ALY6010

R-PRACTICE MODULE 6

Name: Shreyansh Bhalodiya

NUID: 002664707

Instructor name: Mohsen Soltanifar

Dataset

- We have a dataset with 8 columns. Looking at the columns we can see that it's about the treatment a person has gone through with different features like smoker, time, wbc, hrt_months age and outcome.
- Cleaning data from null values.

A tibble: 6 × 8

id	time	treatment	smoker	hrt_months	wbc	age	outcome
<dbl>	<dbl>	<chr>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
165	0	Gabapentin	1	3	5.7	61.4	14.6
165	1	Gabapentin	1	3	5.7	61.4	8.7
165	2	Gabapentin	1	3	5.7	61.4	8.3
165	3	Gabapentin	1	3	5.7	61.4	6.9
165	4	Gabapentin	1	3	5.7	61.4	6.4
166	0	Placebo	0	4	7.8	55.2	20.6

```
df <- na.omit(df)
sum(is.na(df$age))
### https://sparkbyexamples.com/r-programming/remove-rows-with-na-in-r/
```

```
0
```

PART 1

Creating Dummy variables for the treatment column.

```
unique(df$treatment)
```

'Gabapentin' · 'Placebo'

```
## creating dummy variable for each of the unique values in column treatment.
df$treatment_gabapentin <- ifelse(df$treatment == "Gabapentin", 1, 0)
df$gender_Placebo <- ifelse(df$treatment == "Placebo", 1, 0)
```

id	time	treatment	smoker	hrt_months	wbc	age	outcome	treatment_gabapentin	gender_Placebo	treament_Placebo	treatment_Placebo
<dbl>	<dbl>	<chr>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
165	0	Gabapentin	1	3	5.7	61.4	14.6	1	0	0	0
165	1	Gabapentin	1	3	5.7	61.4	8.7	1	0	0	0
165	2	Gabapentin	1	3	5.7	61.4	8.3	1	0	0	0
165	3	Gabapentin	1	3	5.7	61.4	6.9	1	0	0	0
165	4	Gabapentin	1	3	5.7	61.4	6.4	1	0	0	0
166	0	Placebo	0	4	7.8	55.2	20.6	0	1	1	1

- We have only two unique values in the treatment column that is gabapentin and placebo hence we will create dummy variables for only these two values. We have treament_gabapentin and treatment_placebo.
- Let's rerun the two separate regression lines one without dummy variables and other with dummy variables.

```
library(jtools) # Load jtools
 # Telling R we want to use this data
fit <- lm(wbc ~ time + age + smoker + hrt_months+ outcome + treatment_gabapentin + treatment_Placebo, data = df)
summ(fit)
```

```
library(jtools) # Load jtools
 # Telling R we want to use this data
fit <- lm(wbc ~ time + age + smoker + hrt_months+ outcome, data = df)
summ(fit)
```

PART 2

Results

MODEL INFO:
Observations: 956
Dependent Variable: wbc
Type: OLS linear regression

MODEL FIT:
$F(6,949) = 11.32$, $p = 0.00$
$R^2 = 0.07$
Adj. $R^2 = 0.06$

Standard errors: OLS

	Est.	S.E.	t val.	p
(Intercept)	5.01	0.64	7.84	0.00
time	-0.02	0.04	-0.49	0.63
age	0.02	0.01	1.88	0.06
smoker	1.00	0.17	5.82	0.00
hrt_months	0.01	0.00	4.75	0.00
outcome	-0.01	0.00	-1.36	0.18
treatment_gabapentin	0.29	0.11	2.77	0.01
treatment_Placebo				

MODEL INFO:
Observations: 956
Dependent Variable: wbc
Type: OLS linear regression

MODEL FIT:
$F(5,950) = 11.96$, $p = 0.00$
$R^2 = 0.06$
Adj. $R^2 = 0.05$

Standard errors: OLS

	Est.	S.E.	t val.	p
(Intercept)	5.22	0.64	8.20	0.00
time	-0.02	0.04	-0.54	0.59
age	0.02	0.01	1.81	0.07
smoker	1.01	0.17	5.90	0.00
hrt_months	0.01	0.00	4.40	0.00
outcome	-0.01	0.00	-1.60	0.11

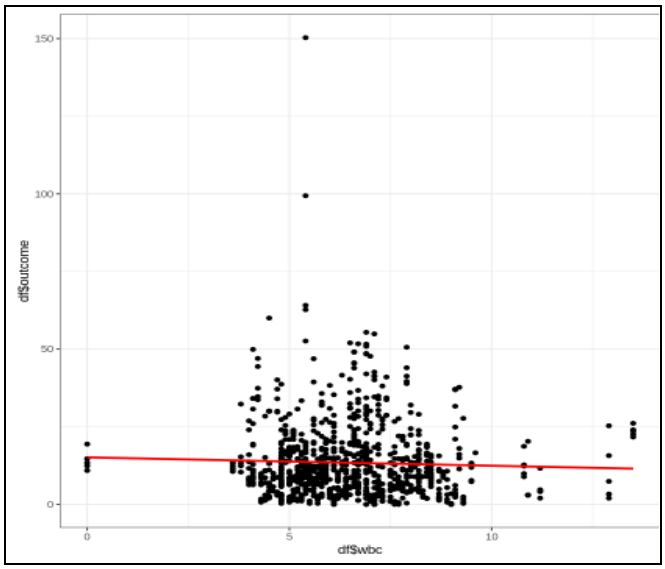
- On the left we have linear regression results with dummy variables and on the right we have results without dummy variables we can clearly see that the R-square value drops down by almost 10% points.
- Also we can see similar trends for adjusted R-square values.
- Hence we can clearly conclude that creating dummy variables out of categorical columns can boost our models evaluation scores.

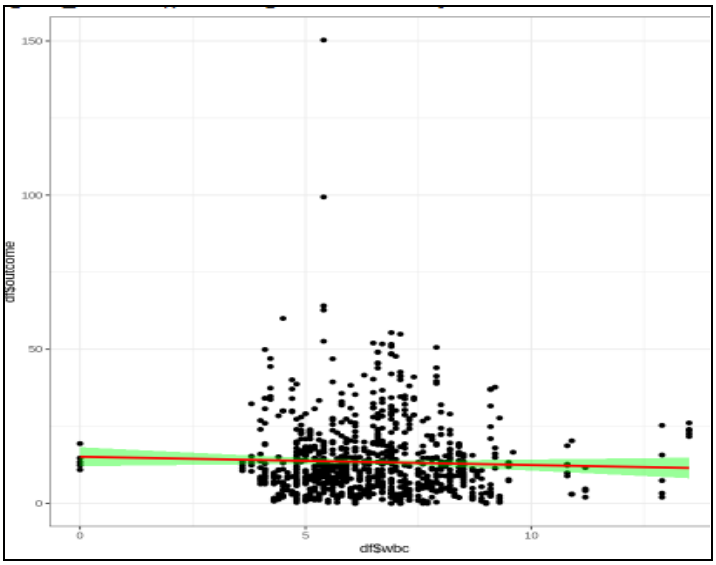
Separate Linear Regression Lines for each subset

```
plt <- ggplot(df, aes(x=df$wbc,y=df$outcome)) + geom_point(color="black")+theme_bw()
```

```
plt2 <- plt + geom_smooth(method = lm, color="red",se=FALSE)
```

```
plt3 <- plt + geom_smooth(method = lm , color = "red", fill="green",se= TRUE)
plt3
```





- **Using the ggplot library we have plotted linear regression linear for outcome and wbc columns.**
- **First we plotted only scatter plot then scatter plot with regression line further regression line with green fill.**

```
fit <- lm(df$wbc ~ df$time + df$age + df$smoker + df$hrt_months+ df$outcome)

summary(fit)


Call:
lm(formula = df$wbc ~ df$time + df$age + df$smoker + df$hrt_months +
    df$outcome)

Residuals:
    Min      1Q  Median      3Q     Max
-7.5175 -1.0575  0.0001  0.8483  7.1641

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)    5.221532   0.636801   8.200 7.77e-16 ***
df$time       -0.020252   0.037729  -0.537   0.5916
df$age         0.021698   0.011956   1.815   0.0699 .
df$smoker      1.014084   0.172018   5.895 5.19e-09 ***
df$hrt_months  0.005631   0.001280   4.398 1.21e-05 ***
df$outcome    -0.007043   0.004411  -1.597   0.1107
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.621 on 950 degrees of freedom
Multiple R-squared:  0.05921,    Adjusted R-squared:  0.05426
F-statistic: 11.96 on 5 and 950 DF,  p-value: 3.034e-11
```

- We have fitted multiple columns and checking the fit summary we got multiple R-squared values as 0.059 and Adjusted R-squared values as 0.05423.

References

GeeksforGeeks. (2020, August 5). ***Dummy Variables in R Programming.***
https://www.geeksforgeeks.org/dummy-variables-in-r-programming/