Effect Size: Cohen d vs Correlation r

Note and Disclaimer:

(1) This PDF is part of YouTube tutorials (https://youtu.be/0ZvL_Bl-Fd4). This PDF is for individual, personal usage only.
(2) The author accepts no responsibility for the topicality, correctness, completeness or quality of the information provided.

Effect size - Cohen's *d*

Cohen d is used to quantify the size of the difference between two groups, taking into account the variability within each group.

Population

$$d = \frac{\mu_2 - \mu_1}{\sigma}$$

Sample

We can use a sample to estimate the d by replacing with sample means and sample standard deviation.

$$d = \frac{m_2 - m_1}{s_{pooled}} = \frac{m_2 - m_1}{\sqrt{\frac{s_1^2 + s_2^2}{2}}}$$

s_1 is the sample standard deviation for group 1, whereas s_2 is the sample standard deviation for group 2.

Effect Size - Pearson Product Moment *r*

The Pearson product-moment correlation coefficient, often referred to as the Pearson correlation coefficient (ρ) , quantifies the strength of a linear relationship between two variables.

Population

$$\rho = \frac{cov(X, Y)}{\sigma_X \sigma_Y}$$

Sample

We can use a sample correlation coefficient, namely r, to estimate population correlation coefficient ρ.

$$r_{xy} = \frac{cov(x, y)}{s_x s_y} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}$$

Comparing Cohen's *d* and Pearson's *r*

1. Basic Idea

The following effect size numbers are based on Jacob Cohen's *Statistical Power Analysis for the Behavioral Sciences* (Second Edition, p. 82)

Effect Size	d	r
Small	0.2	0.1
Medium	0.5	0.3
Large	0.8	0.5

2. Additional comments

Cohen's d can assume values ranging from 0 to infinity (i.e., can be greater than 1), whereas Pearson's r is confined to a range between -1 and 1.

Note that, based on my best understanding (not sure), Cohen's d can be negative as well, depending on how to define m_1 and m_2 and which order to put them.

3. Convert *r* to *d*, and vice versa

While Cohen's d looks different from Pearson's r, there is internal connection between these two.

The basic idea is that, the membership for group 1 and group 2 may be considered to be a dichotomy or a two point scale.

Thus, you can convert r to d, and vice versa.

Convert *r* to *d*

$$d = \frac{2r}{\sqrt{1-r^2}}$$

Convert *d* to *r*

$$d^2 = \frac{4r^2}{1-r^2}$$

$$d^2 = 4r^2 + r^2d^2 = r^2(4+d^2)$$

$$r^2 = \frac{d^2}{4+d^2}$$

$$r = \sqrt{\frac{d^2}{4+d^2}} = \frac{d}{\sqrt{d+d^2}}$$

For the formula converting d to r, you can refer to Jacob Cohen's *Statistical Power Analysis for the Behavioral Sciences* (Second Edition, p.23).

According to Cohen, sometimes, people might want to think of effect sizes for mean differences d in terms of r. Thus, you can convert d to r if needed.

For instance, group 1 and group 2 are two experimental manipulations. The effect size typically is d. But you can convert it into r.

I will provide data to demonstrate the connection between d and r.

Q1: What is the use case of effect size?

Effect size can be used to decide sample size.

As effect size decreases, the need for a larger sample size increases.

1. Effect size = 0.40

```
# Load the pwr package (if not already installed)
# install.packages("pwr")

library(pwr)
pwr.t.test(n=NULL,
           d = 0.4,
           sig.level = 0.05,
           power = 0.80,
           type = "two.sample",
           alternative="two.sided")
```

```
##
##      Two-sample t test power calculation
##
##               n = 99.08032
##               d = 0.4
##       sig.level = 0.05
##           power = 0.8
##     alternative = two.sided
##
## NOTE: n is number in *each* group
```

2. Effect size = 0.30

```
pwr.t.test(n=NULL,
           d = 0.3,
           sig.level = 0.05,
           power = 0.80,
           type = "two.sample",
           alternative="two.sided")
```

```
##
##      Two-sample t test power calculation
##
##               n = 175.3847
##               d = 0.3
##       sig.level = 0.05
##           power = 0.8
##     alternative = two.sided
##
## NOTE: n is number in *each* group
```

Changing effect size from 0.4 to 0.3, the required sample size increases from 99 to 135.

Q2: For effect size, any difference between correlation and linear regression?

1. Theoretical part

Correlation:

$$r_{xy} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}$$

Estimated slope in simple linear regression:

$$\hat{b} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2}$$

Standardized regression coefficient:

$$\hat{\beta} = \hat{b}\frac{s_x}{s_y} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2}\frac{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}}{\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}$$

Thus, correlation coefficient is equal to the standardized regression coefficient for the same X and Y in simple linear regressions (i.e., a regression model with one X and one Y).

2. Simulate a data for correlation / simple linear regression

```
# set the seed for reproducibility
set.seed(123)

# specify marginal means and standard deviations
mean_x <- 5
mean_y <- 6
sd_x <- 12
sd_y<- 14

## specify coefficient, which must be between -1 and 1
corcoef <- 0.5

## covariance between two variables
covariance <- corcoef * sd_x * sd_y

## variance-covariance matrix
sigma <- matrix(c(sd_x^2, covariance, covariance, sd_y^2), nrow = 2)

xy <- MASS::mvrnorm(n = 50, mu = c(mean_x, mean_y), Sigma = sigma)
colnames(xy) <- c("x", "y")
df <- data.frame(xy)
head(df)
```

```
##           x          y
## 1 -2.188776  0.1167428
```

```
## 2  3.014416  2.8711033
## 3 20.168055 25.9184510
## 4 -4.307292 14.2689957
## 5  7.878421  6.4575243
## 6 10.288163 36.3225845
```

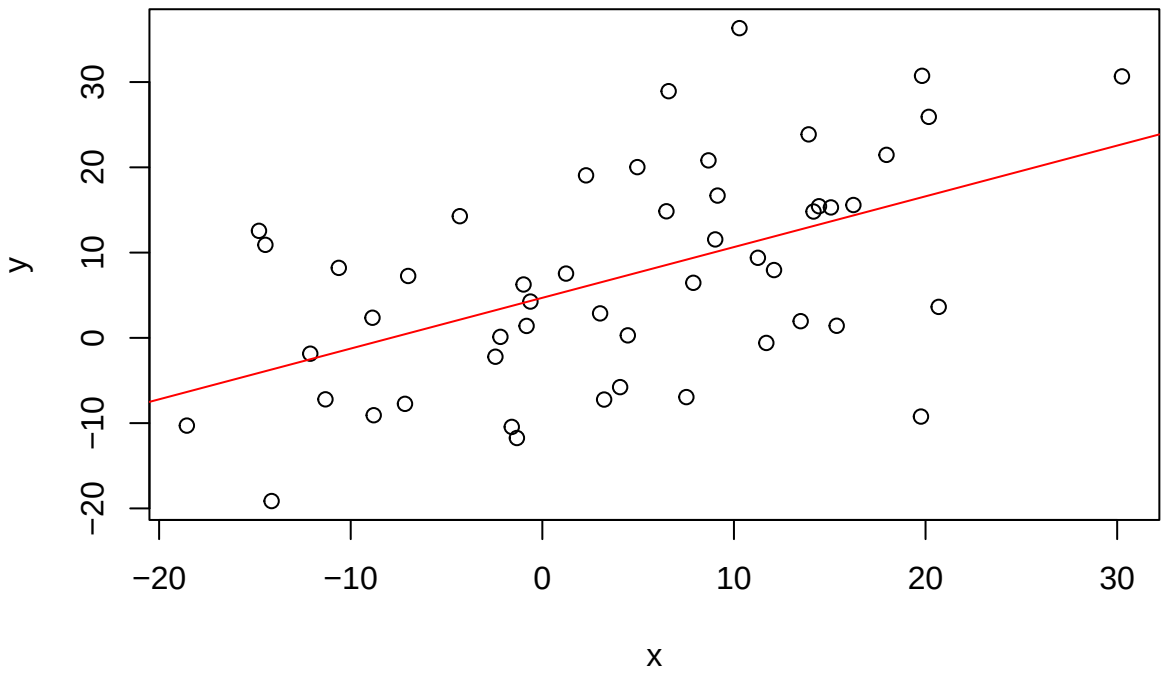
```
# check the means
sapply(df, mean)
```

```
##        x        y
## 4.260339 7.231809
```

```
# check the standard deviation
sapply(df, sd)
```

```
##        x        y
## 11.20860 12.75766
```

```
# plot it
plot(df)
abline(lm(y ~ x,data=df), col = "red")
```



Correlation

```
# check the correlation coefficient
cor.test(df$x,df$y)
```

```
##
##  Pearson's product-moment correlation
##
## data:  df$x and df$y
## t = 4.2497, df = 48, p-value = 9.8e-05
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.2861663 0.6994209
## sample estimates:
##       cor
## 0.5228658
```

Simple linear regression

```
# simple linear regression
#raw x and y
lm(y ~ x,data=df)
```

```
##
## Call:
## lm(formula = y ~ x, data = df)
##
## Coefficients:
## (Intercept)            x
##      4.6964       0.5951
```

```
# simple linear regression
#standardized both x and y
lm(scale(y) ~ scale(x),data=df)
```

```
##
## Call:
## lm(formula = scale(y) ~ scale(x), data = df)
##
## Coefficients:
## (Intercept)     scale(x)
##  -2.463e-17    5.229e-01
```

Observation

Thus, correlation coefficient is the same as the standardized regression coefficient for the same X and Y.

Q3: Why is r different from d?

Effect Size	d	r
Small	0.2	0.1
Medium	0.5	0.3
Large	0.8	0.5

```
# dichotomize x
df$dichotomized_x<- ifelse(df$x >mean(df$x), 1, 0)
head(df)
```

```
##           x          y dichotomized_x
## 1 -2.188776  0.1167428              0
## 2  3.014416  2.8711033              0
## 3 20.168055 25.9184510              1
## 4 -4.307292 14.2689957              0
## 5  7.878421  6.4575243              1
## 6 10.288163 36.3225845              1
```

Since we got a column of dichotomized x, we can actually use the formula mentioned to calculate Cohen's d. That is, we can calculate the means for groups 1 and 2. Then, we can calculate the sample standard deviations.

$$d = \frac{m_2 - m_1}{s_{pooled}} = \frac{m_2 - m_1}{\sqrt{\frac{s_1^2 + s_2^2}{2}}}$$

```
# calculate group means
means<-aggregate(df$y, list(df$dichotomized_x), FUN=mean)

# calculate sample standard deviation - pooled
variances<- aggregate(df$y, list(df$dichotomized_x),FUN=var)
s_pooled=sqrt((variances$x[1]+variances$x[2])/2)

d_calculated<-(means$x[2]-means$x[1])/s_pooled
print(d_calculated)
```

```
## [1] 1.252203
```

$$d = \frac{2r}{\sqrt{1 - r^2}}$$

```
# check the correlation coefficient
cor_result<-cor.test(df$x,df$y)
correlation_coefficient<-cor_result$estimate
d_converted<-2*correlation_coefficient/sqrt(1-correlation_coefficient^2)
print(d_converted)
```

```
##      cor
## 1.226787
```

References

1. What is the relation between the effect size and correlation?

https://stats.stackexchange.com/questions/412590/what-is-the-relation-between-the-effect-size-and-correlation

2. What is Effect Size and Why Does It Matter? (Examples)

https://www.scribbr.com/statistics/effect-size/

3. How to simulate a strong correlation of data with R

https://stackoverflow.com/questions/72894192/how-to-simulate-a-strong-correlation-of-data-with-r

4. FAQ How is effect size used in power analysis?

https://stats.oarc.ucla.edu/other/mult-pkg/faq/general/effect-size-power/faqhow-is-effect-size-used-in-power-analysis/