DecL Cheat Sheet for Compound Distributions

The DecL specification of an `Aggregate` compound distribution object has eight clauses:

`agg <NAME> <EXPOSURE> <LIMIT*> <SEVERITY> <OCC_RE*> <FREQUENCY> <AGG_RE*> <NOTE*>`

Key: `<INPUT>` denotes user input(s); clauses marked with an asterisk are optional; `lower_case` is a DecL keyword; `CLAUSE_TYPE` is a valid clause; options: `a|b|c`.

1. Name Clause

`agg NAME`

Name of the compound
Access from knowledge as `agg.NAME`.
Names must match regex
`r'[a-zA-Z][\._: a-zA-Z0-9\-]*'`.

2. Exposure Clause

`<EXP_LOSS> loss`
`<PREMIUM> premium at <LR> lr`
`<EXPOSURE> exposure at <RATE> rate`
`<CLAIMS> claims`
`dfreq <OUTCOMES> <PROBABILITIES*>`

Outcomes entered `[1 2 3 4]` or `[2:10:2]` and probabilities `[.5 .25 1/8 1/9]` or omitted for equally likely.

3. Limit Clause (optional)

`<LIMIT> xs <ATTACHMENT>`

Occurrence limits applied to ground-up severity, unlimited reinstatements, losses conditional on attaching layer by default.

4. Severity Clause

`sev <DIST_NAME> <MEAN> cv <CV>`
`sev <DIST_NAME> <SHAPE1> <SHAPE2>`
`dsev <OUTCOMES> <PROBABILITIES>`

`<SCALE> * SEV + <LOC>`.

`SEV splice [<LB> <UB>]` *conditional in layer*

`SEV !` *unconditional, when* `ATTACHMENT` *> 0*

5. Occurrence Reinsurance Clause

`occurrence ceded to LAYERS`
`occurrence net of LAYERS`
`LAYER=SHARE so LAYER xs ATTACH |`
`PARTICIPATION po LAYER xs ATTACH`
`LAYERS=LAYER1 and LAYER2 and ...`

Amount share of (`so`); participation (placed) $0 \le po \le 1$ *part of (`po`)*

`LAYERS=tower[250 500 1000]`

Specify layer breaks, expands to `250 xs 0`, `250 xs 250`, and `500 xs 500`; ground-up layer automatically added.

6. Frequency Clause

`poisson`, `bernoulli`, `fixed`, `geometric`,
`logarithmic`, `binomial XX`, `negbin var_mult`,
`pascal XX XX`, `neymana XX`
`mixed <MIXING DIST> <SHAPE1> <SHAPE2>`
`MIXING DIST=gamma|delaporte|ig|sichel|beta`
`FREQ zt`
`FREQ zm p0`

zero truncated, zero modified with $\Pr(N = 0) = p0$

7. Aggregate Reinsurance Clause

`aggregate ceded to LAYERS`
`aggregate net of LAYERS`
`aggregate (net of|ceded to) tower [<BREAKS>]`

8. Note

`note{prems op A curve, effective 1/1/2024;}`
`note{bs=100; log2=17; normalize=False}`

Add hints for updating

9. Vectorization

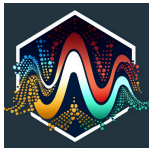
Exposure clause
`[1 2 3] claims`
`[100 200 300] loss`
`[100 200 300] premium at [.8 .7 .65] lr`
Layers clause
`[250 250 500] xs [0 250 500]` *zip layers*
Severity clause
`[1 3] * expon 1 wts [.6 .4]`
`[1 3] * [gamma lognorm] [4 1.25] wts [.6 .4]`
Vectors are broadcast; layers, exposure etc. are zipped.

10. Mathematical Expressions

Only division, exponentiation, and exponential allowed
`123`, `12.34e2`, `-12.4e-5`, `-12.0`, `12.4%`
`1/2`, `3**4`, `exp(2)`
Warning: *there is no unary minus, minus binds to the number:* $-4^2 = (-4)^2 = 16$.
Scale factor for lognormal μ, σ *entered as* `exp(mu)/exp(sigma**2/2)`.

Notes:

`aggregate 0.18.0`
2023-07-16



`Aggregate` Class Cheat Sheet

The `Aggregate` call signature follows the corresponding DecL clauses, using prefixes for exposure (including limit sub-clause), severity, occurrence reinsurance, frequency, aggregate reinsurance, and note. `sev_xs`, `sev_ps` equal `dsev` outcomes and probabilities, and `(occ|agg)_reins` clauses are lists of (share, limit, attachment) triples.

m `Aggregate(name, exp_el=0, exp_premium=0, exp_lr=0, exp_en=0, exp_attachment=0, exp_limit=np.inf, sev_name=", sev_a=np.nan, sev_b=0, sev_mean=0, sev_cv=0, sev_loc=0, sev_scale=0, sev_xs=None, sev_ps=None, sev_wt=1, sev_conditional=True, occ_reins=None, occ_kind=", freq_name=", freq_a=0, freq_b=0, freq_zm=False, freq_p0=np.nan, agg_reins=None, agg_kind=", note=")`[0]

The following tables show all m `methods`, and fields or properties (used interchangeably). Comments elucidate the meaning of more obscure entries.

1. Specification & creation

`name`, `limit`, `attachment`, `freq_name`, `freq_a`, `freq_b`, `freq_p0`, `freq_zm`, `note`, `sev_pick_attachments`, `sev_pick_losses`, `program` (DecL program), `pprogram` (pretty printed), `spec` (constructor kwarg dictionary; `Aggregate(**spec)` re-creates the object), `spec_ex` (adds meta elements)

2. Update

`log2`, `bs`, `sev_calc` (discrete=round, forward, backwards), `discretization_calc` (distribution, survival, both), `normalize`, `padding`, `tilt_vector`, `approximation` (exact, slognorm, sgamma), `fzapprox` (frozen approximation rv), m `picks`, m `discretize`, m `easy_update`, m `recommend_bucket`, m `rescale` (homogeneous severity or inhomogeneous frequency rescaling), m `update`, m `update_work`

3. Moments

`est` prefix=estimated from FFT approximation

`agg_m`, `agg_cv`, `agg_sd`, `agg_var`, `agg_skew`, `est_m`, `est_cv`, `est_sd`, `est_var`, `est_skew`, `sev_m`, `sev_cv`, `sev_sd`, `sev_var`, `sev_skew`, `est_sev_m`, `est_sev_cv`, `est_sev_sd`, `est_sev_var`, `est_sev_skew`, m `freq_moms`, m `freq_pmf`, m `freq_pgf`, `panjer_ab` (Panjer parameters), m `prn_eq_0` ($\Pr(N=0)$ unmodified), `n` (frequency), `en` (vector), `unmodified_mean` (when ZT or ZM)

4. Statistical functions

`sevs` (list of `Severity`s), m `cdf`, m `sf` (survival), m `pdf`, m `pmf`, m `q` (lower quantile=VaR), m `tvar`, m `sev` (exact severity cdf, sf, pdf), m `q_sev`, m `tvar_sev`, m `var_dict`[1], m `sample`

5. Validation

`describe` (validation statistics), `valid` (true="not unreasonable" or false), `validation_eps` (validation epsilon threshold 1e-04), m `qt` ("quick test" validation details), m `aggregate_error_analysis` (agg errors over range of bs), m `severity_error_analysis` (truncation and discretization errors by severity component)

6. Output dataframes

`density_df`[1] (main output), `report_df` (component, mixture & empirical stats), `agg_density`, `agg_density_ceded`, `agg_density_gross`, `agg_density_net`, `sev_density`, `sev_density_ceded`, `sev_density_gross`, `sev_density_net`, `ftagg_density`, `xs`, `statistics_df` (row, by component), `statistics_total_df` (row, indep. vs. mixed), `statistics` (cols, combined, better index), `audit_df` (deprecated), `report_ser` (internal, series), *see also Reinsurance.*

7. Reinsurance

`agg_kind` (net of or ceded to), `agg_reins` (list), `agg_reins_df` (gcn loss and dists), `occ_kind`, `occ_reins`, `occ_reins_df`, m `agg_ceder`, m `agg_netter`, m `apply_agg_reins`, m `occ_ceder`, m `occ_netter`, m `apply_occ_reins`, m `reinsurance_description` (text rendering of re), `reinsurance_kinds` (None, occ, agg, occ & agg), `reinsurance_audit_df` (stats by gcn, splits severity for occ), `reinsurance_occ_layer_df` (aggregate gcn stats for occ layers), `reinsurance_df` (all combinations of gcn occ and agg densities), `reinsurance_report_df` (m, cv, sd, skew for `reinsurance_df`)

8. Visualization

m `plot`, m `reinsurance_occ_plot` `figure` (return last figure), m `limits` (suggest axis limits for plotting),

9. Risk and pricing

m `apply_distortion`, m `price(p, dist)` m `cramer_lundberg aka pollaczeck_khinchine` (probability of eventual ruin vs. initial capital and margin)

10. Approximations

Method of moments (shifted gamma or lognormal), or minimum entropy approximations.

m `approximate`, m `entropy_fit`

11. Meta

`aggregate_keys` (internal), m `more(regex)` (print all methods and fields matching `regex`), `info` (text meta info), m `html_info_blob` (internal), m `json` (persist to json), m `snap`[1] (snap argument to index)

Notes:

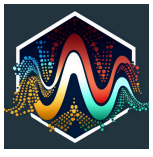
[0]: Arguments `sev_pick_attachments=None`, `sev_pick_losses=None`, omitted; see help.

[1]: matches `Portfolio`

Any vectorizable input accepts numeric or iterable datatypes.

Abbreviations: gcn=gross (subject), ceded, and net; stats: m=mean, cv=coefficient of variation, sd=standard deviation, var=variance, skew(ness); VaR=value-at-risk

`aggregate 0.18.0`
2023-07-16



`Portfolio` Class Cheat Sheet

The `Portfolio` call signature is straightforward. `spec_list` is a DecL program, list of `Aggregate` objects or kwargs, or names known to the Underwriter, or a pandas DataFrame sample.

(m) `Portfolio(self, name, spec_list, uw=None)`

The following tables show all (m) `methods`, (s) `static methods`, and fields or properties (used interchangeably). Comments elucidate the meaning of more obscure entries. Internal methods and fields are not shown.

1. Specification & creation

`name`, `n_units`, `agg_list` (list of `Aggregate` objects), `line_names`, `line_names_ex`, `unit_names` (unit ← line), `unit_names_ex`, `line_name_pipe`, `program` (DecL program), `pprogram` (pretty printed), `spec` (constructor kwarg dictionary; `Aggregate(**spec)` re-creates the object), `spec_ex` (adds meta information), (m) `nice_program`, (s) `from_DataFrame`, (s) `from_dict_of_aggs`, (s) `from_Excel`, (s) `create_from_sample`

2. Update

`log2`, `bs`, `sev_calc` (discrete=round, forward, backwards), `discretization_calc` (distribution, survival, both), `normalize`, `padding`, `tilt_amount`, `approx_freq_ge`, `approx_type` (exact, slognorm, sgamma), (m) `best_bucket`, (m) `recommend_bucket`, (m) `update`, (m) `add_exa`, (m) `add_exa_details`, (m) `add_exa_sample`, (m) `trim_df`, (m) `ft` & (m) `ift` (FFT and inverse FFT), (m) `remove_fuzz`, (m) `set_a_p`

3. Moments

`est` prefix=estimated from FFT approximation

`agg_m`, `agg_cv`, `agg_sd`, `agg_var`, `agg_skew`, `est_m`, `est_cv`, `est_sd`, `est_var`, `est_skew`, `ex`

4. Statistical functions

(m) `cdf`, (m) `sf` (survival), (m) `pdf`, (m) `pmf`, (m) `q` (lower quantile=VaR), (m) `tvar`, (m) `tvar_threshold`, (m) `var`, (m) `var_dict`, (m) `density_sample`, (m) `percentiles`, (m) `sample`, (m) `sample(_density)_compare`,

5. Validation

`describe` (validation statistics),
`valid` (true=all components and total "not unreasonable" or false),
`validation_eps` (validation epsilon threshold 1e-04),
(m) `audits`, (m) `uat`, (m) `uat_differential`,
(m) `uat_interpolation_functions`

6. Output dataframes

`density_df`[1] (main output),
`report_df` (component, mixture & empirical stats),
`statistics`, `statistics_df`, `audit_df`,
`augmented_df`, `independent_audit_df`,
`independent_density_df`, `priority_analysis_df`,
(m) `make_audit_df`, (m) `make_all`, (m) `report`

7. Reinsurance

None – applies at the component level

8. Visualization & exhibits

(m) `plot`, (m) `scatter`, (m) `twelve_plot`,
(m) `biv_contour_plot`,
(m) `analyze_distortion_plots`,
(m) `natural_profit_segment_plot`,
(m) `profit_segment_plot`, `figure` (return last figure),
(m) `limits`, `line_renamer`,
`premium_capital_renamer`, `renamer`,
(m) `short_renamer`, `stat_renamer`, `tm_renamer`,
(m) `show_enhanced_exhibits`,
`EX_accounting_economic_balance_sheet`,
`EX_multi_premium_capital`, `EX_premium_capital`

9. Risk and pricing

(m) `accounting_economic_balance_sheet`,
(m) `analysis_collateral`, (m) `analysis_priority`,
(m) `analyze_distortion(s|_add_comps)`,
(m) `apply_distortion(s)`, `assets_2_epd`,
(m) `bodoff`, (m) `calibrate_blends`,
(m) `calibrate_distortion(s)`, (m) `cotvar`,
`dist_ans`, `distortion`, `distortion_df`, `dists`,
`epd_2_assets`, (m) `equal_risk_epd`,
(m) `equal_risk_var_tvar`, (m) `gamma`, (m) `gradient`,
(m) `merton_perold`, (m) `multi_premium_capital`,
(m) `premium_capital`, (m) `price`, (m) `price_ccoc`,
(m) `pricing_bounds`, `priority_capital_df`,
(m) `stand_alone_pricing`,

10. Approximations

(m) `approximate`, (m) `as_severity`, (m) `collapse`

11. Meta

`audit_percentiles`, `hash_rep_at_last_update`, `info` (text meta info), (m) `json` (persist to json), `last_update`, (m) `more(regex)` (print all methods and fields matching `regex`), (m) `save`, (m) `snap`[1] (snap argument to index)

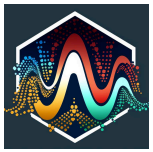
Notes:

[1]: matches `Aggregate`

Any vectorizable input accepts numeric or iterable datatypes.

Abbreviations: gcn=gross (subject), ceded, and net; stats: m=mean, cv=coefficient of variation, sd=standard deviation, var=variance, skew(ness); VaR=value-at-risk

`aggregate 0.18.0`
2023-07-16



`Severity` Class Cheat Sheet

The `Severity` call signature, `sev_xs`, `sev_ps` equal `dsev` outcomes and probabilities,

m `Severity(name, sev_name='', sev_a=np.nan, sev_b=0, sev_mean=0, sev_cv=0, sev_loc=0, sev_scale=0, sev_xs=None, sev_ps=None, sev_wt=1, sev_lb, sev_ub, sev_conditional=True)`

The following tables show all m `methods`, and fields or properties (used interchangeably). Comments elucidate the meaning of more obscure entries.

1. Specification & creation

`a`, `attachment`, `b`, `badvalue`, `conditional`, `detachment`, `extradoc`, `limit`, `long_name`, `name`, `note`, `numargs`, `program`, `sev_loc`, `sev_name`, `sev_wt`, `shapes`,

2. Update

m `cv_to_shape`, m `mean_to_scale`, `pattach`, `pdetach`,

3. Moments

m `generic_moment`, m `mean`, m `median`, m `moment`, m `moment_type`, m `moms`, `sev1`, `sev2`, `sev3`, m `stats`, m `std`, m `support`, m `var`,

4. Statistical functions

m `cdf`, m `entropy`, m `expect`, m `interval`, m `isf`, m `logcdf`, m `logpdf`, m `logsf`, m `nnlf`, m `pdf`, m `ppf`, m `rvs`, m `sf`, m `vecentropy`,

5. Validation

None

6. Output dataframes

None

7. Reinsurance

None

8. Visualization

m `plot`,

9. Risk and pricing

None

10. Approximations

m `fit`, m `fit_loc_scale`, m `freeze`,

11. Meta

`fz`, `random_state`, `xtol`,

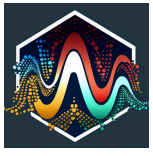
Notes:

[0]: Arguments `sev_pick_attachments=None`, `sev_pick_losses=None`, omitted; see help.

[1]: matches `Portfolio`

Any vectorizable input accepts numeric or iterable datatypes.

Abbreviations: gcn=gross (subject), ceded, and net; stats: m=mean, cv=coefficient of variation, sd=standard deviation, var=variance, skew(ness); VaR=value-at-risk



`aggregate 0.18.0`

2023-07-16

`Underwriter` Class Cheat Sheet

The `Underwriter` call signature follows the corresponding DecL clauses, using prefixes for exposure (including limit sub-clause), severity, occurrence reinsurance, frequency, aggregate reinsurance, and note. `sev_xs, sev_ps` equal `dsev` outcomes and probabilities, and `(occ|agg)_reins` clauses are lists of (share, limit, attachment) triples.

m `Severity(name, exp_el=0, exp_premium=0, exp_lr=0, exp_en=0, exp_attachment=0, exp_limit=np.inf, sev_name=", sev_a=np.nan, sev_b=0, sev_mean=0, sev_cv=0, sev_loc=0, sev_scale=0, sev_xs=None, sev_ps=None, sev_wt=1, sev_conditional=True, occ_reins=None, occ_kind=", freq_name=", freq_a=0, freq_b=0, freq_zm=False, freq_p0=np.nan, agg_reins=None, agg_kind=", note=")`[0]

The following tables show all m `methods`, and fields or properties (used interchangeably). Comments elucidate the meaning of more obscure entries.

1. Specification & creation

None

2. Update

`knowledge`, `lexer`, `log2`, `parser`, m `read_database`, m `read_databases`, m `safe_lookup`, `update`, m `write`, m `write_from_file`,

3. Moments

None

4. Statistical functions

None

5. Validation

m `interpret_program`, m `interpreter_file`, m `interpreter_line`, m `interpreter_list`, m `interpreter_test_suite`, m `test_suite`,

6. Output dataframes

None

7. Reinsurance

None

8. Visualization

None

9. Risk and pricing

None

10. Approximations

None

11. Meta

m `build`, `case_dir`, `databases`, `debug`, `default_dir`, m `dir`, m `factory`, m `logger_level`, m `more`, `name`, m `qlist`, m `qshow`, m `show`, `site_dir`, `template_dir`, `test_suite_file`, `version`,

Notes:

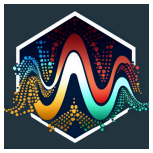
[0]: Arguments `sev_pick_attachments=None, sev_pick_losses=None,` omitted; see help.

[1]: matches `Portfolio`

Any vectorizable input accepts numeric or iterable datatypes.

Abbreviations: gcn=gross (subject), ceded, and net; stats: m=mean, cv=coefficient of variation, sd=standard deviation, var=variance, skew(ness); VaR=value-at-risk

`aggregate 0.18.0`
2023-07-16



`Distortion` Class Cheat Sheet

The `Distortion` call signature follows the corresponding DecL clauses, using prefixes for exposure (including limit sub-clause), severity, occurrence reinsurance, frequency, aggregate reinsurance, and note. `sev_xs, sev_ps` equal `dsev` outcomes and probabilities, and `(occ|agg)_reins` clauses are lists of (share, limit, attachment) triples.

m `Severity(name, exp_el=0, exp_premium=0, exp_lr=0, exp_en=0, exp_attachment=0, exp_limit=np.inf, sev_name=", sev_a=np.nan, sev_b=0, sev_mean=0, sev_cv=0, sev_loc=0, sev_scale=0, sev_xs=None, sev_ps=None, sev_wt=1, sev_conditional=True, occ_reins=None, occ_kind=", freq_name=", freq_a=0, freq_b=0, freq_zm=False, freq_p0=np.nan, agg_reins=None, agg_kind=", note=")`[0]

The following tables show all m `methods`, and fields or properties (used interchangeably). Comments elucidate the meaning of more obscure entries.

1. Specification & creation

`assets`, m `average_distortion`, `col_x`, `col_y`, `df`, `display_name`, m `distortions_from_params`, `has_mass`, `mass`, `name`, `premium_target`, `r0`, m `s_gs_distortion`, `shape`,

2. Update

m `bagged_distortion`, `error`,

3. Moments

None

4. Statistical functions

m `g`, m `g_dual`, m `g_inv`, m `g_prime`, m `wtd_tvar`,

5. Validation

m `test`,

6. Output dataframes

None

7. Reinsurance

None

8. Visualization

m `plot`,

9. Risk and pricing

m `price`, m `price2`,

10. Approximations

None

11. Meta

m `available_distortions`, m `convex_example`, `renamer`,

Notes:

[0]: Arguments `sev_pick_attachments=None, sev_pick_losses=None,` omitted; see help.

[1]: matches `Portfolio`

Any vectorizable input accepts numeric or iterable datatypes.

Abbreviations: gcn=gross (subject), ceded, and net; stats: m=mean, cv=coefficient of variation, sd=standard deviation, var=variance, skew(ness); VaR=value-at-risk

`aggregate 0.18.0`
2023-07-16