

Lab_1.R

r3382487

2025-09-10

```
##### PSY 503
##### Lab_1
##### Diana Kim

### 1 ###
install.packages("coursekata")

## Installing package into '/cloud/lib/x86_64-pc-linux-gnu-library/4.5'
## (as 'lib' is unspecified)
install.packages("tidyverse")

## Installing package into '/cloud/lib/x86_64-pc-linux-gnu-library/4.5'
## (as 'lib' is unspecified)
install.packages("datasauRus")

## Installing package into '/cloud/lib/x86_64-pc-linux-gnu-library/4.5'
## (as 'lib' is unspecified)
library(coursekata)

## Registered S3 method overwritten by 'mosaic':
##   method           from
##   fortify.SpatialPolygonsDataFrame ggplot2

## Loading required package: dslabs
## Loading required package: Lock5withR
## Loading required package: fivethirtyeightdata
## Loading required package: fivethirtyeight
## Loading required package: Metrics
## Loading required package: lsr
## Loading required package: mosaic
##
## The 'mosaic' package masks several functions from core packages in order to add
## additional features. The original behavior of these functions should not be affected by this.
##
## Attaching package: 'mosaic'

## The following objects are masked from 'package:dplyr':
##
```

```

##      count, do, tally
## The following object is masked from 'package:Matrix':
##
##      mean
## The following object is masked from 'package:ggplot2':
##
##      stat
## The following objects are masked from 'package:stats':
##
##      binom.test, cor, cor.test, cov, fivenum, IQR, median, prop.test,
##      quantile, sd, t.test, var
## The following objects are masked from 'package:base':
##
##      max, mean, min, prod, range, sample, sum
## Loading required package: supernova

## -- CourseKata packages ----- coursekata 0.19.0 --
## v dslabs          0.8.0     v Metrics        0.1.4
## v Lock5withR      1.2.2     v lsr           0.5.2
## v fivethirtyeightdata 0.1.0   v mosaic         1.9.2
## v fivethirtyeight 0.6.2     v supernova      3.0.0
##
## Attaching package: 'coursekata'
##
##
## The following object is masked from 'package:datasets':
##
##      penguins
library(tidyverse)

## -- Attaching core tidyverse packages ----- tidyverse 2.0.0 --
## vforcats    1.0.0     v stringr     1.5.2
## v lubridate  1.9.4     v tibble      3.3.0
## v purrr     1.1.0     v tidyr       1.3.1
## v readr     2.1.5
## -- Conflicts ----- tidyverse_conflicts() --
## x mosaic::count() masks dplyr::count()
## x purrr::cross()  masks mosaic::cross()
## x mosaic::do()    masks dplyr::do()
## x tidyr::expand() masks Matrix::expand()
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## x tidyr::pack()   masks Matrix::pack()
## x mosaic::stat()  masks ggplot2::stat()
## x mosaic::tally() masks dplyr::tally()
## x tidyr::unpack() masks Matrix::unpack()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
library(datasauRus)

```

```

#### 2 ####
lab1_data <- datasaurus_dozen
str(lab1_data)

## # tibble [1,846 x 3] (S3: tbl_df/tbl/data.frame)
## $ dataset: chr [1:1846] "dino" "dino" "dino" "dino" ...
## $ x      : num [1:1846] 55.4 51.5 46.2 42.8 40.8 ...
## $ y      : num [1:1846] 97.2 96 94.5 91.4 88.3 ...
## - attr(*, "spec")=
##   .. cols(
##     .. dataset = col_character(),
##     .. x = col_double(),
##     .. y = col_double()
##   )

# what are these...??
# 'dataset' column shows nominal values, and the other two are probably intervals...??
# ohhhh I figure it out after plotting the dots. They are coordinates of different shapes!!
# so x and y coordinates should be ratio variables I guess

print(lab1_data$dataset)

## #> [1] "dino"      "dino"      "dino"      "dino"      "dino"
## #> [6] "dino"      "dino"      "dino"      "dino"      "dino"
## #> [11] "dino"     "dino"      "dino"      "dino"      "dino"
## #> [16] "dino"     "dino"      "dino"      "dino"      "dino"
## #> [21] "dino"     "dino"      "dino"      "dino"      "dino"
## #> [26] "dino"     "dino"      "dino"      "dino"      "dino"
## #> [31] "dino"     "dino"      "dino"      "dino"      "dino"
## #> [36] "dino"     "dino"      "dino"      "dino"      "dino"
## #> [41] "dino"     "dino"      "dino"      "dino"      "dino"
## #> [46] "dino"     "dino"      "dino"      "dino"      "dino"
## #> [51] "dino"     "dino"      "dino"      "dino"      "dino"
## #> [56] "dino"     "dino"      "dino"      "dino"      "dino"
## #> [61] "dino"     "dino"      "dino"      "dino"      "dino"
## #> [66] "dino"     "dino"      "dino"      "dino"      "dino"
## #> [71] "dino"     "dino"      "dino"      "dino"      "dino"
## #> [76] "dino"     "dino"      "dino"      "dino"      "dino"
## #> [81] "dino"     "dino"      "dino"      "dino"      "dino"
## #> [86] "dino"     "dino"      "dino"      "dino"      "dino"
## #> [91] "dino"     "dino"      "dino"      "dino"      "dino"
## #> [96] "dino"     "dino"      "dino"      "dino"      "dino"
## #> [101] "dino"    "dino"      "dino"      "dino"      "dino"
## #> [106] "dino"    "dino"      "dino"      "dino"      "dino"
## #> [111] "dino"    "dino"      "dino"      "dino"      "dino"
## #> [116] "dino"    "dino"      "dino"      "dino"      "dino"
## #> [121] "dino"    "dino"      "dino"      "dino"      "dino"
## #> [126] "dino"    "dino"      "dino"      "dino"      "dino"
## #> [131] "dino"    "dino"      "dino"      "dino"      "dino"
## #> [136] "dino"    "dino"      "dino"      "dino"      "dino"
## #> [141] "dino"    "dino"      "away"      "away"      "away"
## #> [146] "away"    "away"      "away"      "away"      "away"
## #> [151] "away"    "away"      "away"      "away"      "away"
## #> [156] "away"    "away"      "away"      "away"      "away"
## #> [161] "away"    "away"      "away"      "away"      "away"

```



```

## [1786] "wide_lines" "wide_lines" "wide_lines" "wide_lines" "wide_lines"
## [1791] "wide_lines" "wide_lines" "wide_lines" "wide_lines" "wide_lines"
## [1796] "wide_lines" "wide_lines" "wide_lines" "wide_lines" "wide_lines"
## [1801] "wide_lines" "wide_lines" "wide_lines" "wide_lines" "wide_lines"
## [1806] "wide_lines" "wide_lines" "wide_lines" "wide_lines" "wide_lines"
## [1811] "wide_lines" "wide_lines" "wide_lines" "wide_lines" "wide_lines"
## [1816] "wide_lines" "wide_lines" "wide_lines" "wide_lines" "wide_lines"
## [1821] "wide_lines" "wide_lines" "wide_lines" "wide_lines" "wide_lines"
## [1826] "wide_lines" "wide_lines" "wide_lines" "wide_lines" "wide_lines"
## [1831] "wide_lines" "wide_lines" "wide_lines" "wide_lines" "wide_lines"
## [1836] "wide_lines" "wide_lines" "wide_lines" "wide_lines" "wide_lines"
## [1841] "wide_lines" "wide_lines" "wide_lines" "wide_lines" "wide_lines"
## [1846] "wide_lines"

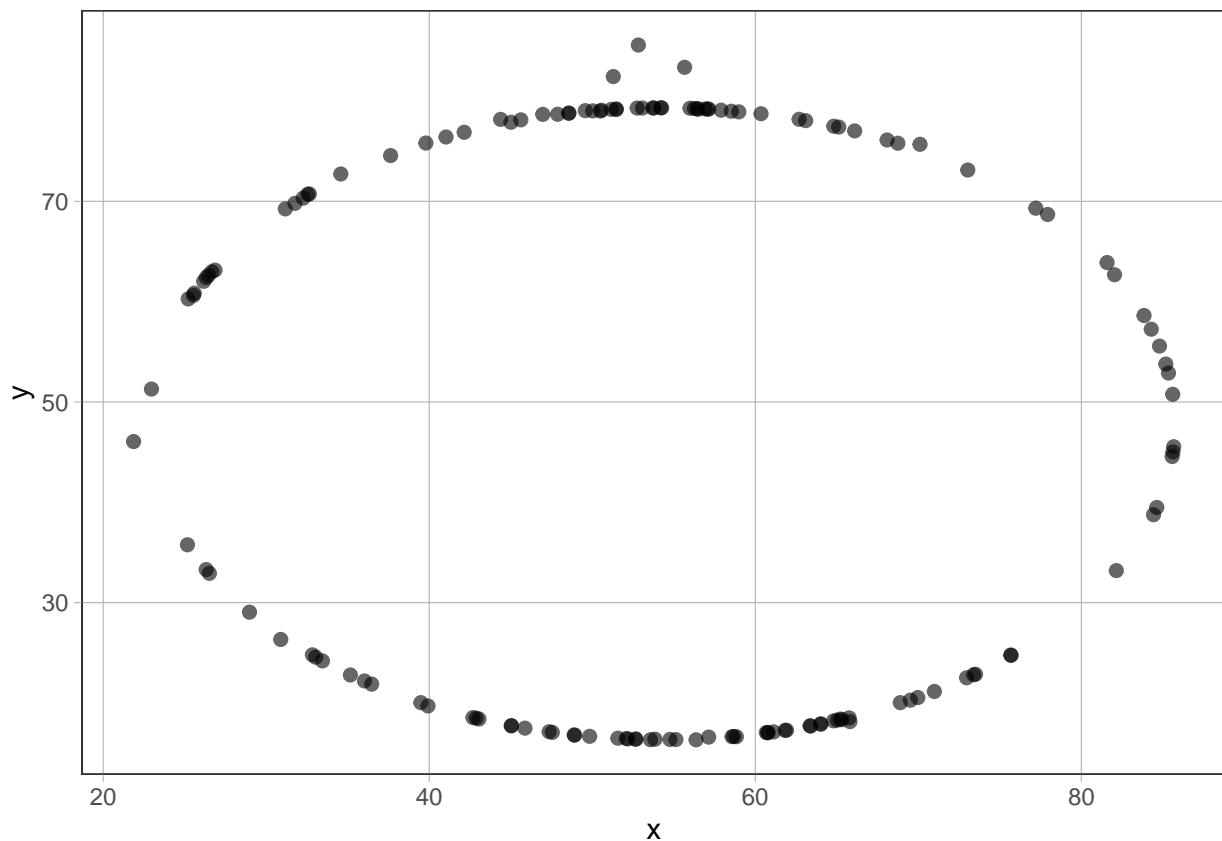
print(unique(lab1_data$dataset))

## [1] "dino"      "away"       "h_lines"    "v_lines"    "x_shape"
## [6] "star"       "high_lines"  "dots"      "circle"     "bullseye"
## [11] "slant_up"   "slant_down"  "wide_lines"
# it's printing only the unique values in 'dataset' column

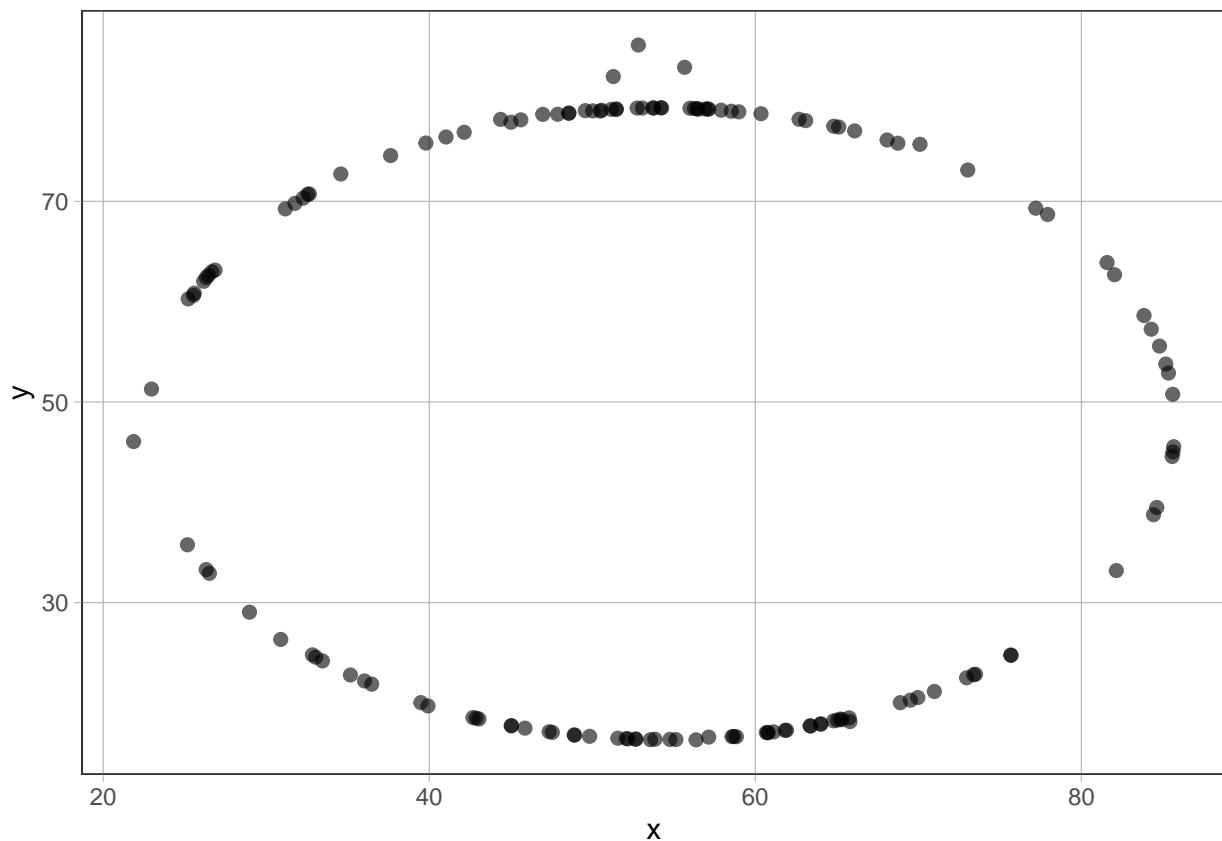
#### 3 ####
library(ggplot2)
temp_data <- lab1_data %>%
  filter(dataset == "circle")

ggplot(data = temp_data, aes(x = x, y = y)) +
  geom_point()

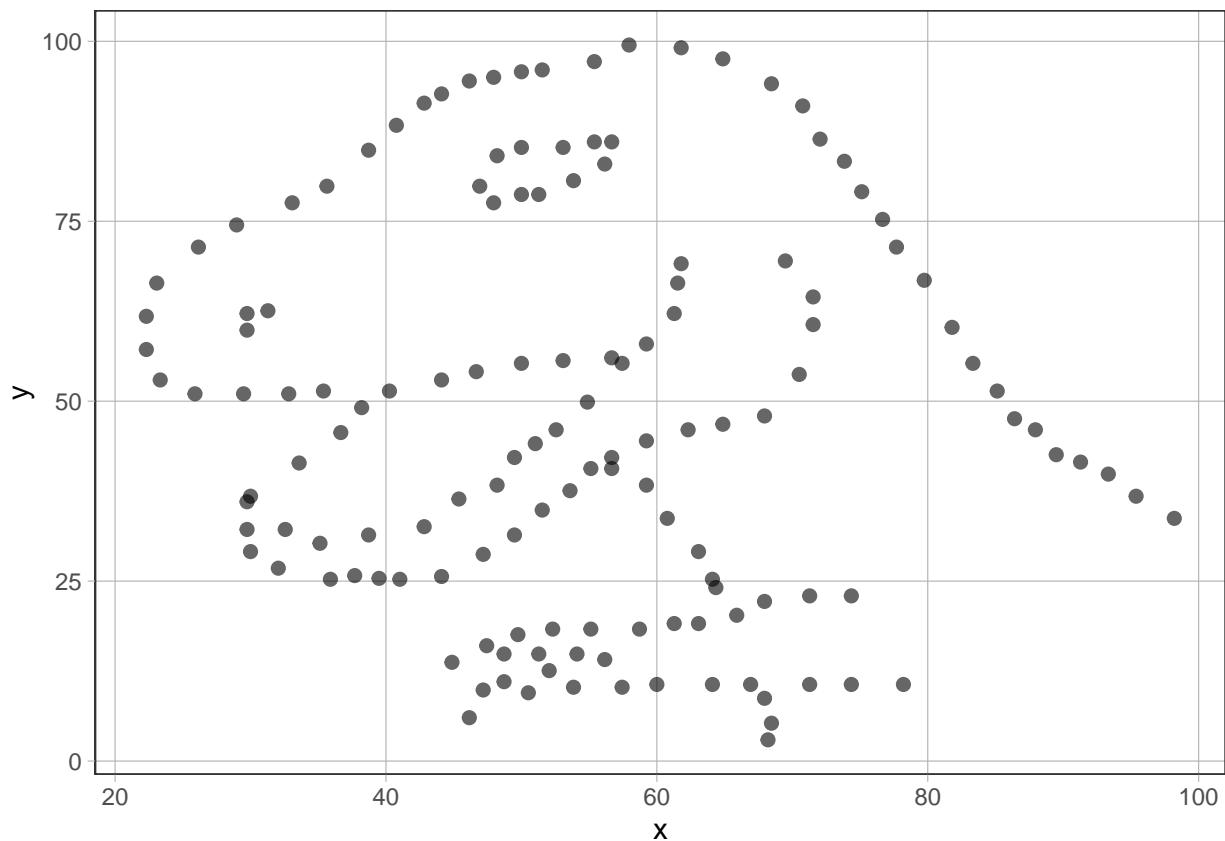
```



```
gf_point (y ~ x, data = temp_data)
```



```
#wow this is cute
dinodata <- lab1_data %>%
  filter(dataset == "dino")
ggplot(data = dinodata, aes(x = x, y = y)) +
  geom_point()
```



```
gf_point (y ~ x, data = dinodata)
```

