Integration of AI and R Programming

Dadong Zhang

Demonstration of inplementing OpenAI in R programming for generating neat document in PDF, Presentation, and Word.

```r
rm(list = ls())
library(tidyverse)
library(ggplot2)
library(readxl)
library(writexl)
library(httr)
library(jsonlite)
```

```
Attaching package: 'jsonlite'

The following object is masked from 'package:purrr':

    flatten
```

```r
library(flextable)
```

```
Attaching package: 'flextable'

The following object is masked from 'package:purrr':

    compose
```

```r
library(table1)
```

```
Attaching package: 'table1'


The following objects are masked from 'package:base':

    units, units<-
```

```r
set.seed(123)
options(width = 70)
source("src/func.R")
```

Ask AI to generate a dummey patient characteristic, demogrphaic, baseline, treatment, and binary clinical response data set

```r
question <- "Can you give R code to generate a dummy data frame for 100
 ↪  patients with their characteristics such as gender, age, bmi, height,
 ↪  race, baseline, treatment, and binary clinical response data set?"
result1 <- chatgpt_query_with_history(question)
cat("Assistant's answer:\n", result1$answer, "\n")
```

~~~
Assistant's answer:
 Sure! Here is R code to generate a dummy data frame for 100 patients with the characteristi

```R
# Set seed for reproducibility
set.seed(123)

# Generate dummy data for 100 patients
gender <- sample(c("Male", "Female"), 100, replace = TRUE)
age <- round(rnorm(100, mean = 50, sd = 10))
bmi <- round(rnorm(100, mean = 25, sd = 5), 1)
height <- round(rnorm(100, mean = 170, sd = 10), 1)
race <- sample(c("White", "Black", "Asian", "Hispanic"), 100, replace = TRUE)
baseline <- round(runif(100, min = 100, max = 200), 1)
treatment <- sample(c("Placebo", "Drug A", "Drug B"), 100, replace = TRUE)
~~~

```
clinical_response <- sample(c(0, 1), 100, replace = TRUE)

# Create a data frame
patients <- data.frame(Gender = gender, Age = age, BMI = bmi, Height = height, Race = race,
                       Baseline = baseline, Treatment = treatment, Clinical_Response = clini
                       
# Display the first few rows of the data frame
head(patients)
```
```

This code will create a data frame `patients` with 100 rows and the following columns: Gende
```

```
# Optionally, extract R code:
eval(parse(text = extract_r_code(result1)))
```

```
  Gender Age  BMI Height     Race Baseline Treatment
1   Male  53 28.9  166.2    Black    147.1   Placebo
2   Male  50 28.8  164.4    White    136.6   Placebo
3   Male  50 26.7  166.6    White    112.1    Drug A
4 Female  64 20.0  170.9    Asian    104.7    Drug B
5   Male  48 24.4  186.0 Hispanic    126.3    Drug B
6 Female  65 23.6  169.1    White    196.9    Drug A
  Clinical_Response
1                 1
2                 0
3                 0
4                 0
5                 0
6                 0
```

Continue asking AI to make a characteristics table for the patients betweo treatment

```
question2 <- "Can you give me the R code to generate the patient
 ↪  characteristics
table using R package table1 between treatment? I already installed the
 ↪  package."
result2 <- chatgpt_query_with_history(question2, history = result1$history)
cat("Assistant's answer:\n", result2$answer, "\n")
```

~~~
Assistant's answer:
 Sure! To generate a patient characteristics table using the `table1` package in R between th

```R
# Load the required library
library(table1)

# Generate the patient characteristics table between treatment groups
table1(~ Gender + Age + BMI + Height + Race + Baseline + Clinical_Response | Treatment, data
```

In this code:
- `~ Gender + Age + BMI + Height + Race + Baseline + Clinical_Response | Treatment` specifies
- `data = patients` specifies that the data for the analysis is stored in the `patients` data

When you run this code, you will get a table summarizing the patient characteristics by trea
~~~

```r
eval(parse(text = extract_r_code(result2)))
```

	Drug A	Drug B	Placebo	Overall
	(N=35)	(N=38)	(N=27)	(N=100)
Gender				
Female	14 (40.0%)	18 (47.4%)	11 (40.7%)	43 (43.0%)
Male	21 (60.0%)	20 (52.6%)	16 (59.3%)	57 (57.0%)
Age				
Mean (SD)	48.5 (10.3)	50.0 (9.17)	49.8 (9.65)	49.5 (9.62)
Median [Min, Max]	48.0 [27.0, 72.0]	49.5 [35.0, 71.0]	51.0 [29.0, 69.0]	49.5 [27.0, 72.0]
BMI				
Mean (SD)	25.4 (3.86)	24.5 (5.40)	25.4 (4.64)	25.1 (4.68)
Median [Min, Max]	25.3 [18.2, 33.4]	23.2 [18.4, 41.2]	25.5 [18.4, 34.8]	24.1 [18.2, 41.2]
Height				
Mean (SD)	173 (9.98)	170 (10.1)	170 (9.53)	171 (9.92)
Median [Min, Max]	174 [153, 190]	171 [150, 193]	170 [153, 190]	171 [150, 193]
Race				
Asian	5 (14.3%)	10 (26.3%)	6 (22.2%)	21 (21.0%)
Black	5 (14.3%)	9 (23.7%)	8 (29.6%)	22 (22.0%)
Hispanic	11 (31.4%)	9 (23.7%)	7 (25.9%)	27 (27.0%)
White	14 (40.0%)	10 (26.3%)	6 (22.2%)	30 (30.0%)
Baseline				
Mean (SD)	152 (30.6)	147 (30.1)	147 (26.3)	149 (29.1)
Median [Min, Max]	157 [100, 199]	149 [101, 199]	149 [101, 181]	152 [100, 199]
Clinical_Response				
Mean (SD)	0.457 (0.505)	0.500 (0.507)	0.444 (0.506)	0.470 (0.502)
Median [Min, Max]	0 [0, 1.00]	0.500 [0, 1.00]	0 [0, 1.00]	0 [0, 1.00]

Continue asking AI to make a bar plot for the response rate between the treatment groups

```r
question3 <- "Can you give me the R code to make a barplot for the response
rates between treatment groups?"
result3 <- chatgpt_query_with_history(question3, history = result1$history)
cat("Assistant's answer:\n", result3$answer, "\n")
```

```
Assistant's answer:
 Sure! Here is the R code to create a barplot showing the response rates between different t
```

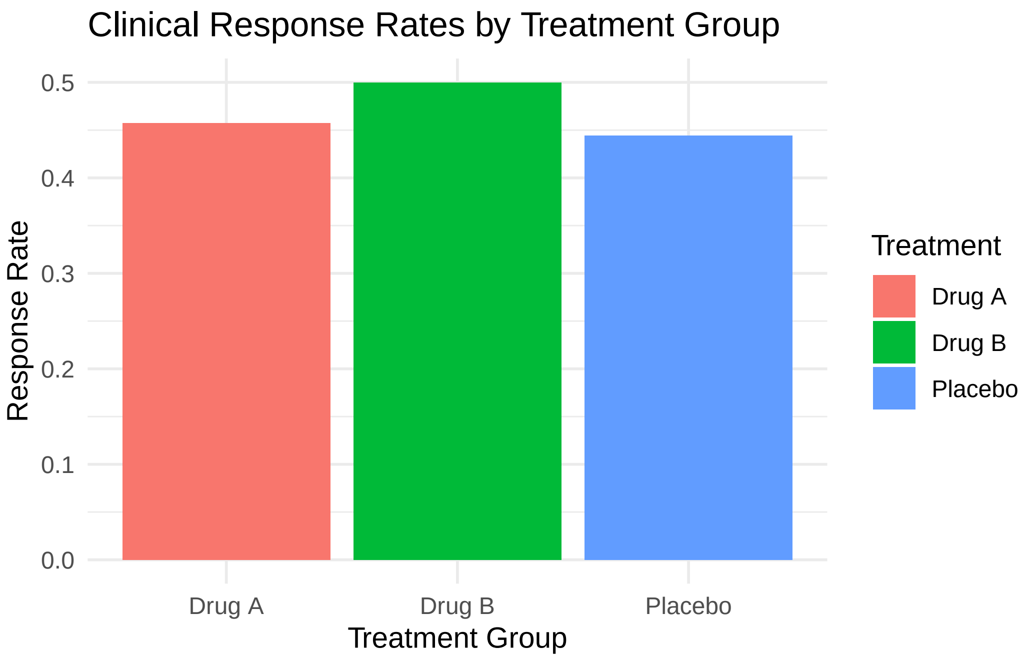
~~~~
```R
# Load the necessary library
library(ggplot2)

# Create a summary table of response rates by treatment group
response_summary <- aggregate(Clinical_Response ~ Treatment, data = patients, FUN = mean)

# Create a bar plot showing response rates by treatment group
ggplot(response_summary, aes(x = Treatment, y = Clinical_Response, fill = Treatment)) +
  geom_bar(stat = "identity") +
  labs(title = "Clinical Response Rates by Treatment Group",
       x = "Treatment Group",
       y = "Response Rate") +
  theme_minimal()
```

This code will generate a barplot displaying the clinical response rates for each treatment g
~~~~

```r
eval(parse(text = extract_r_code(result3)))
```



Continue asking AI to statistical test for the response rate between the treatment groups

```r
question4 <- "Can you give me the R code to evaluate statistical differences
 ↪  for
the response rates between treatment groups?"
result4 <- chatgpt_query_with_history(question4, history = result1$history)
cat("Assistant's answer:\n", result4$answer, "\n")
```

~~~~
Assistant's answer:
 Sure! To evaluate statistical differences in response rates between treatment groups, you ca

```R
# Create a contingency table of Treatment and Clinical_Response
cont_table <- table(patients$Treatment, patients$Clinical_Response)

# Perform a chi-squared test
chi_test <- chisq.test(cont_table)

# Display the results
print(chi_test)
```

This code will create a contingency table of Treatment and Clinical_Response in the `patients
~~~~

```r
eval(parse(text = extract_r_code(result4)))
```

```
	Pearson's Chi-squared test

data:  cont_table
X-squared = 0.23131, df = 2, p-value = 0.8908
```