

This is a lab for Chapter 2.

```
library(tidyverse)
```

```
## -- Attaching packages -----
```

```
## v ggplot2 3.0.0      v purrr  0.2.5
## v tibble  1.4.2      v dplyr  0.7.6
## v tidyr   0.8.1      v stringr 1.3.1
## v readr   1.1.1      v forcats 0.3.0
```

```
## -- Conflicts -----
```

```
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()     masks stats::lag()
```

```
income <- read_csv("http://www-bcf.usc.edu/~gareth/ISL/Income1.csv")
```

```
## Warning: Missing column names filled in: 'X1' [1]
```

```
## Parsed with column specification:
```

```
## cols(
##   X1 = col_integer(),
##   Education = col_double(),
##   Income = col_double()
## )
```

```
head(income)
```

```
## # A tibble: 6 x 3
##       X1 Education Income
##   <int>    <dbl> <dbl>
## 1     1      10   26.7
## 2     2     10.4   27.3
## 3     3     10.8   22.1
## 4     4     11.2   21.2
## 5     5     11.6   15.2
## 6     6     12.1   26.4
```

```
# base R graphics
```

```
x0 <- 16
```

```
h <- 1
```

```
flag <- income$Education > x0 - h & income$Education < x0 + h
```

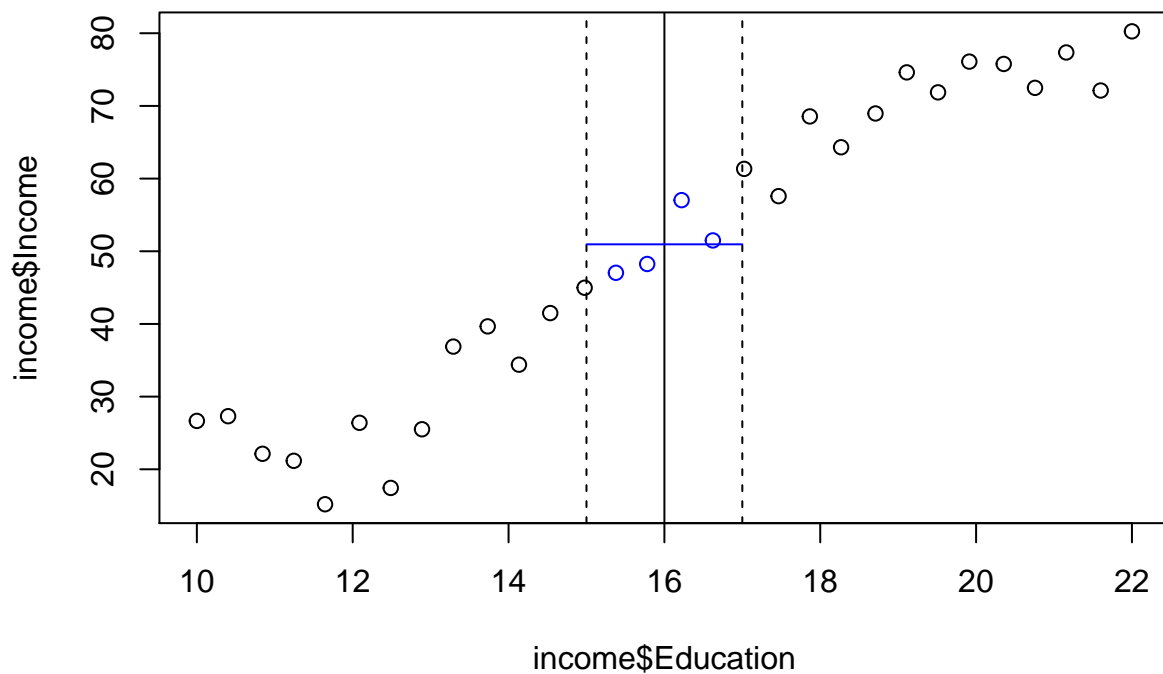
```
plot(income$Education, income$Income, col = c("black", "blue")[flag + 1])
```

```
abline(v = x0)
```

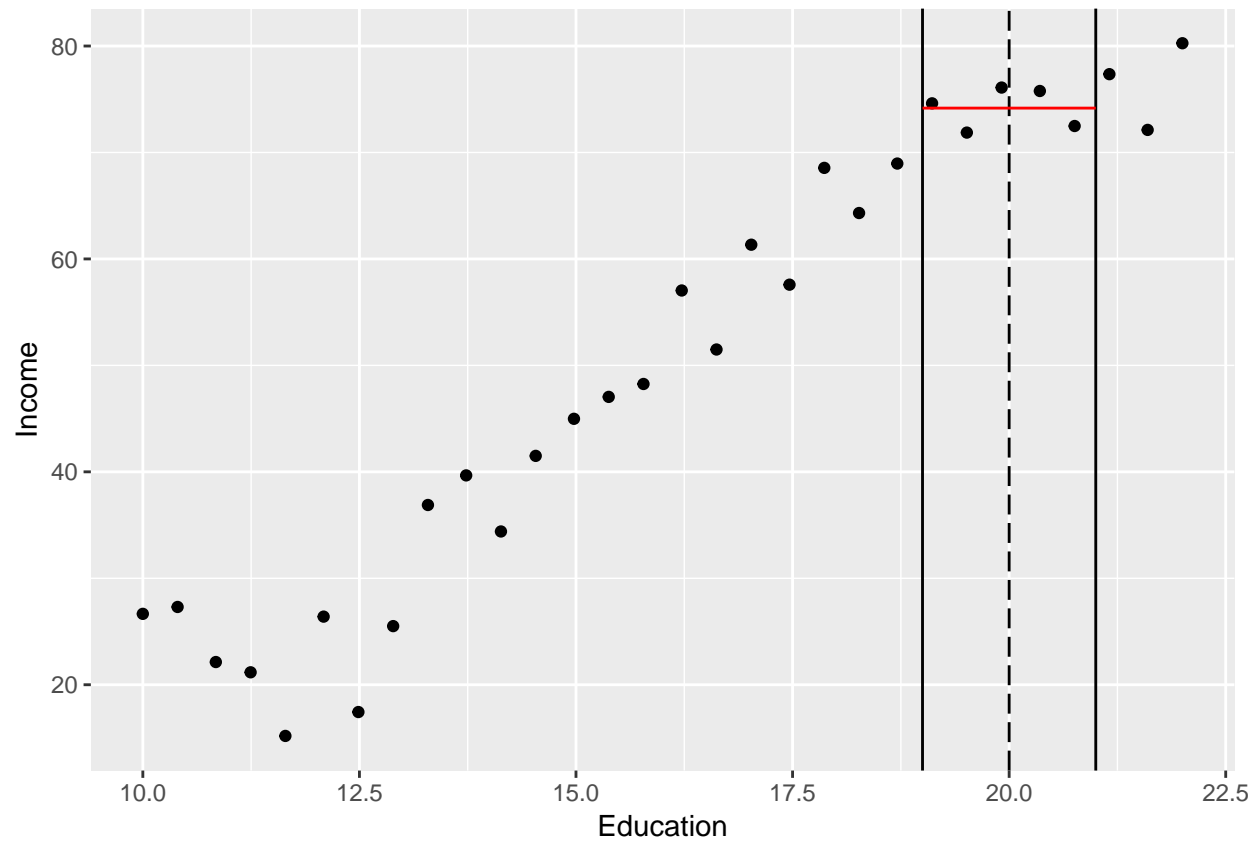
```
abline(v = x0 - h, lty = 2)
```

```
abline(v = x0 + h, lty = 2)
```

```
segments(x0 - h, mean(income$Income[flag]), x0 + h, col = "blue")
```

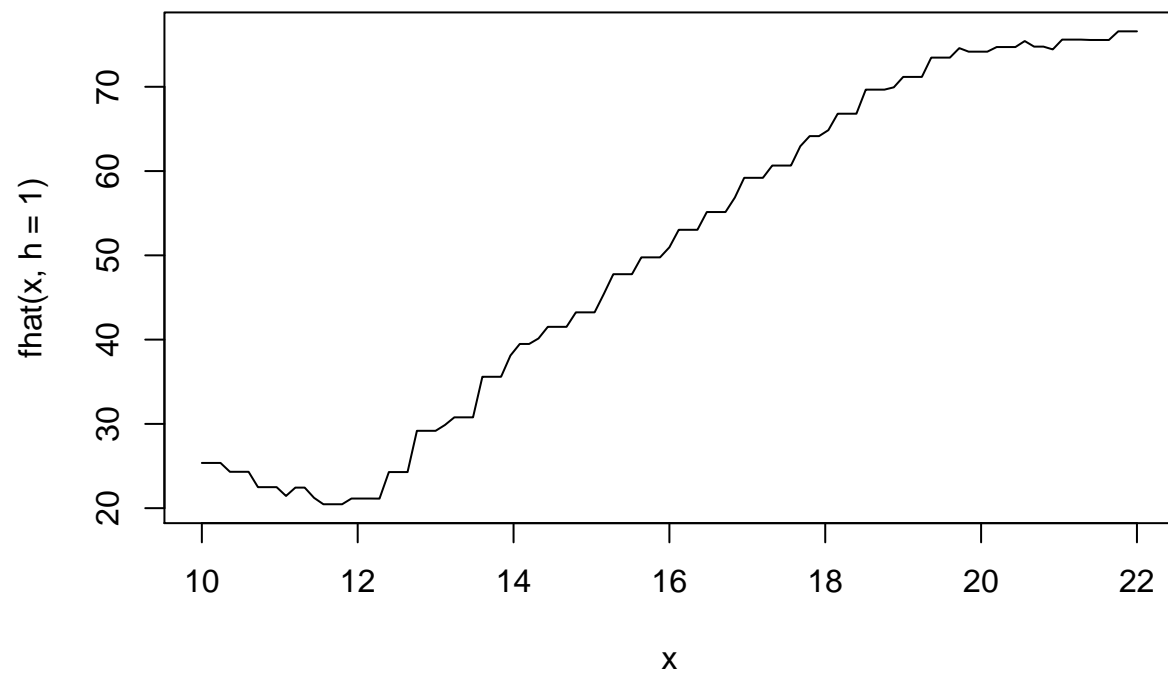


```
# tidyverse style
x0 <- 20
h <- 1
ggplot(income) + geom_point(aes(Education, Income)) +
  geom_vline(xintercept = x0, linetype = "longdash") +
  geom_vline(xintercept = x0 - h) +
  geom_vline(xintercept = x0 + h) +
  geom_segment(data = income %>%
    filter(Education > x0 - h, Education < x0 + h) %>%
    summarize(y = mean(Income)),
    aes(x = x0 - h, y = y, xend = x0 + h, yend = y), color = "red")
```



```
local_avg <- function(x, y){
  f <- function(x0, h) mean(y[x > x0 - h & x < x0 + h])
  return(Vectorize(f))
}

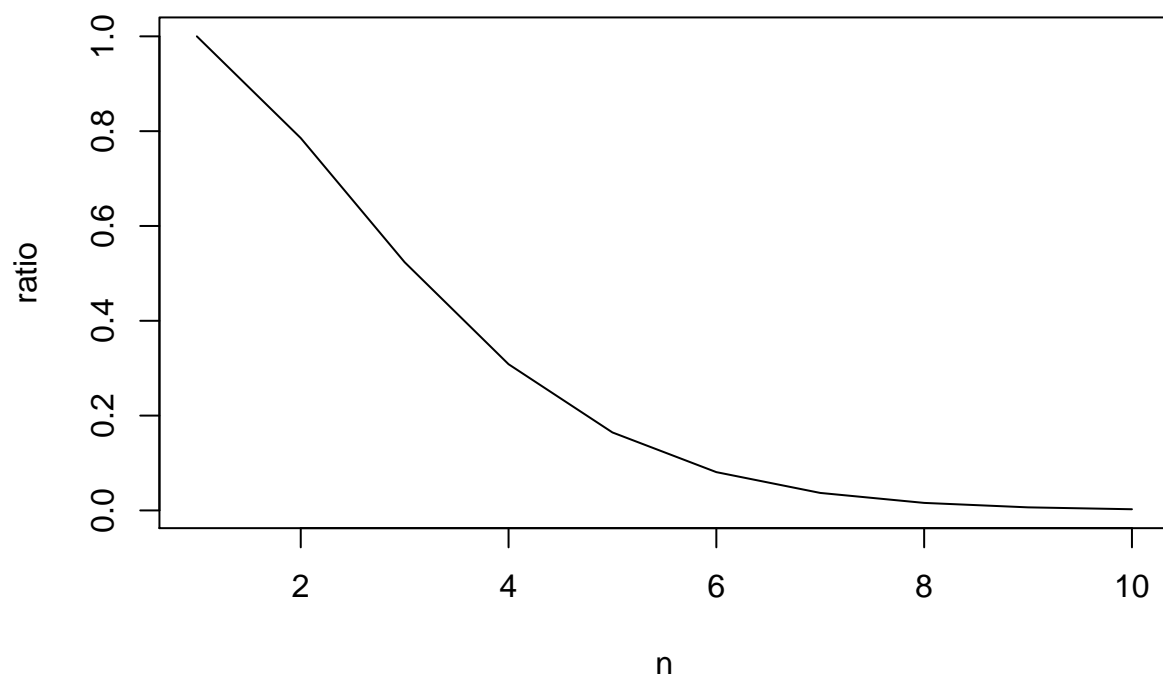
fhat <- local_avg(income$Education, income$Income)
curve(fhat(x, h = 1), xlim = range(income$Education))
```



## Curse of Dimensionality

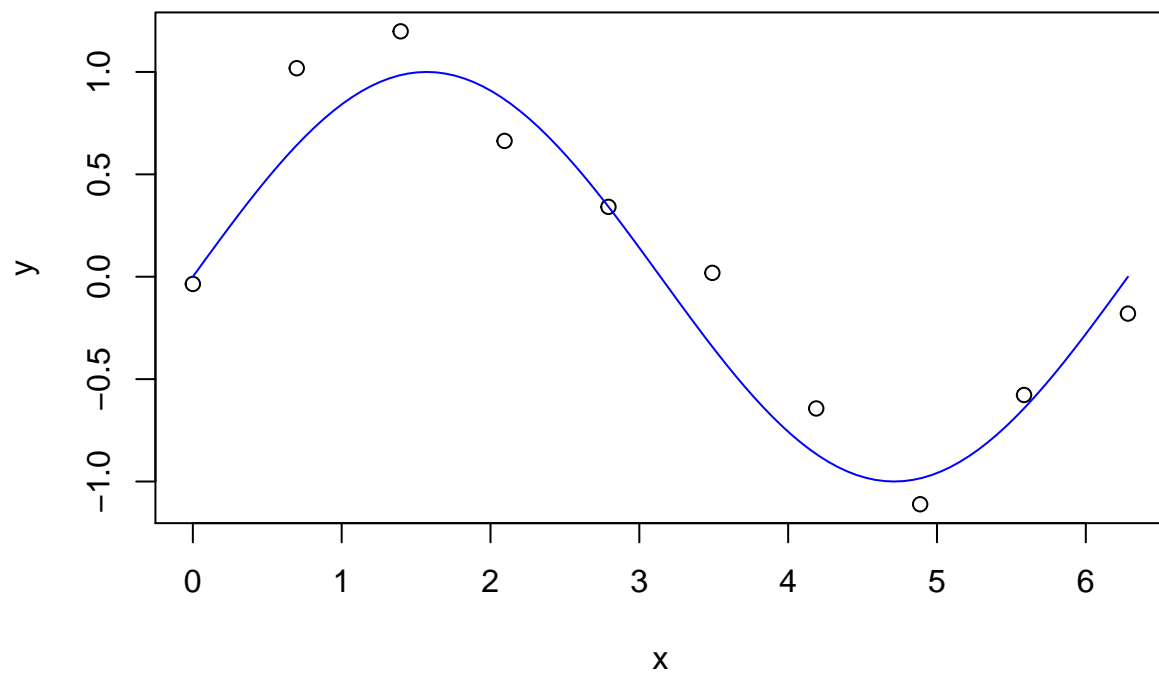
```
vball <- function(n) pi ^ (n / 2) / gamma(n / 2 + 1)
vcube <- function(n) 2^n
```

```
n <- 1:10
ratio <- vball(n) / vcube(n)
plot(n, ratio, type = "l")
```



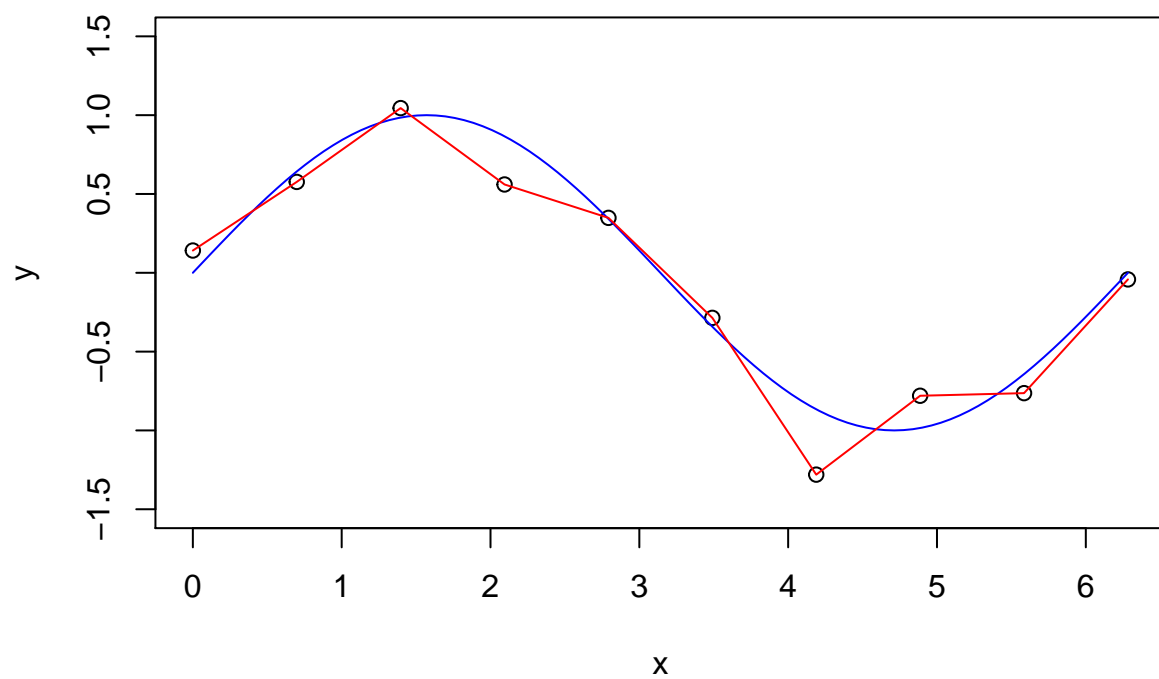
### Bias vs Variance

```
x <- seq(0, 2 * pi, len = 10)
e <- rnorm(10, sd = 0.2)
y <- sin(x) + e
plot(x, y)
curve(sin, add = TRUE, col = "blue")
```



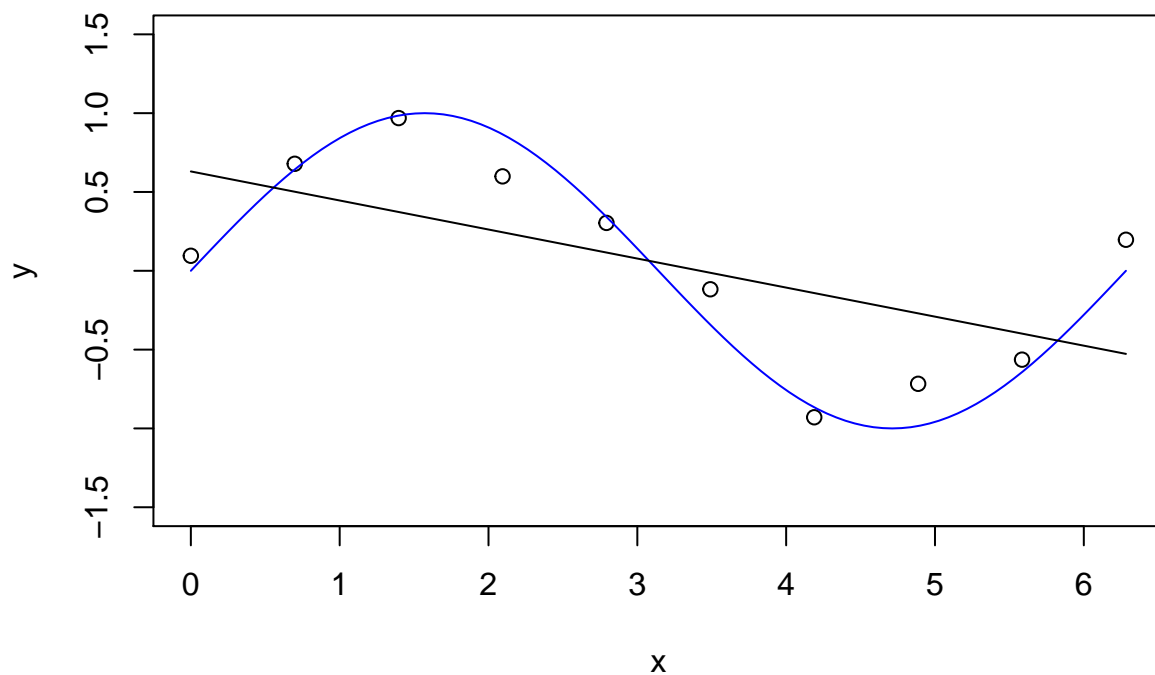
## High variance

```
x <- seq(0, 2 * pi, len = 10)
e <- rnorm(10, sd = 0.2)
y <- sin(x) + e
plot(x, y, ylim = c(-1.5, 1.5))
curve(sin, add = TRUE, col = "blue")
lines(x, y, col = "red")
```



## High bias

```
x <- seq(0, 2 * pi, len = 10)
e <- rnorm(10, sd = 0.2)
y <- sin(x) + e
plot(x, y, ylim = c(-1.5, 1.5))
curve(sin, add = TRUE, col = "blue")
fit <- lm(y ~ x)
lines(x, fitted(fit))
```



### Thin Plate spline

(Actually, we will cover this later)

```
income <- read_csv("http://www-bcf.usc.edu/~gareth/ISL/Income2.csv")
```

```
## Warning: Missing column names filled in: 'X1' [1]
```

```
## Parsed with column specification:
## cols(
##   X1 = col_integer(),
##   Education = col_double(),
##   Seniority = col_double(),
##   Income = col_double()
## )
```

```
head(income)
```

```
## # A tibble: 6 x 4
##       X1 Education Seniority Income
##   <int>    <dbl>    <dbl>  <dbl>
## 1     1     21.6     113.   99.9
## 2     2     18.3     119.   92.6
## 3     3     12.1     101.   34.7
```



```
## 4      4      17.0      188.      78.7
## 5      5      19.9       20      68.0
## 6      6      18.3      26.2     71.5
```

```
library(fields)
```

```
## Loading required package: spam
```

```
## Loading required package: dotCall64
```

```
## Loading required package: grid
```

```
## Spam version 2.2-1 (2018-12-20) is loaded.
## Type 'help( Spam)' or 'demo( spam)' for a short introduction
## and overview of this package.
## Help for individual functions is also obtained by adding the
## suffix '.spam' to the function name, e.g. 'help( chol.spam)'.
```

```
##
## Attaching package: 'spam'
```

```
## The following objects are masked from 'package:base':
##
##      backsolve, forwardsolve
```

```
## Loading required package: maps
```

```
##
## Attaching package: 'maps'
```

```
## The following object is masked from 'package:purrr':
##
##      map
```

```
## See www.image.ucar.edu/~nychka/Fields for
## a vignette and other supplements.
```

```
library(modelr)
fit <- Tps(select(income, Education, Seniority), income$Income)
lonlat <- income %>% do(tibble(lon = seq_range(. $Education, 10), lat = seq_range(. $Seniority, 10)))
grid <- expand(lonlat, lon, lat)
result <- grid %>% add_predictions(fit)
persp(lonlat$lon, lonlat$lat, matrix(result$pred, nc = 10), phi = 45, theta = 45, d = 2)
```

