Statistical Models in R

Yebelay Berehan

Biostatistician

yebelay.ma@gmail.com

2022-06-19

Introduction

- Linear regression models (also known as "Ordinary Least Squares" model) allow us to determine if changing the values on a variable is associated with the values of another variable.
- In other words, if I make a 1-unit change in X, how much does Y change?
- We use linear regression models to test the association between two or more variables where the outcome is a continuous data type.
- linear regression model is called the "Ordinary Least Squares" or OLS model because it minimizes the squared errors (e.g., distance from the best-fit line).

1. Assumptions about the form of the model:
    - The linearity assumption. examining the scatter plot of Y versus X.
2. Assumptions about the errors: $e \sim Niid(0, \sigma^2)$.
    - normality assumption.
    - The errors have mean zero.
    - the constant variance assumption.
    - The independent-errors assumption. the autocorrelation problem.
3. Assumptions about the predictors:
    - nonrandom (assumed fixed).
    - measured without error.
    - linearly independent of each other; collinearity problem.
4. Assumptions about the observations:
    - equally reliable and have approximately equal role in determining the regression results and in influencing conclusions.

- diagnostic methods to check the violation of regression assumption are based on the study of model residuals with the help of various types of graphics.

Simple linear regression

- The structural form of a linear regression model:

$$Y_i = \beta_0 + \beta_1 X_{1i} + \epsilon$$

- Typical notations of the linear regression include:

- Y_i denotes the outcome (or dependent) variable for subject i
- X_{1i} denotes the predictor of interest (X_1) for subject i
- β_0 denotes the Y-intercept when X is zero
- β_1 denotes the slope or the change in Y with a 1-unit change in X
- ϵ denotes the error or residuals

Diabets data

```
library(readr)
diabetes1 <- read_csv("diabetes.data.csv")
```

- The following variables are included in the data

- Pregnancies: Number of times pregnant
- Glucose: Plasma glucose concentration a 2 hours in an oral glucose tolerance test
- BloodPressure: Diastolic blood pressure (mm Hg)
- SkinThickness: Triceps skin fold thickness (mm)
- Insulin: 2-Hour serum insulin (mu U/ml)
- BMI: Body mass index (weight in kg/(height in m)^2)
- DiabetesPedigreeFunction: Diabetes pedigree function
- Age: Age (years)
- Outcome: Class variable (0 = no diabetes or 1 = diabetes)

Correlation

- Correlation analysis is concerned with measuring whether two variables are associated with each other.
  - If two variables tend to change together in the same direction, they are said to be positively correlated.
  - If they tend to change together in opposite directions, they are said to be negatively correlated.

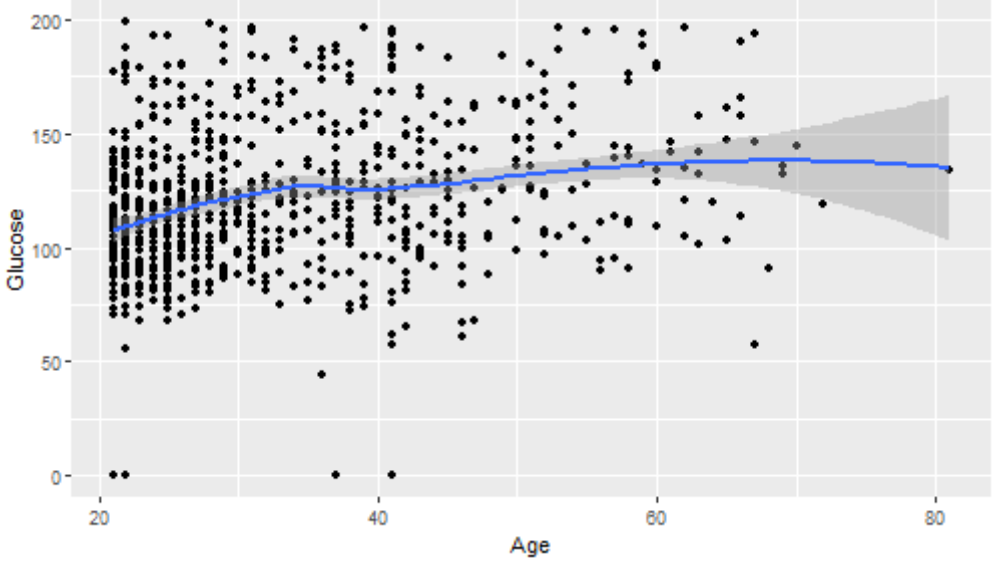
```
diabetes1 %>%
 select(- ...1,- Outcome,-Pregnancies,-DiabetesPedigreeFunction) %>%
 correlate() %>% shave(upper = TRUE) %>%
 fashion(decimals = 2, na_print = ".")
```

```
##             term Glucose BloodPressure SkinThickness Insulin  BMI Age
## 1        Glucose       .             .             .       .    .   .
## 2  BloodPressure     .15             .             .       .    .   .
## 3  SkinThickness     .06           .21             .       .    .   .
## 4        Insulin     .33           .09           .44       .    .   .
## 5            BMI     .22           .28           .39     .20    .   .
## 6            Age     .26           .24          -.11    -.04  .04   .
```

Visualize the association using scatter plot

- Let's look at how Age is related to Glucose level by plotting their relationship.
- As Age increases, the Glucose level also increases. There appears to be a positive relationship between Age and Glucose level.

```
ggplot(diabetes1, aes(x = Age, y = Glucose)) +
  geom_point() +  stat_smooth()
```



- Then, we update our linear regression model's structural form:

$$Glucose_i = \beta_0 + \beta_1 Age_i + \epsilon,$$

- where $Glucose_i$ denotes the expected Glucose level for subject i given the Age of subject i.

- We will use the `lm()` function with Glucose level as the Y variable and Age as the X variable.

- By using the `lm()` function, we can construct the linear regression model: `lm(Glucose ~ Age, data = diabetes.data)`.

Here is how we put all of this together in R:

```
linear.model1 <- lm(Glucose ~ Age, data = diabetes1)
summary(linear.model1)
```

```
##
## Call:
## lm(formula = Glucose ~ Age, data = diabetes1)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -126.453  -20.849   -3.058   18.304   86.159
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 97.08016    3.34095   29.06  < 2e-16 ***
## Age          0.71642    0.09476    7.56 1.15e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 30.86 on 766 degrees of freedom
## Multiple R-squared:  0.06944,    Adjusted R-squared:  0.06822
## F-statistic: 57.16 on 1 and 766 DF,  p-value: 1.15e-13
```

- β_1 coefficient is in the linear regression output as `Age`, which is `0.71642`.
- The `lm()` function does not generate 95% CI, so you will need to use the `confint()` function.

```
### Generate the 95% CI
confint(linear.model1)
```

```
##                   2.5 %      97.5 %
## (Intercept) 90.5216601 103.6386585
## Age          0.5304001   0.9024361
```

Interpret the linear regression output

- We are interested in the coefficients. To make interpreting the output easier, we can create a table to visualize the critical elements using `gtsummary` package.

```
model1 <- tbl_regression(linear.model1, intercept = TRUE)
as_gt(model1) %>%
  gt::tab_header("Table 2. Linear regression model") %>%
  gt::tab_options(table.align='center')
```

Table 2. Linear regression model

Characteristic	Beta	95% CI[1]	p-value
(Intercept)	97	91, 104	<0.001
Age	0.72	0.53, 0.90	<0.001

[1] CI = Confidence Interval

- The `Intercept` denotes the Y intercept when X is equal to zero. In this case, it would be where Glucose level would be on the linear plot when Age is equal to zero, which is 97.08-units of Glucose.

- The `Age` coefficient denotes the change in Glucose level for a one-unit increase in Age. In other words, a 1-year increase in Age is associated with a 0.72-unit increase in Glucose level.

- Since the 95% CI is between 0.53-units and 0.90-units of Glucose, it does not include zero, so this association is statistically significant. We can also look a the p-value of the `Age` coefficient to determine whether this is statistically significant (<0.0001).

    - However, it is preferable to present the 95% CI when describing the association between X and Y.

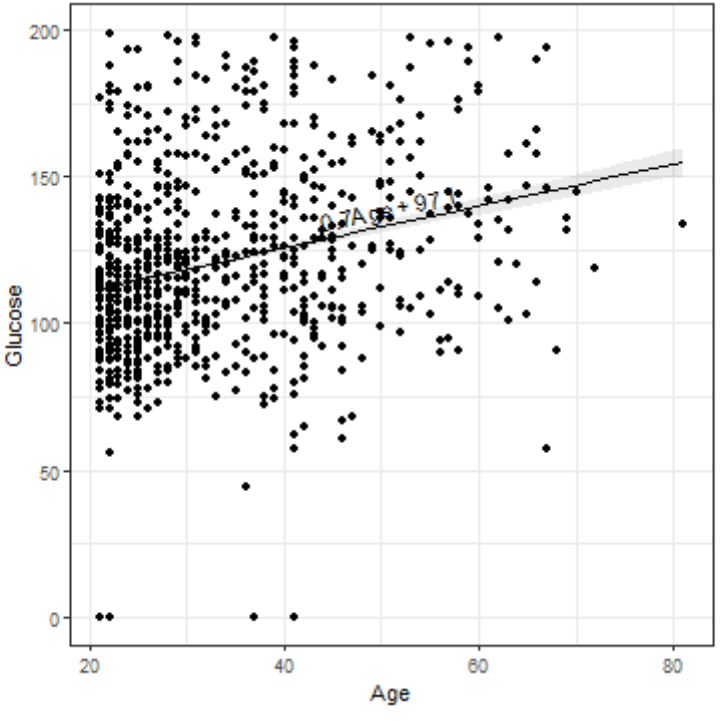
- The `Adjusted R squared` denotes the amount of data that are explained by the linear regression model.

- In other words, the current linear regression model explains 6.8% of the data.

Visualize predicted model

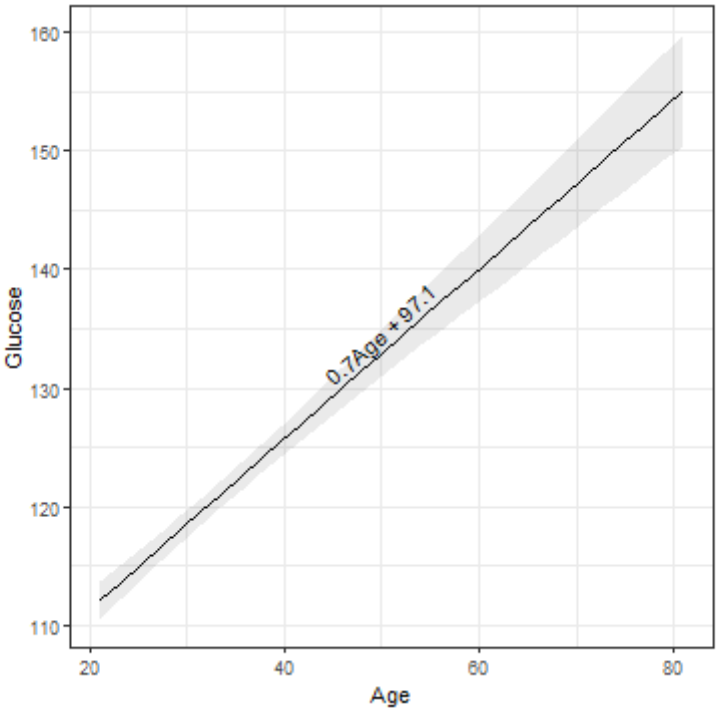
- We can plot the linear form of the model against the actual data using the `ggPredict()` function.

```
ggPredict(linear.model1, digits = 1, show.point = TRUE,
          se = TRUE, xpos = 0.5)
```



- Here is a version without the scatter plot.

```
ggPredict(linear.model1, digits = 1, show.point = FALSE,
          se = TRUE, xpos = 0.5)
```



Multiple regression: Adding a confounder

```
# Generate groups based on pregnancy history
diabetes1$pregnancy.history <- ifelse(diabetes1$Pregnancies == 0, 0,1)
table(diabetes1$pregnancy.history)
```

```
##
##   0   1
## 111 657
```

- We see that there are 657 women who a history of pregnancy and 111 women with no history of pregnancy.
- We need to include a confounder `pregnancy.history` in our linear regression model:

$$Glucose_i = \beta_0 + \beta_1 Age_i + \beta_2 PregnancyHistory_i + \epsilon,$$

where $Glucose_i$ denotes the expected Glucose level for subject i given the Age of subject i controlling for Pregnancy History of subject i.

```
linear.model2 <- lm(Glucose ~ Age + pregnancy.history, data = diabetes1)
summary(linear.model2)
```

```
##
## Call:
## lm(formula = Glucose ~ Age + pregnancy.history, data = diabetes1)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -125.715  -20.546   -2.991   17.316   87.734
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)      102.00752    3.95100  25.818  < 2e-16 ***
## Age                0.76050    0.09638   7.891 1.04e-14 ***
## pregnancy.history  -7.47264    3.22137  -2.320   0.0206 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 30.77 on 765 degrees of freedom
## Multiple R-squared:  0.07594,    Adjusted R-squared:  0.07352
## F-statistic: 31.43 on 2 and 765 DF,  p-value: 7.592e-14
```

- We can present the model output into a table.

```
#### Present the output in a table
model2 <- tbl_regression(linear.model2, intercept = TRUE)
as_gt(model2) %>%
  gt::tab_header("Table 3. Multiple regression model") %>%
  gt::tab_options(table.align='center')
```

Table 3. Multiple regression model

Characteristic	Beta	95% CI[1]	p-value
(Intercept)	102	94, 110	<0.001
Age	0.76	0.57, 0.95	<0.001
pregnancy.history	-7.5	-14, -1.1	0.021

[1] CI = Confidence Interval

- You can see that the `Age` coefficient is slightly different from our first model. It is 0.76 with a 95% CI of 0.57, 0.95. Compare this to the previous model's result, which was 0.72; 95% CI: 0.53, 0.90.

```
#### Merge the two linear regression model's outputs
model1 <- tbl_regression(linear.model1, intercept = TRUE)
model2 <- tbl_regression(linear.model2, intercept = TRUE)
table1 <- tbl_merge(tbls = list(model1, model2),
          tab_spanner = c("**Model 1**", "**Model 2**"))
as_gt(table1) %>%
  gt::tab_header("Table 4. Comparison between regression models")%>%
  gt::tab_options(table.align='center')
```

Table 4. Comparison between regression models

	Model 1			Model 2		
Characteristic	**Beta**	**95% CI[1]**	**p-value**	**Beta**	**95% CI[1]**	**p-value**
(Intercept)	97	91, 104	<0.001	102	94, 110	<0.001
Age	0.72	0.53, 0.90	<0.001	0.76	0.57, 0.95	<0.001
pregnancy.history				-7.5	-14, -1.1	0.021

[1] CI = Confidence Interval

- Model 1 is considered the crude model or the unadjusted model.
- Model 2 is the adjusted model because it is adjusting based on the Pregnancy History confounder.

- Notice that the $\beta_{1,unadjusted}$ is 0.72 which is lower than the $\beta_{1,adjusted}$ result which is 0.76.

- Additionally, the `Adjusted R squared` is higher in model 2 (7.35%) compared to model 1, which was 6.82%.

- This means that Model 2 does a better job of explaining the data than Model 1.
- Let's plot Model 2's results.

```
ggPredict(linear.model2, digits = 1, show.point = FALSE,
          se = TRUE, xpos = 0.5)
```

- We can see that the group that had a history of pregnancy is lower than the group that did not have a history of pregnancy.
- This makes sense when you look at the `pregnancy.history` coefficient.
- It is -7.5, which means that a subject with a history of pregnancy is associated with a 7.5 decrease in Glucose level (95% CI: -13.80, -1.15) compared to a subject without a history of pregnancy controlling for age.
- Therefore, for all ranges of Age, the group with a history of pregnancy will have Glucose levels that are 7.5-unit lower than a group without a history of pregnancy.
- You can visualize this on the plot; the linear lines do not cross and remain constant across all ranges of Age.
- But there is a positive correlation betwteen Age and Glucose level.

Evaluate residual plots

- It is good practice to look at the residuals of the regression model to make sure that the assumptions hold.
- We need to check whether the residual are correlated with fitted or predicted values of Glucose.
- If there is an association, then we have heteroscedasticity, which is a violation of the linear regression model assumption.

```
plot(linear.model1$res ~ linear.model1$fitted)
```

- Upon visual inspection,there doesn't appear to be any evidence of heteroskedasticity. we can see that the residuals are uniform across the "fitted" values.
- We can verify this visual inspection by performing the Breusch-Pagan test of heteroskedasticity.
- We will need to install and load the `lmtest` package and use the `bptest()` function.

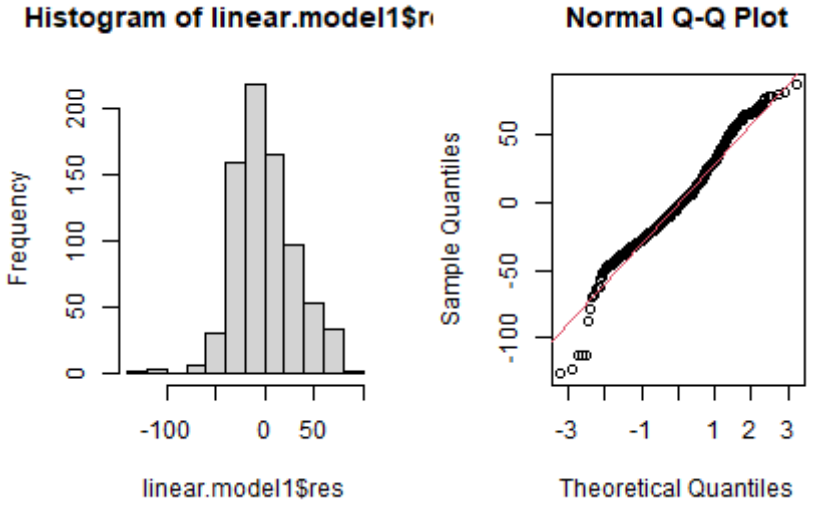
```
bptest(linear.model1)
```

```
##
##     studentized Breusch-Pagan test
##
## data:  linear.model1
## BP = 2.4585, df = 1, p-value = 0.1169
```

- The p-value is 0.1169, we fail to reject the null that the variance of the residuals are constant.

- We can also evaluate if the residuals are normally distributed.
- We can generate a histogram and a Q-Q plot.
- The historgram has a slight left skew and the Q-Q plot has its tails deviate from the neutral line.

```
par(mfrow = c(1, 2))
hist(linear.model1$res); qqnorm(linear.model1$res);
qqline(linear.model1$res, col = "2", lwd = 1, lty = 1)
```



- You only need to use one of these tests. They will generally give the same results.

- We can also test for the normality of the residuals.
- Common tests of normality include the Shapiro-Wilk's test, and the Kolmogorov-Smirnov test.

```
shapiro.test(linear.model1$res)
```

```
##
##     Shapiro-Wilk normality test
##
## data:  linear.model1$res
## W = 0.97467, p-value = 2.838e-10
```

```
lillie.test(linear.model1$res)
```

```
##
##     Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  linear.model1$res
## D = 0.065397, p-value = 2.941e-08
```

- Despite not being normally distributed, the linear regression model is pretty robust to violations of this assumption.

Logistic regression model

The structural form of the logistic regression model:

$$logit(p_i) = ln(\frac{p_i}{1-p_i}) = \beta_0 + \beta_1 X_{1i}$$

- Y_i denotes the outcome (or dependent) variable for subject i; this is a binary variable
- X_{1i} denotes the predictor of interest or the independent variable (X_1) for subject i
- β_0 denotes the Y-intercept when X is zero; this is not informative for logistic regression models
- β_1 denotes the slope or the change in Y with a 1-unit change in X
- p_i denotes the probability of the event occurring

- The logistic regression model is a predictive model for binary data.
- Hence, the logistic regression model can generate probabilities that a sample will have the discrete outcome given an input variable(s).
- The logistic regression model uses maximum likelihood estimation (MLE) which is a conditional probability that classifies the outcome if a certain threshold is met (e.g,. > 0.50).
- Hence, the probability range of a logistic regression model is between 0 and 1.
- Additionally, the logistic regression can include multiple predictors which can be controlled or adjusted in a multivariable logistic regression model.

```r
library(LaplacesDemon)
x <- -10:10
prob <- invlogit(x)
plot(x, prob, type = "l",
     main = "Logistic regression plot",
     ylab = "Probability", xlab = "Values of X")
```

Example - Logistic regression

- We'll use the `mtcar` data to build our logistic regression model.

```
#### Load the libraries
library("ggplot2"); library("gmodels")
library("epitools"); library("tidyverse")
### Create factors
data1 <- within(mtcars, {
           vs <- factor(vs, labels = c("V", "S"))
           am <- factor(am, labels = c("automatic", "manual"))
})
# head(data1)
```

Odds ratio calculation

```
oddsratio(data1$vs, data1$am, conf.level = 0.95, method = "wald")
```

```
## $data
##           Outcome
## Predictor automatic manual Total
##     V             12      6    18
##     S              7      7    14
##     Total         19     13    32
##
## $measure
##           odds ratio with 95% C.I.
## Predictor estimate     lower    upper
##         V        1        NA       NA
##         S        2 0.4764466 8.395484
##
## $p.value
##           two-sided
## Predictor midp.exact fisher.exact chi.square
##         V          NA           NA         NA
##         S   0.3718521    0.4726974  0.3409429
```

- Using the odds ratio, vehicles with a "V" engine had a 2 times higher odds of having an automatic transmission (95% CI: 0.47, 8.40) compared to vehicles with a straight engine.

Logistic regression in R

- We can create a crude logistic regression model to estimate the odds ratio.
- We set the transmission type `am` as the dependent variable and the engine type `vs` as the independent variable.
- The `glm()` command generates coefficients that are interpreted as the log odds of the event occuring.
- We need to exponentiate this to get the odds ratio using the `exp()` command.

```r
logit1<- glm(formula = am ~ vs, data = data1,
             family = "binomial"(link = "logit"))
summary(logit1)
```

```
##
## Call:
## glm(formula = am ~ vs, family = binomial(link = "logit"), data = data1)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.1774  -0.9005  -0.9005   1.1774   1.4823
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -0.6931     0.5000  -1.386    0.166
## vsS           0.6931     0.7319   0.947    0.344
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 43.230  on 31  degrees of freedom
## Residual deviance: 42.323  on 30  degrees of freedom
## AIC: 46.323
##
```

- According to the logistic regression model, vehicles with an "V" engine had a 2.0 times higher odds of having an automatic transmission (95% CI: 0.48, 8.76) compared to vehicles with a straight engine; this is not statistically significant since the odds ratio crosses the null or OR = 1.

```
confint(logit1)### Generate the 95% CI
```

```
##                  2.5 %    97.5 %
## (Intercept) -1.7483158 0.2526876
## vsS         -0.7334126 2.1696774
```

```
exp(coef(logit1))     ### Odds ratio
```

```
## (Intercept)         vsS
##         0.5         2.0
```

```
exp(confint(logit1))  ### 95% CI (odds ratio)
```

```
##                 2.5 %   97.5 %
## (Intercept) 0.1740669 1.287481
## vsS         0.4802672 8.755459
```

Multivariable logistic regression model

- The structural form of the multivariable logistic regression model (this example uses two `X` variables):

$$logit(p_i) = ln(\frac{p_i}{1-p_i}) = \beta_0 + \beta_1 X_{1i} + \beta_2 X_{2i}$$

- Since the logistic regression model can include both continuous and categorical predictors, we can add the engine type `vs` (V versus straight engine) and vehicle weight `wt`.

$$logit(p_i) = ln(\frac{p_i}{1-p_i}) = \beta_0 + \beta_1 (vs)_i + \beta_2 (wt)_i$$

where `vs` is the engine type and `wt` is the vehicle weight.

```r
logit2 <- glm(formula = am ~ vs + wt, data = data1,
              family = "binomial"(link = "logit"))
summary(logit2)
```

```
##
## Call:
## glm(formula = am ~ vs + wt, family = binomial(link = "logit"),
##     data = data1)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -1.72025  -0.25387  -0.04841   0.13220   1.90889
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)   22.143      9.134   2.424   0.0153 *
## vsS           -4.496      2.641  -1.703   0.0887 .
## wt            -6.664      2.640  -2.524   0.0116 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
```

- In this multivariable logistic regression model, the association between engine type `vs` and transmission type `am` is much lower (OR = 0.01; 95% CI: 0.000005, 0.048) controlling for vehicle weight `wt` .
- Controlling for the vehicle's weight reduced odds of the association between engine type `vs` and transmission type `am`.

```
### Exponentiate the coefficients
exp(coef(logit2))     ### Odds ratio
```

```
##  (Intercept)          vsS           wt
## 4.136804e+09 1.114894e-02 1.276657e-03
```

```
exp(confint(logit2))  ### 95% CI (odds ratio)
```

```
##                    2.5 %       97.5 %
## (Intercept) 1.358356e+04 1.439659e+21
## vsS         5.472656e-06 4.898832e-01
## wt          6.028539e-07 5.011505e-02
```

Comparison between models

- We compare the odds ratio between the crude and adjusted logistic regression models.

example -- diabetes dataset

- There was a total of 657 (85.5%) subjects with a history of pregnancies and 111 (14.5%) subjects with no history of pregnancy.

```
### Create factors
data2 <- within(diabetes1, {
            pregnancy.history <- factor(pregnancy.history,
    labels = c("No history of pregnancies", "History of pregnancies"))
})
```

Crude Logistic Regression Model

- We create a crude logistic regression model to evaluate the association of the subject's age `Age` on history of pregnancy `pregnancy.history`.

```
logit3 <- glm(pregnancy.history ~ Age, data = data2,
              family = "binomial"(link = "logit"))
summary(logit3)
```

```
##
## Call:
## glm(formula = pregnancy.history ~ Age, family = binomial(link = "logit"),
##     data = data2)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.9478   0.2902   0.4966   0.6630   0.7500
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.33970    0.39173  -0.867    0.386
## Age          0.06972    0.01343   5.193 2.06e-07 ***
```

Multivariable logistic regression model

```
logit3 <- glm(pregnancy.history ~ Age + BMI + Glucose + SkinThickness,
              data = data2, family = "binomial"(link = "logit"))
summary(logit3)
```

```
##
## Call:
## glm(formula = pregnancy.history ~ Age + BMI + Glucose + SkinThickness,
##     family = binomial(link = "logit"), data = data2)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.3549   0.2632   0.4737   0.6239   1.2282
##
## Coefficients:
##                Estimate Std. Error z value Pr(>|z|)
## (Intercept)    1.516255   0.626257   2.421 0.015472 *
## Age            0.083301   0.014808   5.625 1.85e-08 ***
## BMI           -0.053471   0.015487  -3.453 0.000555 ***
## Glucose       -0.005492   0.003615  -1.519 0.128700
## SkinThickness  0.007768   0.007413   1.048 0.294694
```

Models comparisons

- Baesd on the crude logistic regression model, a 1-unit increase in age was associated with a 7% increase in the odds of having a history of pregnancy (95% CI: 1.05, 1.10), which is statistically significant.
- The odds ratio describing the association between Age and History of pregnancy did not change much between the crude and adjusted logistic regression model.