> set.seed(2)

> x <- rchisq(20, 5)

> e <- rnorm(length(x), 0, 5)

> y <- 3 + 2\*x +e

> x\_bar <- mean(x)

> y\_bar <- mean(y)

> numerator <- sum((x - x\_bar) \* (y - y\_bar))

> denominator <- sum((x - x\_bar)^2)

> b\_hat <- numerator/denominator

> lm(y~x)

Call:

lm(formula = y ~ x)

Coefficients:

(Intercept) x

1.720 2.156

> vector1 <- rbinom(100, 1, 0.5)

> vector2 <- sample(c(-1, 0, 1), 100, replace = TRUE)

> same <- which(vector1 == vector2)

> print(same)

[1] 5 6 17 18 19 20 31 37 39 41 46 51 52 54 57 59 62 73 75 77 78 81 84 85 86 87 91

[28] 92 95

> poker\_vector <- c(140, -50, 20, -120, 240)

> roulette\_vector <- c(-24, -50, 100, 350, 10)

> days\_vector <- c("Mon", "Tues", "Wed", "Thu", "Fri")

> names(poker\_vector) <- days\_vector

> names(roulette\_vector) <- days\_vector

> poker\_midweek <- poker\_vector[c("Tues", "Thu")]

> ave\_poker\_midweek <- mean(poker\_midweek)

> max(poker\_vector)

[1] 240

> which.max(poker\_vector)

Fri

5

> which(poker\_vector > roulette\_vector)

Mon Fri

1 5

> total <- poker\_vector + roulette\_vector

> total\_poker <- sum(poker\_vector)

> total\_roulette <- sum(roulette\_vector)

> total\_poker > total\_roulette

[1] FALSE

> set.seed(pi)

> num <- sample(c(1:10), 1000, replace = TRUE)

> num

[1] 2 9 4 4 7 7 2 3 6 7 6 6 6 6 9 9 2 8 9 3 3 1 2 1 3 8

[27] 6 10 6 8 4 4 2 5 3 4 9 3 6 3 3 8 2 6 5 3 1 2 4 9 3 3

[53] 9 10 9 10 5 3 2 3 9 1 9 2 8 4 8 6 4 1 8 8 10 10 6 6 2 2

[79] 8 8 8 7 4 1 10 9 3 5 7 10 1 3 5 9 9 3 4 4 2 7 8 7 3 8

[105] 7 4 1 5 1 4 4 1 2 2 10 8 9 7 3 5 8 9 9 2 7 5 7 5 9 4

[131] 4 5 4 3 9 6 1 2 6 6 5 4 7 9 9 3 7 9 6 2 8 2 9 4 7 3

[157] 2 2 5 1 10 6 4 9 8 8 9 2 8 6 5 2 6 9 2 6 7 8 2 6 2 4

[183] 2 1 3 4 9 2 4 4 5 9 4 6 8 4 9 10 5 3 8 9 4 5 9 4 8 7

[209] 9 1 8 5 6 3 1 10 3 2 7 8 4 9 4 6 4 8 10 1 3 4 4 5 6 7

[235] 6 9 6 4 5 6 1 8 7 1 7 4 4 10 8 8 3 3 7 7 3 6 1 5 9 10

[261] 8 10 6 7 10 7 1 2 6 3 1 8 10 10 2 4 1 1 2 3 2 1 1 4 2 4

[287] 5 10 9 7 7 7 2 7 3 6 10 9 7 5 1 4 10 8 7 7 9 6 9 10 6 1

[313] 6 7 2 10 1 9 10 7 3 3 9 3 9 6 10 9 10 10 8 6 9 2 2 10 8 1

[339] 9 2 6 3 4 5 6 5 3 1 4 9 9 3 10 9 1 4 1 8 2 10 7 2 2 9

[365] 7 2 6 5 4 3 3 2 1 3 8 10 10 3 4 1 1 4 3 4 8 8 8 5 10 4

[391] 1 9 6 10 5 3 2 9 3 5 1 9 7 6 9 3 1 3 4 6 10 8 1 10 10 7

[417] 1 6 2 6 4 4 2 9 9 6 4 2 8 6 3 1 7 6 10 4 10 8 7 9 6 6

[443] 10 7 2 6 8 8 3 3 10 6 4 5 5 9 9 4 8 9 2 6 4 4 7 6 1 5

[469] 3 9 4 2 2 1 2 7 7 7 4 7 4 9 10 10 7 1 9 4 9 3 8 4 4 5

[495] 7 9 7 9 10 8 3 3 10 4 6 6 3 3 10 9 8 1 1 6 3 9 5 6 1 6

[521] 10 8 8 8 4 2 4 5 7 2 2 4 3 9 6 7 6 2 9 9 7 3 5 10 4 1

[547] 9 8 3 2 8 10 7 4 6 9 9 6 2 6 5 5 5 9 1 6 7 5 10 5 2 8

[573] 10 10 6 4 10 8 2 10 8 10 3 9 3 3 7 8 10 2 7 10 4 1 9 9 4 9

[599] 8 9 2 7 8 9 3 2 4 2 1 6 10 8 2 8 6 1 6 9 2 4 7 4 8 7

[625] 8 1 1 3 2 1 5 1 5 6 4 4 7 4 6 5 2 4 9 9 4 1 6 2 10 6

[651] 6 1 2 3 4 4 6 1 2 9 1 9 4 8 8 10 2 8 8 6 10 5 6 2 10 6

[677] 7 5 3 9 5 8 10 8 9 5 7 3 6 7 6 5 7 6 10 10 8 10 6 7 1 10

[703] 6 7 7 8 7 7 6 8 6 7 1 1 10 5 7 1 7 1 8 6 7 3 1 10 2 1

[729] 9 9 10 10 4 9 7 2 10 8 5 2 10 6 4 9 3 1 5 4 4 6 8 1 3 2

[755] 2 1 7 4 10 5 4 6 2 1 3 4 4 4 6 5 5 9 4 5 1 1 3 4 1 7

[781] 4 6 3 2 5 7 5 3 9 3 2 9 3 1 2 10 5 10 7 4 10 8 9 3 7 10

[807] 6 4 4 7 8 9 10 7 8 6 10 9 10 4 4 3 4 1 4 10 7 5 4 9 3 7

[833] 10 7 3 4 4 1 3 4 8 10 6 10 6 4 5 1 6 2 2 2 2 9 7 9 10 10

[859] 2 9 5 4 3 3 9 7 10 4 2 4 8 8 6 1 4 2 7 4 4 4 6 4 8 5

[885] 9 8 1 5 9 4 2 8 1 1 1 2 8 6 10 3 6 5 1 7 7 3 2 6 9 10

[911] 2 8 5 6 7 3 10 6 6 8 8 9 8 10 4 7 3 2 10 2 8 7 4 6 8 7

[937] 10 6 9 9 8 6 6 3 2 5 5 5 10 9 2 1 9 1 3 7 1 10 2 9 5 1

[963] 10 7 9 7 8 1 9 6 4 8 10 6 3 7 1 7 4 9 10 9 1 5 6 3 10 2

[989] 4 6 9 2 1 1 2 3 2 6 5 6

> a <- sample(c(1:20), 20, replace = TRUE)

> a

[1] 1 4 3 10 11 7 12 4 12 17 12 2 6 6 13 12 7 12 3 17

> replace(num, num == a, NA)

[1] 2 9 4 4 7 NA 2 3 6 7 6 6 NA NA 9 9 2 8 9 3 3 1 2 1 3 8

[27] 6 10 6 8 4 4 2 5 3 4 9 3 6 3 3 8 2 6 5 3 1 2 4 9 3 3

[53] 9 10 9 10 5 3 2 3 9 1 9 2 8 4 8 6 4 1 8 8 10 10 6 6 2 2

[79] 8 8 8 7 4 1 10 9 3 5 7 10 1 3 5 9 9 3 4 4 2 7 8 7 NA 8

[105] 7 4 1 5 1 4 4 1 2 2 10 8 9 7 NA 5 8 9 9 2 7 5 7 5 9 4

[131] 4 5 4 3 9 6 1 2 6 6 5 NA 7 9 9 3 7 9 6 2 8 NA 9 4 7 3

[157] 2 2 5 1 10 6 4 9 8 8 9 2 8 6 5 NA NA 9 2 6 NA 8 2 6 2 NA

[183] 2 1 3 4 9 2 4 4 5 9 4 NA 8 4 9 10 5 3 8 9 4 5 9 4 8 7

[209] 9 1 8 5 NA 3 1 10 3 2 7 8 4 9 4 6 4 8 10 1 3 4 4 5 NA 7

[235] 6 9 6 4 5 6 NA 8 7 1 7 4 4 10 8 8 3 3 7 7 3 6 1 5 9 10

[261] 8 10 6 7 10 NA 1 2 6 3 1 8 10 10 2 4 1 1 2 3 2 1 1 4 2 4

[287] 5 10 9 7 7 7 2 7 3 6 10 9 7 5 NA NA 10 8 7 NA 9 6 9 10 6 1

[313] NA 7 2 10 1 9 10 7 3 3 9 3 9 6 10 9 10 10 8 6 9 2 2 10 8 1

[339] 9 2 6 3 4 5 6 5 3 1 4 9 9 3 10 9 1 4 1 8 2 10 7 2 2 9

[365] 7 2 6 5 4 3 3 NA 1 3 8 10 10 3 4 1 NA NA NA 4 8 8 8 5 10 4

[391] 1 9 NA 10 5 3 2 9 NA 5 NA 9 7 6 9 3 1 3 4 6 10 8 1 10 10 7

[417] 1 6 2 6 4 NA 2 9 9 6 4 2 8 6 3 1 7 NA 10 4 10 8 7 9 6 6

[443] 10 7 2 6 8 8 3 3 10 6 4 5 5 9 9 4 8 9 2 6 4 4 7 6 1 5

[469] 3 9 4 NA 2 1 2 7 NA 7 4 7 4 9 10 NA 7 1 9 NA 9 3 8 4 4 5

[495] 7 9 NA 9 10 8 3 3 10 4 6 6 3 3 10 9 8 1 1 NA 3 9 5 6 1 6

[521] 10 8 8 8 4 2 4 5 7 2 2 4 3 9 6 7 6 2 9 9 7 3 5 NA 4 1

[547] 9 8 3 2 8 10 7 4 6 9 9 6 2 6 5 5 5 9 1 6 7 5 10 5 2 8

[573] 10 10 6 4 10 8 2 10 8 10 NA 9 3 3 7 8 10 2 7 10 4 1 9 9 4 9

[599] 8 9 2 7 8 9 3 2 4 2 1 6 10 8 2 8 6 1 6 9 2 4 7 NA 8 7

[625] 8 1 1 3 2 1 5 1 5 NA 4 4 NA 4 6 5 2 NA 9 9 4 1 6 2 10 6

[651] 6 1 2 3 4 4 6 1 2 9 NA 9 4 8 8 10 2 8 8 6 10 5 NA 2 10 6

[677] NA 5 NA 9 5 8 10 8 9 5 7 3 6 7 6 5 7 NA 10 10 8 10 6 7 NA 10

[703] 6 7 7 8 7 7 6 8 6 7 1 1 10 5 NA 1 7 1 8 6 7 3 1 10 2 1

[729] 9 9 10 10 4 9 7 2 10 8 5 2 10 6 4 9 3 1 5 NA 4 6 8 1 3 2

[755] 2 1 NA 4 10 5 4 6 2 1 3 4 4 NA 6 5 5 9 4 5 1 1 3 4 1 7

[781] 4 6 NA 2 5 NA 5 3 9 3 2 9 3 1 2 10 5 10 7 4 10 8 9 3 7 10

[807] 6 NA 4 7 8 9 10 7 8 6 10 9 10 4 4 3 4 1 4 10 7 5 4 9 3 7

[833] 10 7 3 4 4 1 NA 4 8 10 6 NA 6 4 5 1 6 2 2 NA 2 9 7 9 10 10

[859] 2 9 5 NA NA 3 9 NA 10 NA 2 4 8 8 NA 1 4 2 NA 4 4 4 6 NA 8 5

[885] 9 8 1 5 9 4 2 8 1 1 1 2 8 6 10 3 6 5 1 7 7 3 2 6 9 10

[911] 2 8 5 NA 7 3 10 6 6 8 8 9 8 NA 4 NA 3 2 10 2 8 7 4 NA 8 7

[937] 10 6 9 9 8 6 6 3 2 5 5 5 10 9 2 1 9 1 3 7 1 10 2 9 5 1

[963] 10 7 9 NA 8 1 9 6 4 8 10 NA 3 7 1 7 4 9 10 9 1 5 6 3 10 2

[989] 4 6 9 NA 1 1 2 3 2 6 5 6

> missing <- which(num == a)

> missing

[1] 6 13 14 103 119 142 152 172 173 177 182 194 213 233 241 266 301 302 306 313

[21] 372 381 382 383 393 399 401 422 434 472 477 484 488 497 514 544 583 622 634 637

[41] 642 661 673 677 679 694 701 717 748 757 768 783 786 808 839 844 852 862 863 866

[61] 868 873 877 882 914 924 926 934 966 974 992