

Assignment1

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.

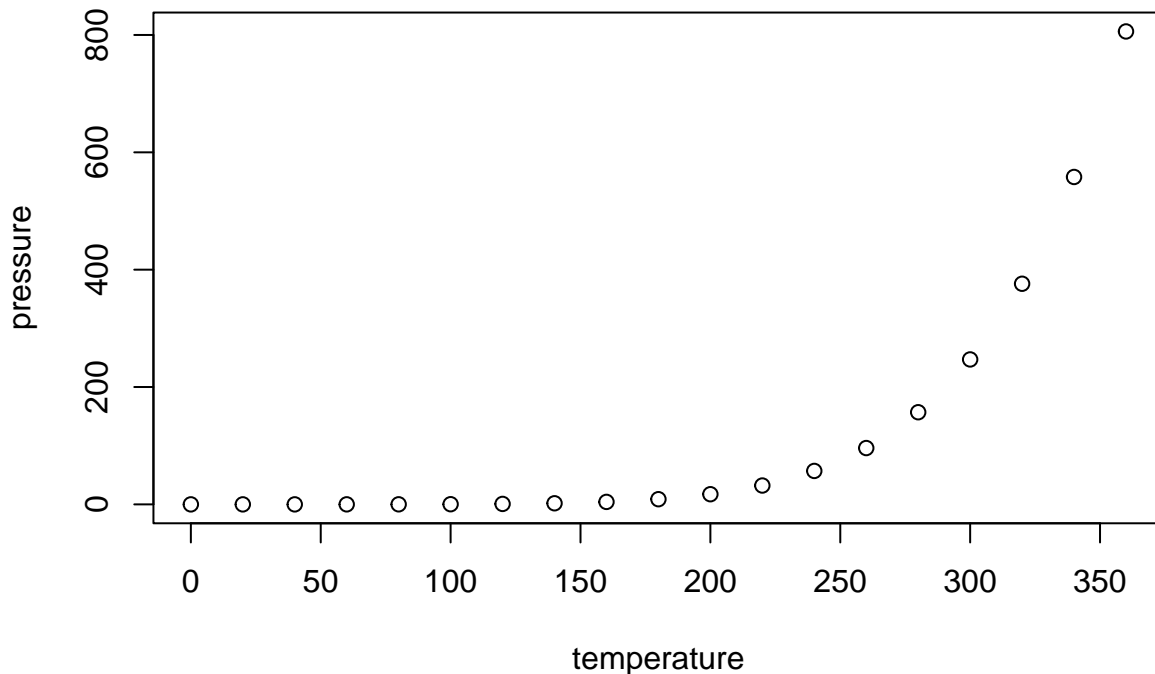
When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```
summary(cars)
```

```
##      speed      dist
##  Min.   : 4.0    Min.   : 2.00
## 1st Qu.:12.0    1st Qu.: 26.00
##  Median :15.0    Median : 36.00
##   Mean  :15.4    Mean   : 42.98
## 3rd Qu.:19.0    3rd Qu.: 56.00
##   Max.  :25.0    Max.    :120.00
```

Including Plots

You can also embed plots, for example:



Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.

```
#hand_function
```

```

sum_special <- function(df_x){

  ## sum_special calculates data summary statistics
  ## the input param df_x is the data frame of input values

  # browser() # browser() will start the debugger
  # if the line is uncommented

  ## test the input data to assure that it is a data frame.
  try(if(!is.data.frame(df_x)) stop("Input data must be a data frame."))

  #calculate means,var,cov,cor
  sp_means <- apply(df_x, MARGIN = 2, FUN = mean)
  sp_var <- apply(df_x, MARGIN = 2, FUN = var)
  sp_cov <- cov(df_x)
  sp_cor <- cor(df_x)

  ## Note that defining a list with the
  ## syntax list(list_name = list_content) produces
  ## named list items
  sp_outputs <- list(sp_means=sp_means,
                    sp_var = sp_var,
                    sp_cov = sp_cov,
                    sp_cor = sp_cor)

  return(sp_outputs)
}

```

```
#car_viz
```

```
library(tidyverse)
```

```

## -- Attaching packages ----- tidyverse 1.3.1 --

## v ggplot2 3.3.5      v purrr   0.3.4
## v tibble  3.1.4      v dplyr   1.0.7
## v tidyr   1.1.3      v stringr 1.4.0
## v readr   2.0.1      v forcats 0.5.1

## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()

```

call built-in data mtcars.

```
data(mtcars)
```

Select only car models where mpg<20

```
mtcars_mpg2 <- mtcars[mtcars$mpg < 20,]
```

Reduce the variables to mpg, cyl, disp, hp, gears

```
mtcars_mpg2 <- mtcars_mpg2[, c(1,2,3,4,10)]
```

read the R file hand_functions.R so that it can be used

notice that with echo = TRUE

```
source(file = "hand_functions(2).R", echo = TRUE)

##
## > sum_special <- function(df_x) {
## +   try(if (!is.data.frame(df_x))
## +     stop("Input data must be a data frame."))
## +   sp_means <- apply(df_ .... [TRUNCATED]
```

Now use the function from hand_functions.R

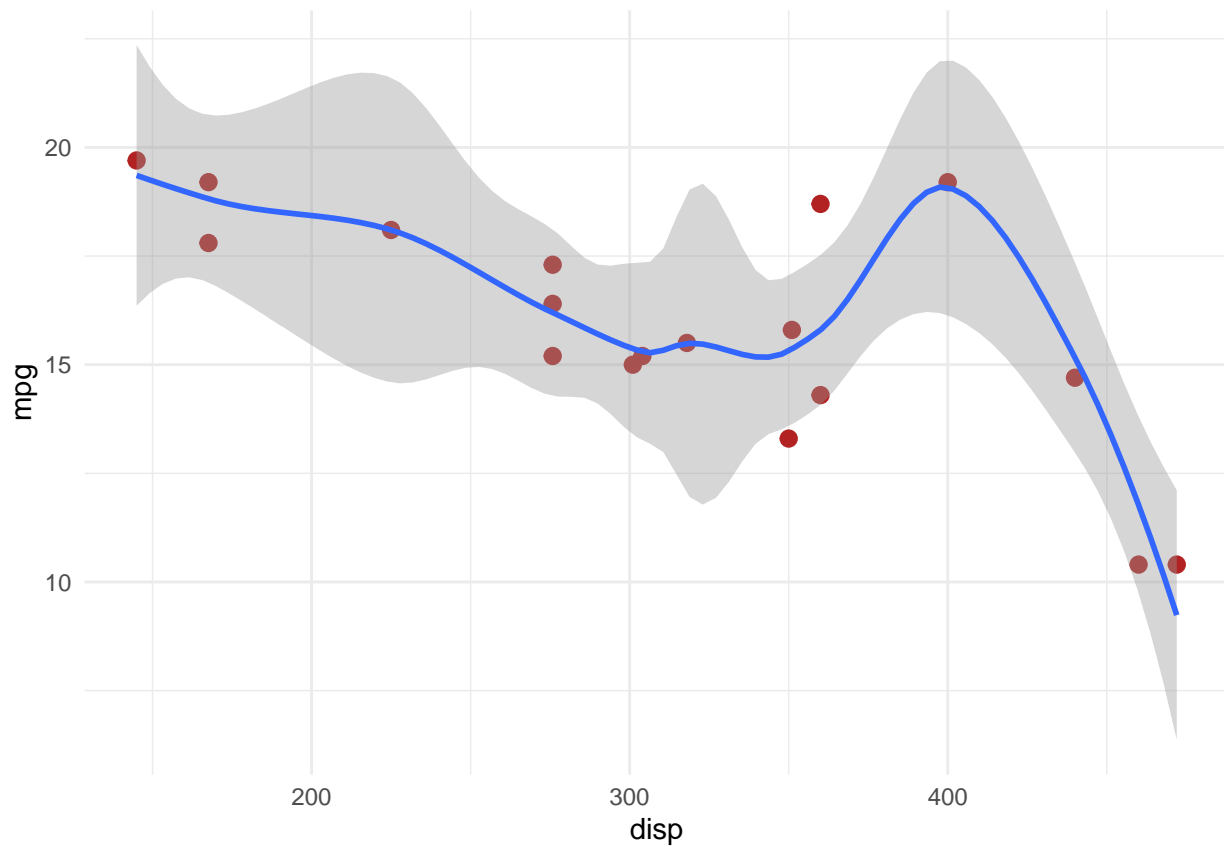
```
sp_out <- sum_special(mtcars_mpg2)
```

library(esquisse)

esquisser(data = mtcars_mpg2, viewer = "browser")

```
#use ggplot to evaluate plot
ggplot(mtcars_mpg2) +
  aes(x = disp, y = mpg) +
  geom_point(shape = "bullet", size = 4L, colour = "#B22222") +
  geom_smooth(span = 0.5) +
  theme_minimal()

## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



note that this boxplot cannot be made with `esquisse()` unless the data is adjusted. What adjustment is needed?

```
#use ggplot to evaluate boxplot
ggplot(mtcars_mpg2, aes(x=as.factor(cyl), y=mpg)) +
  geom_boxplot(fill="slateblue", alpha=0.2) +
  xlab("cyl")
```

