✪ Psychological Statistics ✪

Week 08: *Correlation Analysis*

- Edited by Prof. **Changwei Wu**
- Graduate Institute of Mind, Brain and Consciousness (***GIMBC***), Taipei Medical University

In [ ]:

```
### [ Setup the working directory ]

setwd("/Users/wesley/[Course]/Python/R_Script")
getwd()
```

In [3]:

```
### [ Loading the required libraries ]

library("dplyr")
library("rstatix")
library("ggplot2")
library("psych")
library("ggm")
library("ggcorrplot")
```

In [53]:

```
### Visualization ["Scatter Plot"]

data(msleep)
ggplot(msleep, aes(sleep_total, bodywt)) +
    geom_point(size = 3) +
    labs(x = "Total sleep time (hours)", y = "Body weight (g)")
```

...

(1) Pearson's Correlation (*cor_test* in {rstatix})

In [18]:

```
### 1-1.[ Load data of "mtcars"]

mydata <- mtcars %>% select(mpg, disp, hp, drat, wt, qsec)
head(mydata, 4)
```

A data.frame: 4 × 6

	mpg	disp	hp	drat	wt	qsec
	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
Mazda RX4	21.0	160	110	3.90	2.620	16.46
Mazda RX4 Wag	21.0	160	110	3.90	2.875	17.02
Datsun 710	22.8	108	93	3.85	2.320	18.61
Hornet 4 Drive	21.4	258	110	3.08	3.215	19.44

In [19]:

```
### 1-2.(1) Correlation test between two variables

mydata %>% cor_test(wt, mpg, method = "pearson")
```

A cor_test: 1 × 8

var1	var2	cor	statistic	p	conf.low	conf.high	method
<chr>	<chr>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<chr>
wt	mpg	-0.87	-9.559044	1.29e-10	-0.9338264	-0.7440872	Pearson

In [20]:

```
### 1-2.(2) Correlation of one variable against all

mydata %>% cor_test(mpg, method = "pearson")
```

A cor_test: 5 × 8

var1	var2	cor	statistic	p	conf.low	conf.high	method
<chr>	<chr>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<chr>
mpg	disp	-0.85	-8.747152	9.38e-10	-0.92335937	-0.7081376	Pearson
mpg	hp	-0.78	-6.742389	1.79e-07	-0.88526861	-0.5860994	Pearson
mpg	drat	0.68	5.096042	1.78e-05	0.43604838	0.8322010	Pearson
mpg	wt	-0.87	-9.559044	1.29e-10	-0.93382641	-0.7440872	Pearson
mpg	qsec	0.42	2.525213	1.71e-02	0.08195487	0.6696186	Pearson

In [21]:

```
### 1-2.(3) Pairwise correlation test between all variables

mydata %>% cor_test(method = "pearson")
```

...

(2) Pearson's Correlation Coefficient (*r*)

[Hypothesis] Anxiety level is linearly related with the exam score. *(2-tailed)*

- Null hypothesis H_0 **:** No (linear) relationship between anxiety and exam score → ***r***(Anxiety, Score) = 0
- Alternative hyp. H_1 **:** There is (linear) relationship between anxiety and exam score → ***r***(Anxiety, Score) ≠ 0

In [11]:

```
### [ Step.1 ] Data Loading

# → Loading the dataset
examData = read.delim("ExamAnxiety.dat",  header = TRUE)
examData %>% head(4)
```

A data.frame: 4 × 5

	Code	Revise	Exam	Anxiety	Gender
	<int>	<int>	<int>	<dbl>	<fct>
1	1	4	40	86.298	Male
2	2	11	65	88.716	Female
3	3	27	80	70.178	Male
4	4	53	80	61.312	Male

In [126]:

```
### [ Step.2 ] Check assumptions

#----- (a) Outliers -----#
examData %>% identify_outliers(Anxiety)
#examData %>% identify_outliers(Exam)

#----- (b) Normality -----#
examData %>% shapiro_test(Anxiety)
#examData %>% shapiro_test(Exam)
#examData %>% shapiro_test(Revise)

# → subgroup by Gender
examData %>% group_by(Gender) %>% shapiro_test(Anxiety)
```

A data.frame: 7 × 7

Code	Revise	Exam	Anxiety	Gender	is.outlier	is.extreme
<int>	<int>	<int>	<dbl>	<fct>	<lgl>	<lgl>
15	98	95	34.714	Male	TRUE	FALSE
24	84	90	0.056	Female	TRUE	TRUE
28	72	75	27.460	Female	TRUE	FALSE
33	43	60	43.580	Male	TRUE	FALSE
37	72	85	37.132	Male	TRUE	FALSE
78	2	100	10.000	Male	TRUE	TRUE
83	68	100	20.206	Female	TRUE	TRUE

A tibble: 1 × 3

variable	statistic	p
<chr>	<dbl>	<dbl>
Anxiety	0.8224243	8.650105e-10

A tibble: 2 × 4

Gender	variable	statistic	p
<fct>	<chr>	<dbl>	<dbl>
Female	Anxiety	0.7808172	2.660005e-07
Male	Anxiety	0.8647347	2.895485e-05

In [127]:

```
### [ Step.3 ] Correlation analysis: cor & cor_test

#--- (1) cor {stats} ---#
examData2 <- examData %>% select(Exam, Anxiety, Revise)
cor(examData2, use = "complete.obs", method = 'pearson') %>% round(3)

#--- (2) cor_test {rstatix} ---#
examData2 %>% cor_test(Anxiety, Exam, use = "pairwise.complete.obs", method = 'spear
(cor.mat <- examData2 %>% cor_mat(method = 'spearman'))
#cor.mat %>% cor_mark_significant()
#cor.mat %>% cor_get_pval() %>% pull_lower_triangle()
```

A matrix: 3 × 3 of type dbl

	Exam	**Anxiety**	**Revise**
Exam	1.000	-0.441	0.397
Anxiety	-0.441	1.000	-0.709
Revise	0.397	-0.709	1.000

A cor_test: 1 × 6

var1	**var2**	**cor**	**statistic**	**p**	**method**
<chr>	**<chr>**	**<dbl>**	**<dbl>**	**<dbl>**	**<chr>**
Anxiety	Exam	-0.4	255785.8	2.25e-05	Spearman

A cor_mat: 3 × 4

	rowname	**Exam**	**Anxiety**	**Revise**
	<chr>	**<dbl>**	**<dbl>**	**<dbl>**
1	Exam	1.00	-0.40	0.35
2	Anxiety	-0.40	1.00	-0.62
3	Revise	0.35	-0.62	1.00

In [82]:

```
### [ Step.3 ] Correlation analysis: cor.test

#--- (3) cor.test {stats} ---#

cor.test(examData$Anxiety, examData$Exam, method = 'spearman')
#cor.test(examData$Revise, examData$Exam, method = 'spearman')
#cor.test(examData$Anxiety, examData$Revise, method = 'spearman')
```

```
Warning message in cor.test.default(examData$Anxiety, examData$Exam, m
ethod = "spearman"):
“Cannot compute exact p-value with ties”


        Spearman's rank correlation rho

data:  examData$Anxiety and examData$Exam
S = 255786, p-value = 2.245e-05
alternative hypothesis: true rho is not equal to 0
sample estimates:
       rho
-0.4046141
```

In [72]:

```
### [ Step.4 ] Effect size: R square (coefficient of determination)

(R2 <- cor(examData2, method = 'spearman')^2 * 100)
```

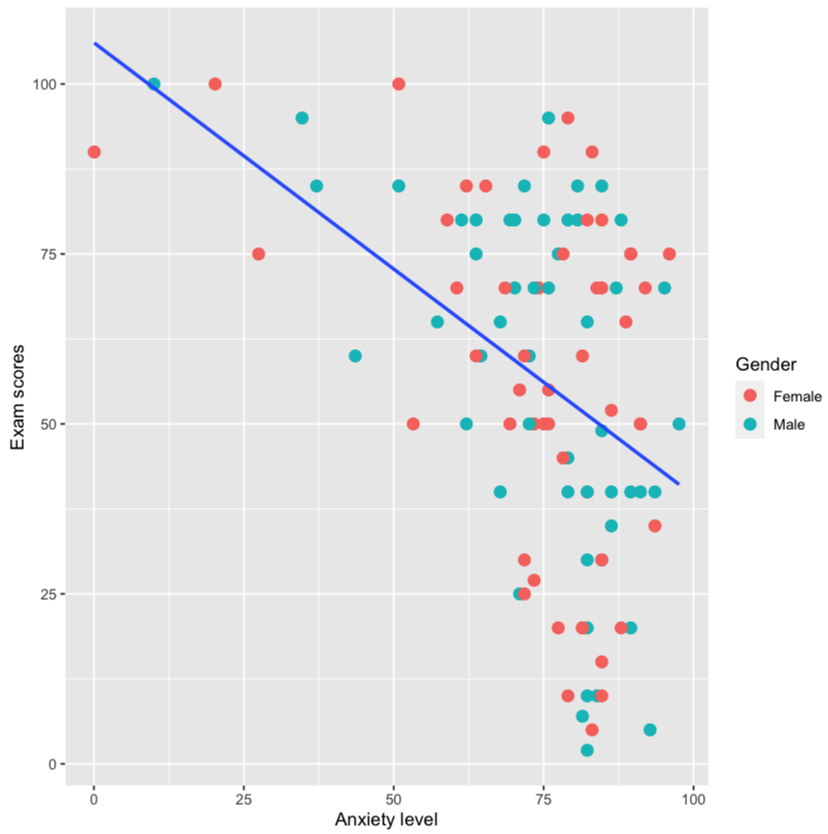
A matrix: 3 × 3 of type dbl

	Exam	Anxiety	Revise
Exam	100.00000	16.37126	12.24264
Anxiety	16.37126	100.00000	38.68459
Revise	12.24264	38.68459	100.00000

In [97]:

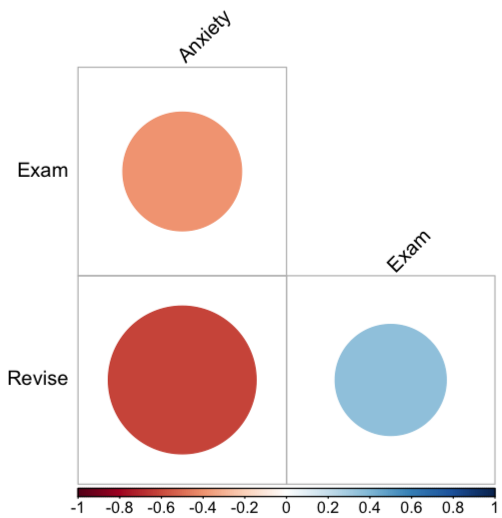
```
### [ Step.5 ] Visualization: Scatter plot

ggplot(examData, aes(Anxiety, Exam)) +
    geom_point(size = 3, aes(colour = Gender, size = Revise)) +
    labs(x = "Anxiety level", y = "Exam scores") +
    geom_smooth(method = "lm", , se=FALSE, level=0.95)
```

```
`geom_smooth()` using formula 'y ~ x'
```



In [117]:

```
### [ Step.5 ] Visualization: correlogram

cor.mat %>%
  cor_reorder() %>%
  pull_lower_triangle() %>%
  cor_plot()
```



~ *Report* ~

- The anxiety level is linearly related with the exam score (*Pearson*'s r_{101} = -0.44, p < .001, R^2 = 0.194).
- **A relationship exhibited between the anxiety level and the exam score (*Spearman*'s ρ_{101} = -0.40, p < .001, R^2 = 0.164).**

---

(3) Partial Correlation

In [86]:

```
### 3-0.[ Partial Correlation analysis ] Controlling Revise Time in previous example

#(1) calculate covariance first
cov(examData2)

#(2) calculate partial correlation {ggm}
(r_pc<-ggm::pcor(c(1,2,3), cov(examData2)) %>% round(3))
r_pc^2

#(3) get statistics
ggm::pcor.test(r_pc, 1, 103)
```

A matrix: 3 × 3 of type dbl

	Exam	Anxiety	Revise
Exam	672.9138	-196.5540	186.8784
Anxiety	-196.5540	295.2163	-221.2909
Revise	186.8784	-221.2909	329.7531

-0.247

0.061009

$tval
-2.54897888796454
$df
100
$pvalue
0.0123236010241439

~ Report of Partial correlation in ExamAnxiety.dat ~

- **After controlling the revise time, the anxiety level is related with the exam score still (r_{100} = -0.25, p < .02, R^2 = 0.061).**
- (Original) The anxiety level is linearly related with the exam score (*Pearson*'s r_{101} = -0.44, p < .001, R^2 = 0.194).

Example: COPC management and English score

[Hypothesis] Controlling IQ, the English score is related with the usage of teaching materials. *(2-tailed)*

- Null hypothesis H_0 **:** Controlling IQ, no relationship between English score and teaching materials.
- Alternative hyp. H_1 **:** Controlling IQ, there is linear relationship between English score and teaching materials.

In [104]:

```
### [ Step.1 ]  Data Loading

CopcData <- read.csv("Copc.csv",header=T)
head(CopcData,4)

CopcData %>% get_summary_stats(type = "mean_sd")
```

A data.frame: 4 × 3

	English	usage	IQ
	<int>	<int>	<int>
1	73	3600	70
2	82	4400	84
3	80	3500	77
4	69	2500	56

A tibble: 3 × 4

variable	n	mean	sd
<chr>	<dbl>	<dbl>	<dbl>
English	10	80.3	10.863
IQ	10	72.8	20.698
usage	10	3870.0	1382.470

In [128]:

```
### [ Step.2 ]  Assumption check

#----- (a) Linearity -----#
# scatter plot

#----- (b) Outliers -----#
CopcData %>% identify_outliers(English)
CopcData %>% identify_outliers(usage)

#----- (c) Normality -----#
CopcData %>% shapiro_test(English)
CopcData %>% shapiro_test(usage)
```

A data.frame: 0 × 5

English	usage	IQ	is.outlier	is.extreme
<int>	<int>	<int>	<lgl>	<lgl>

A data.frame: 0 × 5

English	usage	IQ	is.outlier	is.extreme
<int>	<int>	<int>	<lgl>	<lgl>

A tibble: 1 × 3

variable	statistic	p
<chr>	<dbl>	<dbl>
English	0.9578225	0.7607864

A tibble: 1 × 3

variable	statistic	p
<chr>	<dbl>	<dbl>
usage	0.9528441	0.7022021

In [110]:

```
### [ Step.3 ] Partial Correlation analysis

#(1) calculate covariance first
cov(CopcData)

#(2) correlation between English and usage given IQ {ggm}
(r_pc<-ggm::pcor(c("English", "usage", "IQ"), cov(CopcData)) %>% round(3))

#(3) get statistics
ggm::pcor.test(r_pc, 1, 10)

# original
cor.test(CopcData$English, CopcData$usage)
```

A matrix: 3 × 3 of type dbl

	English	usage	IQ
English	118.0111	13698.89	182.1778
usage	13698.8889	1911222.22	25471.1111
IQ	182.1778	25471.11	428.4000

0.715

$tval
2.70583038657573
$df
7
$pvalue
0.0303816740581674

```
        Pearson's product-moment correlation

data:  CopcData$English and CopcData$usage
t = 6.2949, df = 8, p-value = 0.0002341
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.6637041 0.9793334
sample estimates:
      cor
0.9121543
```

In [112]:

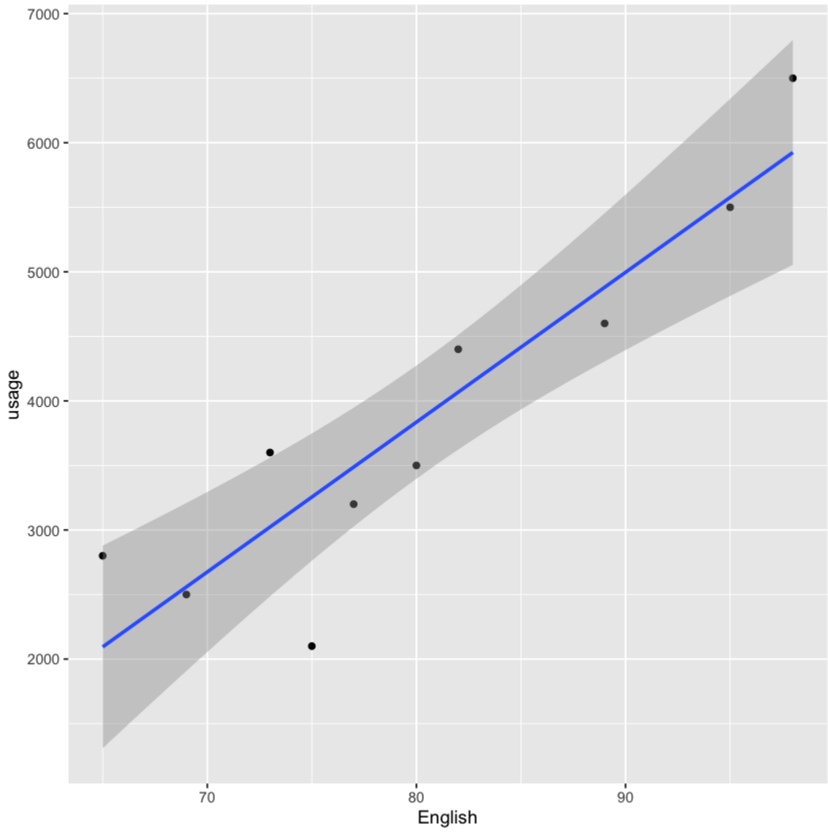
```
### [ Step.4 ] Effect size

r_pc^2 %>% round(3)
```

0.511

In [122]:

```
### [ Step.5 ] Visualization: cor_plot {rstatix}

ggplot(data=CopcData, aes(x=English, y=usage))+geom_point()+geom_smooth(method=lm, s
```

```
`geom_smooth()` using formula 'y ~ x'
```



~ Report ~

- **After contolling IQ, the English score is still positively correlated with the COPC usage (r_7 = 0.715, *p* = .03, *R^2* = 51.1%).**

(4) Comparison between Correlations & GGcorrplot

In [124]:

```
#-------Differences between independent rs-----

zdifference<-function(r1, r2, n1, n2)
{zd<-(atanh(r1)-atanh(r2))/sqrt(1/(n1-3)+1/(n2-3))
    p <-1 - pnorm(abs(zd))
    print(paste("Z Difference: ", zd))
    print(paste("One-Tailed P-Value: ", p))
    }

zdifference(-0.506, -0.381, 52, 51)

psych::r.test(52, -0.506, -0.381, n2=51)
```

```
[1] "Z Difference:  -0.768709306290097"
[1] "One-Tailed P-Value:  0.221032949510287"

Correlation tests
Call:psych::r.test(n = 52, r12 = -0.506, r34 = -0.381, n2 = 51)
Test of difference between two independent correlations
 z value 0.77    with probability  0.44
```

In [125]:

```
#-------Differences between dependent rs-----

tdifference<-function(rxy, rxz, rzy, n)
{df<-n-3
    td<-(rxy-rzy)*sqrt((df*(1 + rxz))/(2*(1-rxy^2-rxz^2-rzy^2+(2*rxy*rxz*rzy))))
    p <-pt(td, df)
    print(paste("t Difference: ", td))
    print(paste("One-Tailed P-Value: ", p))
    }

tdifference(-0.441, -0.709, 0.397, 103)

psych::r.test(103, -.441, .397, -.709)
```

```
[1] "t Difference:  -5.09576822523987"
[1] "One-Tailed P-Value:  8.21913727738007e-07"

Correlation tests
Call:[1] "r.test(n =  103 ,  r12 =  -0.441 ,  r23 =  -0.709 ,  r13 =
0.397 )"
Test of difference between two correlated  correlations
 t value -5.09    with probability < 1.7e-06
```

Example: Stroke Rehabilitation {GGCORRPLOT}

Wu et al., Frontiers in Neuroscience. 2020; 14: 548

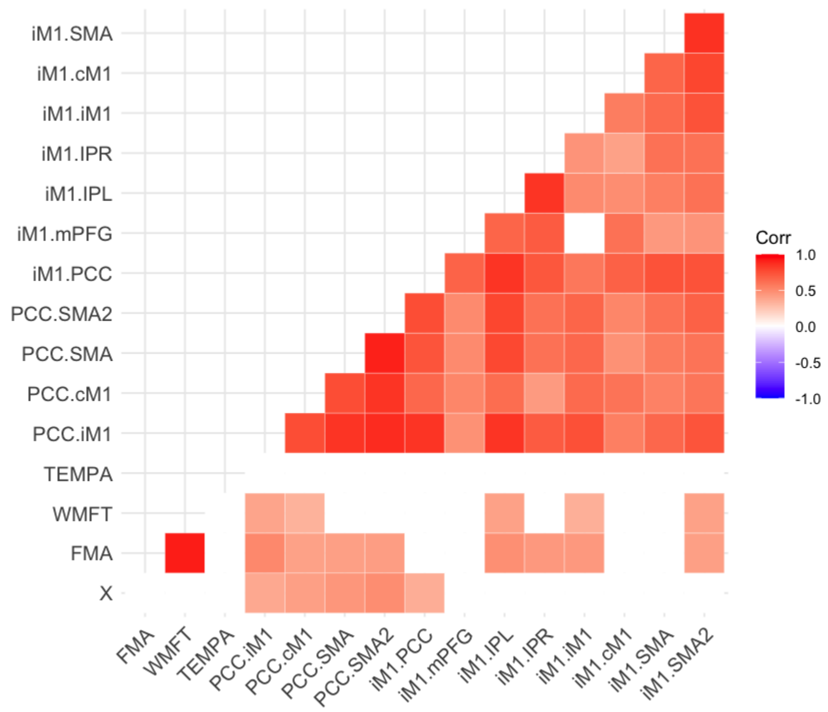
In [ ]:

```
RehabIDX<-read.csv("Stroke.csv", header=T, sep=",")
attach(RehabIDX)
```

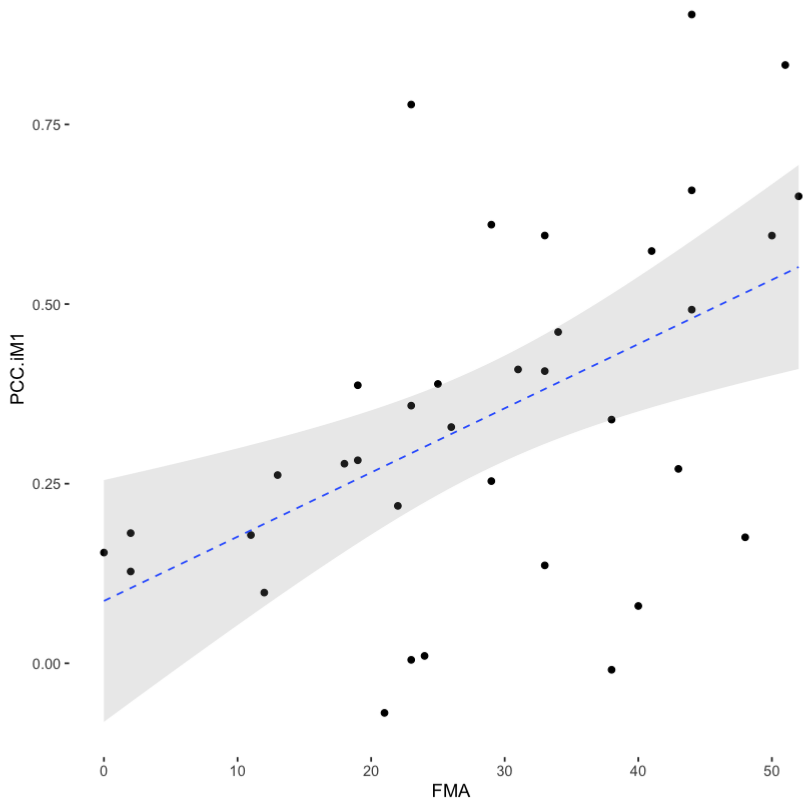
In [123]:

```
able<-cor(RehabIDX)
qr<-round(cor(RehabIDX)^2,2)
al <- round(cor_pmat(RehabIDX),3)
gcorrplot(Rtable,hc.order = FALSE, outline.col = "white")
corrplot(Rtable,hc.order = FALSE, outline.col = "white",type ="lower",p.mat=Pval, in

g<-ggplot(RehabIDX, aes(x=FMA, y=PCC.iM1))
g+geom_point()+stat_smooth(method="lm", se=T, linetype="dashed", size=0.5, alpha=0.2
r.test(RehabIDX$FMA, RehabIDX$PCC.iM1)
```

```
`geom_smooth()` using formula 'y ~ x'
```




```
        Pearson's product-moment correlation

data:  RehabIDX$FMA and RehabIDX$PCC.iM1
t = 3.4574, df = 34, p-value = 0.001485
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.2180204 0.7182151
sample estimates:
      cor
0.5100244
```



0.75
0.50
0.25
0.00
PCC.iM1
0
10
20
30
40
50
FMA