

ECO208


R and RStudio

Week 11 | December 9, 2022



Happy Friday class! 🎉


Please take this class survey.
We will use it as data for our class today.

Week 11

 Yoh Kawano さんが新しい資料を投稿しました: Week 11 Lecture

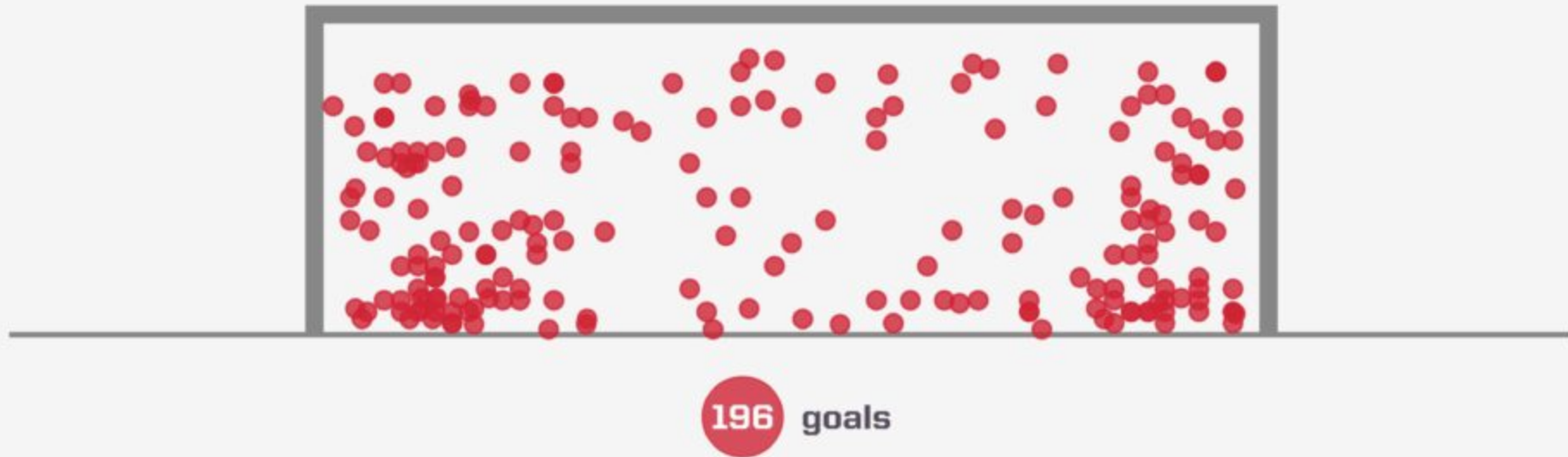
投稿日: 昨日 (最終編集: 11:28)

 ECO208 Final Exam Week 11: December 9, 2022	W11.pdf PDF	 Week 11 Econ Class Survey Google フォーム
 chiba11.csv カンマ区切り		

 クラスのコメントを追加...

World Cup Shootout Penalties Scored

World Cups since 1982



World Cup Shootout Penalties Missed/Saved

World Cups since 1982



World Cup Shootout Penalty Conversion

World Cups since 1982



10/10 100%	8/8 100%	8/8 100%	4/4 100%	5/5 100%	8/8 100%
16/17 94.1%	6/13 46.2%	4/5 80%	2/2 100%	4/6 66.7%	17/23 73.9%
32/37 86.5%	17/30 56.7%	7/14 50%	7/10 70%	12/18 66.7%	29/38 76.3%

World Cup Shootout Penalty Placement

World Cups since 1982



10 3.9%	8 3.1%	8 3.1%	4 1.6%	5 2%	8 3.1%
17 6.6%	13 5.1%	5 2%	2 0.8%	6 2.3%	23 9%
37 14.5%	30 11.7%	14 5.5%	10 3.9%	18 7%	38 14.8%

Last week...

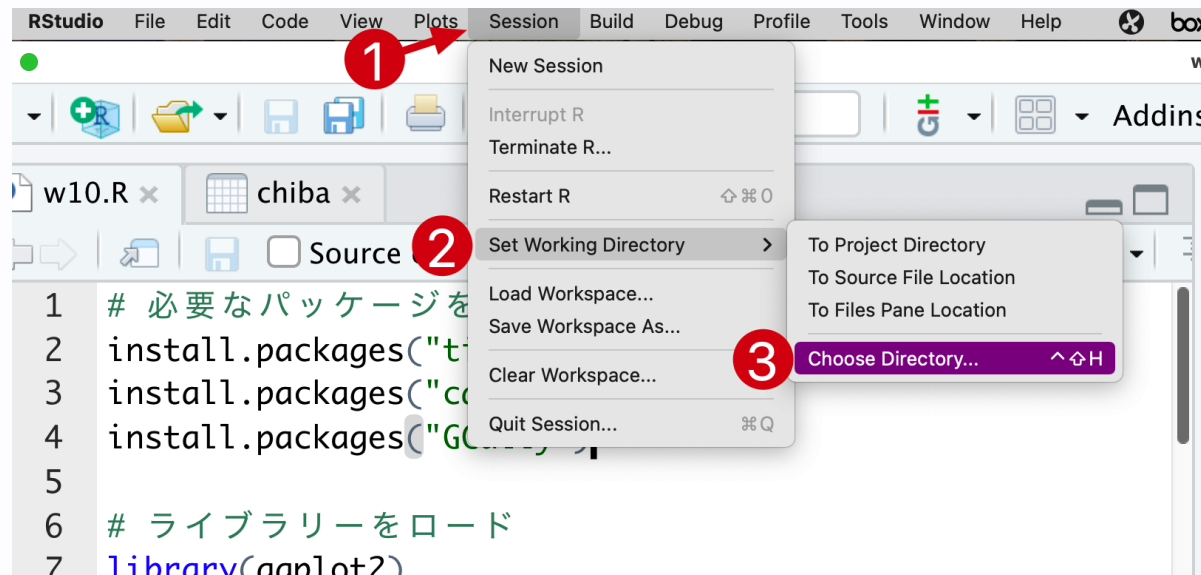
今日は一緒にRStudioでゼロから
重回帰分析をやしましょう

Part 1: Download data

- わかりやすいところに `Week11` フォルダを作る（デスクトップとか）
- クラスサイトから `class.csv` ファイルを `Week11` フォルダにダウンロード

Part 2: R Setup

- Rを立ち上げる
- Working directoryを Week11 にセットする



こうなるはず：

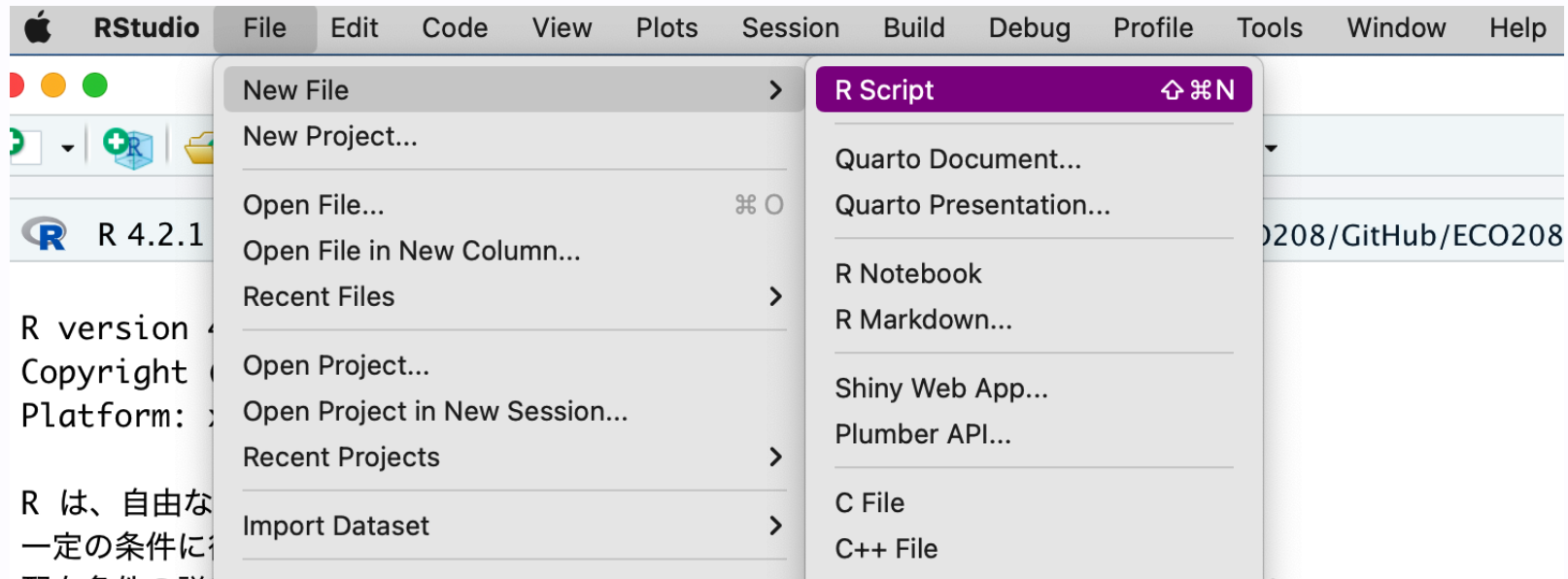
The screenshot shows the RStudio interface with the following components:

- Source Editor:** Displays R version 4.2.1 (2022-06-23) and Japanese text about R's license and usage. The text includes instructions on how to use R, such as `demo()`, `help()`, `help.start()`, and `q()`.
- Environment Pane:** Shows the Global Environment. A red box highlights the 'Sweep' button (a broom icon) in the top right corner of the pane. A red arrow points from the text 'もしここがemptyじゃなかったら sweep ボタンを押す' to this button.
- Files Pane:** Shows the file structure. A red box highlights the file 'chiba11.csv' in the 'week11' directory. A red arrow points from the text 'データファイルがあることを確認' to this file.

Additional text in the image:

- Red text in the Environment pane: **もしここがemptyじゃなかったら sweep ボタンを押す**
- Red text below the Files pane: **データファイルがあることを確認**

新しいR Scriptファイルを作る



Part 3: Get the data into R

```
# import data  
class <- read.csv("class.csv")
```

データの確認

```
summary(class)  
str(class)
```

- ここで数的 (int,float)ではない値 (chrなど) に注意
- 場合によっては統計的に関係のないカラムを削除してからもう一度読み込む

このセッションにデータをattachする

```
attach(class)
```

Part 4: Initial data inspection

plotでデータをinspectする

```
plot(class)
```


データの列が多ければこのようにsubsetする

```
class[c("koma", "study", "happiness")]
```

plotにも使える

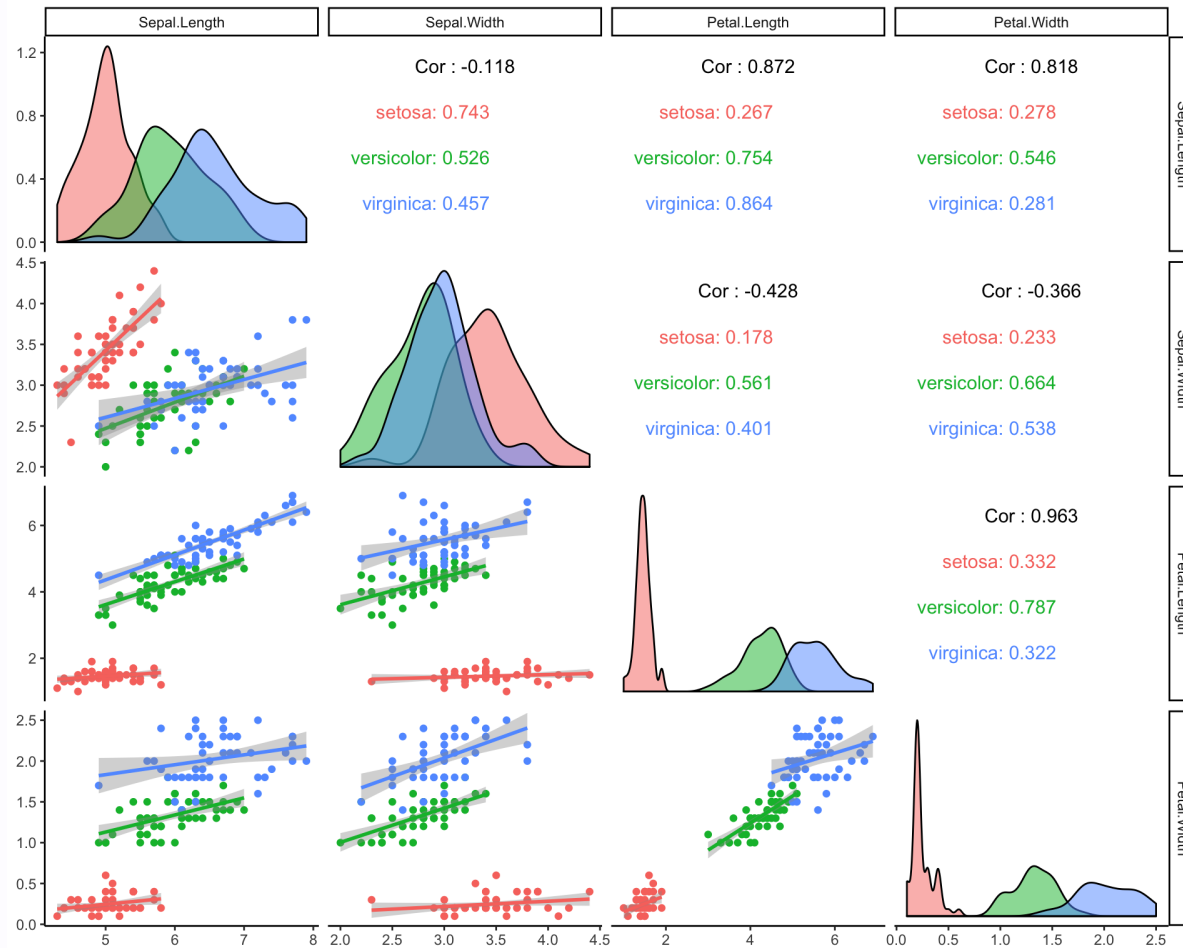
```
plot(class[c("koma", "study", "happiness")])
```

変数にしてからplotに入れるのもあり

```
class_mini <- class[c("koma", "study", "happiness")]  
plot(class_mini)
```

ggpairs()

変数間の関係を1つの図で可視化するスーパー関数



ggpairsを使うにはGGallyパッケージが必要

```
install.packages("GGally")
```

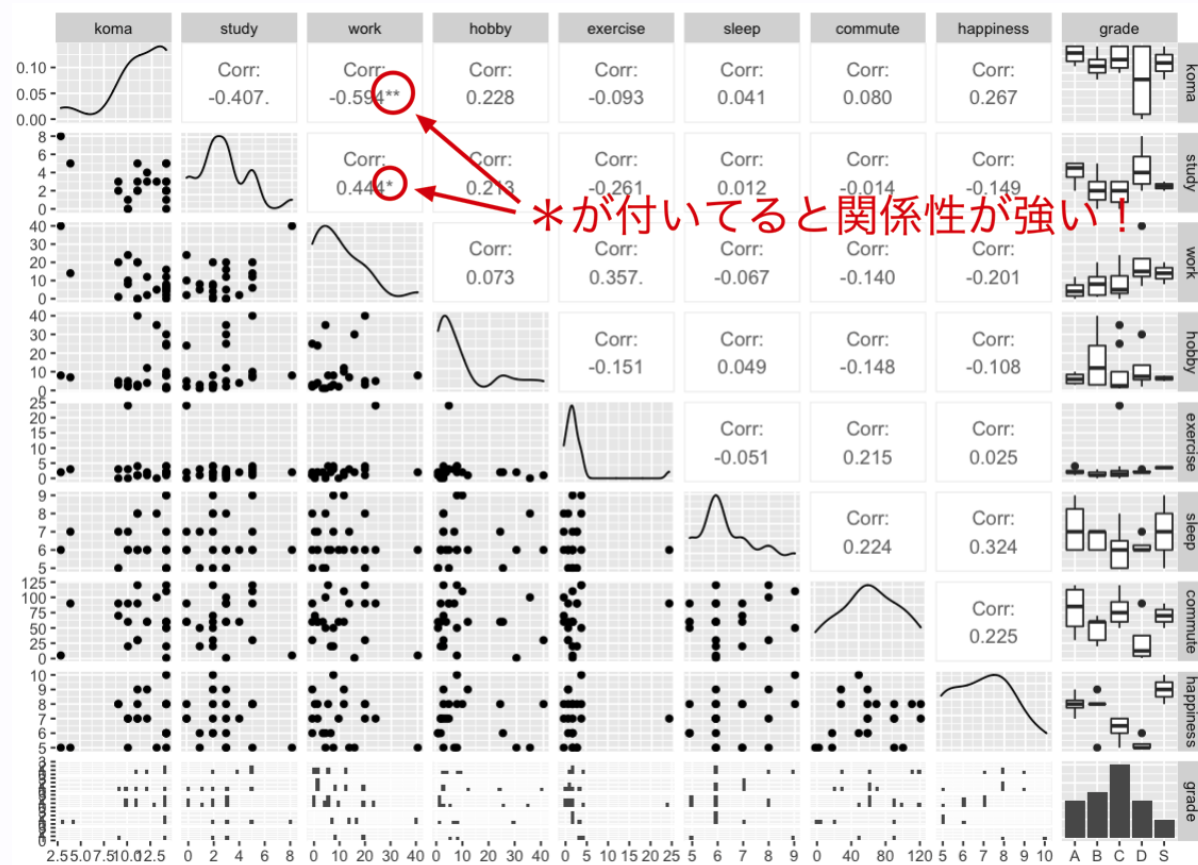
使うときはlibraryを取り込む

```
library(GGally)
```

```
ggpairs(class)
```

カラムが多ければ選べばいい：

```
ggpairs(class, columns = c("study", "work", "happiness", "grade"))
```



色々試して、説明変数、目的変数を決める

Part 5: ではモデルを作ろう

一つの手段としてはとりあえず全部入れてみる

```
model <- lm(happiness ~ ., class)  
summary(model)
```

Part 6: 変数選択

ではどの変数を使えば、最も良いモデルが作れる？


```

Residuals:
    Min       1Q   Median       3Q      Max
-1.03330 -0.23403 -0.03484  0.30832  1.32881

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  4.014853   0.452184   8.879 8.10e-13 ***
age         -0.068594   0.007566  -9.066 3.80e-13 ***
minutes     -0.058808   0.009673  -6.080 7.06e-08 ***
area         0.122917   0.012817   9.590 4.58e-14 ***
flooringない -0.357973   0.161942  -2.211  0.0306 *
konroない    0.294918   0.156907   1.880  0.0647 .
senmen       -0.092264   0.141655  -0.651  0.5171
autolock      0.317114   0.190681   1.663  0.1011
aircon        0.166318   0.199164   0.835  0.4067
bath_toilet  -0.268485   0.151838  -1.768  0.0817 .
parking       0.049833   0.160486   0.311  0.7572
corner       -0.224767   0.133322  -1.686  0.0966 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.4799 on 65 degrees of freedom
Multiple R-squared:  0.9108,    Adjusted R-squared:  0.8958
F-statistic: 60.37 on 11 and 65 DF,  p-value: < 2.2e-16

```

*が付いてるといい！

P値が一番高い変数を一つずつ消していくといいかも。



モデルを作り直す

```
model <- lm(happiness ~ work + study + hobby + exercise , class)  
summary(model)
```

色んな組み合わせで最適な結果が出るまで繰り返す！ 🧐

In a nutshell... (手短かに)

```
Call:
lm(formula = rent ~ age + minutes + area + flooring)
```

Residuals:

Min	1Q	Median	3Q	Max
-1.22332	-0.28299	0.01017	0.24967	1.51507

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)	
(Intercept)	4.262208	0.405264	10.517	3.36e-16	***
age	-0.061535	0.006341	-9.705	1.03e-14	***
minutes	-0.059760	0.009329	-6.406	1.35e-08	***
area	0.112764	0.012024	9.378	4.14e-14	***
flooringない	-0.326137	0.149887	-2.176	0.0328	*

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.5062 on 72 degrees of freedom

Multiple R-squared: 0.8901, Adjusted R-squared: 0.884

F-statistic: 145.9 on 4 and 72 DF, p-value: < 2.2e-16

*が付いてるといい

R-squared: 1 に近ければ近いほどいい

p: 低ければ低いほどいい

重回帰分析はAdjusted R-Squaredを使うように！ ➡

変数を自動的に選択できる方法もある 🤖

```
step(model)
```

この中で一番AICが低いモデルを選ぶ