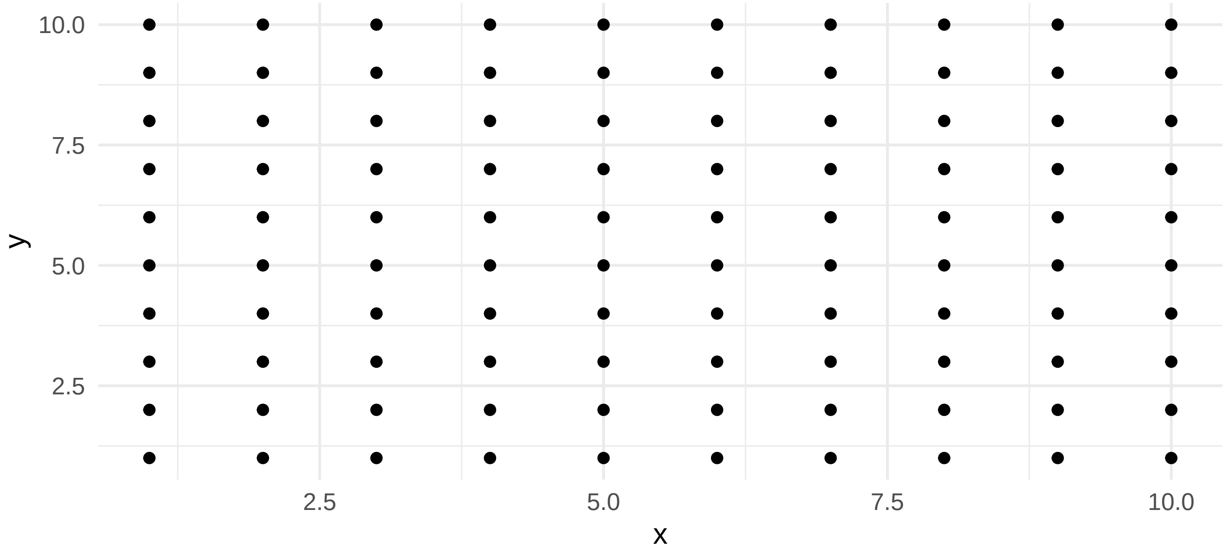
Sonish Lamsal / Bradley Thompson

STA6375: Computational Statistics I

Homework 3

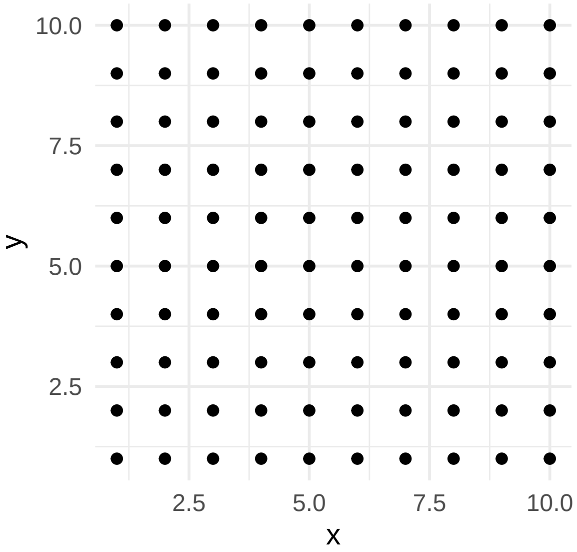
1a.

```
library("tidyverse")



df <- expand.grid("x" = 1:10, "y" = 1:10)
ggplot(df, aes(x, y)) +
  geom_point() +
  theme_minimal()
```



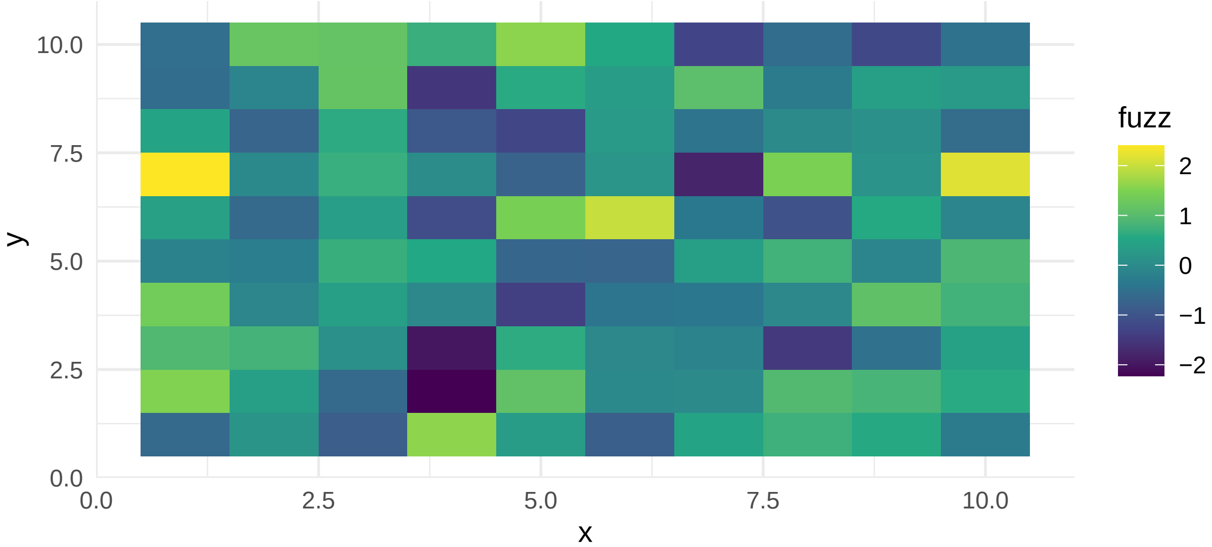
1b.

```
ggplot(df, aes(x, y)) +
  geom_point() +
  theme_minimal() +
  coord_equal()
```



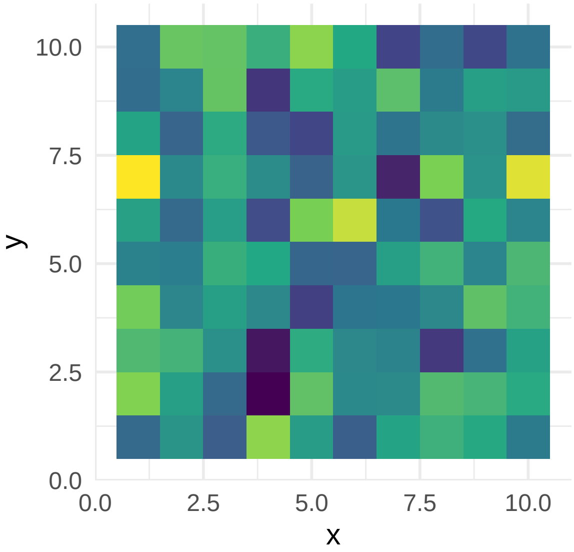
1c.

```
set.seed(1)
fuzz <- rnorm(nrow(df))
  ggplot(df,aes(x,y)) +
  geom_raster(aes(fill=fuzz))+
  theme_minimal()
```



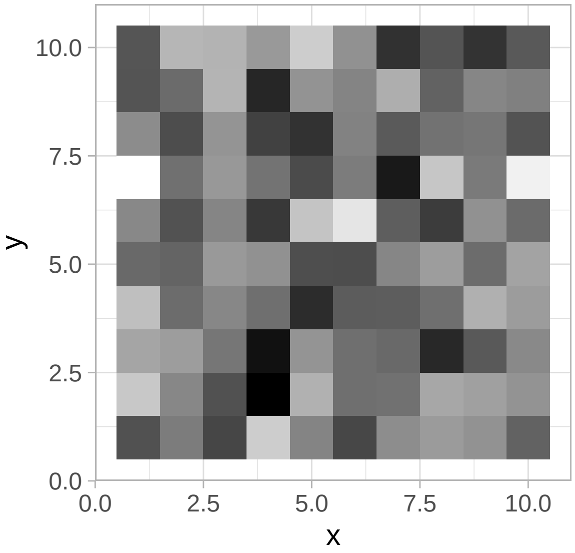
1d.

```
ggplot(df,aes(x,y)) +
  geom_raster(aes(fill=fuzz)) +
  theme_minimal() +
  theme(legend.position="none") +
  coord_equal()
```



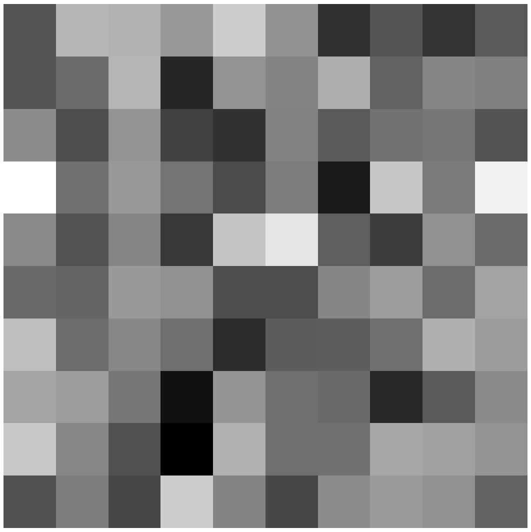
1e

```
ggplot(df, aes(x = x, y = y)) +
  geom_raster(aes(fill = fuzz)) +
  scale_fill_gradient(low = "black", high = "white") +
  coord_equal() +
  theme_light() +
  theme(legend.position = "none")
```



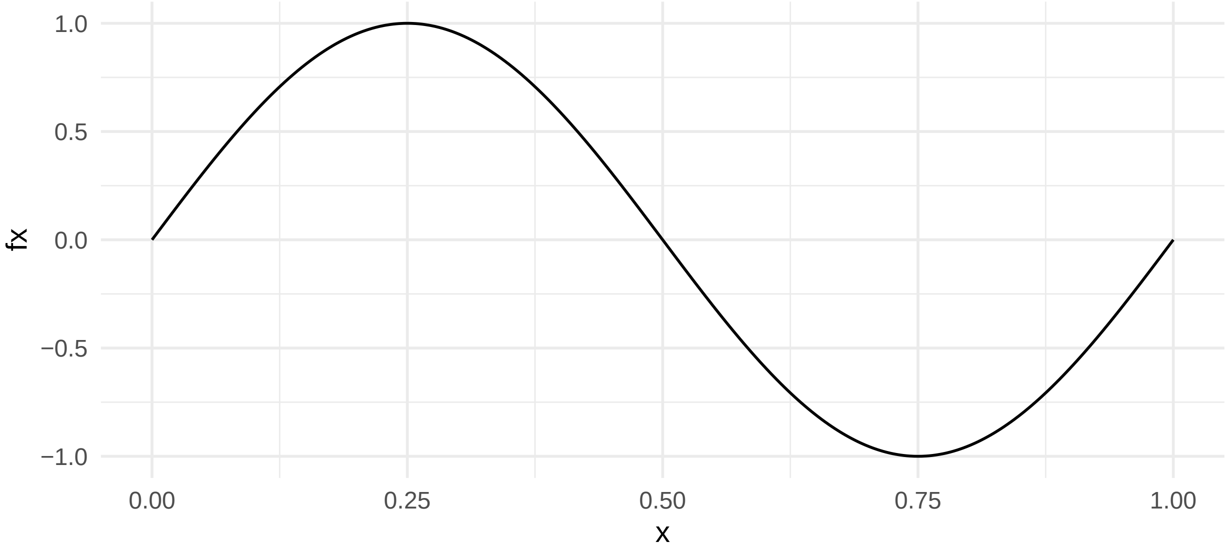
1f.

```
ggplot(df, aes(x = x, y = y)) +
  geom_tile(aes(fill = fuzz)) +
  scale_fill_gradient(low = "black", high = "white") +
  coord_equal() +
  theme_void() +
  theme(legend.position = "none")
```



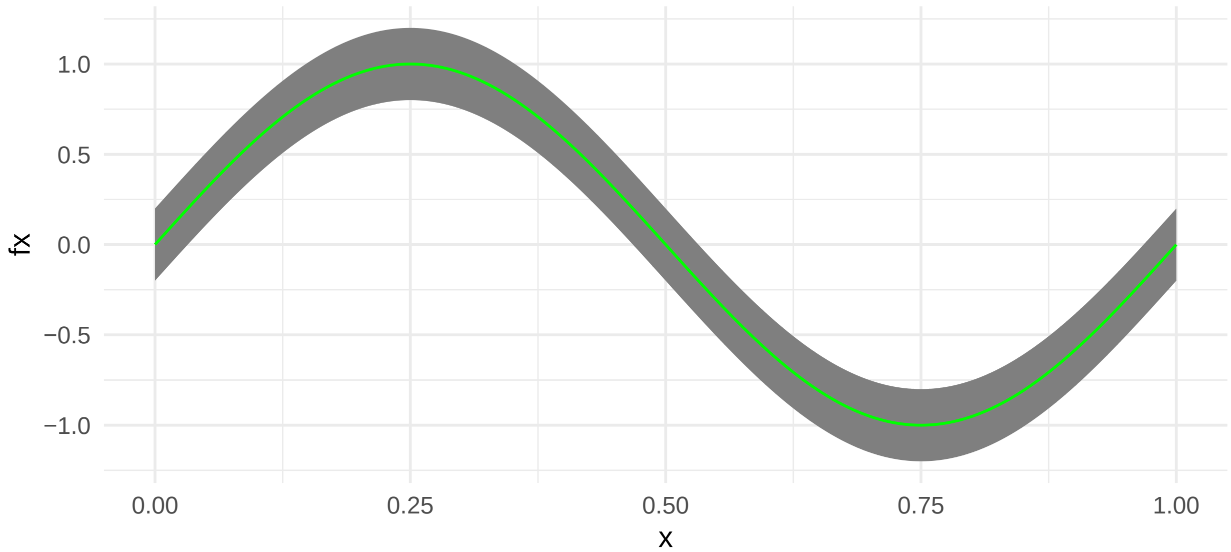
1g

```
x<- seq(0,1, length.out=1001)
FX <- sinpi(2*x)
values<- data.frame(x=c(x),y=c(FX))
ggplot(values,aes(x,y)) +
  geom_line() +
  xlim(0,1) +
  ylim(-1,1) +
  theme_minimal() +
  ylab("fx")
```



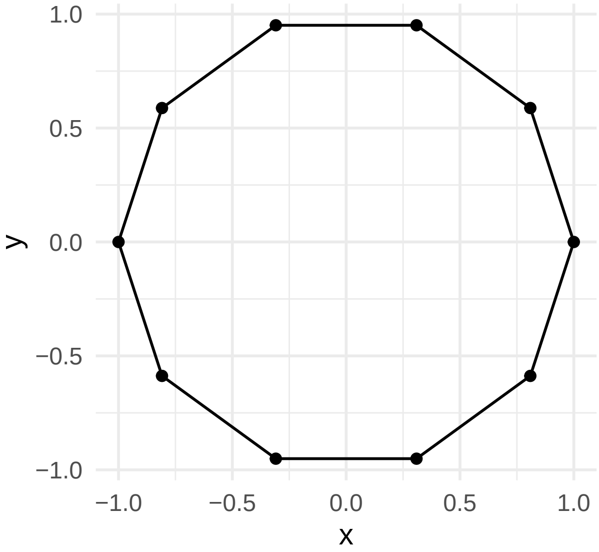
1h

```
ggplot(values,aes(x,y)) +
  geom_ribbon(aes(ymin=y-0.2,ymax=y+0.2),fill="grey50")+
  geom_line(aes(y=y),colour="green") +
  xlim(0,1) +
  ylim(-1.2,1.2) +
  theme_minimal() +
  ylab("fx")
```



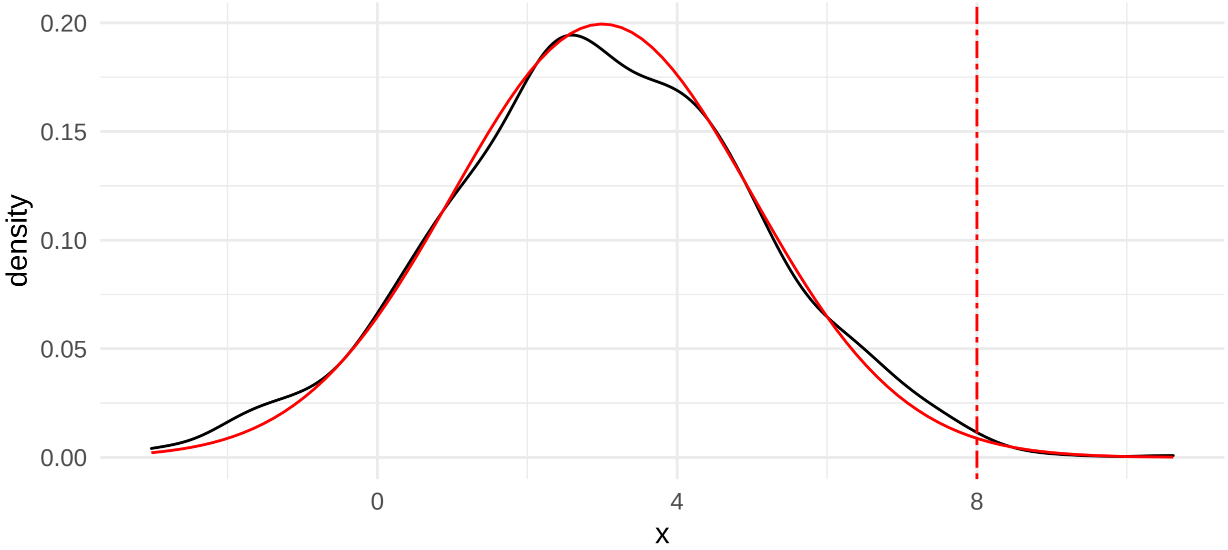
1i

```
x<- vector()
y<-vector()
for (i in 1:10) {
  x[i]=cos((i-1)*36*pi/180)
  y[i]=sin((i-1)*36*pi/180)
}
df<- data.frame(x=x,y=y)
ggplot(df,aes(x,y))+
  geom_polygon(colour="black",fill=NA)+
  theme_minimal()+
  geom_point()+
  coord_equal()
```



1j

```
set.seed(1)
df<- data.frame(x=rnorm(1e3, mean= 3, sd=2 ))
  ggplot(df, aes(x)) +
  geom_density() +
  stat_function(fun=dnorm, args = list(mean=3, sd=2), colour = "red")+
  geom_vline(xintercept = 8, colour="red", linetype="twodash") +
  theme_minimal()
```



2a.

```
# A <- matrix(c(1,3,2,-7,9,1,-2,2,4), nrow = 3, byrow = TRUE)
# ev <- eigen(A)
# V <- ev$vectors
# L <- ev$values
# Lamda <- matrix(c(0,0,0,0,0,0,0,0,0), nrow = 3)
# for(i in 1:length(L)){
```

```
#   Lamda[i,i] <- L[i]
# }
# V_1 <- solve(V)
# A
# V %*% Lamda %*% V_1
```