A26_Hoermann

Paul Hörmann

12/2/2019

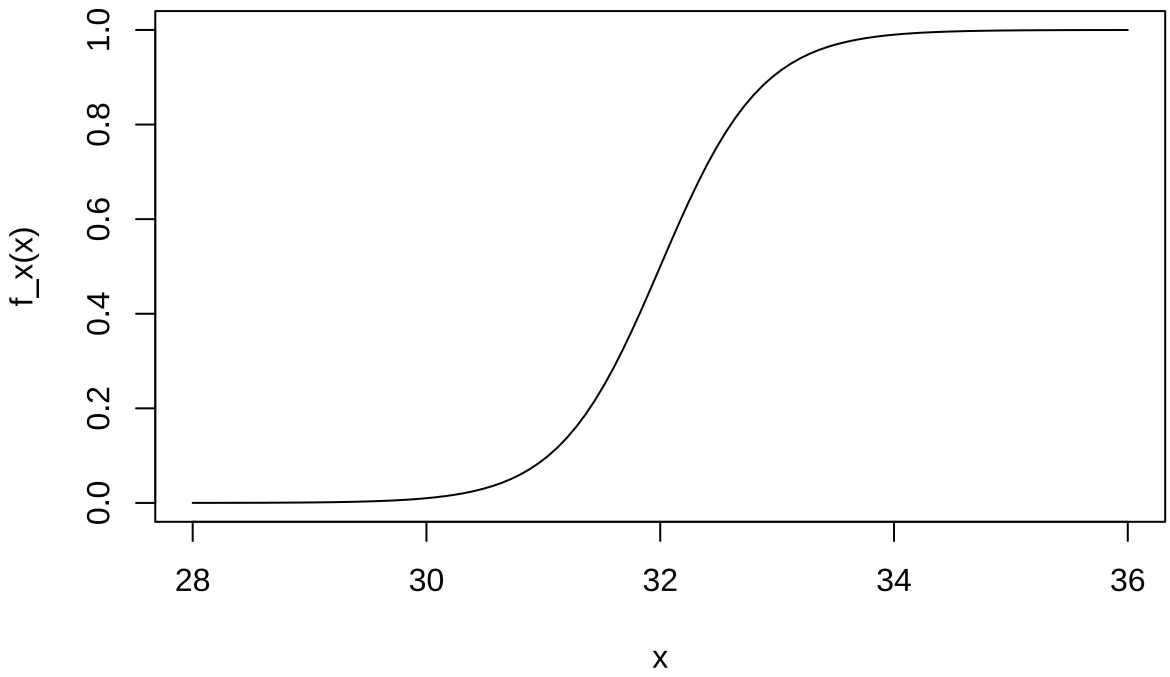
A26

a)

```
a = log(1/.99 - 1) / -2
b = -a * 32
f_x = function(x) {
  1/ (1 + exp(-(a*x + b)))
}
```

Plot

```
curve(f_x,from = 28, to = 36)
```



c)

```
x = seq(30, 34, by = 0.5)
x
```

```
## [1] 30.0 30.5 31.0 31.5 32.0 32.5 33.0 33.5 34.0
```

Get probability Values

```
p_x = lapply(x, f_x)
```

“roll the dice”

```
roll_dice = function(prob) {
  rbinom(1, 1, prob)
}
y = lapply(p_x, roll_dice)
```

combine

```
dt = do.call(rbind, Map(data.frame, x=x, p_x=p_x, y=y))
dt
```

```
##      x        p_x y
## 1 30.0 0.01000000 0
## 2 30.5 0.03087820 0
## 3 31.0 0.09132525 0
## 4 31.5 0.24071200 1
## 5 32.0 0.50000000 0
## 6 32.5 0.75928800 1
## 7 33.0 0.90867475 1
## 8 33.5 0.96912180 1
## 9 34.0 0.99000000 1
```

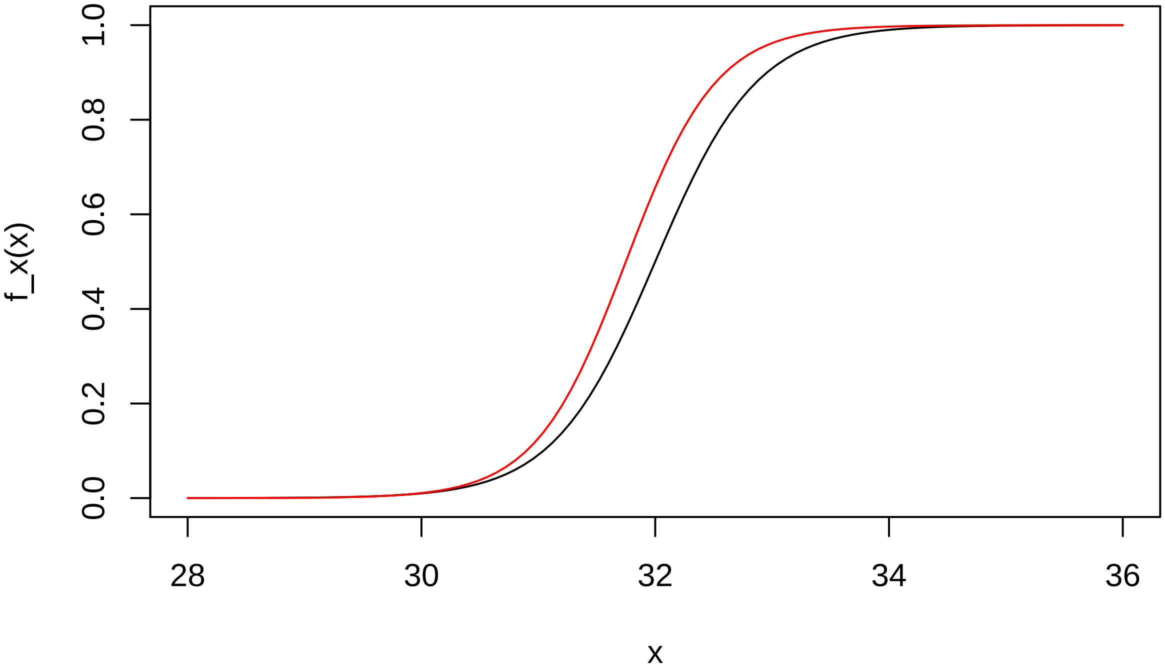
Logistic regression

```
form_x <- unlist(y) ~ x
lr_x = glm(formula = form_x, family = binomial)
lr_x
```

```
##
## Call:  glm(formula = form_x, family = binomial)
##
## Coefficients:
## (Intercept)            x
##     -82.054        2.585
##
## Degrees of Freedom: 8 Total (i.e. Null);  7 Residual
## Null Deviance:       12.37
## Residual Deviance: 5.012     AIC: 9.012
```

Test and compare the new formula

```
curve(f_x,from = 28, to = 36)
b1 = b
a1 = a
b = lr_x$coefficients[1]
a = lr_x$coefficients[2]
p1_x = lapply(x, f_x)
curve(f_x,from = 28, to = 36, add = TRUE, col = "red")
```



```
y1 = lapply(p_x, roll_dice)
dt1 = do.call(rbind, Map(data.frame, x=x, p_x=p_x, y=y, p1_x=p1_x, y1=y1))
dt1
```

```
##       x        p_x y       p1_x y1
## x  30.0 0.01000000 0 0.01078205  0
## x1 30.5 0.03087820 0 0.03817049  0
## x2 31.0 0.09132525 0 0.12625141  0
## x3 31.5 0.24071200 1 0.34473577  0
## x4 32.0 0.50000000 0 0.65700974  0
## x5 32.5 0.75928800 1 0.87459955  0
## x6 33.0 0.90867475 1 0.96211253  1
## x7 33.5 0.96912180 1 0.98930015  1
## x8 34.0 0.99000000 1 0.99703830  1
```