Covariance Realism Assessment Software:
Abbreviated Documentation and User's Guide

Overview

The Covariance Realism Assessment software package examines residual sets to determine whether their associated covariances reasonably represent the residuals' expected distribution. The typical application is to examine, for a particular spacecraft, sets of predicted ephemerides that provide both predicted states and predicted position covariances, in the presence of a definitive (as-flown) ephemeris that will allow residuals between the predicted and definitive ephemerides to be calculated. The accompanying technical memo *Covariance Realism Evaluation Approaches.pdf* gives both abbreviated theoretical treatments and practical information regarding how best to proceed with position covariance realism assessments. The present document repeats some of this information but is more focused on practical issues regarding the use of the software package.

It is an overworn expression to state that a particular endeavor is more an art than a science. At the same time, it is certainly true that there are no hard-and-fast rules or test result sets that indicate whether the covariances contained in a particular predicted ephemeris set should be considered realistic. In the accompanying *Covariance Realism Test Results.pptx* file, each (synthetic) test case represents a clear situation; but to evaluate even these requires some subtlety of examination of results; and data for real ephemerides are often even more ambiguous. The accompanying conference paper *Zaidi and Hejduk 2016.pdf* outlines how a covariance realism evaluation and compensation investigation was performed for a particular satellite constellation set; perhaps this can give some implementation ideas. NASA missions who wish assistance in the interpretation of position covariance realism results should contact NASA CARA.

Software Structure

The software suite is written in MATLAB and tested with version 2019b. As written, the software requires the Statistics and Machine Learning toolbox and is structured to use the Parallel Computing toolbox, although the latter is not strictly necessary and can be eliminated as a consideration by a small change to one module. The included modules are the following:

Module Name	Description
AnalysisDriver.m	A basic driver that loads a test input set and runs the entire covariance realism software string. A good module to examine to understand how the entire software flows.
EvaluateResiduals.m	Checks normalized residuals for conformity to a Gaussian distribution (by component) and Mahalanobis distances (technically the square of Mahalanobis distances) for conformity to a 3-DoF chi-square distribution.

FullSpecDistVec.m	Evaluates a fully-specified empirical distribution's conformity to a hypothesized parent distribution; called by multiple routines
DetermineOptimal ScaleFactor.m	Performs distribution testing iteratively to determine the optimal covariance scale factor to achieve desired distribution matching. This module contains a *parfor* statement that can be changed to a *for* statement if the user lacks the Parallel Computing Toolbox; with this change, the routine will require a longer run-time but will function correctly.
Labelpoints.m	Service routine to produce certain features on output graphs.
CreateTestInput0X.m	Creates one of the test input files. These files are also included, but providing this code allows the user to understand exactly how the test datasets are produced.

Input Data

The generation of input data for this set of routines is treated in more depth in *Covariance Realism Evaluation Approaches.pdf*, but a few brief statements are appropriate here. The *Residuals* input structure is an n x 3 array (n here is the number of ordered triples of residuals) for which the first column is presumed to be a radial residual, the second column an in-track residual, and the third column a cross-track residual. These residuals are presumably calculated by differencing a predicted state to the definitive state for that same time. The *Covariances* input structure is a 3 x 3 x n array for which the first two dimensions is a 3 x 3 position covariance in radial/in-track/cross-track (RIC) space. While RIC is presumed, any orthogonal system is acceptable, although the graph labeling will require modification. Also, any convenient units of length are acceptable; the residuals are presumed to be in length space and the covariance in length-squared space.

The *AnalysisDriver.m* module provides a commented example of setting certain program operating parameters, managing input data, and executing the program. Rather than attempt to explain those items here, it is easier simply to inspect this code.

First-pass Outputs: Component and Composite Tests

The first pass of the evaluation examines the normalized position residuals without any scaling factor applied; this gives a sense of the realism of the covariances as produced, without any artificial alteration. The first step is to normalize the residuals by dividing each individual residual by the standard deviation of the appropriate component from the accompanying covariance. This creates a set of "z-variables" that should have a mean of 0 and a standard deviation of 1. It is presumed that the residuals are unbiased, meaning that the mean of each component's residuals is 0; otherwise, a covariance matrix cannot properly represent the errors. The second step is to produce a series of random samples, without replacement, from the set of residuals (separated by component) in order to produce i sets of j samples, one i x j set for each component. Each of i sets is then tested for normality, using both the more permissive Cramér –

von Mises test and the more demanding Anderson-Darling test. Then the Mahalanobis Distance is computed for each residual group (that is, for each ordered triple of RIC residuals and its associated covariance) is computed. Following the same resampling procedure as described above, one *i* x *j* set of Mahalanobis Distance samples is tested for conformity to a 3-DoF chi-squared distribution. The output plots for the first pass are as follows:

- **Figure 11: Estimated PDFs of Resampled Normalized Residuals**. This chart gives estimated PDFs (from kernel density estimation) of the means of all of the resampled sets, by component. This allows a quick visual confirmation whether the residuals appear to be unbiased. One should not expect the peak of this distribution to be exactly at zero, especially because the PDF estimation is not fully precise; but it does allow one to tell whether there is an appreciable bias.
- **Figure 12: Portion of Samples Passing Normality Test**. This chart gives a CDF by p-value of the results of both the Cramér – von Mises and Anderson-Darling tests. To read these results, determine the p-value with which one is comfortable, and the cumulative percentage value that the graph associates with that p-value is the percentage of cases that achieved that value or a higher value. One should not expect compliance at the 100% level; but below 90% one starts to get increasingly uncomfortable.
- **Figure 13: Portion of Samples Passing 3-DoF Chi-Squared Test**. As in Figure 12 above, a CDF of cumulative percentage vs p-value is given; and the same interpretation guidance applies.

  For the figures above, an empty graph indicates that none of the samples passed the assigned tests. In some of the illustrative test cases, this does happen; in actual practice it is rare to perform truly that badly.

Second-pass Outputs: Scale Factor Profiling and Repeat of First-Pass Tests

In the second pass through the data, an analysis is performed to explore whether scaling the covariance by a single multiplicative factor would improve the situation appreciably. Since the covariance production process can undersize or oversize covariances (with the former being far more common), it is often helpful to know which of these is taking place and the degree to which it is observed. It is not recommended that a scale factor simply be applied operationally as a solution to any apparent covariance mis-sizing; instead, the root cause of the problem should be identified and repaired accordingly. However, the size of the scale factor, and whether it is greater or less than unity, are both helpful data in assessing the type and degree of any mis-sizing problem; and re-running analyses after attempted repairs and examining the scale factor’s size in the post-repair state can help to assess whether the repairs have been successful.

The same resampling technique used in the first-pass analysis is employed here, and a scale factor to be applied to all of the covariances (within each sample group) is calculated so that the selected test statistic (Cramér – von Mises or Anderson-Darling) is minimized.[1] What is produced, therefore, is a set of scale factors, one for each of the sample groups analyzed. This information is used to produce the following output plots:

- **Figure 21: Estimated PDF of Scale Factors**. A KDE of the set of scale factors produced by analyzing each sample is given, along with a point that represents the scale factor obtained by analyzing the entire dataset, without resampling. One can eyeball the peak of the distribution to see what the most common scale factor is and how well that aligns with the full-sample calculation.
- **Figure 31: Portion of Samples Passing Normality Test**. The normality tests of the first pass on the component datasets are re-run, with the scale factor derived from the analysis of the full dataset (plotted as a magenta point in Figure 21) applied to all of the covariances.[2] One can view here how much improvement, if any, is wrought by the application of the scale factor.
- Figure 13: Portion of Samples Passing 3-DoF Chi-Squared Test. The 3-DoF chi-squared test from the first pass on the datasets is re-run with the scale factor for the full dataset applied to all of the covariances. One can view here how much improvement, if any, is wrought by the application of the scale factor.

Test Datasets and Results

The distribution contains a number of synthetic datasets and results. Eventually this test collection may include actual satellite data, but releasibility issues have constrained the test sets for the present. Note that the file extensions in the furnished package are .mmm rather than .mat, in order to keep e-mail filters from stripping them out; so the user will need to change each of the file extensions back to .mat or simply run the MATLAB routines that generate each of these test datasets. The accompanying PowerPoint file shows graphical results obtained when running each of the test datasets through the attached software. The test datasets have the following definitions:

File Name	Contents
TestInput01.mat	Normal distribution, $\mu=0$, $\sigma=1$; covariances reflect this
TestInput02.mat	Normal distribution, $\mu=0$, $\sigma=1$; covariances based on normal distribution with $\mu=0$, $\sigma=4$ (should generate scale factor of ~0.25
TestInput03.mat	Normal distribution, $\mu=0$, $\sigma=1$; covariances based on normal distribution with $\mu=0$, $\sigma=0.333$ (should generate scale factor of ~3

[1] One should be cautioned against choosing the Anderson-Darling factor for minimization; this will work adequately with well-behaved datasets, but the construction of the test is such that it can resist convergence by the iterative solver used by the present software.

[2] It should be noted that application of the scale factor requires pre- and post-multiplying the covariance by the scale factor (analogous to propagating a covariance using a state transition matrix); since the scale factor is a scalar, an equivalent result is obtained by multiplying each covariance by the square of the scale factor.

TestInput04.mat	Student's t-distribution, ν=3; covariances based on sample covariances of this same exponential distribution (should produce failure on the first pass and marginal results on the second pass)
TestInput05.mat	Exponential distribution, μ=1; covariances based on sample covariances of this same exponential distribution (should produce failure on the first pass and marginal results on the second pass)

The graphical results themselves, along with some amplifying notes, are supplied in the accompanying *Covariance Realism Test Results.pptx* PowerPoint file. Because resampling is used in the evaluation software, re-executions of the test cases will not identically match the results given in this PowerPoint file.