lab1

2025-02-16

Bahaa Khaled Mohamed 21010383

Omar Aldawy 21010864

Declare Variables

```
num_var <- 10

int_var <- 7L

char_var <- "Bioinformatics"

complex_var <- 4 + 3i

print(num_var)
```

```
## [1] 10
```

```
print(int_var)
```

```
## [1] 7
```

```
print(char_var)
```

```
## [1] "Bioinformatics"
```

```
print(complex_var)
```

```
## [1] 4+3i
```

Data Type

```
typeof(num_var)
```

```
## [1] "double"
```

```
typeof(int_var)
```

```
## [1] "integer"
```

```
typeof(char_var)
```

```
## [1] "character"
```

```
typeof(complex_var)
```

```
## [1] "complex"
```

Countdown using while loop

```
count <- 10
while (count >= 0) {
  print(count)
  count <- count - 1
}
```

```
## [1] 10
## [1] 9
## [1] 8
## [1] 7
## [1] 6
## [1] 5
## [1] 4
## [1] 3
## [1] 2
## [1] 1
## [1] 0
```

Function to check even or odd

```
check_even_odd <- function(num) {
  if (num %% 2 == 0) {
    print("Even")
  } else {
    print("Odd")
  }
}
```

Create a vector

```
vec <- c(1,2,3,4,5,6,7,8,9,10)

for (element in vec) {
  print(element)
}
```

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

Create a 4D array with random numbers

```
array_4d <- array(runif(16, min=0, max=10), dim = c(2,2,2,2))
print(array_4d)
```

```
## , , 1, 1
##
##          [,1]      [,2]
## [1,] 2.721795 5.8652203
## [2,] 7.956649 0.5142495
##
## , , 2, 1
##
##          [,1]     [,2]
## [1,] 9.246398 3.978844
## [2,] 1.666010 0.601180
##
## , , 1, 2
##
##            [,1]      [,2]
## [1,] 0.75577295 0.1638839
## [2,] 0.03327732 9.7256466
##
## , , 2, 2
##
##           [,1]      [,2]
## [1,] 4.8907710 5.9931232
## [2,] 0.2830289 0.5044351
```

Iris

```
data(iris)

num_rows <- nrow(iris)
num_cols <- ncol(iris)

column_names <- colnames(iris)

filtered_rows <- subset(iris, Petal.Length > 1.5 & Species == "setosa")

print(paste("Number of rows:", num_rows))
```

```
## [1] "Number of rows: 150"
```

```
print(paste("Number of columns:", num_cols))
```

```
## [1] "Number of columns: 5"
```

```
print("Column names:")
```

```
## [1] "Column names:"
```

```
print(column_names)
```

```
## [1] "Sepal.Length" "Sepal.Width"  "Petal.Length" "Petal.Width"  "Species"
```

```r
print(paste("Rows where Petal.Length > 1.5 & Species == Setosa:", nrow(filtered_rows)))
```

```
## [1] "Rows where Petal.Length > 1.5 & Species == Setosa: 13"
```

Dependency

```r
install.packages('tidyverse')
library(tidyverse)
library(dplyr)
```

Read data-set

```r
dataset <- read.csv("BrainCancerMin.csv")

print(paste("-Number of rows =", nrow(dataset)))
```

```
## [1] "-Number of rows = 130"
```

```r
print(paste("-Number of columns =", ncol(dataset)))
```

```
## [1] "-Number of columns = 150"
```

```r
print("-Column names are")
```

```
## [1] "-Column names are"
```

```r
print(colnames(dataset))
```

```
##   [1] "samples"       "type"          "X1007_s_at"    "X1053_at"
##   [5] "X117_at"       "X121_at"       "X1255_g_at"    "X1294_at"
##   [9] "X1316_at"      "X1320_at"      "X1405_i_at"    "X1431_at"
##  [13] "X1438_at"      "X1487_at"      "X1494_f_at"    "X1552256_a_at"
##  [17] "X1552257_a_at" "X1552258_at"   "X1552261_at"   "X1552263_at"
##  [21] "X1552264_a_at" "X1552266_at"   "X1552269_at"   "X1552271_at"
##  [25] "X1552272_a_at" "X1552274_at"   "X1552275_s_at" "X1552276_a_at"
##  [29] "X1552277_a_at" "X1552278_a_at" "X1552279_a_at" "X1552280_at"
##  [33] "X1552281_at"   "X1552283_s_at" "X1552286_at"   "X1552287_s_at"
##  [37] "X1552288_at"   "X1552289_a_at" "X1552291_at"   "X1552293_at"
##  [41] "X1552295_a_at" "X1552296_at"   "X1552299_at"   "X1552301_a_at"
##  [45] "X1552302_at"   "X1552303_a_at" "X1552304_at"   "X1552306_at"
##  [49] "X1552307_a_at" "X1552309_a_at" "X1552310_at"   "X1552311_a_at"
##  [53] "X1552312_a_at" "X1552314_a_at" "X1552315_at"   "X1552316_a_at"
##  [57] "X1552318_at"   "X1552319_a_at" "X1552320_a_at" "X1552321_a_at"
##  [61] "X1552322_at"   "X1552323_s_at" "X1552325_at"   "X1552326_a_at"
##  [65] "X1552327_at"   "X1552329_at"   "X1552330_at"   "X1552332_at"
##  [69] "X1552334_at"   "X1552335_at"   "X1552337_s_at" "X1552338_at"
##  [73] "X1552340_at"   "X1552343_s_at" "X1552344_s_at" "X1552347_at"
##  [77] "X1552348_at"   "X1552349_a_at" "X1552354_at"   "X1552355_s_at"
##  [81] "X1552359_at"   "X1552360_a_at" "X1552362_a_at" "X1552364_s_at"
##  [85] "X1552365_at"   "X1552367_a_at" "X1552368_at"   "X1552370_at"
##  [89] "X1552372_at"   "X1552373_s_at" "X1552375_at"   "X1552377_s_at"
##  [93] "X1552378_s_at" "X1552379_at"   "X1552381_at"   "X1552383_at"
##  [97] "X1552384_a_at" "X1552386_at"   "X1552388_at"   "X1552389_at"
## [101] "X1552390_a_at" "X1552391_at"   "X1552393_at"   "X1552394_a_at"
```

```
## [105] "X1552395_at"   "X1552396_at"   "X1552398_a_at" "X1552399_a_at"
## [109] "X1552400_a_at" "X1552401_a_at" "X1552402_at"   "X1552405_at"
## [113] "X1552408_at"   "X1552409_a_at" "X1552410_at"   "X1552411_at"
## [117] "X1552412_a_at" "X1552414_at"   "X1552415_a_at" "X1552417_a_at"
## [121] "X1552418_at"   "X1552419_s_at" "X1552421_a_at" "X1552422_at"
## [125] "X1552423_at"   "X1552424_at"   "X1552425_a_at" "X1552426_a_at"
## [129] "X1552427_at"   "X1552430_at"   "X1552432_at"   "X1552436_a_at"
## [133] "X1552438_a_at" "X1552439_s_at" "X1552440_at"   "X1552445_a_at"
## [137] "X1552448_a_at" "X1552449_a_at" "X1552450_a_at" "X1552452_at"
## [141] "X1552453_a_at" "X1552455_at"   "X1552456_a_at" "X1552457_a_at"
## [145] "X1552458_at"   "X1552459_a_at" "X1552461_at"   "X1552463_at"
## [149] "X1552466_x_at" "X1552467_at"
```

Data pre-processing

Determining the Working Set

```r
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

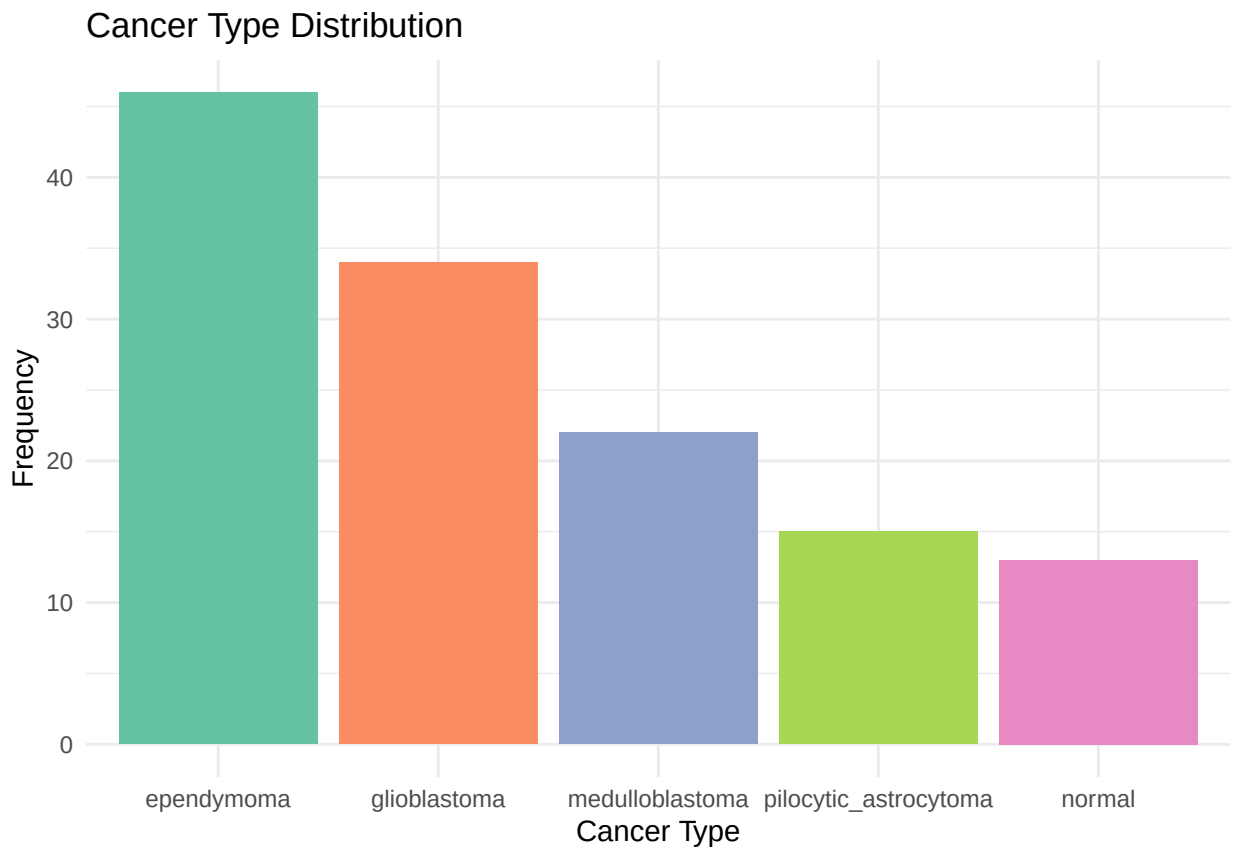
```r
subset_dataset <- dataset %>% select(samples, type, 3:5, 147:150)

type_count <- table(subset_dataset$type)
the_most_occurring_type_of_cancer <- names(which.max(type_count))
print(paste("The most occurring type of cancer is:", the_most_occurring_type_of_cancer))
```

```
## [1] "The most occurring type of cancer is: ependymoma"
```

```r
library(ggplot2)
cancer_dataframe <- as.data.frame(type_count)
colnames(cancer_dataframe) <- c("Type", "Count")

ggplot(cancer_dataframe, aes(x = reorder(Type, -Count), y = Count, fill = Type)) +
  geom_bar(stat = "identity") +
  scale_fill_brewer(palette = "Set2") +  # Use different colors for each type
  labs(title = "Cancer Type Distribution",
       x = "Cancer Type",
       y = "Frequency") +
  theme_minimal() +
  theme(legend.position = "none")  # Hide legend if unnecessary
```



Data Cleaning and Filtering

```
print(paste("-The number of NA in dataset is", sum(is.na(dataset))))
```

```
## [1] "-The number of NA in dataset is 0"
```

```
filtered_dataset <- dataset %>% filter(X1007_s_at > 12)
print(paste("-The number of rows before filtering is", nrow(dataset)))
```

```
## [1] "-The number of rows before filtering is 130"
```

```
print(paste("-The number of rows after filtering is", nrow(filtered_dataset)))
```

```
## [1] "-The number of rows after filtering is 91"
```

Data Analysis

Genes Analysis

```
genes <- dataset %>% select(!(1:2))

mean_summary <- summarise(genes, across(where(is.numeric),
                                        \(x) mean(x, na.rm = TRUE)))

sd_summary <- summarise(genes, across(where(is.numeric),
                                      \(x) sd(x, na.rm = TRUE)))
```

```r
gene_summary <- bind_rows(mean_summary, sd_summary) %>%
  mutate(Summary = c("mean", "sd")) %>%
  select(Summary, everything())
```

##Genes Analysis By Type

```r
library(dplyr)
library(tidyr)
grouped_summary <- dataset %>%
  group_by(type) %>%
  summarise(across(starts_with("X"), list(mean = ~mean(.x, na.rm = TRUE), sd = ~sd(.x, na.rm = TRUE)))) %>%
  pivot_longer(-type, names_to = c("Gene", "Measure"), names_pattern = "(.*)_(mean|sd)") %>%
  pivot_wider(names_from = Gene, values_from = value) %>%
  mutate(Measure = paste(Measure, type, sep = "_")) %>%
  select(-type)

colnames(grouped_summary)[1] <- "measure"

print(grouped_summary)
```

```
## # A tibble: 10 x 149
##    measure      X1007_s_at X1053_at X117_at X121_at X1255_g_at X1294_at X1316_at
##    <chr>             <dbl>    <dbl>   <dbl>   <dbl>      <dbl>    <dbl>    <dbl>
##  1 mean_ependy~     12.8      8.57    7.96    9.19       4.39     8.17     6.72
##  2 sd_ependymo~      0.355    0.523   1.13    0.599      0.573    0.572    0.525
##  3 mean_gliobl~     12.4      9.25    8.21    9.22       4.87     8.08     6.65
##  4 sd_glioblas~      0.484    0.621   0.972   0.607      0.830    0.647    0.481
##  5 mean_medull~     11.2      9.10    6.94    8.95       4.55     7.37     6.88
##  6 sd_medullob~      0.541    0.520   0.533   0.723      0.607    0.321    0.529
##  7 mean_normal      11.3      8.04    7.07    9.07       6.05     7.46     7.35
##  8 sd_normal         0.581    0.578   0.905   0.380      1.07     0.348    0.518
##  9 mean_pilocy~     12.9      8.44    7.60    9.33       5.53     8.43     6.79
## 10 sd_pilocyti~      0.288    0.481   0.565   0.665      0.990    0.405    0.456
## # i 141 more variables: X1320_at <dbl>, X1405_i_at <dbl>, X1431_at <dbl>,
## #   X1438_at <dbl>, X1487_at <dbl>, X1494_f_at <dbl>, X1552256_a_at <dbl>,
## #   X1552257_a_at <dbl>, X1552258_at <dbl>, X1552261_at <dbl>,
## #   X1552263_at <dbl>, X1552264_a_at <dbl>, X1552266_at <dbl>,
## #   X1552269_at <dbl>, X1552271_at <dbl>, X1552272_a_at <dbl>,
## #   X1552274_at <dbl>, X1552275_s_at <dbl>, X1552276_a_at <dbl>,
## #   X1552277_a_at <dbl>, X1552278_a_at <dbl>, X1552279_a_at <dbl>, ...
```

Save summaries to csv files

```r
save_to_csv <- function(ds, path) {
  if(!endsWith(path, ".csv")){
    path <- paste0(path, ".csv")
  }

  write.csv(ds, path, row.names = TRUE)
}

save_to_csv(gene_summary, "gene_summary.csv")
save_to_csv(grouped_summary, "grouped_summary.csv")
```