

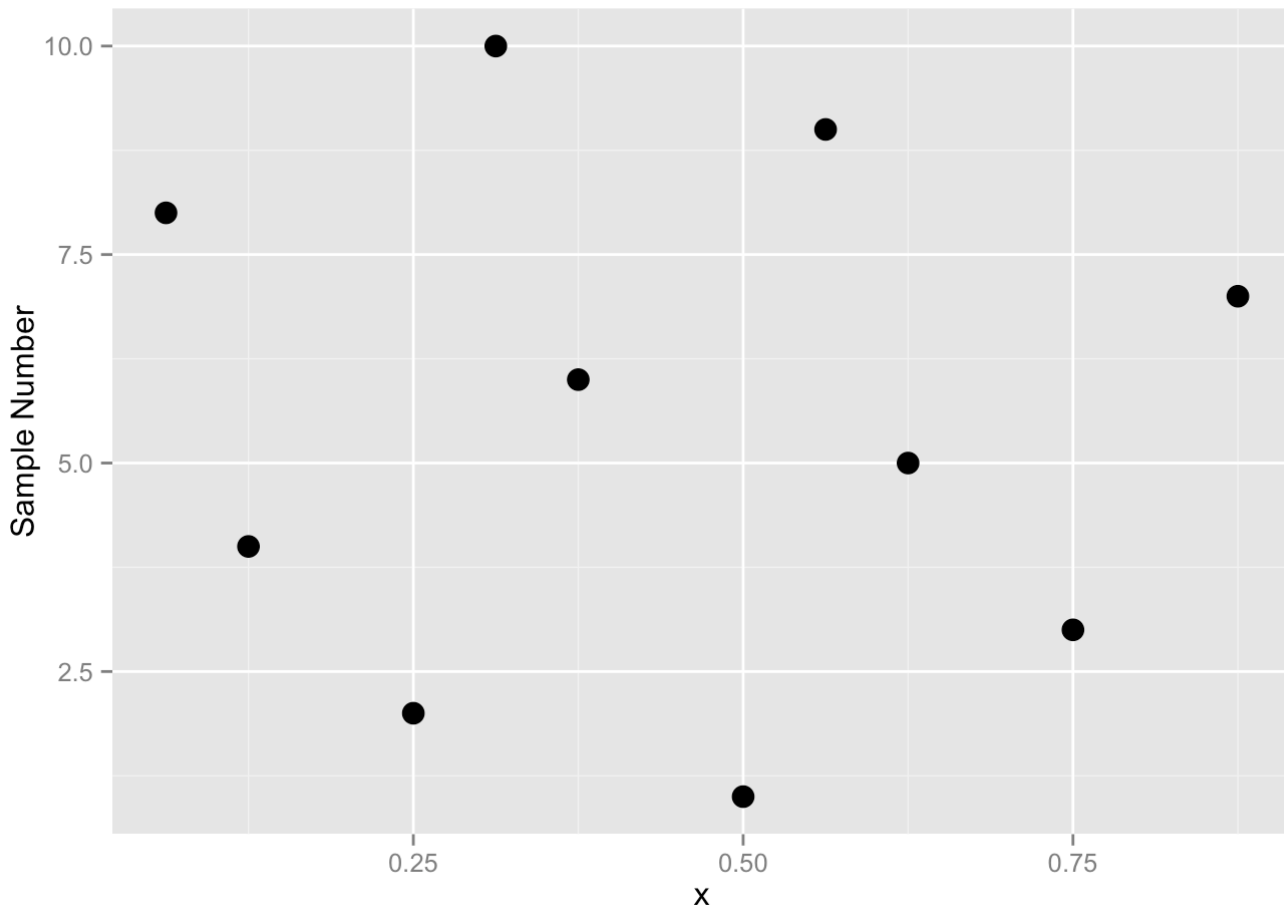
Quasi Monte Carlo

The first example of a QMC is the van Der corput sequence.

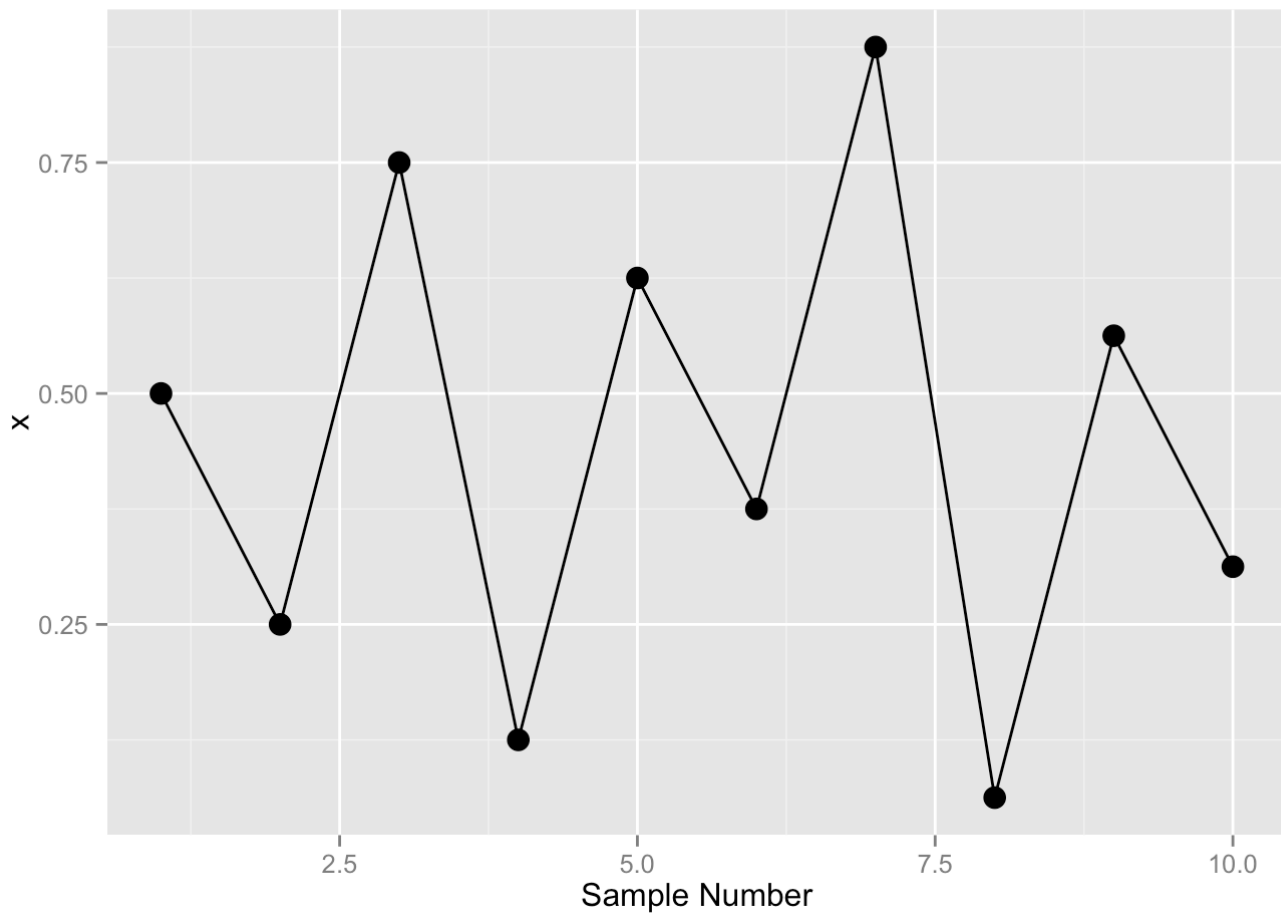
Van der Corput

Let's generate a 10 point van der Corput sequence.

```
npoints = 10  
qmc <- data.frame(j = 1:npoints, x = halton(npoints))  
ggplot(qmc,aes(x=x,y=j)) + geom_point(size=4) + scale_y_continuous('Sample Number')
```



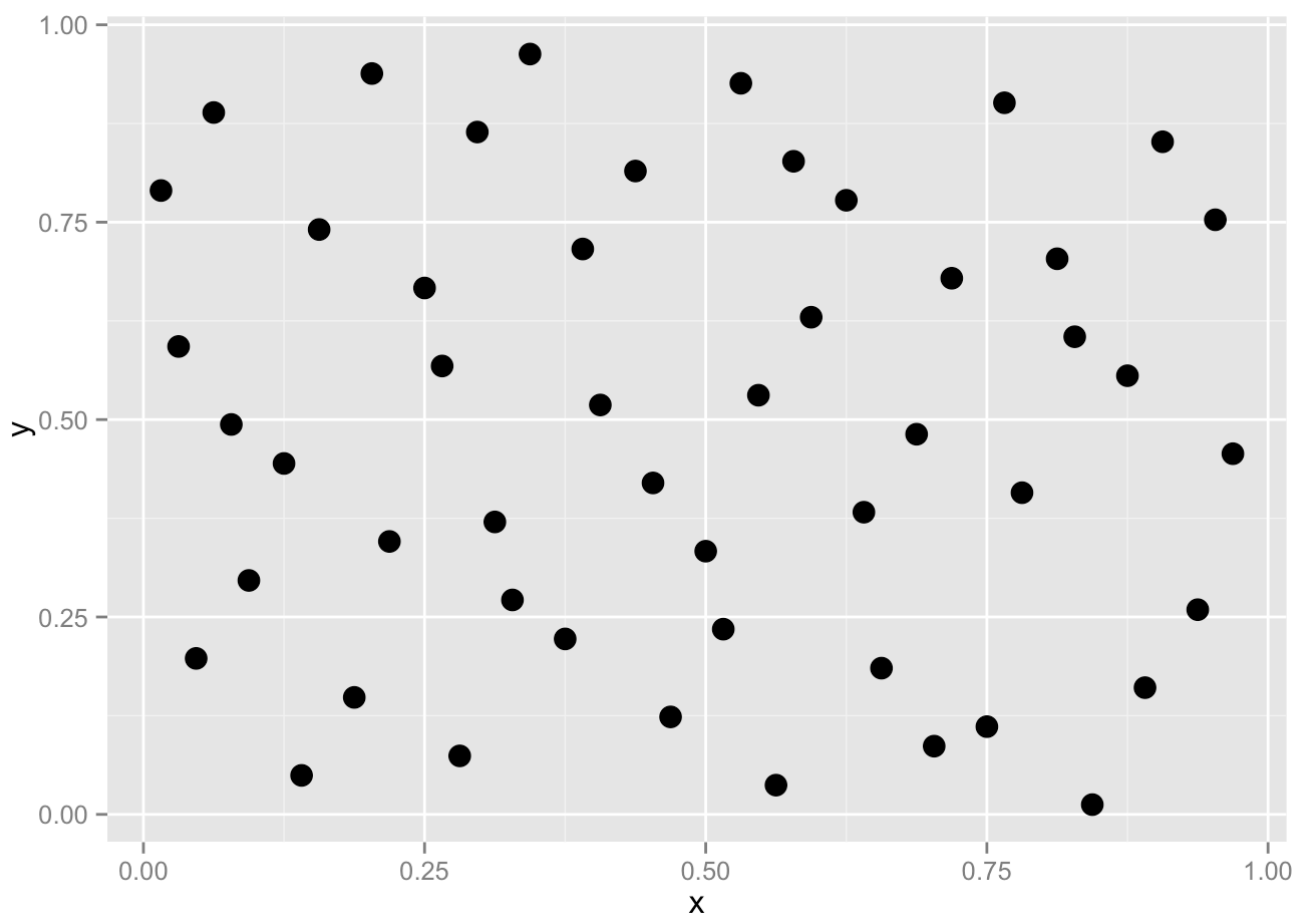
```
ggplot(qmc,aes(x=j,y=x)) + geom_line() + geom_point(size=4) + scale_x_continuous('Sample Number')
```



Halton Sequence

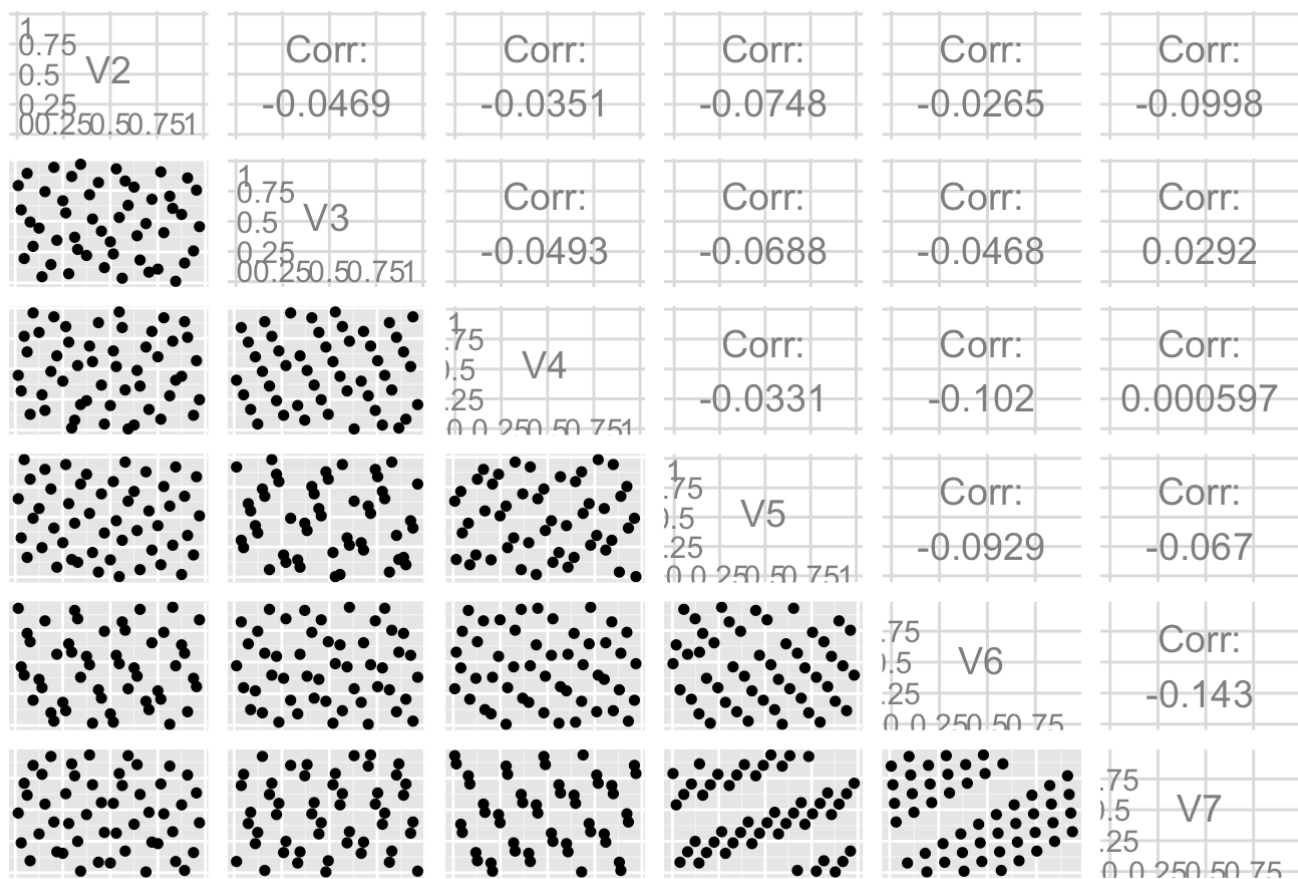
Now a Halton Sequence in 2-D

```
npoints = 50
xy = halton(npoints,dim=2)
qmc <- data.frame(j = 1:npoints)
qmc[,c("x","y")] = xy
ggplot(qmc,aes(x=x,y=y)) + geom_point(size=4)
```



The warts begin to appear in a Halton Sequence in 6-D

```
npoints = 50
xy = halton(npoints,dim=6)
qmc <- data.frame(j = 1:npoints)
qmc[,2:7] = xy
ggpairs(qmc,columns=2:7)
```

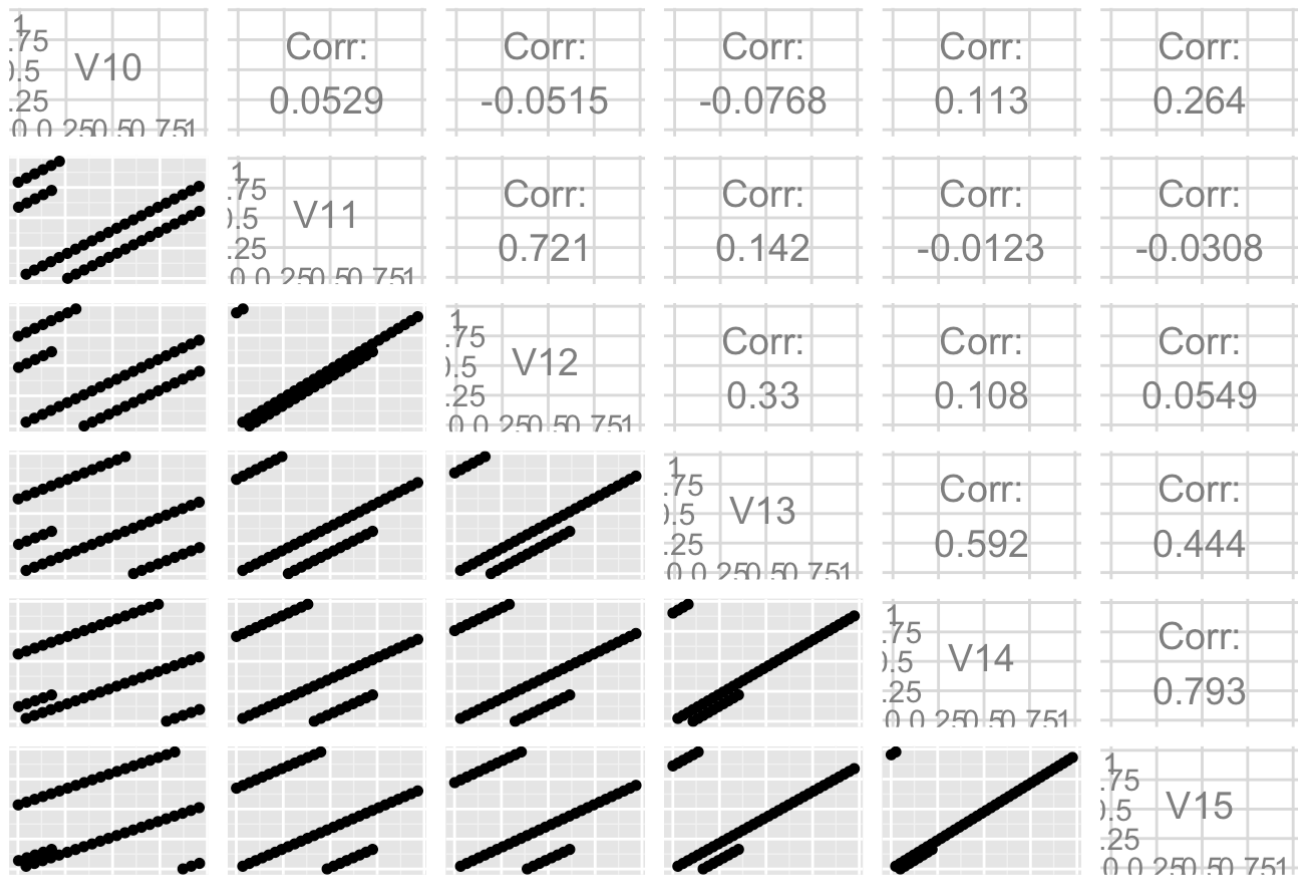


15-D is just plain bad.

```

npoints = 50
xy = halton(npoints,dim=15)
qmc <- data.frame(j = 1:npoints)
qmc[,2:16] = xy
ggpairs(qmc,columns=10:15)

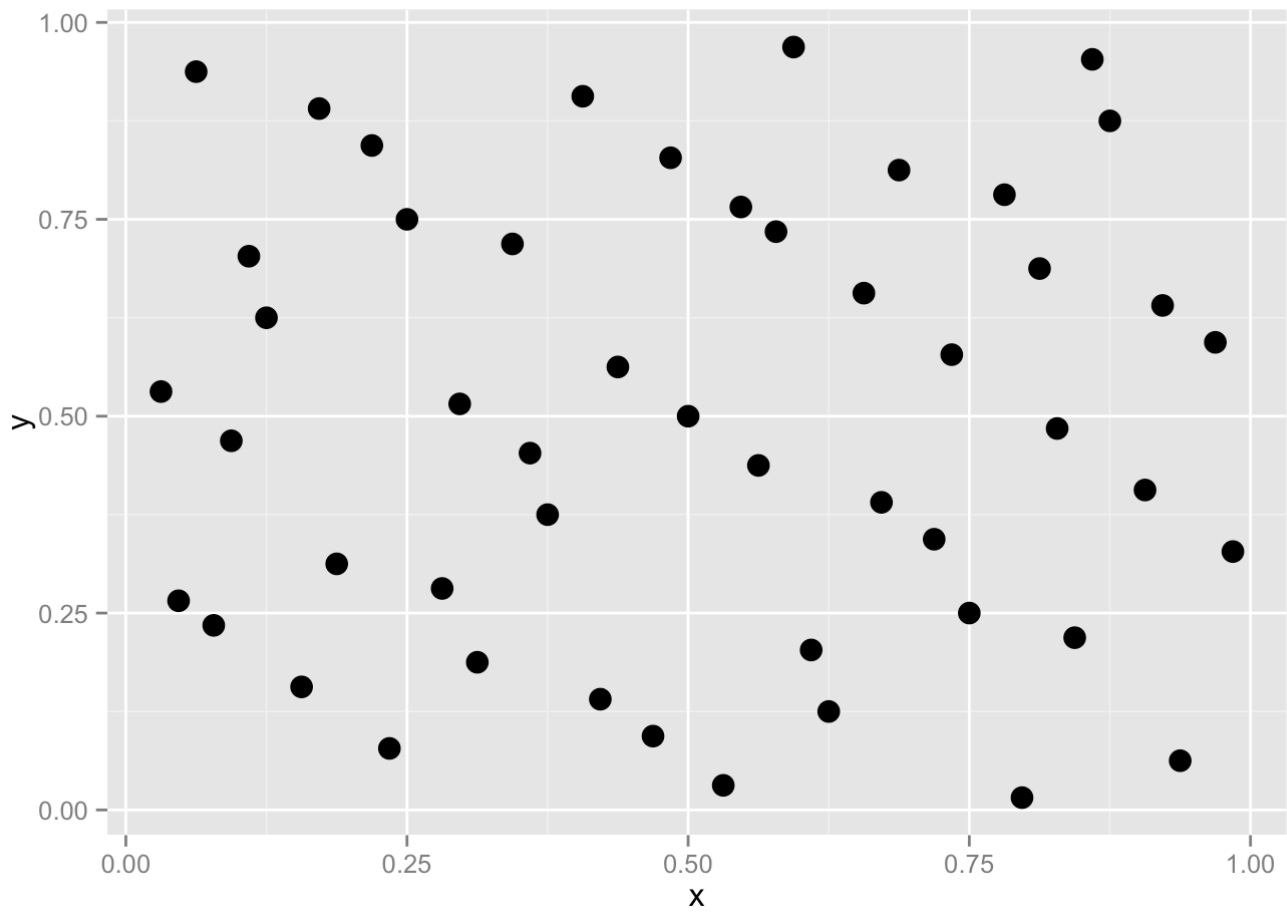
```



Sobol Sequence

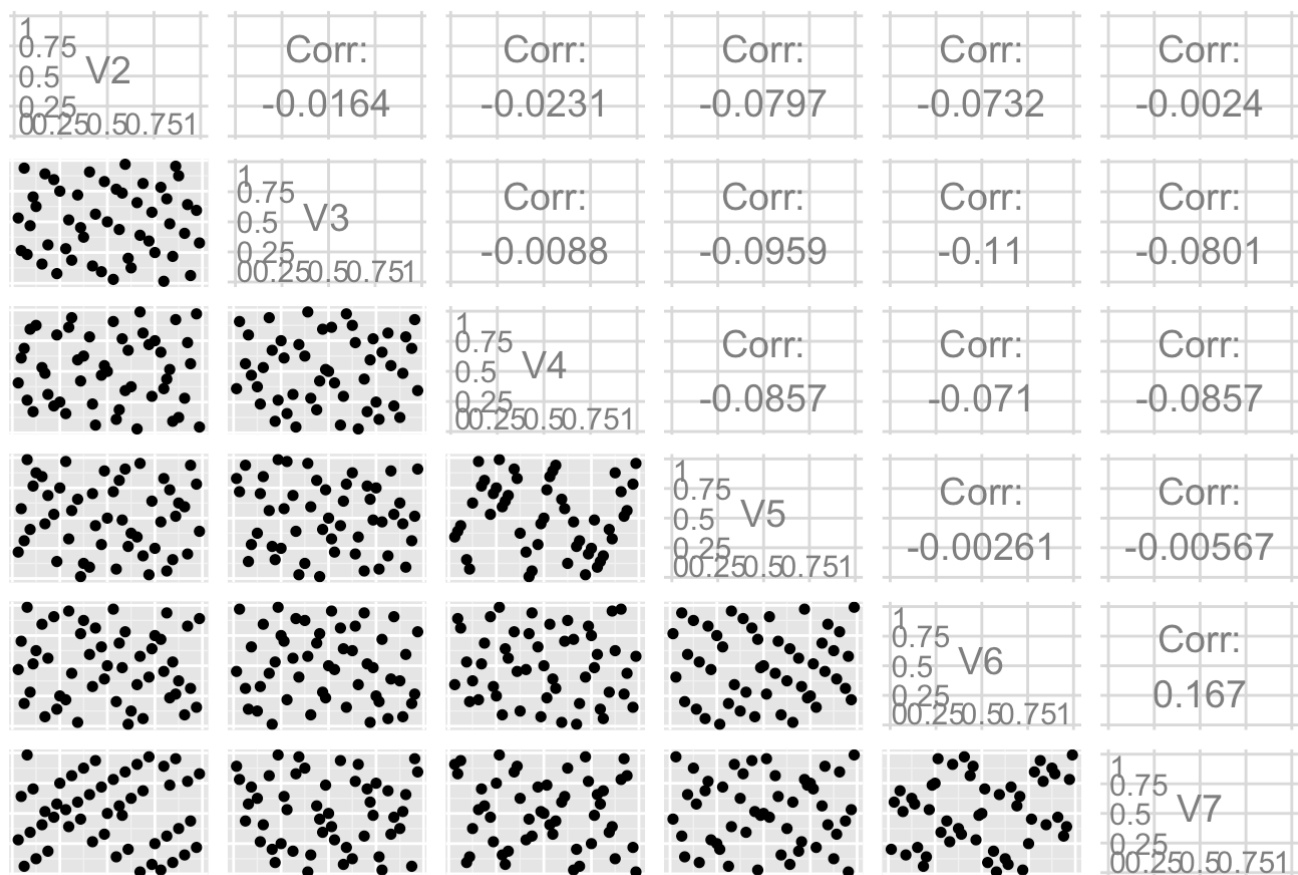
In 2-D the Sobol sequence is similar to Halton.

```
npoints = 50
xy = sobol(npoints,dim=2)
qmc <- data.frame(j = 1:npoints)
qmc[,c("x","y")] = xy
ggplot(qmc,aes(x=x,y=y)) + geom_point(size=4)
```



In 6-D the Sobol appears to be somewhat better than Halton, but still imperfect.

```
npoints = 50
xy = sobol(npoints,dim=6)
qmc <- data.frame(j = 1:npoints)
qmc[,2:7] = xy
ggpairs(qmc,columns=2:7)
```

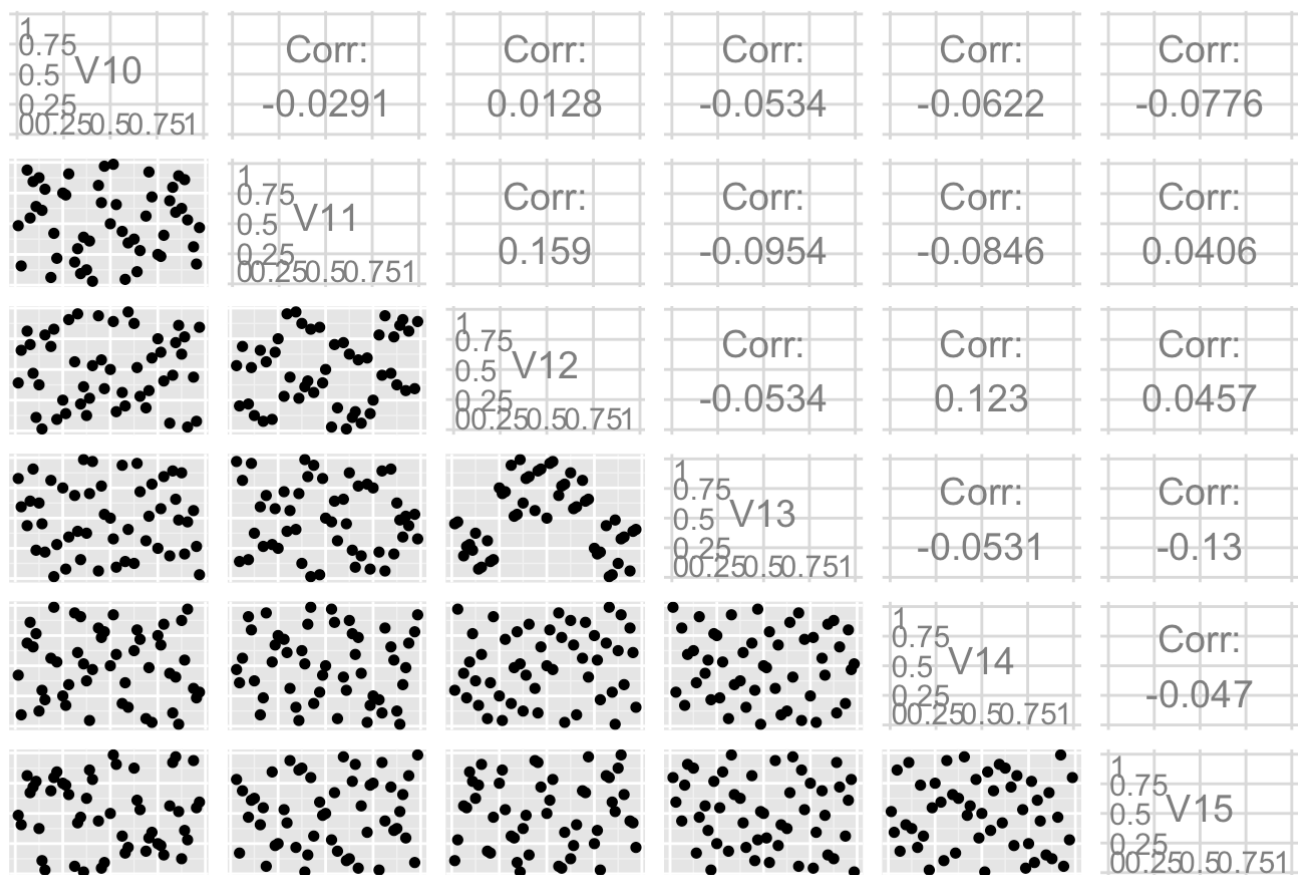


15-D is much better than Halton.

```

npoints = 50
xy = sobol(npoints,dim=15)
qmc <- data.frame(j = 1:npoints)
qmc[,2:16] = xy
ggpairs(qmc,columns=10:15)

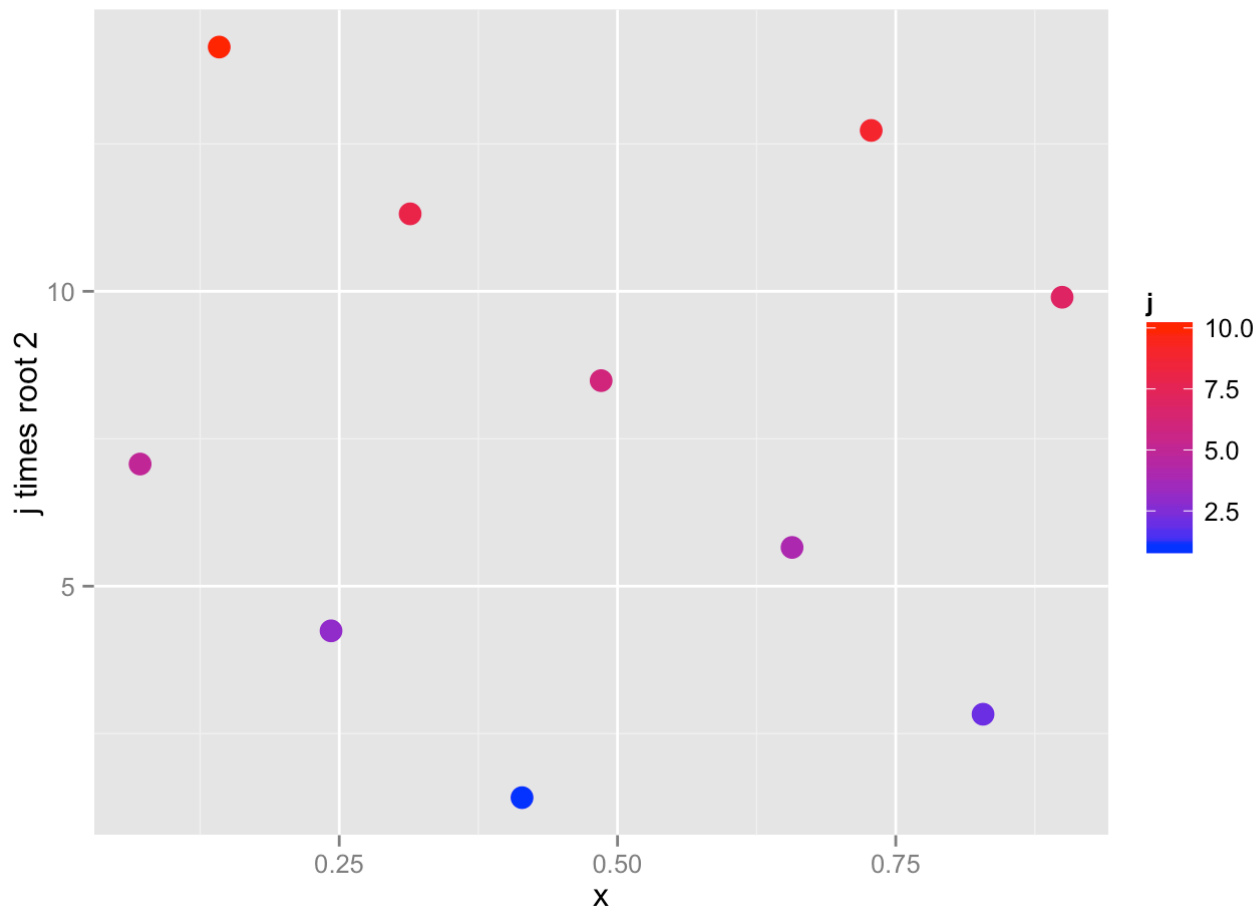
```



Torus

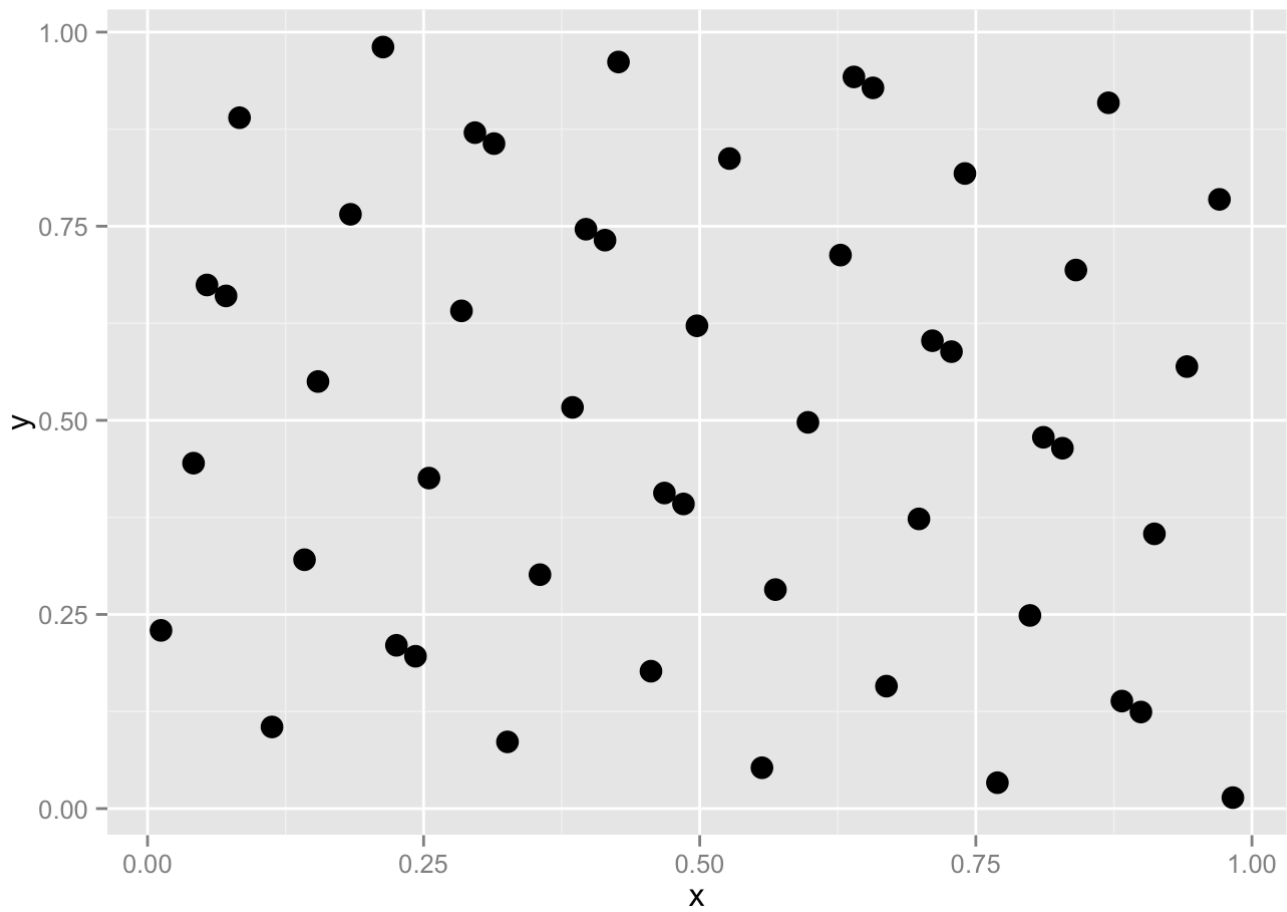
In 1-D the torus is the fractional part of $\sqrt{2}$:

```
npoints = 10
qmc <- data.frame(j = 1:npoints, x = torus(npoints), y = (1:npoints)*sqrt(2))
ggplot(qmc,aes(x=x,y=y,color=j)) + geom_point(size=4) + scale_y_continuous('j times root 2') + scale_colour_
gradient(low="blue",high="red")
```

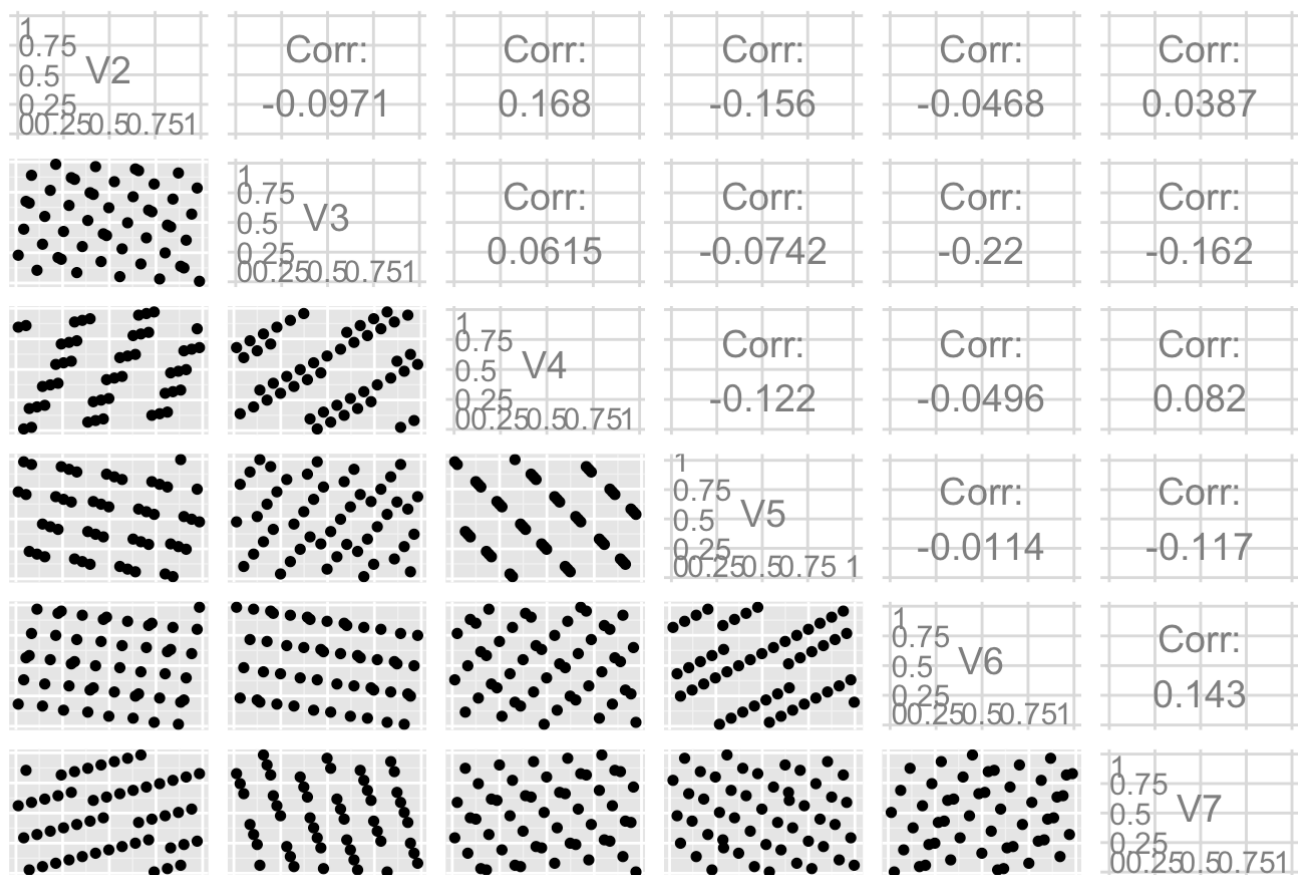
In 2-D the torus doesn't appear too random:

```
npoints = 50
xy = torus(npoints,dim=2)
qmc <- data.frame(j = 1:npoints)
qmc[,c("x","y")] = xy
ggplot(qmc,aes(x=x,y=y)) + geom_point(size=4)
```



In 6-D torus is probably the worst

```
npoints = 50
xy = torus(npoints,dim=6)
qmc <- data.frame(j = 1:npoints)
qmc[,2:7] = xy
ggpairs(qmc,columns=2:7)
```



15-D

```

npoints = 50
xy = torus(npoints,dim=15)
qmc <- data.frame(j = 1:npoints)
qmc[,2:16] = xy
ggpairs(qmc,columns=10:15)

```

