Design Matrix (Collinearity):

- `rho<-tanh(seq(from=-1.8,to=0, length.out=20))`
- `n_obs<-c(5, 31, 400)`
- Distribution of $\boldsymbol{X}$ (normal) `X<-mvtnorm::rmvnorm(...)`
- `mean_x1<-0.6; mean_x2<-3.8`
- `sd_x1<-0.1; sd_x2<-2.1`
- Number of simulations B`<-1211`

Aim:
How does collinearity influence the output $(\hat{\boldsymbol{t}}, \hat{\boldsymbol{\beta}}, \text{se}(\hat{\boldsymbol{\beta}}), \text{power})$ of `lm`, `tram::Lm` ?

Regression parameters:

- `beta_0<-c(51.4)`
- `beta_1<-c(-46.1, 0)`
- `beta_2<-c(-0.9, 0)`

Noise parameters:

- `set.seed(...)`
- `eps_y<-rnorm(0,1,n=n_obs)`
- `s_y<-c(2, 5, 8.2)`

True linear model

$$\mathbf{X} \cdot \beta + \boldsymbol{\varepsilon_y} \cdot s_y = \boldsymbol{y}$$

Conclusions

Data-Generating-Process:

```
for(k in 1:B){
  for(i in 1:nrow(experimental_factors)){
    X<-mvtnorm::rmvnorm(n = n_obs[i],
      mean = c(mean_x1,mean_x2),
      sigma = matrix(c(
      sd_x1^2, rho[i]*sd_x1*sd_x2,
      rho[i]*sd_x1*sd_x2, sd_x2^2
      ), ncol = 2))
    X<-cbind(1,X)
    cond_nu<-max(
      Collinearity::Var_decom_mat(X)[,"cond_ind"])
    E<-Collinearity::equilibrate_matrix(X)
    trouble<-diag(solve(t(E)%*%E))
    eps_y<-rnorm(0,1,n=n_obs[i])
    y<-X%*%c(beta_0[i],beta_1[i],beta_2[i])+eps_y*s_y[i]
    df_list[[length(df_list)+1]]<-data.frame(y,X[,-1])
  }
}
```

Performance Evaluation:
Metrics:

- Wald statistics $\hat{\boldsymbol{t}}$
- Bias $\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}$
- Relative Bias $\left(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}\right) / \boldsymbol{\beta}$
- Empirical standard error se $\left(\hat{\boldsymbol{\beta}}\right)$
- Proportion of p-values $\leq \alpha$ (Power)

Figures:

- Trace plots with quantiles
- Spatial plots

Experimental factors (full factorial):
`experimental_factors<-expand.grid(...)`

id	n_obs	rho	beta_0	beta_1	beta_2	s_y
1	5	-0.9	51.4	-46.1	-0.9	8.2
1212	31	-0.9	51.4	-46.1	-0.9	8.2
...	...	...	...	...	...	...
301540	5	-0.9	51.4	-46.1	-0.9	2
...	...	...	...	...	...	...
870710	400	0	51.4	0	0	5
...	...	...	...	...	...	...
871920	400	0	51.4	0	0	5

Estimands:

hbeta_0	se_0	t_0	...	cond_nu
-13	366.3	0	...	122.1
9.1	64.6	0.1	...	69.1
...	...	...	...	...
-51.1	21.1	-2.4	...	26.5
...	...	...	...	...
0	2.7	0	...	15.5
...	...	...	...	...
-2.7	2.2	-1.2	...	13.8

Simulated Data
(n = 871920)
`df_list[[1]]`

Estimand-Generating-Process:

```
# paired design
m<-lm(data,y~x1+x2)
m<-tram::Lm(data,y~x1+x2)
```