SEM Skewness Study: Indicator vs. Latent (Exponential Margins, Gaussian Copula)

Table of contents

```r
suppressPackageStartupMessages({
  library(dplyr)
  library(tidyr)
  library(readr)
  library(ggplot2)
  library(stringr)
  library(knitr)
})
```

```
# Paths for the SEM study
DATA_DIR   <- "data"
RES_DIR    <- "results_sem"
EXPORT_DIR <- file.path(RES_DIR, "exported_tables")
dir.create(EXPORT_DIR, showWarnings = FALSE, recursive = TRUE)

files <- list(
  cond   = file.path(RES_DIR,  "summary_conditions_sem.csv"),
  rep    = file.path(RES_DIR,  "summary_replications_sem.csv"),
  design = file.path(DATA_DIR, "sim_conditions_sem.rds")
)

if (!all(file.exists(unlist(files)))) {
  stop("Missing input(s): expecting results_sem/summary_*.csv and data/sim_conditions_sem.rds
       "Please run analysis_sem.R first.")
}

# Load design and summaries
design <- readRDS(files$design) |>
  dplyr::select(condition_id, sem_study, direction, T, rho)

cond_raw <- read_csv(files$cond, show_col_types = FALSE)
rep_raw  <- read_csv(files$rep,  show_col_types = FALSE)

# Join design fields onto the summaries
cond <- cond_raw |> left_join(design, by = "condition_id")
rep_df <- rep_raw |> filter(!is.na(param)) |> left_join(design, by = "condition_id")

# Pretty labels and factor levels
model_labs <- c(EI = "Indicator-skew (EI)", EL = "Latent-skew (EL)")
rep_df <- rep_df |>
  mutate(
    Model = factor(model, levels = c("EI","EL"), labels = model_labs),
    sem_study = factor(sem_study, levels = c("A_indicator","B_latent"),
                       labels = c("A: indicator-skew","B: latent-skew")),
    T  = factor(T),
    rho = factor(rho),
    direction = factor(direction, levels = c("++","--","+-"))
  )

cond <- cond |>
  mutate(
```

```r
    Model = factor(model, levels = c("EI","EL"), labels = model_labs),
    sem_study = factor(sem_study, levels = c("A_indicator","B_latent"),
                       labels = c("A: indicator-skew","B: latent-skew")),
    T  = factor(T),
    rho = factor(rho),
    direction = factor(direction, levels = c("++","--","+-"))
  )

# Parameter groups
core_params  <- c("mu[1]","mu[2]","phi11","phi12","phi21","phi22","rho")
extra_params <- c("sigma_exp[1]","sigma_exp[2]")

# Simple ggplot theme
theme_standard <- theme_bw(base_size = 13)

`%||%` <- function(a,b) if (!is.null(a)) a else b
```

0. tl;dr

Question. How do estimates and uncertainty behave when **skewness lives at different layers** of a SEM/VAR(1)?

- **Study A (EI):** skewed **measurement errors** (exponential margins), Gaussian state
- **Study B (EL):** skewed **state innovations** (exponential margins), no measurement error

Key expectations (to confirm with the figures below):

- **Layer matters.** When fitted at the correct layer, both EI and EL should recover the VAR dynamics (Φ) and intercepts (μ) well at ($T=100$).
- **(ρ) sensitivity.** The Gaussian copula correlation (ρ) is estimated on the **active layer**; misspecifying the layer (fitting EI to EL data or vice versa) tends to attenuate ($\hat{\rho}$).
- **Diagnostics vs. inference.** Occasional divergences (near one-sided bounds) need not imply poor inference for (Φ) or (μ), but do check coverage by parameter.

1. Introduction

We compare two bivariate SEM/VAR(1) formulations with **exponential one-sided margins** and a **Gaussian copula**:

- **EI (indicator-skew):**

  $$\mathbf{s}_t = \mu + \mathbf{B}, \mathbf{s}_{t-1} + \eta_t, \quad \eta_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \qquad \mathbf{y}_t = \mathbf{s}_t + \varepsilon_t.$$

  Skewness and (ρ) live in (ε_t).

- **EL (latent-skew):**

  $$\mathbf{y}_t = \mu + \mathbf{B}, \mathbf{y}_{t-1} + \zeta_t.$$

  Skewness and (ρ) live in (ζ_t).

Each margin uses an **exponential** law with **sign** (`direction`): right-skew ((+)): (e $\geq$ -s), left-skew ((-)): (e $\leq$ s); true **scale** (s=1). A **Gaussian copula** with correlation ($\rho \in 0.00, 0.30$) couples the two margins at each (t) on the **active layer**.

1.1 Simulation Design

Table 1: SEM study design (short grid).

Factor	Levels
SEM Study (active layer)	A: indicator-skew (measurement), B: latent-skew (innovations)
Skew Direction (per margin)	`++`, `--`, `+-` (right/right, left/left, right/left)
Copula Correlation (ρ) at active layer	0.00, 0.30
Time Series Length (T)	100
VAR(1) Coefficients ($\mathbf{B}$)	Fixed as $\begin{pmatrix} 0.55 & 0.10 \\ 0.10 & 0.25 \end{pmatrix}$
Replications / cell	10
Total Conditions	12

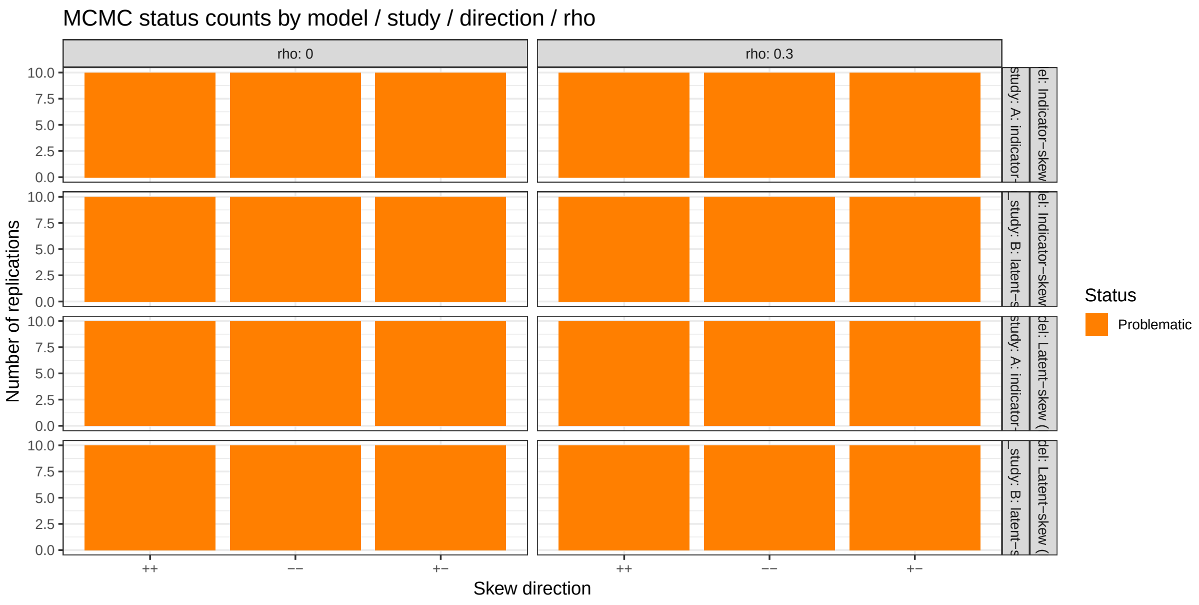
2. Data loading and preparation

```r
# Condition-level helper
cond <- cond |>
  mutate(RMSE = sqrt((mean_bias %||% 0)^2 + (emp_sd %||% 0)^2))

# Replication-level MCMC classification
RHAT_THRESHOLD <- 1.01
rep_df <- rep_df |>
```

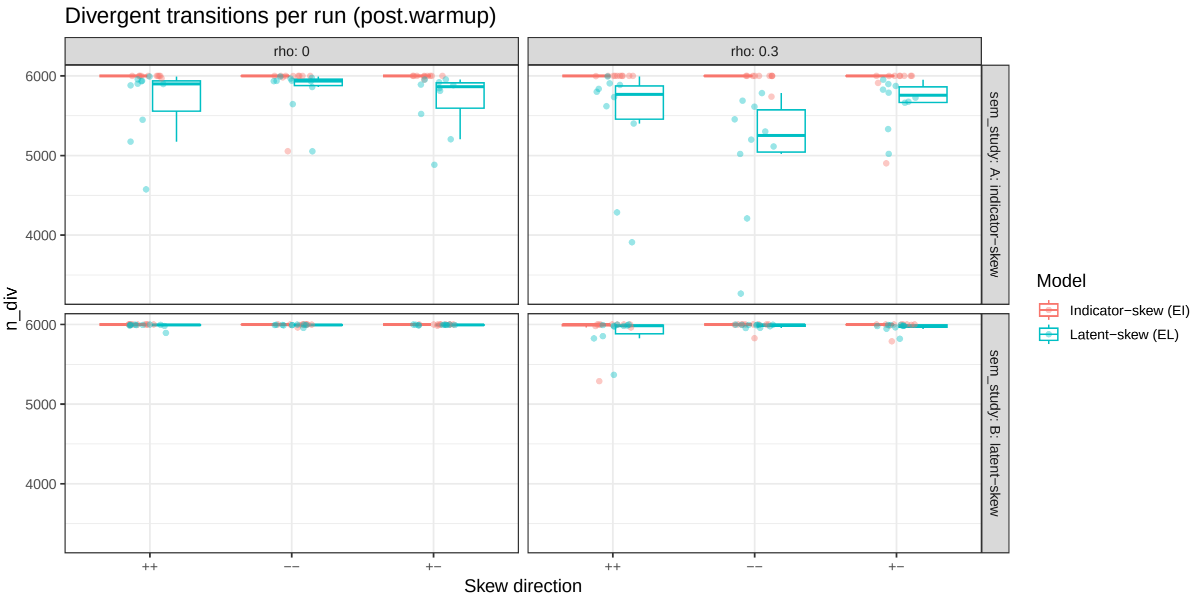
```
  mutate(
    mcmc_status = dplyr::case_when(
      is.na(max_rhat) | status != "ok" ~ "Failed/Error",
      max_rhat > RHAT_THRESHOLD | (n_div %||% 0) > 0 ~ "Problematic",
      TRUE ~ "Clean"
    ),
    mcmc_status = factor(mcmc_status, levels = c("Clean","Problematic","Failed/Error"))
  )
```

2.1 MCMC overview

```
mcmc_counts <- rep_df |>
  distinct(condition_id, rep_id, Model, sem_study, direction, rho, T, mcmc_status) |>
  count(Model, sem_study, direction, rho, T, mcmc_status, name = "N")

ggplot(mcmc_counts, aes(x = direction, y = N, fill = mcmc_status)) +
  geom_col() +
  facet_grid(Model + sem_study ~ rho, labeller = label_both) +
  scale_fill_manual(values = c("Clean"="#4daf4a","Problematic"="#ff7f00","Failed/Error"="#e4
  labs(title = "MCMC status counts by model / study / direction / rho",
       x = "Skew direction", y = "Number of replications", fill = "Status") +
  theme_standard
```



```r
div_data <- rep_df |>
  filter(param == "rho") |>
  distinct(condition_id, rep_id, Model, sem_study, direction, rho, T, n_div, mcmc_status) |>
  filter(mcmc_status != "Failed/Error")

ggplot(div_data, aes(x = direction, y = n_div, color = Model)) +
  geom_boxplot(outlier.shape = NA, alpha = 0.6) +
  geom_jitter(width = 0.15, alpha = 0.4, size = 1.5) +
  facet_grid(sem_study ~ rho, labeller = label_both) +
  labs(title = "Divergent transitions per run (post-warmup)",
       x = "Skew direction", y = "n_div") +
  theme_standard
```



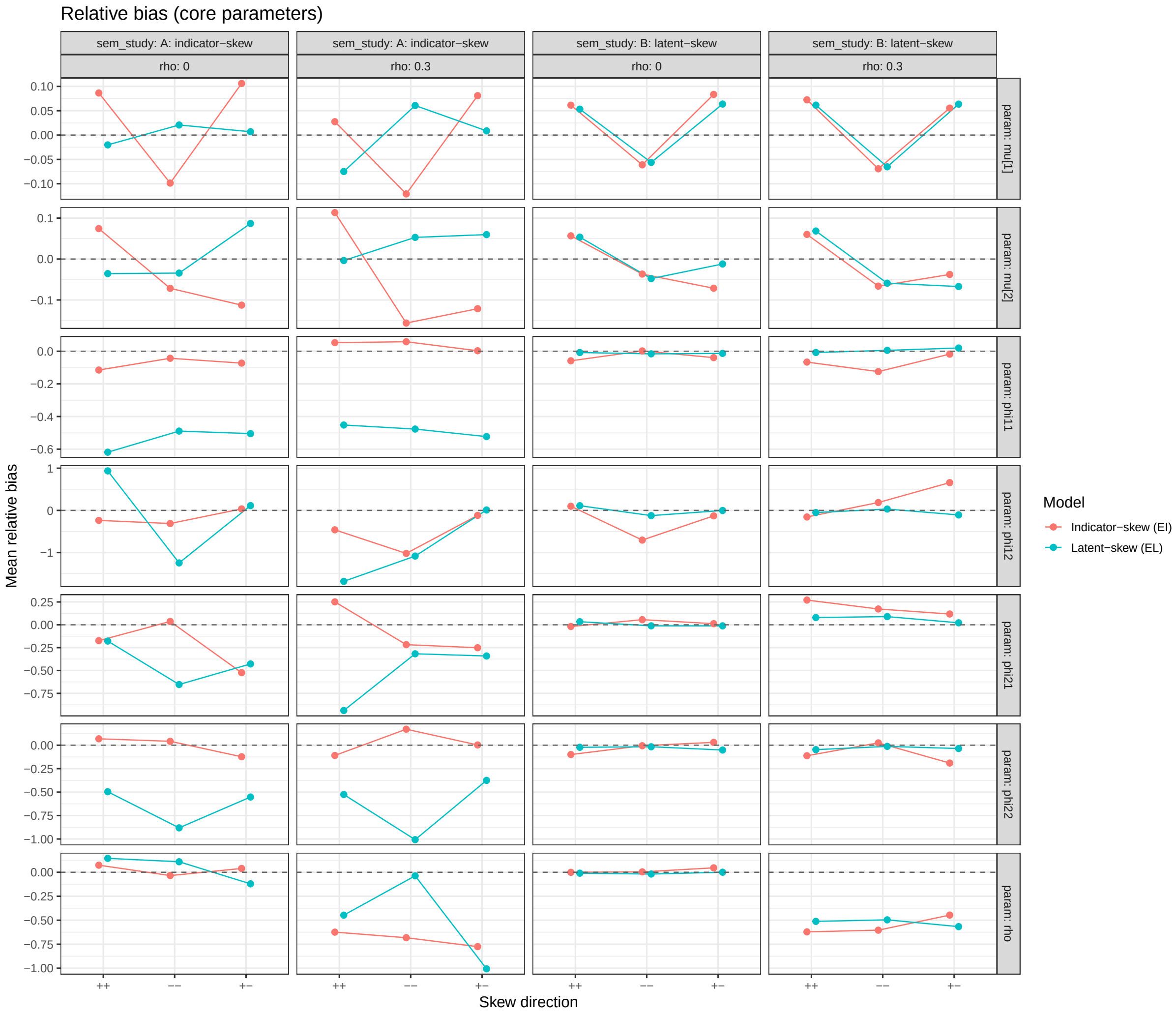
3. Core parameter accuracy ((Φ), (μ), (ρ))

We summarize **bias**, **coverage**, and **uncertainty calibration** for the core parameters $(\mu_1, \mu_2, \phi_{11}, \phi_{12}, \phi_{21}, \phi_{22}, \rho)$.

```r
plot_metric <- function(df, metric_col, title, ylab,
                        use_free_y = FALSE, ylims = NULL) {
  d <- df |> filter(param %in% core_params, !is.na(.data[[metric_col]]))
  if (nrow(d) == 0) return(NULL)
  p <- ggplot(d, aes(x = direction, y = .data[[metric_col]], color = Model, group = Model)) +
    geom_line(aes(group = Model), position = position_dodge(0.25)) +
    geom_point(position = position_dodge(0.25), size = 2.2) +
    facet_grid(param ~ sem_study + rho, labeller = label_both, scales = ifelse(use_free_y,"fi
    labs(title = title, x = "Skew direction", y = ylab, color = "Model") +
    theme_standard
  if (metric_col %in% c("mean_rel_bias","sd_bias")) {
    p <- p + geom_hline(yintercept = 0, linetype = "dashed", color = "grey40")
  } else if (metric_col == "coverage_95") {
    p <- p + geom_hline(yintercept = 0.95, linetype = "dashed", color = "grey40")
  }
  if (!is.null(ylims)) p <- p + coord_cartesian(ylim = ylims)
  p
}

cond_core <- cond |> filter(param %in% core_params)
```

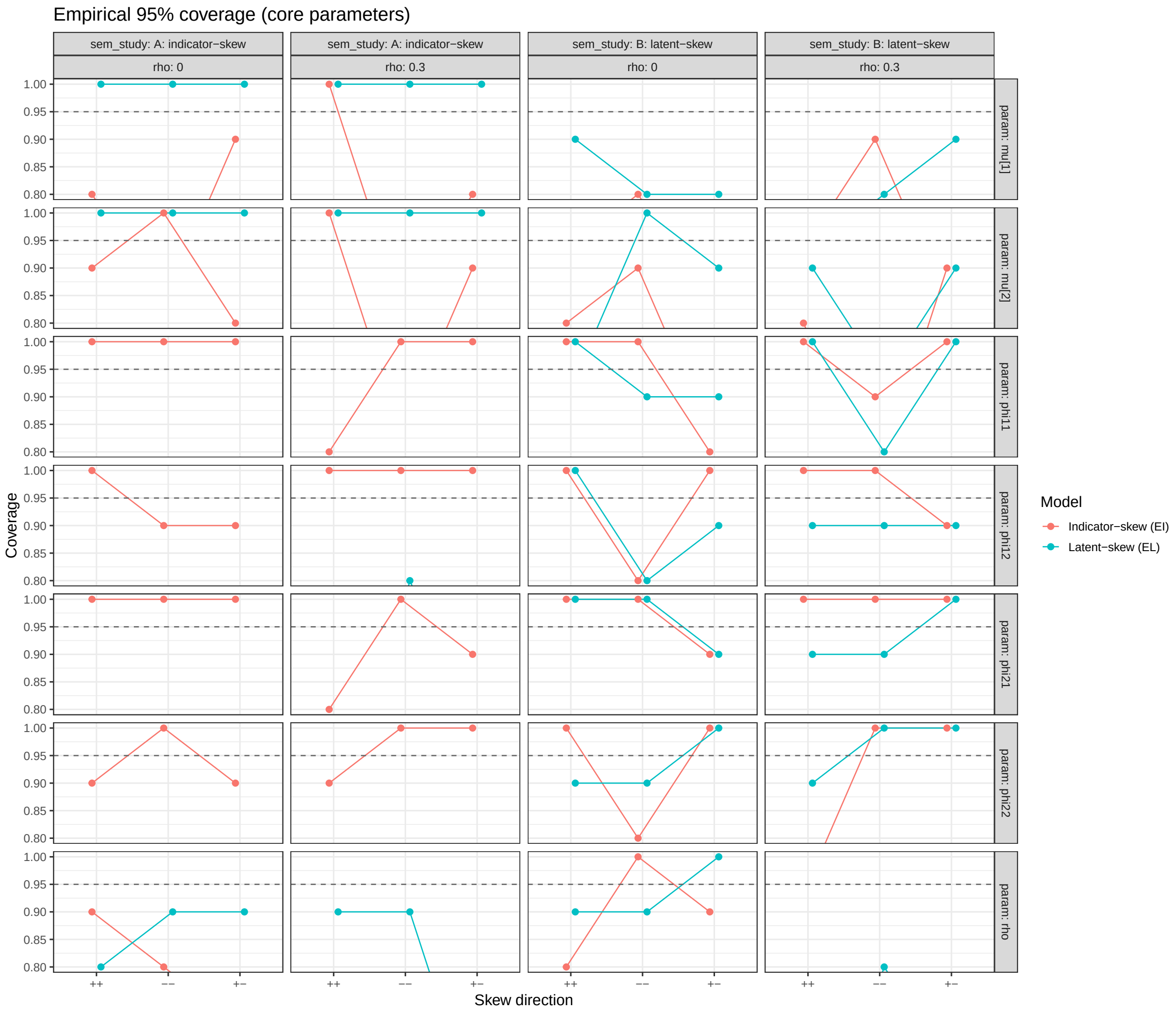
3.1 Relative bias

```
plot_metric(cond_core, "mean_rel_bias", "Relative bias (core parameters)", "Mean relative bia
```



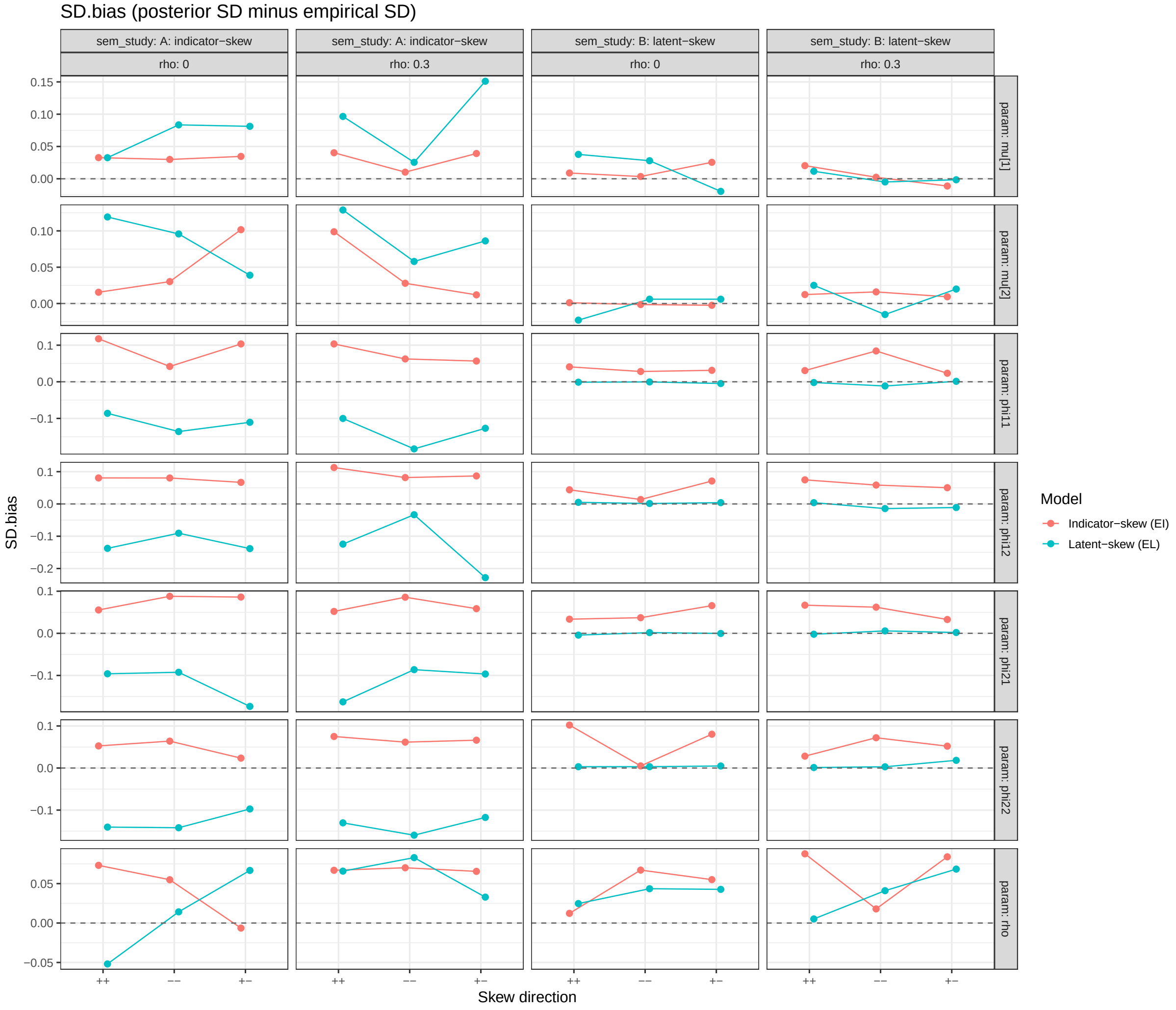
3.2 95% coverage

```
plot_metric(cond_core, "coverage_95", "Empirical 95% coverage (core parameters)", "Coverage"
```


Empirical 95% coverage (core parameters)

3.3 SD-bias (posterior SD − empirical SD)

```
plot_metric(cond_core, "sd_bias", "SD-bias (posterior SD minus empirical SD)", "SD-bias", us
```



4. Marginal scale parameters ((σ_{exp}))

Both EI and EL estimate (σ_{exp}) but on different layers:

- **EI:** (σ_{exp}) is the **measurement-error** scale.
- **EL:** (σ_{exp}) is the **innovation** scale.

Truth is ($\sigma_{\text{exp}} = 1$) for each margin. Bias near zero and coverage near 0.95 indicate good calibration.

```
cond_sigma <- cond |> filter(param %in% c("sigma_exp[1]","sigma_exp[2]"))
```

```
p1 <- plot_metric(cond_sigma, "mean_rel_bias", "Relative bias (sigma_exp)", "Mean relative bi
p2 <- plot_metric(cond_sigma, "coverage_95",   "Empirical 95% coverage (sigma_exp)", "Coverag

print(p1); print(p2)
```

```
NULL

NULL
```

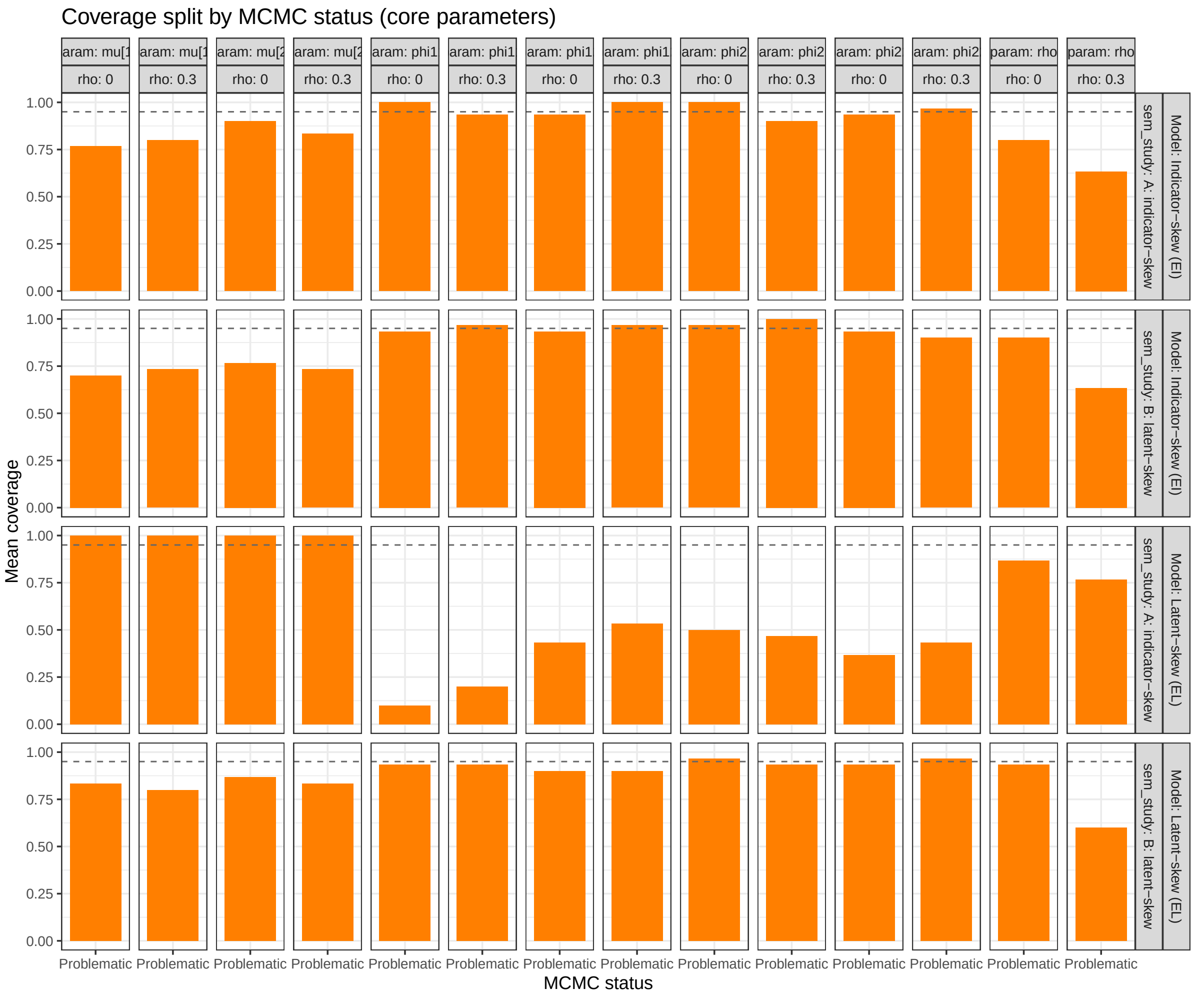
5. Clean vs. Problematic: does it matter for inference?

We re-aggregate replication-level metrics **by MCMC status** (Clean vs. Problematic) to check robustness.

```
aggregate_by_status <- function(df) {
  df |>
    filter(mcmc_status != "Failed/Error", param %in% core_params) |>
    group_by(condition_id, Model, param, mcmc_status, sem_study, direction, rho, T) |>
    summarise(
      N_valid      = n(),
      mean_rel_bias= mean(rel_bias, na.rm = TRUE),
      coverage_95  = mean(cover95, na.rm = TRUE),
      mean_post_sd = mean(post_sd, na.rm = TRUE),
      emp_sd       = sd(post_mean, na.rm = TRUE),
      mean_bias    = mean(bias, na.rm = TRUE),
      .groups      = "drop"
    ) |>
    mutate(
      emp_sd = ifelse(is.na(emp_sd), 0, emp_sd),
      sd_bias = mean_post_sd - emp_sd,
      RMSE = sqrt((mean_bias %||% 0)^2 + (emp_sd %||% 0)^2)
    )
}

cond_status <- aggregate_by_status(rep_df)
```

```
status_overview <- cond_status |>
  group_by(Model, param, sem_study, rho, mcmc_status) |>
  summarise(mean_coverage = mean(coverage_95, na.rm = TRUE), .groups = "drop")
```

```
ggplot(status_overview,
       aes(x = mcmc_status, y = mean_coverage, fill = mcmc_status)) +
  geom_col() +
  facet_grid(Model + sem_study ~ param + rho, labeller = label_both) +
  geom_hline(yintercept = 0.95, linetype = "dashed", color = "grey40") +
  scale_fill_manual(values = c("Clean"="#4daf4a","Problematic"="#ff7f00")) +
  labs(title = "Coverage split by MCMC status (core parameters)",
       x = "MCMC status", y = "Mean coverage") +
  theme_standard +
  theme(legend.position = "none")
```



6. Export tidy tables

```
# 1) Main condition-level summary (with design joined)
export_cond <- cond |>
  dplyr::select(condition_id, Model, sem_study, direction, rho, T, param,
                N_valid, N_truth_avail, mean_rel_bias, coverage_95, RMSE,
                mean_post_sd, emp_sd, sd_bias, mean_n_div, prop_div, mean_rhat)

readr::write_csv(export_cond, file.path(EXPORT_DIR, "sem_analysis_conditions.csv"))

# 2) Coverage by MCMC status (core only)
export_status <- cond_status |>
  dplyr::select(Model, sem_study, direction, rho, T, param, mcmc_status,
                N_valid, mean_rel_bias, coverage_95, RMSE, mean_post_sd, emp_sd, sd_bias)

readr::write_csv(export_status, file.path(EXPORT_DIR, "sem_analysis_status_split_core.csv"))
```

7. Notes on interpretation

- **Layer-specific copula:** (ρ) is identified at the **active layer**. Fitting the “wrong” model (EI on EL data, or EL on EI data) can distort the PIT on that layer and bias ($\hat{\rho}$) toward zero (attenuation).
- **One-sided support:** exponential margins imply hard bounds (e.g., right-skew requires (e $\geq$ -s)). Chains that frequently propose off-support values tend to report divergences; nevertheless, coverage for (Φ) and (μ) can remain adequate if mixing elsewhere is good.
- **Scales:** both EI and EL estimate ($\sigma_{\text{exp}}[j]$) with truth (=1); bias/coverage for these parameters help diagnose whether the model is matching the marginal one-sidedness.