

Hi,

Just to play a bit with the [gganimate](https://github.com/thomasp85/gganimate) package, and to celebrate National Coupling Day, the above plot shows different couplings of two univariate Normal distributions, Normal(0,1) and Normal(2,1). That is, each point is a pair (x,y) where x follows a Normal(0,1) and y follows a Normal(2,1). Below I’ll recall briefly how each coupling operates, in the Normal case. The code is available at the end of the post.

To create the plots, x-values were drawn from a Normal(0,1) distribution and kept fixed. Only the y-values change from a coupling to the next.

The optimal transport coupling is the simplest here. The y-values are simply defined by y = x + 2, i.e. they are translated by 2, the difference between the means of Normal(0,1) and Normal(2,1). It leads to the smallest expected value for the distance between y and x, over all couplings of the given Normals. The multivariate Normal case is well explained in [this blog post](http://djalil.chafai.net/blog/2010/04/30/wasserstein-distance-between-two-gaussians/)Ã‚Â byÃ‚Â Djalil ChafaÃƒÂ¯.

The reflection coupling is also simple. Each point is reflected with respect to the midpoint between the means. Here it just amounts to defining y = 2 – x. Reflection couplings can be used to study stochastic processses, for instance multivariate diffusionsÃ‚Â as in [this important 1986 paper by Lindvall and Rogers](https://projecteuclid.org/euclid.aop/1176992442), and [this more recent one by Eberle in 2016](https://link.springer.com/article/10.1007/s00440-015-0673-1).

A variant of the reflection coupling, termed “reflection-maximal” coupling in the above plot, appears in this [recent study of Hamiltonian Monte Carlo](https://arxiv.org/abs/1805.00452)Ã‚Â by Bou-Rabee, Eberle and Zimmer. It’s a maximal coupling with a reflection element to it (see Eq. (15) in that article for a precise definition). Essentially, there is an attempt at accepting x as a draw from a Normal(2,1). If this does not succeed, then y is defined as 2 – x (a reflection). The procedure is a maximal coupling because the probability of the event {x = y} is maximal over all possible couplings.

Finally the coupling referred to as “maximal coupling” in the plot is obtained with the procedure described in [this earlier post](https://statisfaction.wordpress.com/2017/09/06/sampling-from-a-maximal-coupling/). It is also a maximal coupling (these are not unique), but the distribution of y given that “y is not x” is independent of x.

That’s it! The code is below.

library(ggplot2)

library(gganimate)

set.seed(1)

# normal means

mu1 <- 0

mu2 <- 2

# std deviation

sigma <- 1

# number of samples

nsamples <- 1000

# reflection-maximal coupling

reflmax\_samples <- matrix(nrow = nsamples, ncol = 2)

# draw x components first

reflmax\_samples[,1] <- rnorm(nsamples)

# this follows the notation of Bou-Rabee et al, 2018, roughly

z <- mu1 - mu2

normz <- sqrt(sum(z^2))

e <- z / normz

utilde <- runif(nsamples, 0, 1)

accepts <- (log(utilde) < (dnorm(e \* reflmax\_samples[,1] + normz, 0, 1, log = TRUE) - dnorm(e\*reflmax\_samples[,1], log = TRUE)))

eta <- reflmax\_samples[,1]

eta[accepts] <- reflmax\_samples[accepts,1] + z

eta[!accepts]<- reflmax\_samples[!accepts,1] - 2 \* (e \* reflmax\_samples[!accepts,1]) \* e

reflmax\_samples[,2] <- mu2 + eta

df <- data.frame(coupling = rep("reflection-maximal", nsamples), x = reflmax\_samples[,1], y = reflmax\_samples[,2])

# reflection coupling

refl\_samples <- matrix(0, nrow = nsamples, ncol = 2)

refl\_samples[,1] <- reflmax\_samples[,1]

refl\_samples[,2] <- (mu2-mu1)-refl\_samples[,1]

df <- rbind(df, data.frame(coupling = rep("reflection", nsamples), x = refl\_samples[,1], y = refl\_samples[,2]))

# optimal transport coupling

transport\_samples <- matrix(0, nrow = nsamples, ncol = 2)

transport\_samples[,1] <- reflmax\_samples[,1]

transport\_samples[,2] <- mu2 - mu1 + transport\_samples[,1]

df <- rbind(df, data.frame(coupling = rep("optimal transport", nsamples), x = transport\_samples[,1], y = transport\_samples[,2]))

# max coupling

max\_samples <- matrix(0, nrow = nsamples, ncol = 2)

max\_samples[,1] <- reflmax\_samples[,1]

dp <- function(x) dnorm(x, mean = mu1, sd = 1, log = TRUE)

dq <- function(x) dnorm(x, mean = mu2, sd = 1, log = TRUE)

rq <- function(n) rnorm(n, mean = mu2, sd = 1)

for (isample in 1:nsamples){

x <- max\_samples[isample,1]

if (dp(x) + log(runif(1)) < dq(x)){

max\_samples[isample,2] <- x

} else {

reject <- TRUE

y <- NA

while (reject){

y <- rq(1)

reject <- (dq(y) + log(runif(1)) < dp(y))

}

max\_samples[isample,2] <- y

}

}

df <- rbind(df, data.frame(coupling = rep("maximal", nsamples), x = max\_samples[,1], y = max\_samples[,2]))

## Scatter plots and marginals

# ggplot(df, aes(x=x, y=y, group = coupling, colour = factor(coupling))) +

# geom\_point()+

# theme\_minimal() + viridis::scale\_color\_viridis(discrete=T)

#

# ggplot(df, aes(x=x, group = coupling, fill = factor(coupling))) + geom\_histogram(aes(y = ..density..), position = position\_dodge()) +

# theme\_minimal() + viridis::scale\_fill\_viridis(discrete=T)

#

# ggplot(df, aes(x=y, group = coupling, fill = factor(coupling))) + geom\_histogram(aes(y = ..density..), position = position\_dodge()) +

# theme\_minimal() + viridis::scale\_fill\_viridis(discrete=T)

# gganimate

ggplot(df, aes(x = x, y = y)) + xlim(-4, 4) + ylim(-5, 6) +

geom\_point()+ geom\_text(data = data.frame(coupling = unique(df$coupling)), aes(label = coupling, x = -1, y = -4), size = 10) +

theme\_minimal() +

transition\_states(coupling, 3, 1)