Model Answers Set1

Kishore Hari

2022-11-26

## Section 1

1. a
2. b
3. d
4. b,c,d
5. a
6. a,b,c
7. character, numeric, integer, complex, logical
8. a
9. d

## Section 2

1. a, d
2. a
3. a
4. b
5. Sorting

## Section 3

df <- read\_csv("ds\_salaries.csv")

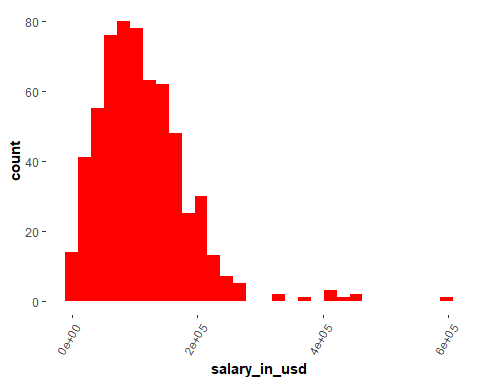
## New names:  
## Rows: 607 Columns: 12  
## -- Column specification  
## -------------------------------------------------------- Delimiter: "," chr  
## (7): experience\_level, employment\_type, job\_title, salary\_currency, empl... dbl  
## (5): ...1, work\_year, salary, salary\_in\_usd, remote\_ratio  
## i Use `spec()` to retrieve the full column specification for this data. i  
## Specify the column types or set `show\_col\_types = FALSE` to quiet this message.  
## \* `` -> `...1`

#15  
ncol(df)

## [1] 12

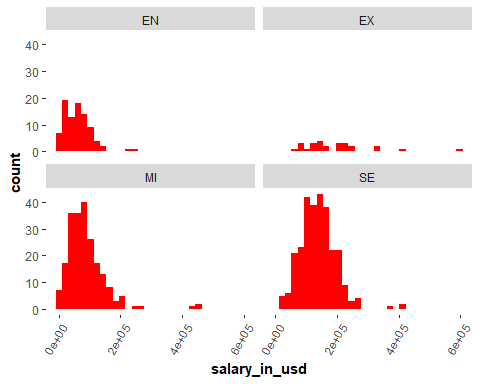
#16  
df %>% ggplot(aes(x = salary\_in\_usd)) +   
 geom\_histogram(fill = "red") +   
 theme(axis.text.x = element\_text(angle = 60, hjust = 1, vjust = 1),  
 panel.background = element\_blank(),   
 axis.title = element\_text(face = "bold"),  
 panel.grid = element\_blank())

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



#17  
df %>% ggplot(aes(x = salary\_in\_usd)) +   
 geom\_histogram(fill = "red") +   
 theme(axis.text.x = element\_text(angle = 60, hjust = 1, vjust = 1),  
 panel.background = element\_blank(),   
 axis.title = element\_text(face = "bold"),  
 panel.grid = element\_blank()) +  
 facet\_wrap(~experience\_level, ncol = 2)

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



#18  
d <- df %>%   
 group\_by(company\_location) %>%  
 summarise(Avg = mean(salary\_in\_usd)) %>%  
 arrange(desc(Avg))  
  
#19  
print("Russia")

## [1] "Russia"

print("User choice")

## [1] "User choice"

#20  
df1 <- df %>%   
 mutate(conversion\_rate = salary/salary\_in\_usd)  
  
#21  
df2 <- df1 %>%  
 group\_by(salary\_currency) %>%  
 summarise(Avg = mean(conversion\_rate)) %>%  
 filter(salary\_currency == "INR")  
print(df2$Avg)

## [1] 74.26327

## Section 4

#22  
  
calculator <- function(num1, num2, char) {  
 if (!is.numeric(num1) || !is.numeric(num2) || !is.character(char)) {  
 print("Invalid inputs")  
 }  
 else if (char == "-") {  
 num1-num2  
 }  
 else if (char == "+") {  
 num1 + num2  
 }  
 else {  
 print("Invalid character")  
 }  
}  
calculator(2,3,"+")

## [1] 5

calculator(3, "4", "-")

## [1] "Invalid inputs"

calculator(3, "a", "-")

## [1] "Invalid inputs"

calculator(5,6,"a")

## [1] "Invalid character"

#23  
  
denominations <- function(a) {  
 d <- c(100, 50, 10, 5, 2,1)  
 nO <- numeric(length(d))  
 for (i in 1:length(d)) {  
 nO[i] <- a%/%d[i]  
 a <- a-nO[i]\*d[i]  
 }  
 if (a != 0) {  
 print("Insufficient d")  
 }  
 return(nO)  
}  
denominations(111)

## [1] 1 0 1 0 0 1

denominations(3000)

## [1] 30 0 0 0 0 0

denominations(23232)

## [1] 232 0 3 0 1 0

denominations(4354)

## [1] 43 1 0 0 2 0

#24  
  
crossProduct <- function(x, y) {  
 if (length(x) < length(y)) {  
 x <- c(x, rep(0, length(y) - length(x)))  
 }  
 if (length(x) > length(y)) {  
 y <- c(y, rep(0, length(x) - length(y)))  
 }  
 dot <- x\*y  
 magx <- sum(x^2) %>% sqrt  
 magy <- sum(y^2) %>% sqrt  
 theta <- acos(dot/(magx\*magy))  
 return(theta)  
}