> head(Fraud\_check)

# A tibble: 6 × 6

Undergrad Marital.Status Taxable.Income City.Population Work.Experience Urban

<chr> <chr> <int> <int> <int> <chr>

1 NO Single 68833 50047 10 YES

2 YES Divorced 33700 134075 18 YES

3 NO Married 36925 160205 30 YES

4 YES Single 50190 193264 15 YES

5 NO Married 81002 27533 28 NO

6 NO Divorced 33329 116382 0 NO

> summary(Fraud\_check)

Undergrad Marital.Status Taxable.Income City.Population Work.Experience

Length:600 Length:600 Min. :10003 Min. : 25779 Min. : 0.00

Class :character Class :character 1st Qu.:32872 1st Qu.: 66967 1st Qu.: 8.00

Mode :character Mode :character Median :55075 Median :106494 Median :15.00

Mean :55208 Mean :108747 Mean :15.56

3rd Qu.:78612 3rd Qu.:150114 3rd Qu.:24.00

Max. :99619 Max. :199778 Max. :30.00

Urban

Length:600

Class :character

Mode :character

> category <- ifelse(Taxable.Income <= 30000 , "Risky", "Good")

|  |
| --- |
| > table(customer$category)  Good Risky  476 124 |
|  |
| |  | | --- | | > | |

> customer <- customer[,c(-3)]

> names(customer)

[1] "Undergrad" "Marital.Status" "City.Population" "Work.Experience"

[5] "Urban" "category"

> set.seed(2)

> train <- sample(1:nrow(customer),nrow(customer)/2)

> length(train)

[1] 300

> test <- -train

> length(test)

[1] 300

> training\_data <- customer[train,]

> testing\_data <- customer[test,]

> names(testing\_data)

[1] "Undergrad" "Marital.Status" "City.Population" "Work.Experience"

[5] "Urban" "category"

> testing\_High <- category[test]

> length(testing\_High)

[1] 300

> dim(training\_data)

[1] 300 6

> dim(testing\_data)

[1] 300 6

> summary(training\_data)

Undergrad Marital.Status City.Population Work.Experience

Length:300 Length:300 Min. : 25779 Min. : 0.00

Class :character Class :character 1st Qu.: 69852 1st Qu.: 9.75

Mode :character Mode :character Median :111166 Median :15.00

Mean :111227 Mean :15.87

3rd Qu.:154842 3rd Qu.:24.00

Max. :199174 Max. :30.00

Urban category

Length:300 Good :233

Class :character Risky: 67

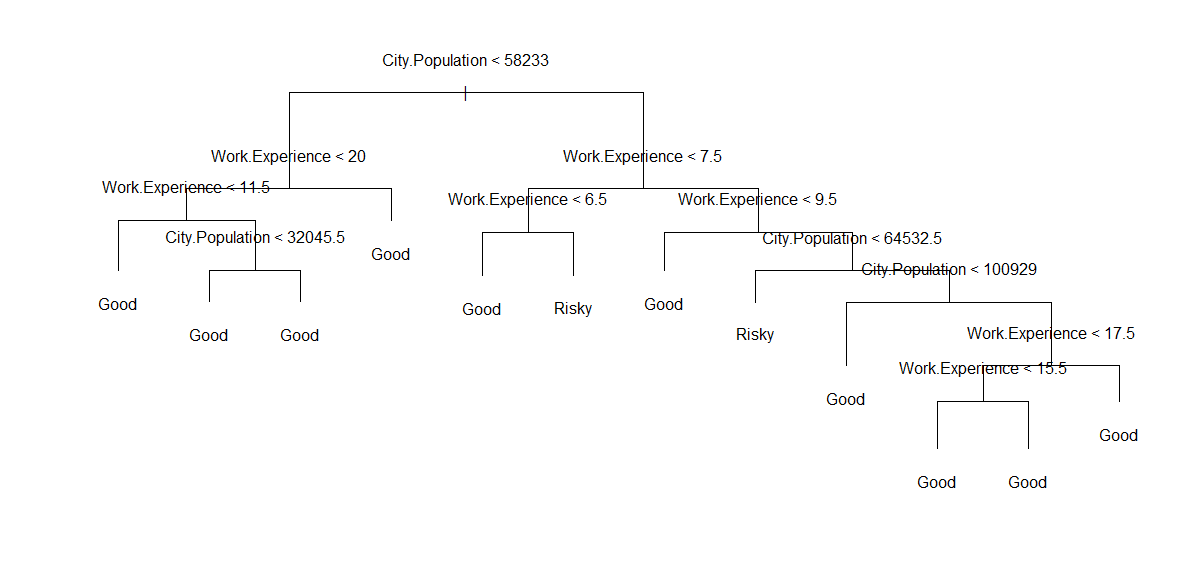
Mode :character

> library(tree)

> tree\_model <- tree(training\_data$category ~ . , training\_data)

> plot(tree\_model)

> text(tree\_model,pretty = 0)



> Decision\_tree\_prediction <- predict(tree\_model, testing\_data, type = "class")

> dim(testing\_data)

[1] 300 6

> mean( Decision\_tree\_prediction != testing\_High)

[1] 0.2333333

> set.seed(3)

> Cross\_Val\_Tree <- cv.tree(tree\_model, FUN = prune.misclass)

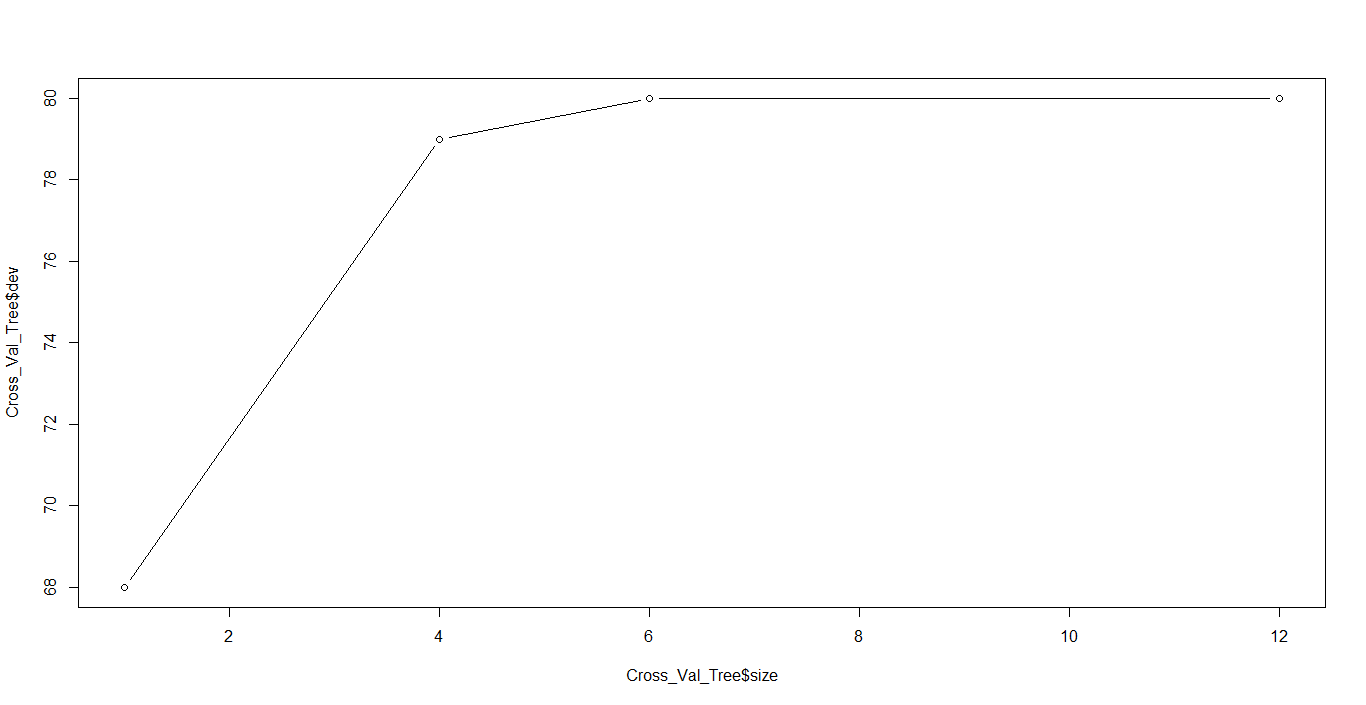
There were 20 warnings (use warnings() to see them)

> names(Cross\_Val\_Tree)

[1] "size" "dev" "k" "method"

> plot(Cross\_Val\_Tree$size, Cross\_Val\_Tree$dev)

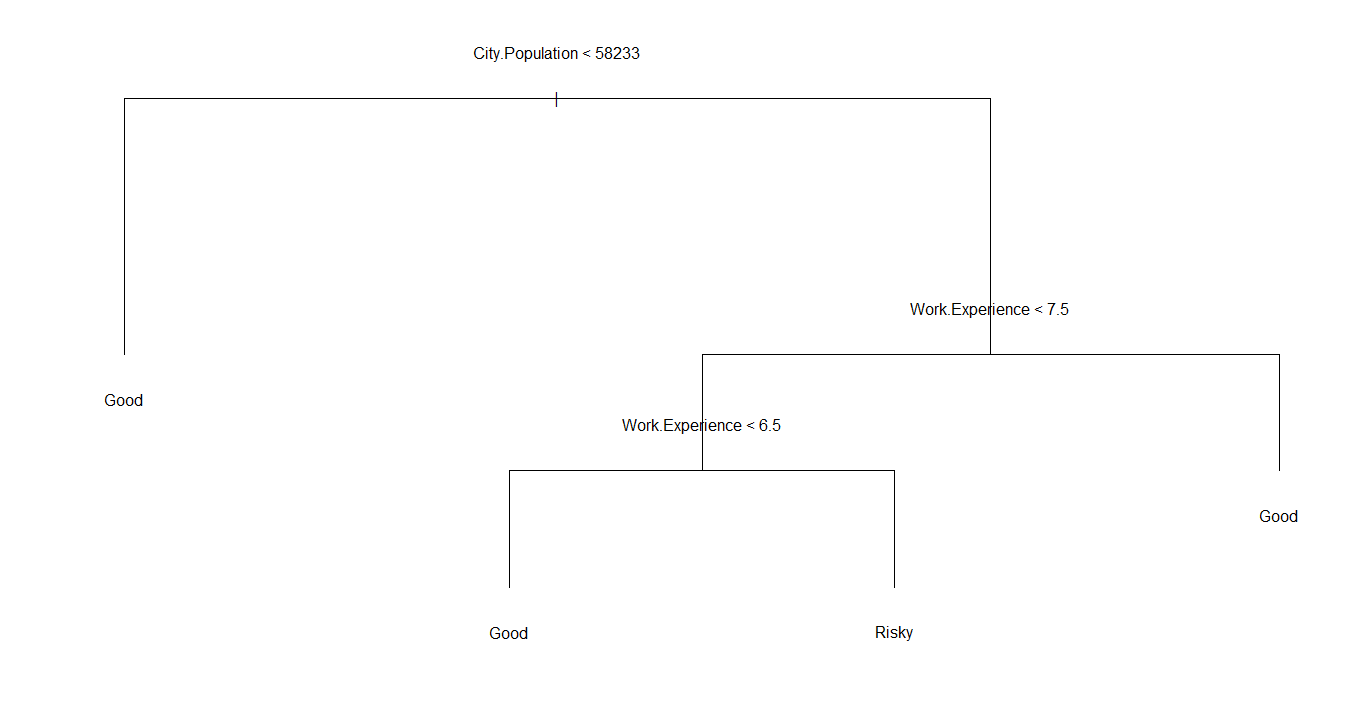
> plot(Cross\_Val\_Tree$size, Cross\_Val\_Tree$dev, type = "b")



> Decision\_Tree\_P\_Model <- prune.misclass(tree\_model,best = 2)

> plot(Decision\_Tree\_P\_Model)

> text(Decision\_Tree\_P\_Model, pretty = 0)



> Decision\_tree\_prediction <- predict(Decision\_Tree\_P\_Model, testing\_data, type = "class")

> mean(Decision\_tree\_prediction != testing\_High)

[1] 0.22