Web Table 1: Description of indirect/direct approaches for applying SIMEX to MSMs in single time interval setting

λ = A vector of values used to control artificial error, e.g. {0.5, 1.0, 1.5, 2.0}
X = A binary indicator variable of exposure status
Y = A continuous response variable
L = An error-free (“true”) continuous time-varying covariate
*L** = An error-prone (“observed”) continuous time-varying covariate, i.e. *L** = *L* + error
ε = Random error
σ_δ^2 = Variance of measurement error

Treatment model: $\text{logit}\left(P\left(X=1\middle|L^*\right)\right)=\alpha_0+\alpha_1 L^*$ Outcome model: $Y=\beta_0+\beta_1 X+\varepsilon$

Note: We describe a single time interval setting; the same general SIMEX approach is appropriate given two or more intervals.

i. Direct Approach

1) Add additional (artificial) error to time-varying covariate *L** from normal distribution with mean of zero and variance $\lambda\times\sigma_\delta^2$. Compute IPT-weighted estimates using *L** with additional error. Estimate the MSM parameter $\hat{\beta}$.
2) Repeat step (1) a total of *B* times, and then average all *B* estimates to obtain $\overline{\hat{\beta}(\lambda)}$.
3) Repeat steps (1) and (2) for all values of λ.
4) Fit chosen regression function $f\left(\hat{\beta}(\lambda)\right)$, then extrapolate back to $\lambda=-1$ to obtain corrected $\overline{\hat{\beta}}^{SIMEX}(\lambda=-1)$.

ii. Indirect Approach

1) Add additional error to *L** from normal distribution with μ = 0 and variance = $\lambda\times\sigma_\delta^2$. Estimate treatment model parameter $\hat{\alpha}$.
2) Repeat step (1) *B* times, and average all *B* estimates to obtain $\overline{\hat{\alpha}}(\lambda)$.
3) Repeat steps (1) and (2) for all values of λ. Compute $\overline{\hat{\alpha}}(\lambda)$ and obtain $\overline{\hat{\alpha}}^{SIMEX}(\lambda=-1)$.
4) Fit chosen regression function $f\left(\hat{\alpha}(\lambda)\right)$, then extrapolate back to $\lambda=-1$ to obtain corrected $\overline{\hat{\alpha}}^{SIMEX}(\lambda=-1)$.
5) Use the corrected treatment parameter estimates obtained in step 4, to compute $\hat{w}^{SIMEX}$ as $\text{logit}^{-1}\left(\overline{\hat{\alpha}}_0^{SIMEX}+\overline{\hat{\alpha}}_1^{SIMEX}L^*\right)$ for the treated, and $\left(1\text{-logit}\left(\overline{\hat{\alpha}}_0^{SIMEX}+\overline{\hat{\alpha}}_1^{SIMEX}L^*\right)\right)^{-1}$ for the untreated (stabilizing if desired).
6) Use $\hat{w}^{SIMEX}$ as the weights in a regression model to obtain MSM parameter estimates.

Data-generating model for primary studies

i. One-interval study (corresponds to Figure 1a)		*ii. Two-interval study (corresponds to Figure 1b)*[‡]	
L	$\sim$ *Uniform*[-2, 3]	L_1	$\sim$ *Uniform*[-2, 3]
L^*	$\sim L + Norm(0,\ \sigma_\delta^2)$	L_1^*	$\sim L_1 + Norm(0,\ \sigma_\delta^2)$
X	$\sim Bern < \mathrm{logit}^{-1}(0.25 + \alpha_1 L)$	X_1	$\sim Bern < \mathrm{logit}^{-1}(0.25 + \alpha_1 L_1)$
	where $\alpha_1 \in \{0.25, 0.50, 0.75\}$		*where* $\alpha_1 \in \{0.50, 0.75\}$
Y	$\sim 0 + 1.0X + 1.25L + Norm(0, 1)$	L_2	$\sim 0.3L_1 + 1.25X_1 + Norm(0, 1)$
		L_2^*	$\sim L_2 + Norm(0,\ \sigma_\delta^2)$
		X_2	$\sim Bern < \mathrm{logit}^{-1}(0.25 + \alpha_1 L_2 -1.25X_1)$
			where $\alpha_1 \in \{0.50, 0.75\}$
		Y	$\sim 0 + 1.0X_1 + 1.25X_2 + 0.8L_1 + 0.3L_2 + Norm(0, 1)$

Summary of fitted models

i. One-interval study	*First interval model for X*
Treatment (denominator)	$\mathrm{logit}(P(X = 1 \mid L^*)) = \alpha_0 + \alpha_1 L^*$
Treatment (numerator)	1
Outcome	$Y = \beta_0 + \beta_1 X + \varepsilon,\ \varepsilon \sim N(0,1)$

ii. Two-interval study[‡]	*First interval model for X_1*	*Second interval model for X_2*
Treatment (denominator)	$\mathrm{logit}(P(X_1 = 1 \mid L_1^*)) = \alpha_0 + \alpha_1 L_1^*$	$\mathrm{logit}(P(X_2 = 1 \mid X_1 + L_2^*)) = \alpha_0 + \alpha_1 X_1 + \alpha_2 L_2^*$
Treatment (numerator)	$P(X_1 = 1) = \mathrm{Mean}(X_1\ )$	$\mathrm{logit}(P(X_2 = 1 \mid X_1)) = \alpha_0 + \alpha_1 X_1^*$
Outcome	$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \eta,\ \eta \sim N(0,1)$	

[‡] For the secondary study (scenario corresponding to Figure 1c) in which exposure depends upon the error-prone covariate, X_1 was generated using L_1^* (instead of L_1), and X_2 depended on L_2^* (instead of L_2).

Web Appendix 1: Reasoning for extrapolation to λ = -1

In the presence of additive error, the variance of the observed covariate L^* is equivalent to the sum of the variances of the true covariate L and its measurement error. The expression below illustrates this relationship (measurement error denoted by δ):

$$Var\left(L^*\right) = \sigma_L^2 + \sigma_\delta^2 + \left(\lambda \times \sigma_\delta^2\right)$$

It can be seen that this introduces bias most plainly in the linear regression setting, where the estimator is available in closed form.

Consider, for example, a simple linear regression of *Y* on *L**; the unbiased regression parameter estimates are given by θ = Cov(*Y*,*L*)/Var(*L*) but, in the presence of measurement error, estimated by the empirical covariance of *Y* and *L** (which is consistent for Cov(*Y*,*L*)) divided by the empirical variance of *L**, which is larger than the variance of *L* (by exactly σ_δ^2). Thus, substituting -1 for the value of λ will cancel out the two measurement error variance terms, and the variance of the error-prone covariate is now equivalent to that of the true (unobserved) covariate.

Consequently, extrapolating f(λ) to -1 yields the SIMEX estimator for the covariate in the absence of measurement error. While this result cannot be shown in closed form for estimators outside a linear regression setting, SIMEX has been used extensively in a variety of other applications (1-5).

Web Appendix 2: R code for direct and indirect SIMEX correction

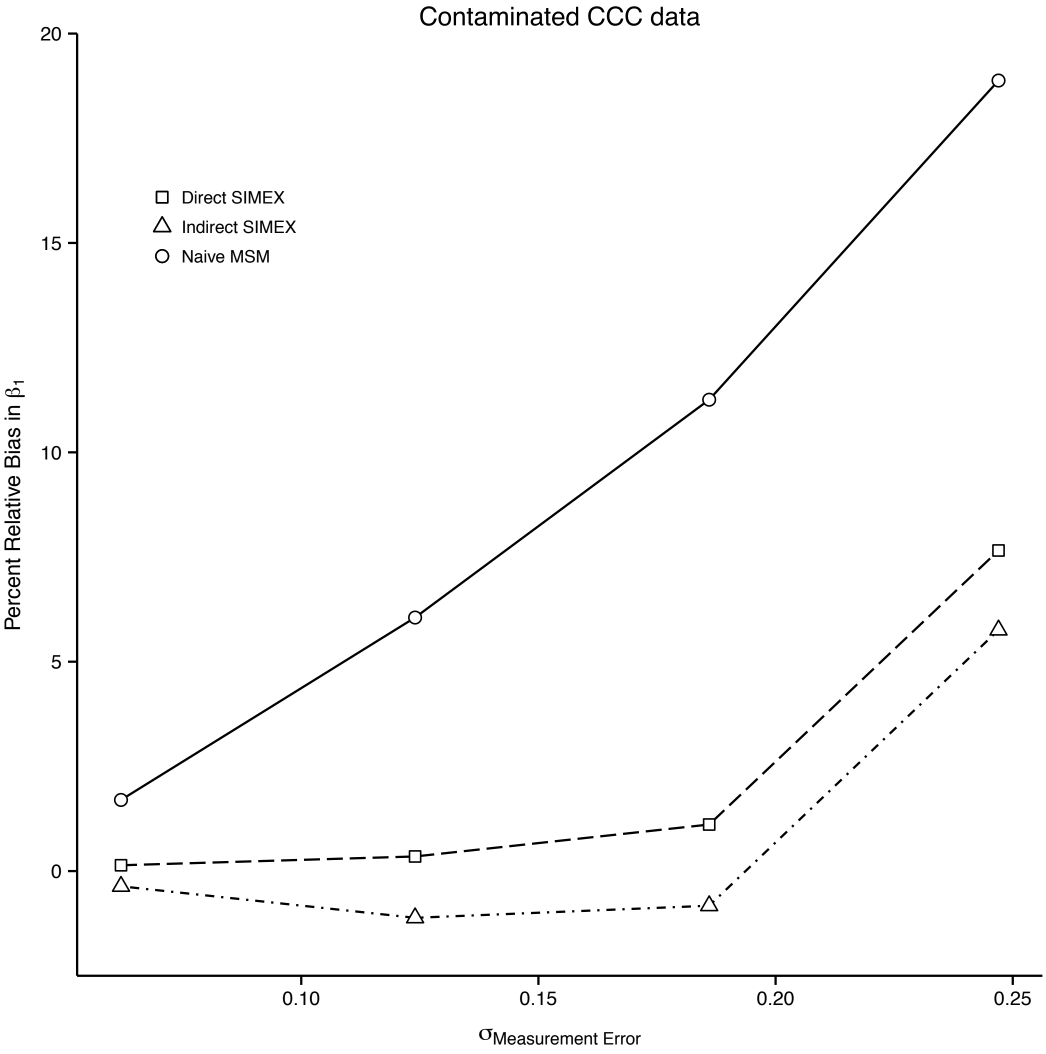
```
# Generate B datasets for SIMEX adjustment in apply_simex function
# (requires foreach package)

simex_data <- foreach(i = sqrt(lambda)) %do% replicate(B,
            transform(indata, Lstar1 = Lstar1 + rnorm(n = nrow(indata),
                mean = 0, sd = i * specerr), Lstar2 = Lstar2 +
                rnorm(n = nrow(indata), mean = 0, sd = i * specerr)))

# Apply both direct and indirect SIMEX methods to address measurement error
# N.B. Implementation of the indirect SIMEX using R’s simex package is far
#      more convenient; however, we wished to obtain direct SIMEX results,
#      while reducing the execution time. For reference, our code is included
#      below.

apply_simex <- function (simex_data, lambda, B)
{
    numlam <- length(lambda)
    outcome_model_params <- replicate(numlam, list(matrix(data = NA,
        ncol = B, nrow = 3)))
    treatment_t1_params <- replicate(numlam, list(matrix(data = NA,
        ncol = B, nrow = 2)))
    treatment_t2_params <- replicate(numlam, list(matrix(data = NA,
        ncol = B, nrow = 3)))
    y_x1mat <- simex_data[[1]][["X1", 1]]
    y_x2mat <- simex_data[[1]][["X2", 1]]
    y_ymat <- simex_data[[1]][["Y", 1]]
    x_x <- cbind(1, X1 = simex_data[[1]][["X1", 1]], X2 = simex_data[[1]][["X2", 1]])
    for (i in 1:numlam) {
        for (j in 1:B) {
            x_lstar1 <- cbind(1, Lstar1 = simex_data[[i]][[c("Lstar1"),
                j]])
            x_lstar2n <- cbind(1, X1 = simex_data[[i]][["X1",
                j]])
            x_lstar2 <- cbind(1, X1 = simex_data[[i]][["X1",
                j]], Lstar2 = simex_data[[i]][["Lstar2", j]])
            model_err1 <- glm.fit(x = x_lstar1, y = y_x1mat,
            family = binomial())
        pred.model_err1 <- ifelse(y_x1mat == 1, model_err1$fitted.values,
            1 - model_err1$fitted.values)
        mod_werr1 <- (ifelse(simex_data[[i]][["X1", j]] ==
            1, (mean(simex_data[[i]][["X1", j]] == 1))/(pred.model_err1),
            (1 - mean(simex_data[[i]][["X1", j]] == 1))/(pred.model_err1)))
        model_err2n <- glm.fit(x = x_lstar2n, y = y_x2mat,
            family = binomial())
        model_err2 <- glm.fit(x = x_lstar2, y = y_x2mat,
            family = binomial())
        pred.model_err2n <- ifelse(y_x2mat == 1, model_err2n$fitted.values,
            1 - model_err2n$fitted.values)
        pred.model_err2 <- ifelse(y_x2mat == 1, model_err2$fitted.values,
            1 - model_err2$fitted.values)
        mod_werr2 <- (pred.model_err2n/(pred.model_err2))
        comb_werr <- mod_werr1 * mod_werr2
        model_werr <- lm.wfit(x = x_x, y = y_ymat, w = comb_werr)
        outcome_model_params[[i]][, j] <- model_werr$coeff
        treatment_t1_params[[i]][, j] <- model_err1$coeff
        treatment_t2_params[[i]][, j] <- model_err2$coeff
    }
}
lambda_matrix <- cbind(1, lambda + 1, ((lambda + 1)^2))
simex_dir_coefs_list <- lapply(outcome_model_params, rowMeans)
simex_dir_coefs_mat <- do.call(rbind, simex_dir_coefs_list)
simex_dir_int <- simex_dir_coefs_mat[, 1]
simex_dir_x1 <- simex_dir_coefs_mat[, 2]
simex_dir_x2 <- simex_dir_coefs_mat[, 3]
simex_dir_int_adj <- lm.fit(x = lambda_matrix, y = simex_dir_int)
simex_dir_x1_adj <- lm.fit(x = lambda_matrix, y = simex_dir_x1)
simex_dir_x2_adj <- lm.fit(x = lambda_matrix, y = simex_dir_x2)
simex_ind_t1_coefs_list <- lapply(treatment_t1_params, rowMeans)
simex_ind_t1_coefs_mat <- do.call(rbind, simex_ind_t1_coefs_list)
simex_ind_t1_int <- simex_ind_t1_coefs_mat[, 1]
simex_ind_t1_Lstar1 <- simex_ind_t1_coefs_mat[, 2]
simex_ind_t2_coefs_list <- lapply(treatment_t2_params, rowMeans)
simex_ind_t2_coefs_mat <- do.call(rbind, simex_ind_t2_coefs_list)
simex_ind_t2_int <- simex_ind_t2_coefs_mat[, 1]
```

```
    simex_ind_t2_x1 <- simex_ind_t2_coefs_mat[, 2]
    simex_ind_t2_Lstar2 <- simex_ind_t2_coefs_mat[, 3]
    simex_ind_t1_int_adj <- lm.fit(x = lambda_matrix, y = simex_ind_t1_int)
    simex_ind_t1_Lstar1_adj <- lm.fit(x = lambda_matrix, y = simex_ind_t1_Lstar1)
    simex_ind_t2_int_adj <- lm.fit(x = lambda_matrix, y = simex_ind_t2_int)
    simex_ind_t2_x1_adj <- lm.fit(x = lambda_matrix, y = simex_ind_t2_x1)
    simex_ind_t2_Lstar2_adj <- lm.fit(x = lambda_matrix, y = simex_ind_t2_Lstar2)
    x_lstar1 <- cbind(1, Lstar1 = simex_data[[1]][[c("Lstar1"),
        1]])
    x_lstar2 <- cbind(1, X1 = simex_data[[1]][["X1", 1]], Lstar2 =
simex_data[[1]][["Lstar2",
        1]])
    x_lstar2n <- cbind(1, X1 = simex_data[[1]][["X1", 1]])
    corrected.wts1 <- ifelse(y_x1mat == 1, (mean(simex_data[[1]][["X1",
        1]] == 1))/inv.logit(x_lstar1 %*% rbind(simex_ind_t1_int_adj$coeff[1],
        simex_ind_t1_Lstar1_adj$coeff[1])), (1 - mean(simex_data[[1]][["X1",
        1]] == 1))/(1 - inv.logit(x_lstar1 %*% rbind(simex_ind_t1_int_adj$coeff[1],
        simex_ind_t1_Lstar1_adj$coeff[1]))))
    cw2.numerator <- ifelse(y_x2mat == 1, model_err2n$fitted.values,
        1 - model_err2n$fitted.values)
    cw2.denominator <- ifelse(y_x2mat == 1, inv.logit(x_lstar2 %*%
        rbind(simex_ind_t2_int_adj$coeff[1], simex_ind_t2_x1_adj$coeff[1],
            simex_ind_t2_Lstar2_adj$coeff[1])), 1 - (inv.logit(x_lstar2 %*%
        rbind(simex_ind_t2_int_adj$coeff[1], simex_ind_t2_x1_adj$coeff[1],
            simex_ind_t2_Lstar2_adj$coeff[1]))))
    corrected.wts2 <- cw2.numerator/cw2.denominator
    comb_werr <- corrected.wts1 * corrected.wts2
    ind_msm <- lm.wfit(x = x_x, y = y_ymat, w = comb_werr)
    return(list(int_dir = simex_dir_int_adj$coeff[1], x1_dir =
simex_dir_x1_adj$coeff[1],
        x2_dir = simex_dir_x2_adj$coeff[1], int_ind = ind_msm$coeff[1],
        x1_ind = ind_msm$coeff[2], x2_ind = ind_msm$coeff[3]))
}
```

Web Appendix 3: Additional analyses comparing direct and indirect SIMEX

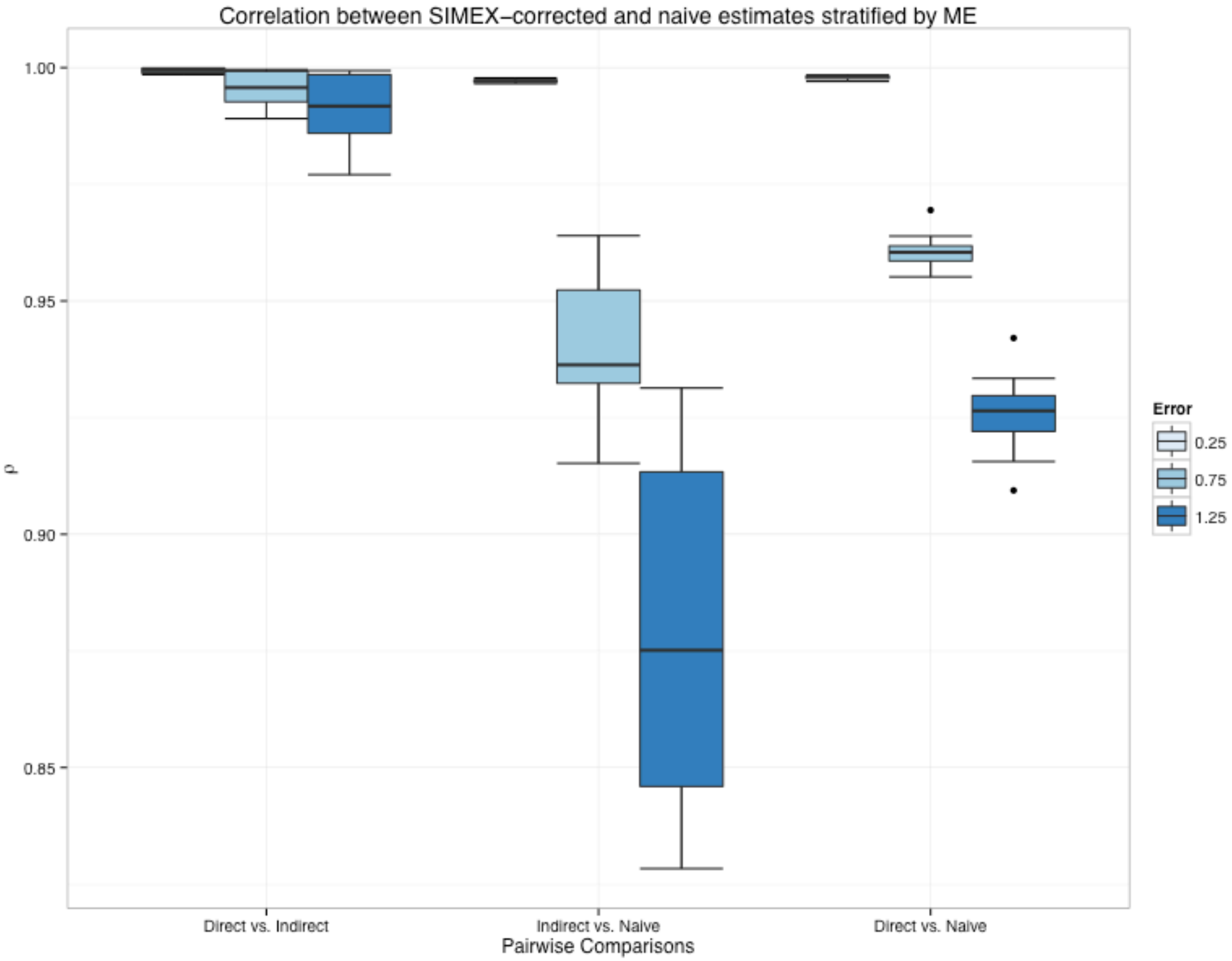
Because there is no closed form solution for MSM estimators found using a SIMEX correction, whether direct or indirect, a theoretical proof of unbiasedness is infeasible. This is the case for adaptations of SIMEX to generalized linear models (3), non-linear regression models (2), quantile regression models (5), accelerated failure time models (9), and generalized linear mixed models (1). We describe additional analyses we conducted to explore and compare performance of both SIMEX approaches (Web Figures 1 and 2), and our choice of extrapolant function (Web Figure 3).

Web Figure 1: *Simulations using Canadian Co-infection Cohort data (2003-2014).* We explored the impact of increasing measurement error on the performance of the two approaches using simulations based on a real dataset, which we also used in the case study. Here we assume that the naïve model from that analysis is the "true" model, and contaminate the measures of GGT with random error drawn from an *N*(0, σ_δ) distribution, with standard deviations σ_δ equal to 0.25x, 0.5x, 0.75x and 1 x 0.247, the measurement error of GGT applied in our previous analysis. Estimates are based on 300 datasets for each specified measurement error variance.



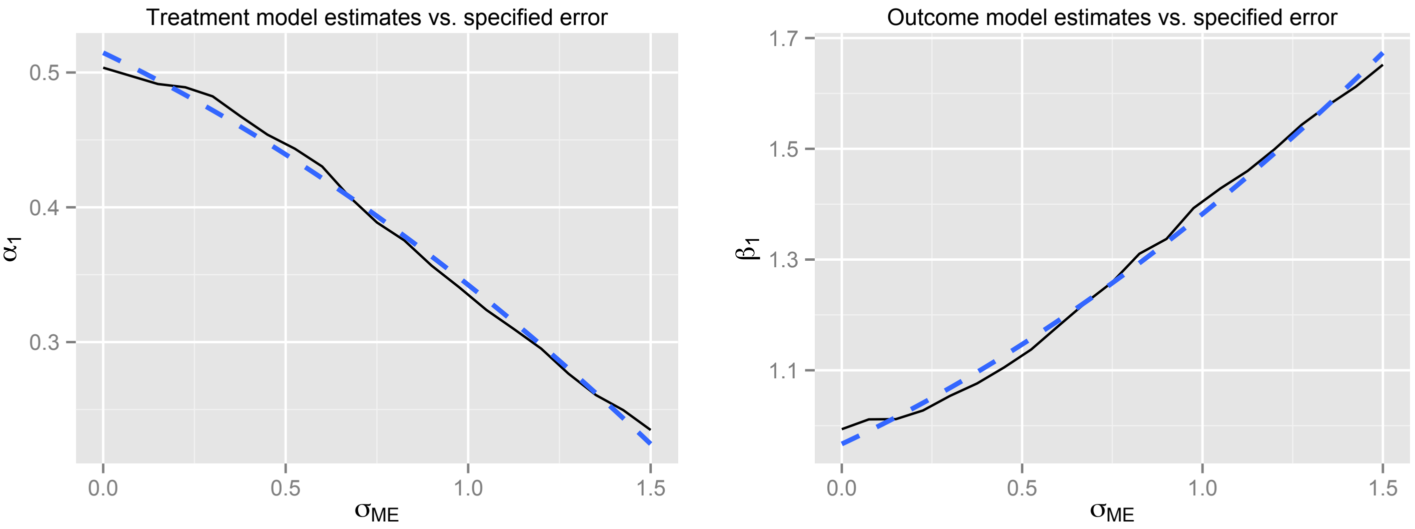
Given the figure above, the results for the two SIMEX approaches remain nearly unbiased when the ratio of the specified versus original measurement error variance is less than 1.0, while the naïve MSM exhibits up to 20% bias relative to the “true” model.

Web Figure 2: *Pairwise correlations between SIMEX and naïve estimates.* We computed pairwise correlations for the direct and indirect SIMEX, indirect SIMEX and naïve MSM, and direct SIMEX and naïve MSM across all 27 scenarios described in Table 1 and Supplementary Table 1c. As seen in the figure below, correlation between direct and indirect SIMEX estimates remained consistently high (ρ ≥ 0.95) regardless of the degree of specified measurement error.



Web Figure 3: *Evaluating the accuracy of the quadratic extrapolation.* To better understand the impact of measurement error on estimates from the naïve model (without any SIMEX correction), we modeled the functional relationship between $\beta(\lambda)$ and λ for the direct SIMEX, and $\alpha(\lambda)$ and λ for the indirect SIMEX. Given 21 gradually decreasing values of measurement error standard deviation (ranging from 1.50 to 0.00), we produced plots for $\beta(\lambda)$ from the outcome model and $\alpha(\lambda)$ from the treatment model versus λ. For each round of simulations, we generated 100 datasets containing 5000 observations each, with α_0 = 0.25, and α_1 = 0.50.

The dotted line on each plot shows the fit provided by a quadratic smoother. A comparison of the two fitted lines suggests that while the quadratic extrapolant may not provide a perfect fit, it offers a reasonable approximation to the observed relationship.



Web Table 2: *Single time interval simulation results for error-free L and error-prone L**

Scenario				**Model using error-free *L***				**Model using error-prone *L****			
	Values			Bias	MSE	MCSE	95% Boot Cov.	Bias	MSE	MCSE	95% Boot Cov.
	N	α_1	σ_δ	β_1				β_1			
A	100	0.25	0.25	-0.003	0.042	0.206	0.953	0.012	0.048	0.219	0.950
B			0.75					0.127	0.089	0.271	0.930
C			1.25					0.267	0.173	0.319	0.883
D	500		0.25	0.008	0.009	0.095	0.930	0.026	0.010	0.099	0.937
E			0.75					0.145	0.036	0.123	0.800
F			1.25					0.286	0.104	0.149	0.453
G	1000		0.25	0.003	0.005	0.071	0.923	0.020	0.006	0.075	0.927
H			0.75					0.135	0.027	0.092	0.623
I			1.25					0.272	0.085	0.106	0.223
J	100	0.50	0.25	-0.006	0.057	0.239	0.953	0.029	0.062	0.248	0.957
K			0.75					0.253	0.148	0.290	0.890
L			1.25					0.528	0.387	0.330	0.667
M	500		0.25	0.006	0.013	0.113	0.920	0.039	0.015	0.116	0.917
N			0.75					0.260	0.085	0.134	0.457
O			1.25					0.534	0.309	0.153	0.077
P	1000		0.25	-0.001	0.006	0.08	0.920	0.031	0.008	0.084	0.917
Q			0.75					0.250	0.073	0.099	0.217
R			1.25					0.524	0.287	0.112	0.003
S	100	0.75	0.25	0.022	0.105	0.323	0.923	0.071	0.111	0.326	0.920
T			0.75					0.383	0.274	0.357	0.763
U			1.25					0.778	0.741	0.370	0.443
V	500		0.25	0.002	0.020	0.14	0.930	0.044	0.022	0.143	0.887
W			0.75					0.355	0.151	0.156	0.343
X			1.25					0.754	0.597	0.167	0.017
Y	1000		0.25	0.003	0.009	0.096	0.940	0.045	0.012	0.099	0.907
Z			0.75					0.354	0.137	0.111	0.113
AA			1.25					0.751	0.578	0.120	0.000

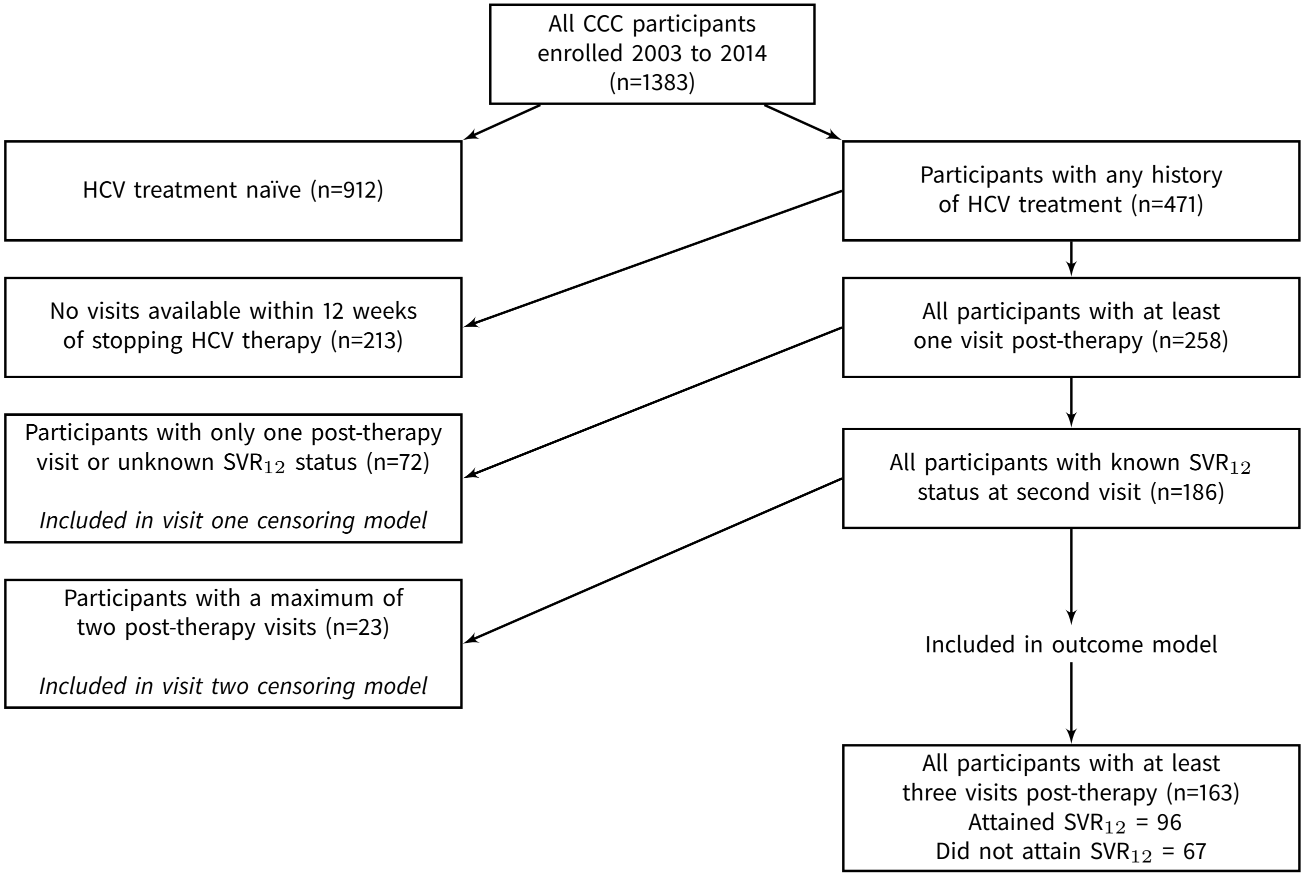
Web Table 3: *Summary of two-interval simulation study results for four scenarios (rows), varying the coefficient for the confounder L (α_1), and the standard deviation of measurement error σ_δ for models including an error-free L and uncorrected error-prone L*. Bootstrap coverage was obtained using the percentiles approach. Mean squared error abbreviated as MSE, Monte Carlo standard error abbreviated as MCSE. Sample size of N = 1000 used for all simulations.*

Scenario			**Model using error-free L_1, L_2**							**Model using error-prone L_1^*, L_2^***						
	Values		Bias		MSE		MCSE		95% Bootstrap Coverage	Bias		MSE		MCSE		95% Bootstrap Coverage
	α_1	σ_δ	β_1	β_2	β_1	β_2	β_1	β_2	β_1, β_2	β_1	β_2	β_1	β_2	β_1	β_2	β_1, β_2
1	0.50	0.25	-0.004	0.009	0.009	0.011	0.094	0.106	0.910, 0.973	0.025	0.03	0.009	0.012	0.094	0.106	0.917, 0.957
2		1.25	-0.004	0.009	0.009	0.011	0.094	0.106	0.907, 0.970	0.411	0.238	0.179	0.067	0.097	0.103	0.010, 0.400
3	0.75	0.25	0.006	0.013	0.014	0.018	0.12	0.135	0.930, 0.927	0.045	0.044	0.016	0.02	0.121	0.133	0.903, 0.927
4		1.25	0.001	-0.001	0.014	0.019	0.118	0.138	0.940, 0.947	0.561	0.319	0.325	0.115	0.098	0.116	0.000, 0.233

Web Appendix 4: Additional details related to case study analyses of CCC data

We consulted trace line plots to assess convergence of imputation models, given 10 iterations and 30 imputations per iteration; we specified these values based on advice provided by Bodner (6). We imputed missing values for continuous variables using predictive mean matching, and applied logistic regression to impute missing indicators of treatment status, as suggested by van Buuren et al. (7, 8). Censoring models included all time-varying covariates except for GGT, which did not strongly predict censoring.

Web Figure 4: *Participant flow diagram for Case Study using Canadian Co-infection Cohort Study data*


All CCC participants enrolled 2003 to 2014 (n=1383)
HCV treatment naïve (n=912)
Participants with any history of HCV treatment (n=471)
No visits available within 12 weeks of stopping HCV therapy (n=213)
All participants with at least one visit post-therapy (n=258)
Participants with only one post-therapy visit or unknown SVR_{12} status (n=72)
Included in visit one censoring model
All participants with known SVR_{12} status at second visit (n=186)
Participants with a maximum of two post-therapy visits (n=23)
Included in visit two censoring model
Included in outcome model
All participants with at least three visits post-therapy (n=163)
Attained SVR_{12} = 96
Did not attain SVR_{12} = 67

Web Table 4: *Characteristics of 186 participants in the Canadian Co-infection Cohort Study (2003-2014) with at least one visit following hepatitis C virus (HCV) therapy and known sustained virologic response status at second visit*

Variable	**SVR attained at 12 weeks**		***P* value**[†]
	Yes **(n = 111)**	**No** **(n = 75)**	
Age in years (mean (SD))	45.85 (8.72)	47.27 (7.90)	0.25
Female (%)	18 (16)	10 (13)	0.59
Duration of HCV infection in years (mean (SD))	16.52 (11.10)	19.79 (10.52)	0.04
HCV genotype of 2/3/4 (%)	42 (38)	18 (24)	0.048

[†] Computed from *t*-test for continuous variables, and chi-square test for categorical variables.

Baseline characteristics and standardized differences prior to and subsequent to weighting

Web Table 5: *Comparison of baseline characteristics between those attaining or not attaining SVR12 prior to weighting, and subsequent to weighting using naïve model with quadratic terms for GGT and HIV RNA levels.*

	Prior to weighting			**Weighted using naïve weights, with quadratic $\log_{10}$(GGT), $\log_{10}$(HIV RNA) terms**		
	SVR_{12}			SVR_{12}		
	No	Yes	Standardized difference	No	Yes	Standardized difference
Variable	N=75	N=111		N=181.63	N=188.79	
$\log_{10}$(GGT)	1.94 (0.36)	1.58 (0.35)	1.01	1.74 (0.39)	1.73 (0.40)	<0.01
BMI	24.84 (3.63)	25.83 (6.85)	0.18	24.79 (3.41)	25.17 (5.83)	0.08
$\log_{10}$(HIV RNA)	2.03 (0.99)	1.87 (0.80)	0.18	1.96 (1.40)	2.14 (1.43)	0.09
Time since stopping HCV therapy	125.41 (146.57)	106.44 (123.56)	0.14	101.10 (10.06)	106.92 (10.34)	0.05
IDU in past 6 months	14 (19%)	15 (20%)	0.14	24.41 (13%)	26.57 (14%)	0.02
Any alcohol use in past 6 months	36 (48%)	62 (56%)	0.16	79.51 (44%)	97.79 (52%)	0.16
Currently receiving HIV anti-retroviral therapy	67 (89%)	101 (91%)	0.06	157.72 (87%)	160.93 (85%)	0.05

N.B. Mean (standard deviation) reported for continuous variables, while N (%) is reported for dichotomous variables.

Web Table 6: *Distribution of missingness prior to imputation. Percentages based on N of uncensored individuals at each visit.*

	HCV genotype	**γ-glutamyl transferase**		**BMI**		**HIV RNA**		**Any IDU in past 6 months**		**Any alcohol use in past 6 months**		**Currently on anti-retroviral therapy**		**APRI**
		Visit 1	*Visit 2*	*Visit 1*	*Visit 2*	*Visit 1*	*Visit 2*	*Visit 1*	*Visit 2*	*Visit 1*	*Visit 2*	*Visit 1*	*Visit 2*	*Visit 3*
Missing	22	22	57	25	60	11	43	2	35	6	43	0	34	7
(%)	(9%)	(9%)	(31%)	(10%)	(32%)	(4%)	(23%)	(1%)	(19%)	(2%)	(23%)	(0%)	(18%)	(4%)

N.B. We did not impute SVR_{12}; for reference, the number of participants with missing SVR status is available in Web Figure 4 above.

Web References

1. Wang N, Lin X, Gutierrez RG, Carroll RJ. Bias Analysis and SIMEX Approach in Generalized Linear Mixed Measurement Error Models. J Am Stat Assoc. 1998;93(441):249-61.
2. Carroll RJ, Küchenhoff H, Lombard F, Stefanski LA. Asymptotics for the SIMEX Estimator in Nonlinear Measurement Error Models. J Am Stat Assoc. 1996;91(433):242-50.
3. Li Y, Lin X. Functional Inference in Frailty Measurement Error Models for Clustered Survival Data Using the SIMEX Approach. J Am Stat Assoc. 2003;98(461):191-203.
4. Kim J, Gleser LJ. SIMEX approaches to measurement error in ROC studies. Communications in Statistics: Theory and Methods. 2000;29(11):2473-91.
5. Shang Y. Measurement Error Adjustment Using the SIMEX Method: An Application to Student Growth Percentiles. Journal of Educational Measurement. 2012;49(4):446-65.
6. Bodner TE. What improves with increased missing data imputations? Structural Equation Modeling. 2008;15(4):651-75.
7. Buuren SV, Groothuis-Oudshoorn K. Multivariate imputation by chained equations in R. J Stat Softw. 2011;45(3):1-67.
8. van Buuren S. Flexible imputation of missing data. Boca Raton, Florida: Chapman & Hall/CRC; 2012.
9. He W, Yi GY, Xiong J. Accelerated failure time models with covariates subject to measurement error. Stat Med. 2007;26:4817-32.