Risk Management Toolbox™ Release Notes

MATLAB®

MathWorks®

How to Contact MathWorks

Latest news: www.mathworks.com

Sales and services: www.mathworks.com/sales_and_services

User community: www.mathworks.com/matlabcentral

Technical support: www.mathworks.com/support/contact_us

Phone: 508-647-7000

The MathWorks, Inc.
1 Apple Hill Drive
Natick, MA 01760-2098

Risk Management Toolbox™ Release Notes

Contents

R2022b

R2022a

R2021b

R2021a

R2020b

R2020a

R2019b

R2019a

R2018b

R2018a

R2017b

R2017a

R2016b

R2022b

Version: 2.1

New Features

Bug Fixes

Credit Models: Custom PD models

Support for custom probability of default (PD) models using `customLifetimePDModel`. After creating a `customLifetimePDModel` object, you can use the following PD functions:

- `predict`
- `predictLifetime`
- `modelDiscrimination`
- `modelDiscriminationPlot`
- `modelAccuracy`
- `modelAccuracyPlot`

For examples using `customLifetimePDModel`, see “Create Custom Lifetime PD Model for Credit Scorecard Model with Function Handle” and “Create Custom Lifetime PD Model for Decision Tree Model with Function Handle”.

Examples: Climate change and credit risk

The “Measure Transition Risk for Loan Portfolios with Respect to Climate Scenarios” example demonstrates how different climate scenarios change the market shares of different energy sectors in different geographies.

The “Assess Physical and Transition Risk for Mortgages” example shows an approach to assess physical and transition risks for mortgages. Physical and transition risks are the two main categories of climate change risks.

Credit Scorecard Examples: Fairness mitigation

The “Bias Mitigation in Credit Scoring by Reweighting” example shows how to remove bias from a credit scorecard model to make it fair by using the bias mitigation process.

The “Bias Mitigation in Credit Scoring by Disparate Impact Removal” example shows how to reduce bias in a credit scorecard model by using disparate impact removal as a preprocessing technique.

Lifetime Credit Risk Analysis: Beta regression for LGD and EAD models

You can use `fitLGDModel` to create a `Beta` model. After creating a `Beta` model object, you can use the following loss given default (LGD) functions:

- `predict`
- `modelDiscrimination`
- `modelDiscriminationPlot`
- `modelAccuracy`
- `modelAccuracyPlot`

You can use `fitEADModel` to create a `Beta` model. After creating a `Beta` model object, you can use the following exposure at default (EAD) functions:

- `predict`

- `modelDiscrimination`
- `modelDiscriminationPlot`
- `modelAccuracy`
- `modelAccuracyPlot`

R2022a

Lifetime Credit Risk Analysis: Calculate lifetime expected credit loss

The lifetime expected credit loss (ECL) calculator supports `portfolioECL` to compute the ECL value at the individual or portfolio level.

ECL Example: Calculate lifetime ECL using Econometrics Toolbox

The Economic Scenarios and Expected Credit Loss Calculations example shows how to generate macroeconomic scenarios and perform expected credit loss (ECL) calculations for a portfolio of loans.

Credit Scorecard Example: Explore fairness metrics for bias

The Explore Fairness Metrics for Credit Scoring Model example demonstrates how to calculate fairness metrics for the data and model levels. The example also explores the metrics for the presence of bias.

Lifetime Credit Risk Analysis: Updates to credit models

The following updates:

- `Regression` and `Tobit` loss given default (LGD) models support a `modelDiscriminationPlot` reference LGD outside of `[0,1]` range.
- `modelDiscrimination` for probability of default (PD) models, `modelDiscrimination` for LGD models, and `modelDiscrimination` for exposure at default (EAD) models have an additional name-value pair for `ShowDetails` to indicate if the `DiscMeasure` output includes columns for `Segment` value and the `SegmentCount`.
- `modelAccuracy` for PD models has an additional column for `AccData` for `GroupCount`.

R2021b

Lifetime Credit Risk Analysis: Exposure at default (EAD) models

You can use `fitEADModel` to create a `Regression` or `Tobit` model for exposure at default. After creating a `Regression` or `Tobit` model object, you can use the following functions:

- `predict`
- `modelDiscrimination`
- `modelDiscriminationPlot`
- `modelAccuracy`
- `modelAccuracyPlot`

Deep Learning Examples: Credit risk analysis using Deep Learning Toolbox

Two examples demonstrate using Deep Learning Toolbox™ for credit risk analysis:

- The Compare Deep Learning Networks for Credit Default Prediction (Deep Learning Toolbox) example shows how to create, train, and compare three deep learning networks for predicting credit default probability.
- The Interpret and Stress-Test Deep Learning Networks for Probability of Default example shows how to train a model for credit risk for probability of default (PD) prediction using a deep neural network. The example also shows how to use locally interpretable model-agnostic explanations (LIME) and Shapley values interpretability techniques to understand the predictions of the model.

Insurance Example: Mean square error of prediction (MSEP) for estimated ultimate claims

The Mean Square Error of Prediction for Estimated Ultimate Claims example demonstrates a workflow to measure the quality of the estimated ultimate claims by calculating the MSEP.

Insurance Example: Bootstrap using chain ladder method to estimate ultimate claims

The Bootstrap Using Chain Ladder Method example demonstrates a workflow using the chain ladder bootstrap method to generate several `developmentTriangle` objects and estimate the ultimate claims.

Live Editor Task: Predictor screening

Use the Live Editor task **Threshold Predictors** to interactively set credit scorecard predictor thresholds for one or more risk metrics computed for a set of predictors, or features.

Lifetime Credit Risk Analysis: Cox lifetime probability of default (PD) model

You can perform lifetime credit analysis using `fitLifetimePDModel` to create a `Cox` model for survival analysis of a lifetime probability of default model. You can then use the following functions:

- `predict`
- `predictLifetime`
- `modelDiscrimination`
- `modelDiscriminationPlot`
- `modelAccuracy`
- `modelAccuracyPlot`

R2021a

Lifetime Credit Risk Analysis: Create Loss Given Default (LGD) models

You can use `fitLGDModel` to create a `Regression` or `Tobit` model for loss given default. After creating a `Regression` or `Tobit` model object, you can use the following functions:

- `predict`
- `modelDiscrimination`
- `modelDiscriminationPlot`
- `modelAccuracy`
- `modelAccuracyPlot`

Binning Explorer app: New user interface design for improved usability and performance

The **Binning Explorer** app is updated with a new user interface. The new design includes an **Overview** pane that makes the app easier to use and speeds up your workflow.

Lifetime Credit Risk Analysis: Plot discrimination and accuracy of lifetime probability of default (PD) models

You can create validation plots for lifetime PD models using `modelDiscriminationPlot` and `modelAccuracyPlot`.

Insurance Analysis: Plot link ratios and claims for development triangles

Using a `developmentTriangle` object, you can generate plots using `linkRatiosPlot` and `claimsPlot`.

Insurance Analysis: Estimate unpaid claims using Cape Cod technique

You can use `developmentTriangle` objects for reported and paid claims to create a `capeCod` object. You can then use the following functions:

- `ibnr`
- `unpaidClaims`
- `ultimateClaims`
- `summary`

R2020b

Market Risk: Backtest expected shortfall (ES) models using minimally biased Acerbi-Szekely tests

You can perform minimally biased Acerbi-Szekely tests with the `minBiasRelative` and `minBiasAbsolute` functions when using an `esbacktestbysim` object.

Market Risk: Expected shortfall (ES) model VaR level extended to 99.9%

The VaR level is extended to 99.9% for ES backtesting with an `esbacktest` object that uses precomputed tables of critical values.

Lifetime Credit Analysis: Probability of default models and examples

You can perform lifetime credit analysis using `fitLifetimePDModel` to create a `Logistic` or `Probit` lifetime probability of default model. You can then use the `predict`, `predictLifetime`, `modelDiscrimination`, and `modelAccuracy` functions to analyze the probability of default model. For more information, see examples:

- Compare Logistic Model for Lifetime PD to Champion Model
- Compare Lifetime PD Models Using Cross-Validation
- Expected Credit Loss (ECL) Computation

Insurance Analysis: Chain ladder, expected claims, and Bornhuetter-Fergurson techniques for analyzing insurance claims reserves

You can use a development triangle with the chain ladder technique, expected claims technique, or Bornhuetter-Fergurson technique to calculate important measures of insurance risk.

- The `developmentTriangle` object supports the following functions:
  - `view`
  - `linkRatios`
  - `linkRatioAverages`
  - `cdfSummary`
  - `ultimateClaims`
  - `fullTriangle`
- The `chainLadder` object supports the following functions:
  - `ibnr`
  - `unpaidClaims`
  - `summary`
- The `expectedClaims` object supports the following functions:
  - `ultimateClaims`
  - `ibnr`

- `unpaidClaims`
- `summary`

- The `bornhuetterFerguson` object supports the following functions:

  - `ultimateClaims`
  - `ibnr`
  - `unpaidClaims`
  - `summary`

R2020a

Consumer Credit Risk: Screen credit scorecard predictors on data that is too big to fit in memory using tall Arrays

Tall variable support for `screenpredictors`.

Random number generation for credit copula classes has changed

In R2019b and previous releases, when using the `creditDefaultCopula` and `creditMigrationCopula` classes to perform nonparallel simulations, the MATLAB® global random number generator was used to generate scenarios.

Compatibility Considerations

In R2020a, the random number generator for the `creditDefaultCopula` and `creditMigrationCopula` classes is set to `Threefry` for both parallel and nonparallel code paths.

R2019b

Market Risk: Backtest expected shortfall (ES) models using Du and Escanciano tests

The following new functions provide support for performing expected shortfall (ES) backtests by Du and Escanciano:

- `esbacktestbyde`
- `summary`
- `runtests`
- `unconditionalDE`
- `conditionalDE`
- `simulate`

New examples demonstrating workflows using expected shortfall (ES) backtests by Du and Escanciano:

- Workflow for Expected Shortfall (ES) Backtesting by Du and Escanciano
- Rolling Windows and Multiple Models for Expected Shortfall (ES) Backtesting by Du and Escanciano

Consumer Credit Risk: Validation of compact credit scorecards using validatemodel

Work with compact credit scorecards using the `compactCreditScorecard` class and then use `validatemodel` to validate a compact credit scorecard.

Credit Scorecard: Example comparing credit score using logistic regression and decision trees

A new example compares credit score values using logistic regression and decision trees (Comparison of Credit Scoring Using Logistic Regression and Decision Trees).

Credit Scorecard: Example using reject inference to incorporate credit rejection data into creditscorecard workflow

A new example demonstrates two approaches for using reject inference techniques to incorporate credit rejection data as part of the `creditscorecard` modeling workflow (Use Reject Inference Techniques with Credit Scorecards).

Consumer Credit Risk: Example comparing probability of default using through-the-cycle and point-in-time models

A new example compares the probability of default using through-the-cycle (TTC) and point-in-time (PIT) models (Comparison of Probability of Default Using Through-the-Cycle and Point-in-Time Models).

Consumer Credit Risk: Example fitting different types of models to loss given default (LGD) data

A new example demonstrates fitting different types of models to loss given default (LGD) data (Model Loss Given Default).

Calculation of p-value for bin has changed

In R2019a and previous releases, the binomial VaR backtest reports the tail probability as the *p*-value and compares the reported *p*-value to half of the *alpha* (`1- TestLevel`) of the `bin` test.

In R2019b, the binomial VaR backtest `bin` calculates the *p*-value contained in the `TestResults` output using the 2*tail probability convention and the *p*-value can be compared to *alpha*. For more information, see Algorithms.

R2019a

Consumer Credit Risk: Predictor screening for credit scorecards

Perform predictor screening for credit scorecards using the `screenpredictors` function. For more information, see Feature Screening with `screenpredictors`.

Consumer Credit Risk: Support for compact credit scorecards for easier deployment and reduced memory usage

Work with compact credit scorecards using the `compactCreditScorecard` class along with associated functions for `displaypoints`, `score`, and `probdefault`. In addition, you can create a compact credit scorecard by using the `compact` function from Financial Toolbox™ with a `creditscorecard` object. For more information, see `compactCreditScorecard` Object Workflow and Validate the Quality of a Compact Credit Scorecard Model.

R2018b

Binning Explorer: Bin data automatically using merge and split algorithms

The Binning Explorer app supports the merge and split algorithms. For more information, see **Binning Explorer**.

Binning Explorer: Bin missing data in a separate bin

The **Binning Explorer** app supports binning missing data for a predictor in a separate <missing> bin. For more information, see **Binning Explorer**.

Binning Explorer: Load data from the command line

The Binning Explorer app supports loading data or a creditscorecard object from the command line. For more information, see **Binning Explorer**.

Corporate Credit Risk: Perform parallel simulations using creditDefaultCopula and creditMigrationCopula

The creditDefaultCopula and creditMigrationCopula classes support the 'UseParallel' property for parallel simulations when using the simulate and riskContributions functions. You can set the 'UseParallel' property when creating creditDefaultCopula or creditMigrationCopula objects only if you have Parallel Computing Toolbox™.

creditCopula object removed

The creditCopula object is removed.

Compatibility Considerations

Object Name	What Happens When You Use It	Use This Instead	Compatibility Considerations
creditCopula	Removed	creditDefaultCopula	Replace all instances of creditCopula object with a creditDefaultCopula object using creditDefaultCopula.

R2018a

Corporate Credit Risk: Calculate standard deviation and value-at-risk contributions for each counterparty in a credit portfolio

The `riskContribution` function for `creditDefaultCopula` and the `riskContribution` function for `creditMigrationCopula` support returned information for counterparty contributions for standard deviation of the losses (`Std`) and value at risk (`VaR`) at the threshold `VaRLevel`.

R2017b

Corporate Credit Risk: Compute regulatory capital and value-at-risk using an asymptotic single risk factor (ASRF) model

The `asrf` function provides an ASRF model for credit risk analysis. `asrf` accepts the risk characteristics of a portfolio of credit-sensitive instruments as input and computes the necessary capital using an ASRF model.

Corporate Credit Risk: Perform credit portfolio simulation with random loss given default (LGD)

Support is provided for specifying random LGD (loss given default) for `creditDefaultCopula` and `creditMigrationCopula` objects. You can now specify the `LGD` input argument as a `NumCounterparties`-by-2 matrix, where the first column contains the LGD mean values and the second column contains the LGD standard deviations. In this case, LGD values are drawn randomly from a beta distribution with the parameters provided for the defaulting counterparty.

Market Risk: Backtest expected shortfall models

The following tools support expected shortfall (ES) backtesting for table-based tests for the unconditional Acerbi-Szekely test.

- `esbacktest`
- `summary`
- `runtests`
- `unconditionalNormal`
- `unconditionalT`

The following tools support expected shortfall (ES) backtesting for distribution tests for normal and `t` distributions.

- `esbacktestbysim`
- `summary`
- `runtests`
- `conditional`
- `unconditional`
- `quantile`
- `simulate`

Consumer Credit Risk: Specify weights in credit scorecards using Binning Explorer

Specify weights in a credit scorecard when using the Binning Explorer app. For more information on defining weights for a `creditscorecard` object, see the optional name-value pair argument `WeightsVar` for `creditscorecard`.

creditCopula object renamed

The `creditCopula` object is renamed to the `creditDefaultCopula` object.

Compatibility Considerations

Object Name	What Happens When You Use This Object	Use This Object Instead	Compatibility Considerations
`creditCopula`	Errors	`creditDefaultCopula`	Replace all instances of `creditCopula` object with `creditDefaultCopula` object using `creditDefaultCopula`. **Note** The `CounterpartyLosses` property of the `creditCopula` object is removed in the `creditDefaultCopula` object. To obtain counterparty losses, use the `getScenarios` function.

R2017a

Corporate Credit Risk: Estimate the probability of credit rating migration based on multifactor copula model

The following tools support corporate credit portfolio analysis for credit migration simulation using a `creditMigrationCopula` object for copula-based simulations:

- `creditMigrationCopula`
- `simulate`
- `portfolioRisk`
- `riskContribution`
- `confidenceBands`
- `getScenarios`

Corporate Credit Risk: Quantify credit concentration risk by Herfindahl index and other concentration measures

The `concentrationIndices` function supports the following concentration indices:

- CR — Concentration ratio
- Deciles — Deciles of the portfolio weights distribution
- Gini — Gini coefficient
- HH — Herfindahl-Hirschman index
- HK — Hannah-Kay index
- HT — Hall-Tideman index
- TE — Theil entropy index

Corporate Credit Risk: Model corporate default risk using Merton model

The `mertonmodel` and `mertonByTimeSeries` functions estimate the default probability using Merton's model.

creditCopula object renamed

The `creditCopula` object is renamed to the `creditDefaultCopula` object.

Compatibility Considerations

Object Name	What Happens When You Use This Object	Use This Object Instead	Compatibility Considerations
`creditCopula`	Warns	`creditDefaultCopula`	Replace all instances of `creditCopula` object with `creditDefaultCopula` object using the `creditDefaultCopula` constructor. **Note** The `CounterpartyLosses` property of `creditCopula` object is removed in the `creditDefaultCopula` object. To obtain counterparty losses, use the `getScenarios` function.

R2016b

Version: 2.4

New Features

Consumer Credit Risk: Binning Explorer for Credit Scorecards

Binning Explorer is an app for developing and modifying binning assignments for a `creditscorecard` object. For more information, see Binning Explorer.

Corporate Credit Risk: Copula-based simulation framework

The following tools support corporate credit portfolio analysis using a `creditCopula` object for copula-based simulations:

- `creditCopula` — Creates a `creditCopula` object.
- `simulate` — Simulates credit defaults using a `creditCopula` object.
- `portfolioRisk` — Generates portfolio-level risk measurements for a `creditCopula` object.
- `confidenceBands` — Generates confidence interval bands for a `creditCopula` object.
- `riskContribution` — Generates risk contributions for each counterparty in the `creditCopula` object.

Market Risk: Value-at-Risk Backtesting Tools

Value-at-risk (VaR) is an important measure of financial risk. VaR is an estimate of how much value a portfolio can lose in a given time period with a given confidence level. VaR backtesting tools assess the accuracy of VaR models. The following VaR backtesting tools are supported:

- `varbacktest` — Creates a `varbacktest` object using portfolio outcomes data and corresponding value-at-risk (VaR) data.
- `bin` — Binomial test.
- `cc` — Christoffersen's conditional coverage mixed test.
- `cci` — Christoffersen's conditional coverage independence test.
- `pof` — Kupiec's proportion of failures test.
- `tbf` — Haas's time between exceptions independence test.
- `tbfi` — Haas's mixed time between exceptions (independence and frequency) test.
- `tl` — Traffic light test.
- `tuff` — Kupiec's time until the first failure test.
- `summary` — Summary report on the given `varbacktest` data.
- `runtests` — Runs all tests and reports the final test results.