curves

0.1 Libraries

```
library(tidyverse)
library(latex2exp)
```

0.2 Define function

```
f <- function(x,b0,b1){
  a <- exp(b0 + b1*x)
  b <- 1 + a
  c <- a/b
  return(c)
}
# Example 12.32
f(2,-2,0.5)
```

```
[1] 0.2689414
```

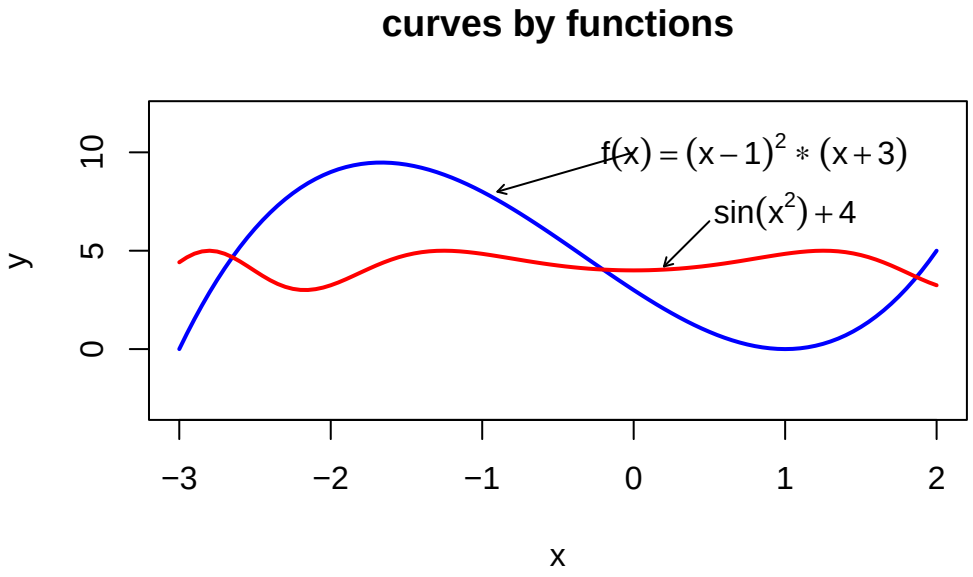
```
f(8,-2,0.5)
```

```
[1] 0.8807971
```

0.3 BaseR:: curve()

`curve()` is a useful baseR function for plotting continuous functions in R. The ggplot equivalent is `geom_function()` as shown below.

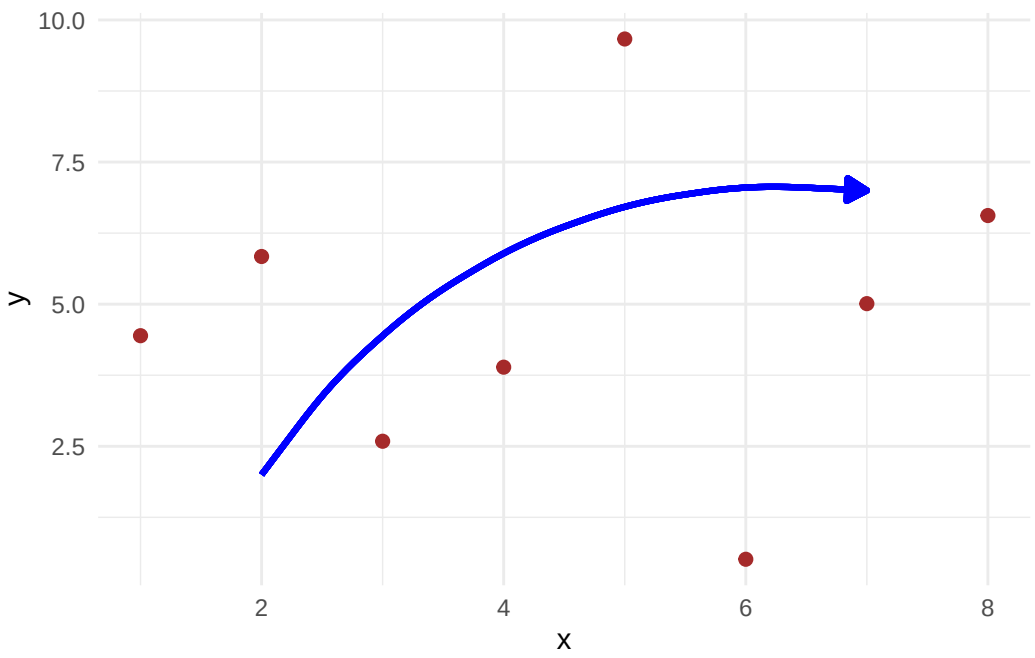
```
ylim=c(-3,12)
curve({(x-1)^2}*(x+3), from=-3, to=2, col="blue", lwd=2, ylim=ylim, ylab="y",
      main="curves by functions")
text(0.8,10,label=TeX(r'($f(x)=(x-1)^2*(x+3)$)'))
arrows(0,10,-0.9,8, length=.05)

curve(sin(x^2)+4, from=-3, to=2, col="red", lwd=2, add=TRUE)
text(1,7,label=expression(sin(x^2)+4))
arrows(0.5,6.5,0.2,4.2, length=.05)
```



0.4 ggplot2::geom_curve()

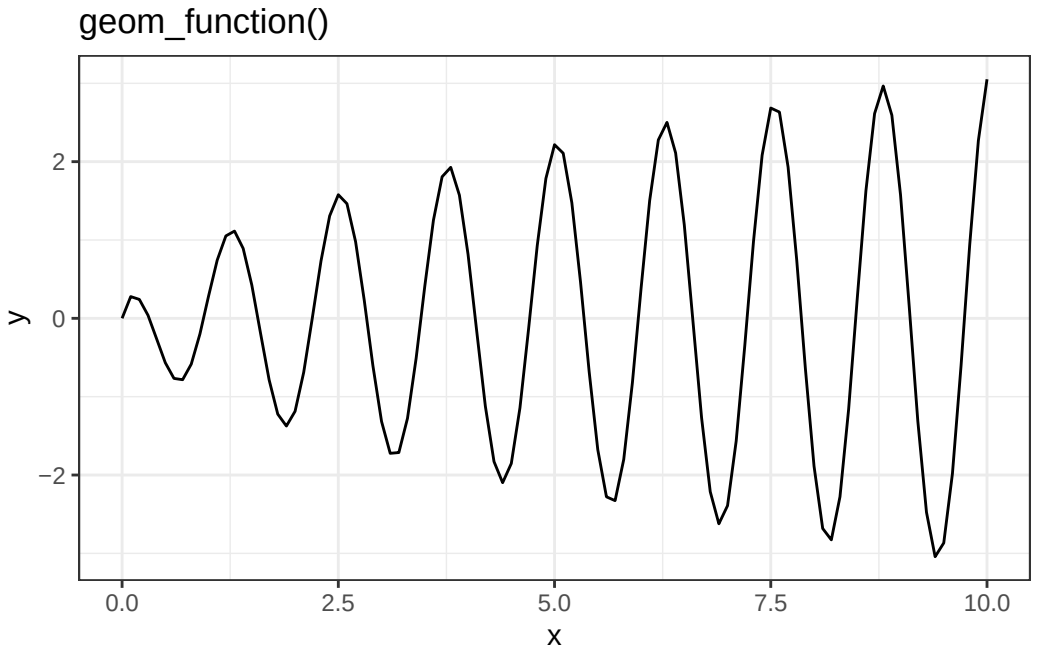
```
df <- data.frame(
  x = 1:8,
  y = rnorm(8, mean = 5, sd = 2)
)
ggplot(df, aes(x, y)) +
  geom_point(size=2, color="brown") +
  geom_curve(aes(x = 2, y = 2, xend = 7, yend = 7),
    arrow = arrow(length = unit(0.3, "cm"), type = "closed"),
    color = "blue",
    size = 1.1,
    curvature = -0.3
```

```
  ) +
  theme_minimal()
```



0.5 ggplot2::geom_function()

```
x <- seq(0,10,0.5)
fn <- function(x){
  sqrt(x)*cos(5*x)
}
ggplot(data.frame(x), aes(x=x))+
  geom_function(fun=fn)+
  theme_bw()+
  ggtitle("geom_function()")
```


geom_function()
2
0
−2
y
0.0
2.5
5.0
7.5
10.0
x