Name : Karan Kurani
UID : U01932963
Subject : Computational Statistics
Professor : Paul Dantzig (Adjunct Professor)

Homework 5 -- Chapter 6: Learning checks 6.1 - 6.3.

(LC6.1) Compute the observed values, fitted values, and residuals not for the interaction model as we just did, but rather for the parallel slopes model we saved in score_model_parallel_slopes.

Answer :
library(tidyverse)
library(moderndive)
library(skimr)
library(ISLR)

regression_points_parallel <- get_regression_points(score_model_parallel_slopes)
regression_points_parallel

```
RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function    Addins
Console  Terminal  Background Jobs
R 4.3.1 · ~/
> library(tidyverse)
> library(moderndive)
> library(skimr)
> library(ISLR)
> regression_points_parallel <- get_regression_points(score_model_parallel_slopes)
> regression_points_parallel
# A tibble: 463 × 6
      ID score   age gender score_hat residual
   <int> <dbl> <int> <fct>      <dbl>    <dbl>
 1     1   4.7    36 female      4.17    0.528
 2     2   4.1    36 female      4.17   -0.072
 3     3   3.9    36 female      4.17   -0.272
 4     4   4.8    36 female      4.17    0.628
 5     5   4.6    59 male        4.16    0.437
 6     6   4.3    59 male        4.16    0.137
 7     7   2.8    59 male        4.16   -1.36
 8     8   4.1    51 male        4.23   -0.132
 9     9   3.4    51 male        4.23   -0.832
10    10   4.5    40 female      4.14    0.363
# ℹ 453 more rows
# ℹ Use `print(n = ...)` to see more rows
>
```

(LC6.2) Conduct a new exploratory data analysis with the same outcome variable y debt but with credit_rating and age as the new explanatory variables x 1 and x 2 . What can you say about the relationship between a credit card holder's debt and their credit rating and age?

Answer :

credit %>%
 select(debt, credit_rating, age) %>%
 head()

```
R 4.3.1 · ~/
> credit %>%
+     select(debt, credit_rating, age) %>%
+     head()
# A tibble: 6 × 3
   debt credit_rating   age
  <int>         <int> <int>
1   333           283    34
2   903           483    82
3   580           514    71
4   964           681    36
5   331           357    68
6  1151           569    77
>
```

skim_with(numeric = list(hist = NULL), integer = list(hist = NULL))

```
> skim_with(numeric = list(hist = NULL), integer = list(hist = NULL))
Creating new skimming functions for the following classes: hist.
They did not have recognized defaults. Call get_default_skimmers() for more information.
function (data, ..., .data_name = NULL)
{
    if (is.null(.data_name)) {
        .data_name <- rlang::expr_label(substitute(data))
    }
    if (!inherits(data, "data.frame")) {
        data <- as.data.frame(data)
    }
    stopifnot(inherits(data, "data.frame"))
    selected <- names(tidyselect::eval_select(rlang::expr(c(...)),
        data))
    if (length(selected) == 0) {
        selected <- names(data)
    }
    grps <- dplyr::groups(data)
    if (length(grps) > 0) {
        group_variables <- selected %in% as.character(grps)
        selected <- selected[!group_variables]
    }
    else {
        attr(data, "groups") <- list()
    }
    skimmers <- purrr::map(selected, get_final_skimmers, data,
        local_skimmers, append)
    types <- purrr::map_chr(skimmers, "skim_type")
    unique_skimmers <- reduce_skimmers(skimmers, types)
    combined_skimmers <- purrr::map(unique_skimmers, join_with_base,
        base)
    ready_to_skim <- tibble::tibble(skim_type = unique(types),
        skimmers = purrr::map(combined_skimmers, mangle_names,
            names(base$funs)), skim_variable = split(selected,
            types)[unique(types)])
    grouped <- dplyr::group_by(ready_to_skim, .data$skim_type)
    nested <- dplyr::summarize(grouped, skimmed = purrr::map2(.data$skimmers,
        .data$skim_variable, skim_by_type, data))
    structure(tidyr::unnest(nested, "skimmed"), class = c("skim_df",
        "tbl_df", "tbl", "data.frame"), data_rows = nrow(data),
        data_cols = ncol(data), df_name = .data_name, dt_key = get_dt_key(data),
        groups = dplyr::group_vars(data), base_skimmers = names(base$funs),
        skimmers_used = get_skimmers_used(unique_skimmers))
}
<bytecode: 0x000001b93cd5ccf0>
<environment: 0x000001b941f1be68>
```

U01932963

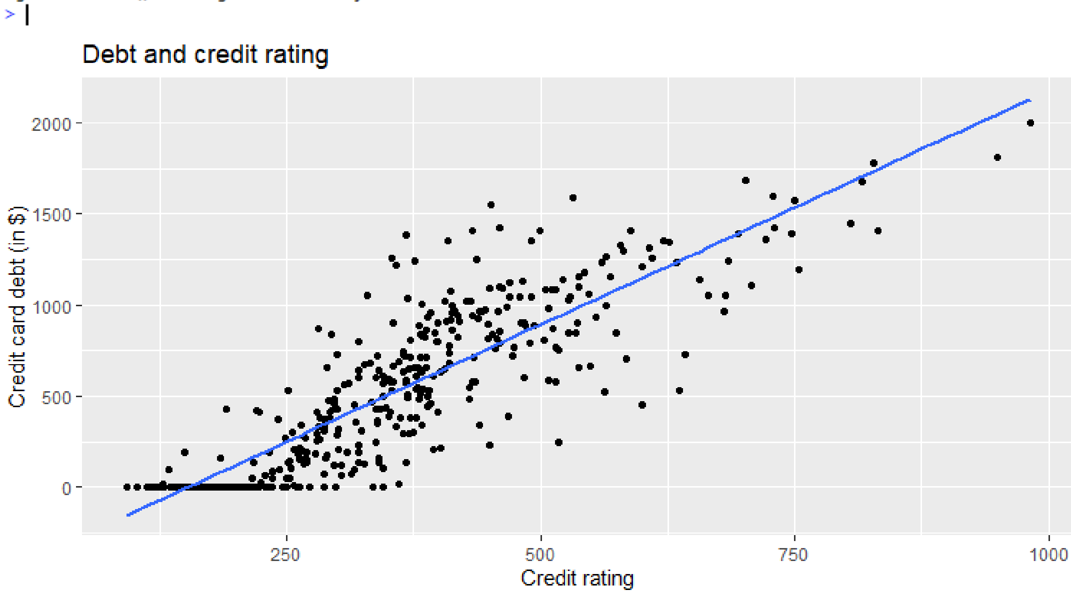
```
credit %>%
  select(debt, credit_rating, age) %>%
  skim()
```

```
> credit %>%
+     select(debt, credit_rating, age) %>%
+     skim()
── Data Summary ────────────────────────
                           Values
Name                       Piped data
Number of rows             400
Number of columns          3
_______________________
Column type frequency:
  numeric                  3
________________________
Group variables            None

── Variable type: numeric ──────────────────────────────────────────────────────
  skim_variable n_missing complete_rate  mean    sd p0   p25  p50  p75 p100 hist
1 debt                  0             1 520.  460.   0  68.8 460. 863  1999 ▇▃▂▂▁
2 credit_rating         0             1 355.  155.  93 247.  344  437.  982 ▇▇▂▁▁
3 age                   0             1  55.7  17.2 23  41.8  56   70    98 ▇▇▇▇▁
>
```

```
ggplot(credit, aes(x = credit_rating, y = debt)) +
  geom_point() +
  labs(
    x = "Credit rating", y = "Credit card debt (in $)",
    title = "Debt and credit rating"
  ) +
  geom_smooth(method = "lm", se = FALSE)
```

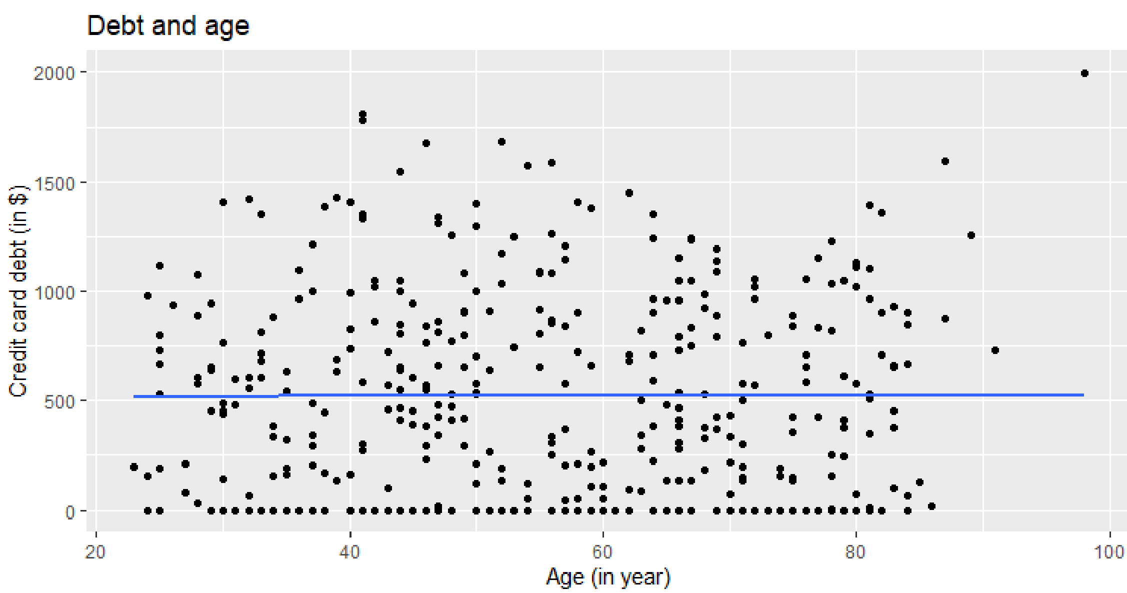
```
> ggplot(credit, aes(x = credit_rating, y = debt)) +
+     geom_point() +
+     labs(
+         x = "Credit rating", y = "Credit card debt (in $)",
+         title = "Debt and credit rating"
+     ) +
+     geom_smooth(method = "lm", se = FALSE)
`geom_smooth()` using formula = 'y ~ x'
>
```



U01932963

```
ggplot(credit, aes(x = age, y = debt)) +
 geom_point() +
 labs(
  x = "Age (in year)", y = "Credit card debt (in $)",
  title = "Debt and age"
 ) +
 geom_smooth(method = "lm", se = FALSE)
```

```
> ggplot(credit, aes(x = age, y = debt)) +
+     geom_point() +
+     labs(
+         x = "Age (in year)", y = "Credit card debt (in $)",
+         title = "Debt and age"
+     ) +
+     geom_smooth(method = "lm", se = FALSE)
`geom_smooth()` using formula = 'y ~ x'
>
```



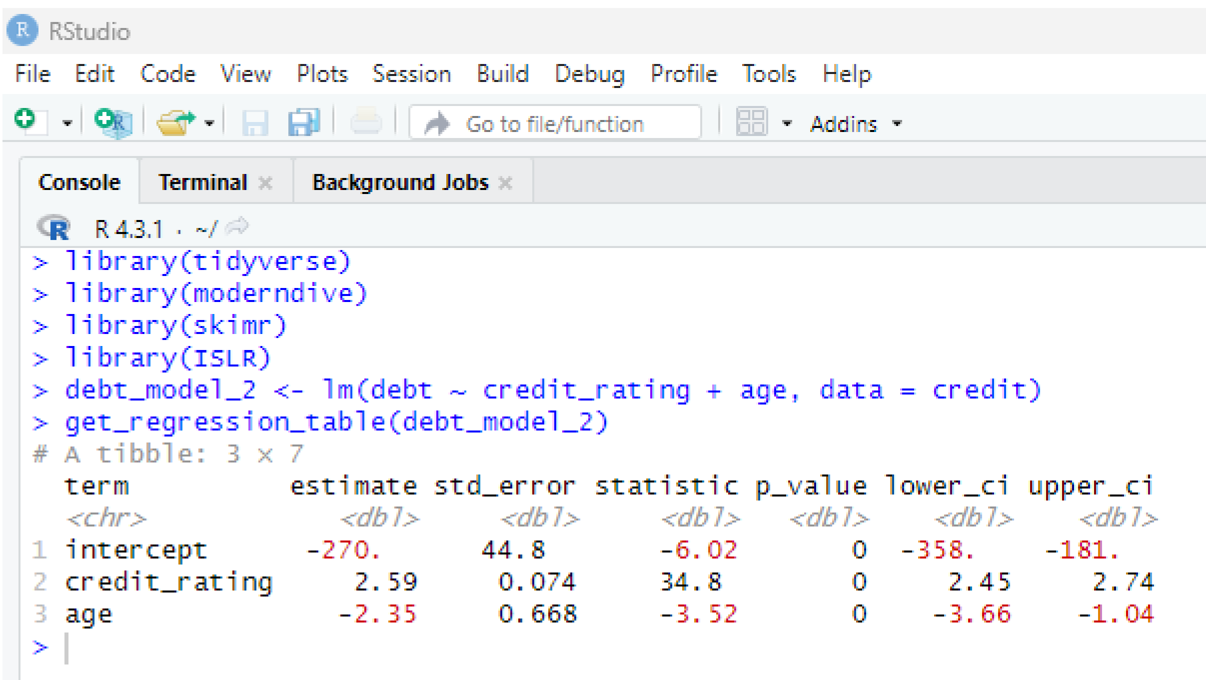
This is a favorable association between one's credit rating and debt, but a minimal relationship between one's age and debt.

(LC6.3) Fit a new simple linear regression using lm(debt ~ credit_rating + age, data = credit_ch6) where credit_rating and age are the new numerical explanatory variables x1 and x2 . Get information about the "best-fitting" regression plane from the regression table by applying the get_regression_table() function. How do the regression results match up with the results from your previous exploratory data analysis?

Answer :

debt_model_2 <- lm(debt ~ credit_rating + age, data = credit)

get_regression_table(debt_model_2)



```
RStudio
File  Edit  Code  View  Plots  Session  Build  Debug  Profile  Tools  Help
Go to file/function   Addins
Console  Terminal  Background Jobs
R 4.3.1 · ~/
> library(tidyverse)
> library(moderndive)
> library(skimr)
> library(ISLR)
> debt_model_2 <- lm(debt ~ credit_rating + age, data = credit)
> get_regression_table(debt_model_2)
# A tibble: 3 × 7
  term          estimate std_error statistic p_value lower_ci upper_ci
  <chr>            <dbl>     <dbl>     <dbl>   <dbl>    <dbl>    <dbl>
1 intercept      -270.      44.8       -6.02       0  -358.     -181.
2 credit_rating     2.59     0.074     34.8        0     2.45      2.74
3 age              -2.35     0.668     -3.52       0    -3.66     -1.04
>
```

Both new numerical explanatory variables x1 and x2, credit_rating and age, have coefficients of 2.59 and 2.35, respectively, indicating that debt and credit_rating are positively associated while debt and age are negatively connected. This corresponds to the findings of your prior exploratory data analysis.