

Supplemental Material S1. Stan code.

```
// Pearson Correlation
data {
  int<lower=0> n;
  vector[2] x[n];
}
parameters {
  vector[2] mu;
  vector<lower=0>[2] sigma;
  real<lower=-1,upper=1> r;
}
transformed parameters {
  cov_matrix[2] T;
  // Reparameterization
  T[1,1] <- square(sigma[1]);
  T[1,2] <- r * sigma[1] * sigma[2];
  T[2,1] <- r * sigma[1] * sigma[2];
  T[2,2] <- square(sigma[2]);
}
model {
  // Priors
  mu ~ normal(0, 100);
  sigma ~ cauchy(0,3);
  r ~ uniform(-1, 1);
  // Data
  x ~ multi_normal(mu, T);
}
generated quantities {
  vector<lower=-5>[1] bias;
  vector<lower=-5>[1] varErr;
  vector<lower=-5>[1] rmsd;
  vector<lower=-5>[1] diffsigma;
  bias[1] = mu[1] - mu[2];
  varErr[1] = sqrt(T[2,2] + T[1,1] - 2*T[1,2]);
  rmsd[1] = sqrt(bias[1]^2 + varErr[1]^2);
  diffsigma[1] = sigma[1] - sigma[2];
}

myinits <- list(
  list(r=.5, mu=c(1, 2), sigma=c(1, 0.1)),
  list(r=-.5, mu=c(0, -2), sigma=c(.1, 1)),
  list(r=.9, mu=c(.5, -0.5), sigma=c(0.5, 5)),
  list(r=-.9, mu=c(-1, 3), sigma=c(0.001, 3)))
```

parameters monitored:

```
parameters <- c("r", "mu", "sigma", "T", "bias", "varErr", "rmsd", "diffsigma")
```

The following command calls Stan with specific options.

For a detailed description type "?rstan".

```
samples<- stan(model_code=model,  
  data=data,  
  init=myinits, # If not specified, gives random inits  
  pars=parameters,  
  iter=8000,  
  warmup = 3000,  
  chains=4,  
  thin=1,  
  cores = 4,  
  seed = 100  
)
```