope/breath pairs.
2. For each scope/breadth pair, make a raw value vector by randomly selecting a number of rows that equals $breadth_i$ times the length of the subcomponent (rounded to an integer as needed).
3. Multiply this raw value vector by $scope_j$ (scalar).
4. Multiply each element of this vector by a N(0, 1) Gaussian random variable.
5. Add this vector to the raw data values. This step is to ensure the zero entries in the scope/breadth vectors has no effect and retains raw values.
6. As indexed by the random selection process, replace raw values with perturbed values in the subcomponent column.
7. Do this for all nonzero subcomponents in a model, for each category in a model, for all scope/breadth pairs. Figure 2 below shows two examples of perturbed data. As determined by the scope/breadth index (i and j in the chart titles), the left is hardly affected, the right is dramatically affected.

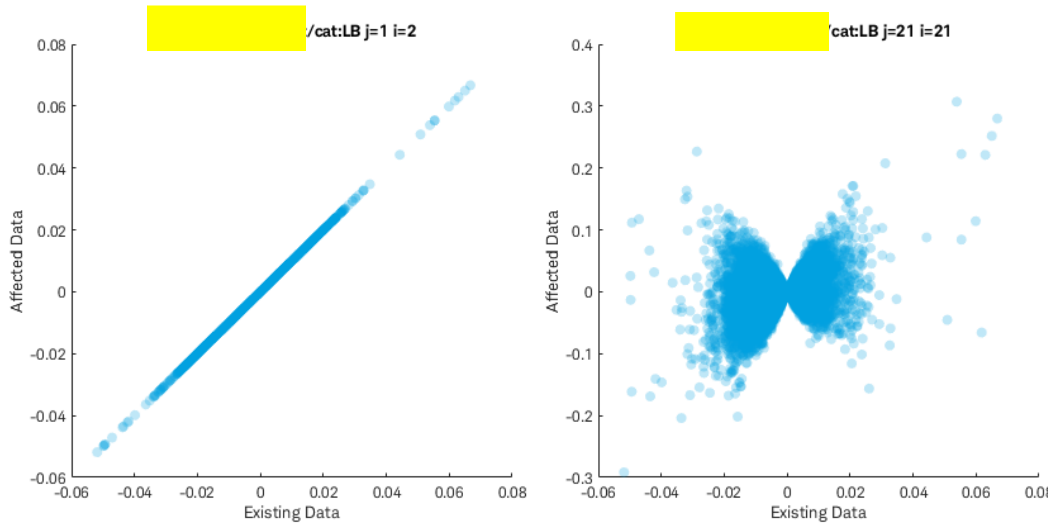

Figure 2

d. The use of outlier scores
    i. We need a single metric to describe how each scope/breath pair has perturbed the subcomponent values. The outlier detection and machine learning literature provide a battery of modeling choices that suit this purpose. Future versions of this research (and how its applied during production) may include these approaches. For now we opt for simplicity and ease of communication across groups and simply select z-score, which measures how many standard deviations a data point is from the mean in a distribution.
    ii. Using the procedure described above, for each nonzero weighted subcomponent in a sub-model, for each category in a sub-model, we now have length(breadth) times length(scope) samples of incrementally perturbed subcomponent values. For each sample, the outlier score is the average of the top quintile of z-scores in the affected data array, computed using the moments of the complete unperturbed raw values column.

b. Obtaining ranks, scores, and grades
    i. For each sample of perturbed data, we proceed in the typical FURS fashion: [redacted] This process includes samples that are unperturbed due to the zero entries ion the scope and breadth vectors. These samples function as a baseline when measuring outputs.

e. Measuring outputs
    i. Grade migration rate
        1. Using the grades derived from unperturbed samples we compute the grade migration rate for each scope/breadth pair i.e. the count of funds that changed grades divided by the count of all graded funds. [redacted]

[redacted]

ii. Change in IC

1. The historical data file also contains 36-month future returns so we can compute IC for each sample. Using the IC derived from unperturbed data we compute the change in IC rate for each scope/breadth pair.

f. Surface plots

i. Figure X below shows a single subcomponent for all categories in a single sub-model. Charts for all models, subcomponents, and categories are included in appendix X.

ii. Figure X shows how we can simultaneously tie scope/breadth pairs to outlier scores, grade migration rate, and change in IC. The scope/breath pairs make up the x- and y-axis of all three plots. Which means all three z-axes are related. This has multiple implications for input and output validation process used during ratings, which we delve into next.

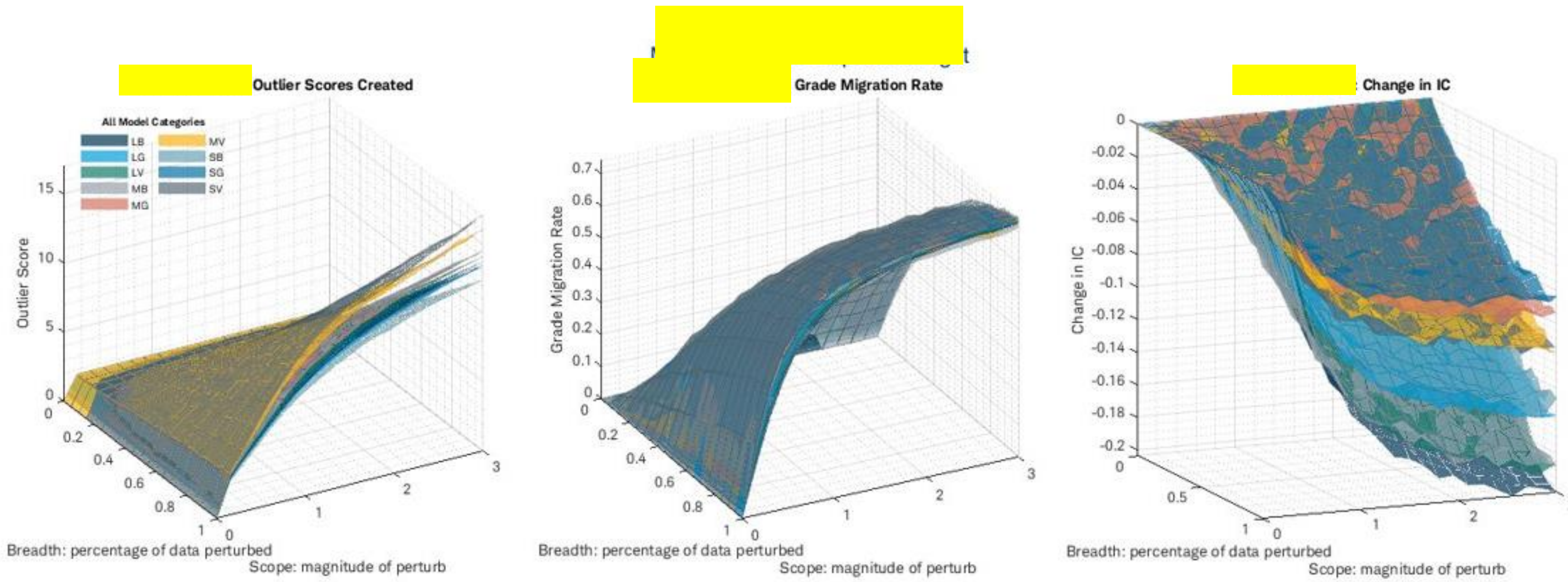
Outlier Scores Created
All Model Categories
LB
LG
LV
MB
MG
MV
SB
SG
SV
Outlier Score
Breadth: percentage of data perturbed
Scope: magnitude of perturb
Grade Migration Rate
Grade Migration Rate
Breadth: percentage of data perturbed
Scope: magnitude of perturb
Change in IC
Change in IC
Breadth: percentage of data perturbed
Scope: magnitude of perturb

Figure 3

2. Tying sensitivity analysis to FURS QA processes
   a. Output validation
      i. The output validation report requires approval before making ratings official. The report uses a transition matrix to assess period-over-period output. Here we review percentages of no grade change, no rank change, small rank change, and large rank change across various lookbacks and fund groupings. Grade migration rate is approximately the complement of no grade change. Sensitivity analysis allows us to assign meaning to grade migration rate: the extent to which it could affect future performance. Revising FURS output validation, however, is not in QMR's near-term queue. We expect to pick this up sometime mid-2024.
   b. Input validation
      i. The input validation report requires approval before proceeding to ratings. The report throws data quality flags on large period-over-period changes. The existing report reviews both percentage change and absolute differences across multiple data sources (plus missing data and category changes). The thresholds for these flags are not currently based on research. Sensitivity analysis can help us assign meaning to these thresholds.
      ii. Inspecting the surface plots shown above in Figure X, we can locate some level of unacceptable IC loss, let us call it 0.03 of IC loss. We can find the x-y coordinates of at least 0.03 IC loss on the furthest right plot. We can take those x-y coordinates to the chart on the furthest left and determine the outliers scores present when 0.03 of IC loas begin to occur. Additionally, we can assess the quantity of those scores with the breadth coordinate. These two figures (the outlier score and the breadth coordinate) are thresholds we can take to input validation.
      iii. Referring again to Table 1, there are nine unique entries in the "Intake State" column. In addition to score and breadth thresholds, we can assign a relative sensitivity descriptor to the nine data sources. Grade migration rates or IC are more sensitive to model subcomponents than others. That is, it takes smaller and fewer input outliers to generate materially distorted output. (Spoiler alert: it's the weights.) A relative sensitivity descriptor like high, medium, or low could help the analyst/approver better discern data quality flags and decide more quickly to investigate or move on.
      iv. Figure 4 below shows how these thresholds and relative sensitivity rating could be determined. Here again, for every model, category, and subcomponent, we find the x-y coordinate of unacceptable IC loss of 0.03 then collect the minimum score and breath thresholds where that loss occurs.

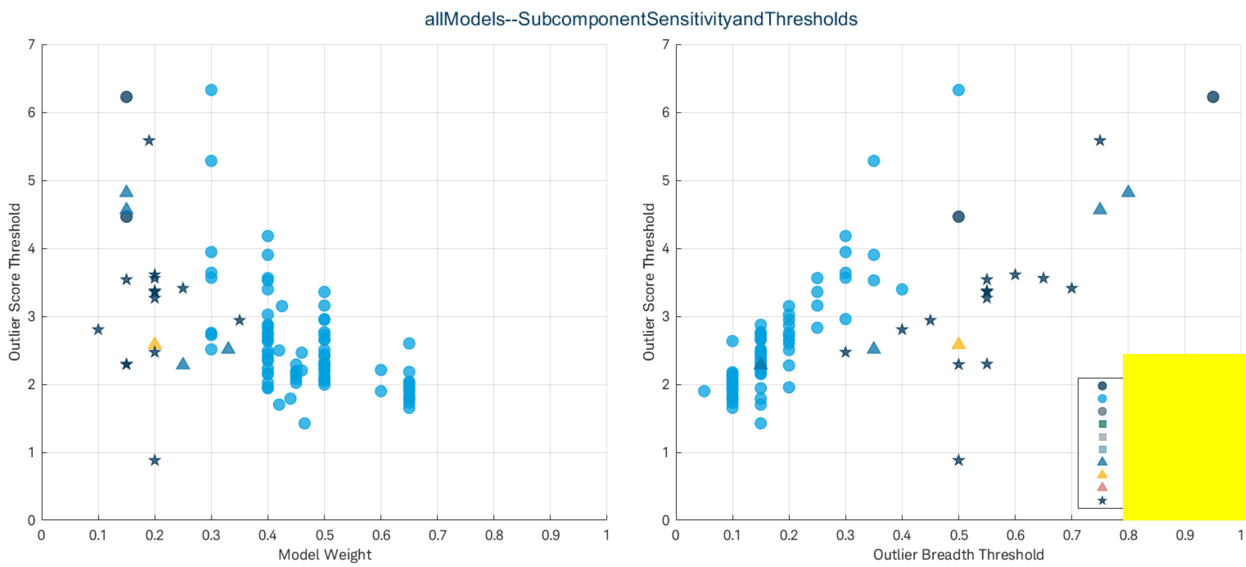

Figure 4

v. Evaluating Figure 4

1. The markers in both scatters represent categories. Subcomponents are designated by color and shape as shown in the legend.
2. In the left scatter, we show subcomponent weight versus its outlier score threshold. In general we see a down-and-to-the-right clustering. This is quite intuitive as of course it would take higher outlier scores in lower-weighted subcomponents to cause unacceptable IC loss. Same for lower outlier scores and higher-weighted subcomponents.
3. In the right scatter, we show breadth threshold versus its outlier score threshold. In general we see an up-and-to-the-right clustering. Knowing the weights from the left scatter, this too is intuitive as it suggests is takes less of higher-weighted subcomponent score breaches to cause unacceptable IC loss. Same for more of lower-weighted subcomponent score breaches.
4. In both scatters, each marker represents the same subcomponent and category (n=111). This is much less than the total amount of categories and subcomponents (n=576). This is because all other categories and subcomponents did not cause 0.03 of IC loss. Some threshold would be taken to input validation for the categories and subcomponents that did not cause 0.03 of IC loss, but we can see that in general it takes a lot of bad data to cause model-wide unacceptable IC loss.

3. Input validation examples

a. Reference table

i. Table X below shows an example reference table that could be collected for every category's nonzero component weight then used in input validation. Here we show [redacted]

[redacted] We show score and breadth thresholds and relative sensitivity rating. Further research may show we can simply average (or minimum, median, etc.) thresholds to use one figure per model because intra-model category thresholds tend to be similar.

b. Data source analysis
    i. Thresholds
        1. Figures X and X below show how we use thresholds to throw data quality flags. Here we show two kinds of input data (fields from the "Intake State" column in Table 1) in terms of outlier score. We outlier scores in two dimensions because a fund outlier score (time series) is different from a category outlier score (cross sectional). We show outlier scores across time and in a scatter. In the scatter, existing data is the gray markers and new data is colored markers.
    ii. Data quality signals
        1. In Figure X we planted a single entry of made-up data: one fund is randomly selected and we mark its most recent return as 100%. If there exists an outlier score threshold breach only, the data quality is signaled yellow. We see the yellow rating in the yellow markers in Figure X.
        2. In Figure X we randomly select half the entries for [redacted] and multiply them by some implausible scaler, here we use 25. Like Figure X, we are just trying to pollute the real [redacted] value column with simulated bad data. The bad data's subsequent outlier scores breach both the score threshold and the breadth threshold within their respective categories. If there exists both an outlier score threshold breach and outlier breadth threshold breach, the data quality is signaled red. We see the red signal in the red markers in Figure X.
        3. If there exists neither an outlier score threshold breach nor an outlier breadth threshold breach, the data quality is signaled green. We would see the green signal in green markers.
        4. Note that all breached threshold tie to a fund. When there is a breach, offending funds are printed in a table and the analyst/approver can review and investigate the nature of the breach.
    iii. Data dashboard
        1. Figure X below shows how the nine unique entries in the "Intake State" column could feed into a data quality dashboard. Appearing at the top of the input validation report, the analyst/approver can get a broad sense of the period's data quality. The columns show the nine data sources. The rows show each sub-model. The circles show data status: green for neither score nor breadth breach, yellow for score breach only, red for score and breadth breach. If breached we also show a high, medium, or low relativity sensitivity descriptor so the analyst/approver can quickly get sense of the breach's potential impact on ratings.
        2. Note it takes only a single category in breach to turn markers from green to yellow or red in the scatters. Likewise, it takes only a single

category in breach to turn a model's circle yellow or red in the data dashboard.

iv. A hypothetical draft of the revised input validation report is embedded in Appendix X.

[redacted]

Table 3

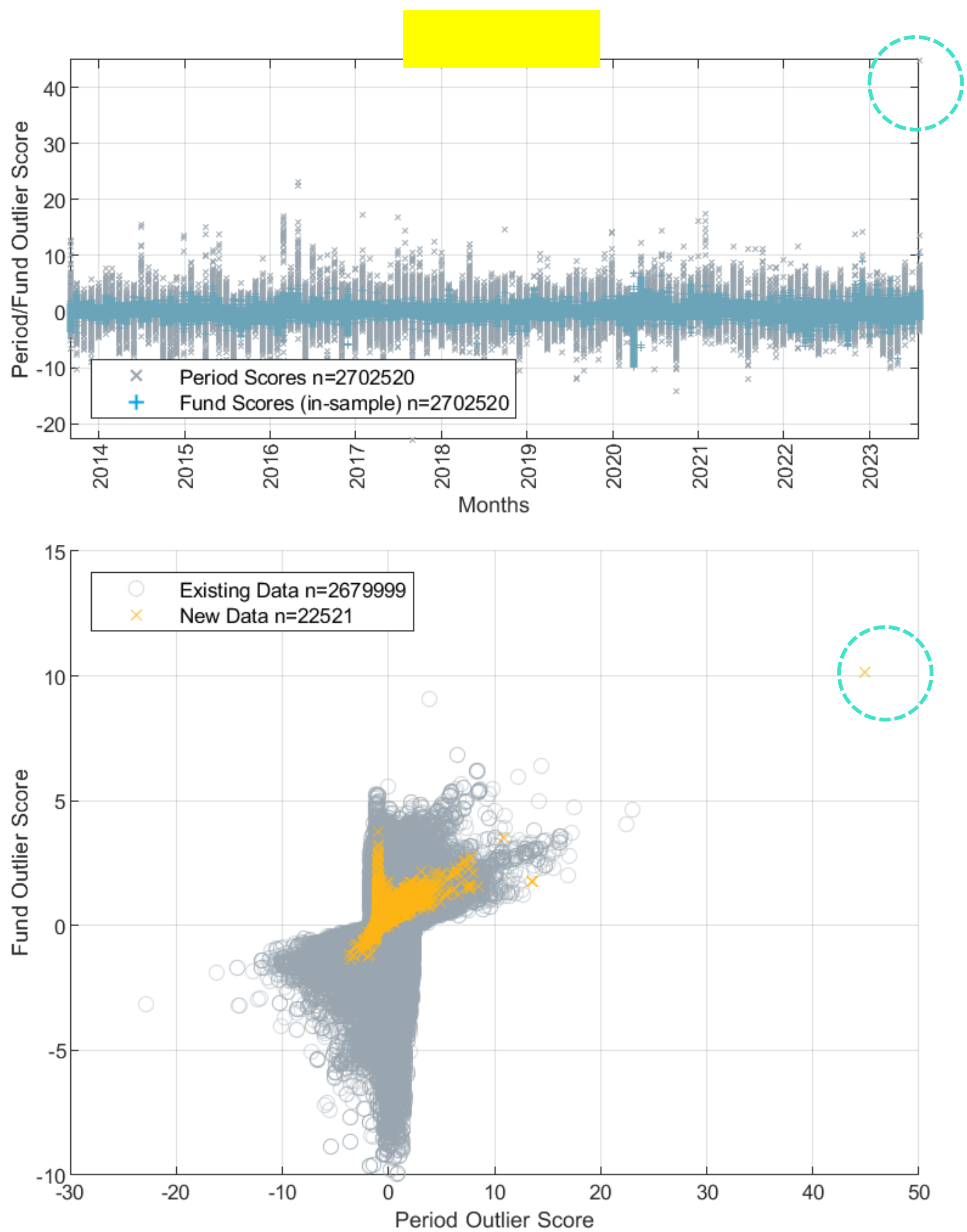
Period/Fund Outlier Score
Months
Period Scores n=2702520
Fund Scores (in-sample) n=2702520
Fund Outlier Score
Period Outlier Score
Existing Data n=2679999
New Data n=22521

Figure 5

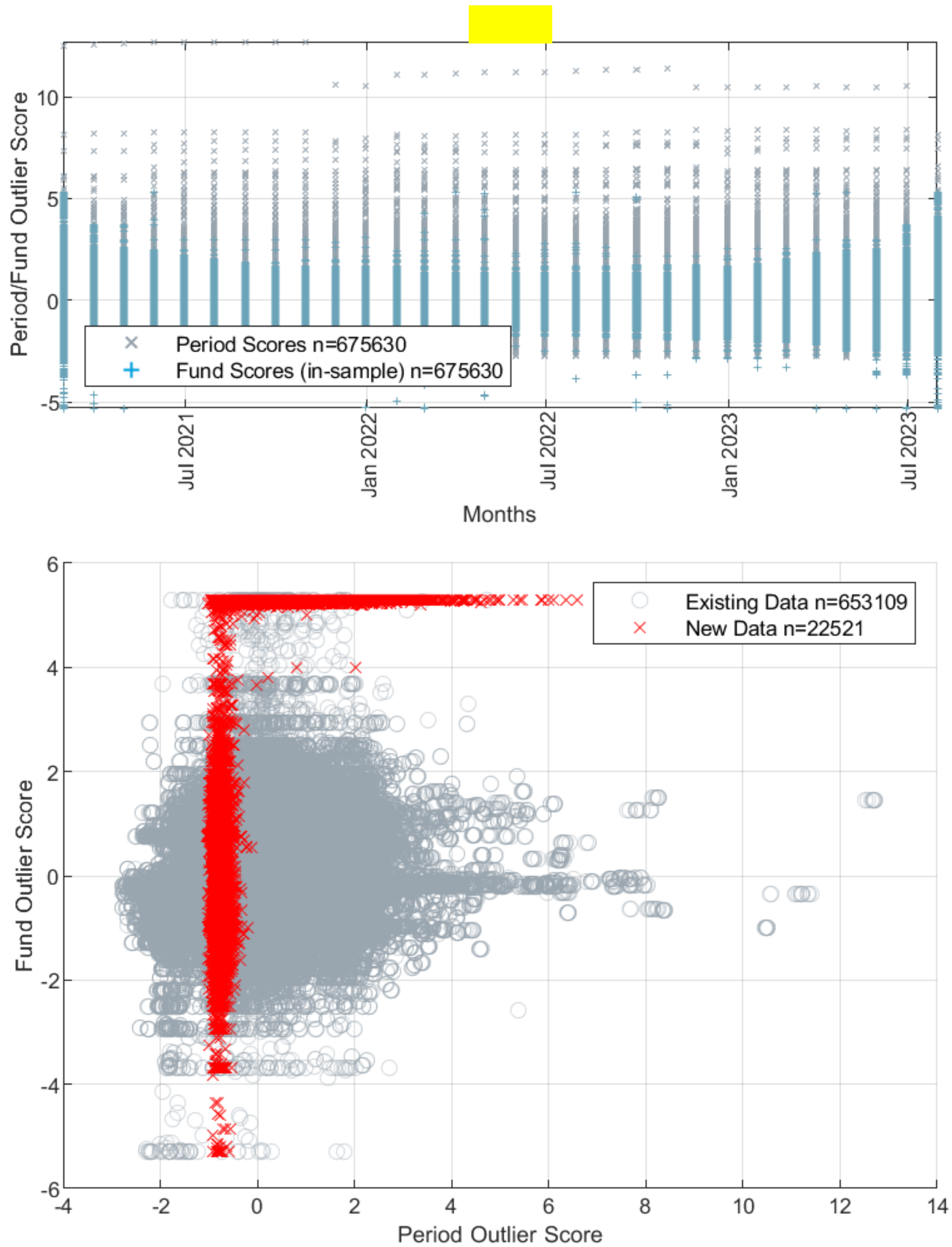
Period/Fund Outlier Score
Period Scores n=675630
Fund Scores (in-sample) n=675630
Jul 2021
Jan 2022
Jul 2022
Jan 2023
Jul 2023
Months
Fund Outlier Score
Existing Data n=653109
New Data n=22521
Period Outlier Score

Figure 6


Input Validation Data Quality Dashboard
HIGH/MEDIUM/LOW: Sensitivity Rating
Green: No outlier score breach
Yellow: Score breach only
Red: Score breach and breadth breach
HIGH
HIGH
HIGH
MEDIUM
HIGH
MEDIUM

Figure 7