

心理・教育測定法 基礎編 B 多変量解析入門

回帰分析

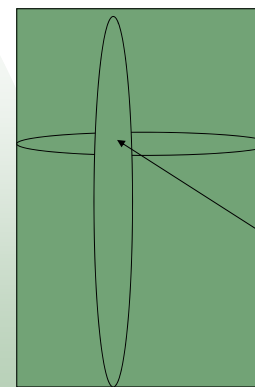
その他

多変量データ

j 列 $j=1,2,\dots,p$

i 行

$i=1, 2, 3, \dots, n$



y_{ij}

個体 i の変数 j による測定値

2014年度 心理・教育測定法 A・B、統計演習 A・B

心理・教育測定法(基礎編) A・B
金曜日 5・6限:W9-935

人間行動システム統計演習 A・B
木曜日 5・6限 W9-607

心理・教育測定法 A・B		人間行動システム統計演習 A・B	
担当		担当	
4月11日	オリエンテーション	4月10日	オリエンテーション
18日	統計量の計算・相関と回帰	17日	データフレームの操作等
25日	統計的推測の基礎	24日	ヒストグラム、クロス集計等
5月2日	平均値に関する推測(1)	5月15日	相関と回帰
9日	平均値に関する推測(2)	22日	平均値に関する推測
23日	分散分析(1)	29日	分散分析
30日	分散分析(2)	6月5日	分散分析
6月6日	イントロ	6月12日	イントロ
13日	多変量データの取り扱い	19日	多変量グラフ
20日	回帰分析	26日	回帰分析
27日	クラスター分析	7月3日	クラスター分析
7月4日	多次元尺度法	10日	多次元尺度法
11日	因子分析(1)	17日	因子分析等
18日	因子分析(2)	24日	因子分析等、Amos, R

人間行動システム統計演習 A・B には、ノート PC をご持参下さい

多変量解析法の分類

表 1: 多変量解析法の分類

		説明変数 X は観測されているか	
		観測されている (顕在変数)	観測されていない (潜在変数)
説明変数 X は	連続	回帰分析 冗長性分析 (判別分析) (ロジスティック回帰)	因子分析 主成分分析 (IRT) (カテゴリカルデータの因子分析) (数量化 III 類, CA)
	連続か 離散か	分散分析 (対数線形モデル)	クラスター分析 (潜在クラスモデル)

表中、括弧の中は、基準変数 Y がカテゴリカルな場合、それ以外は、基準変数が連続の場合を表す。また、説明変数が観測されている場合、その連続離散の区別はあまり重要ではない。

Cars93 データ

```
> data(Cars93,package="MASS")
> str(Cars93)
'data.frame': 93 obs. of 27 variables:
 $ Manufacturer : Factor w/ 32 levels "Acura","Audi",...: 1 1 2 2 3 4 4 4 4 5 ...
 $ Model       : Factor w/ 93 levels "100","190E","240",...: 49 56 9 1 6 24 54 74 73 35 ...
 $ Type       : Factor w/ 6 levels "Compact","Large",...: 4 3 1 3 3 3 2 2 3 2 ...
 $ Min.Price  : num 12.9 29.2 25.9 30.8 23.7 14.2 19.9 22.6 26.3 33 ...
 $ Price     : num 15.9 33.9 29.1 37.7 30 15.7 20.8 23.7 26.3 34.7 ...
 $ Max.Price  : num 18.8 38.7 32.3 44.6 36.2 17.3 21.7 24.9 26.3 36.3 ...
 $ MPG.city  : int 25 18 20 19 22 22 19 16 19 16 ...
 $ MPG.highway : int 31 25 26 26 30 31 28 25 27 25 ...
 $ AirBags   : Factor w/ 3 levels "Driver & Passenger",...: 3 1 2 1 2 2 2 2 2 2 ...
 $ DriveTrain : Factor w/ 3 levels "4WD","Front",...: 2 2 2 2 3 2 2 3 2 2 ...
 $ Cylinders  : Factor w/ 6 levels "3","4","5","6",...: 2 4 4 4 2 2 4 4 4 5 ...
 $ EngineSize : num 1.8 3.2 2.8 2.8 3.5 2.2 3.8 5.7 3.8 4.9 ...
 $ Horsepower : int 140 200 172 172 208 110 170 180 170 200 ...
 $ RPM       : int 6300 5500 5500 5500 5700 5200 4800 4000 4800 4100 ...
 $ Rev.per.mile : int 2890 2335 2280 2535 2545 2565 1570 1320 1690 1510 ...
 $ Man.trans.avail : Factor w/ 2 levels "No","Yes": 2 2 2 2 2 1 1 1 1 1 ...
 $ Fuel.tank.capacity : num 13.2 18 16.9 21.1 21.1 16.4 18 23 18.8 18 ...
 $ Passengers  : int 5 5 5 6 4 6 6 6 5 6 ...
 $ Length     : int 177 195 180 193 186 189 200 216 198 206 ...
 $ Wheelbase  : int 102 115 102 106 109 105 111 116 108 114 ...
 $ Width     : int 68 71 67 70 69 69 74 78 73 73 ...
 $ Turn.circle : int 37 38 37 37 39 41 42 45 41 43 ...
 $ Rear.seat.room : num 26.5 30 28 31 27 28 30.5 30.5 26.5 35 ...
 $ Luggage.room : int 11 15 14 17 13 16 17 21 14 18 ...
 $ Weight     : int 2705 3560 3375 3405 3640 2880 3470 4105 3495 3620 ...
 $ Origin     : Factor w/ 2 levels "USA","non-USA": 2 2 2 2 2 1 1 1 1 1 ...
 $ Make      : Factor w/ 93 levels "Acura Integra",...: 1 2 4 3 5 6 7 9 8 10 ...
```

Manufacturer::Manufacturer.
Model::Model.
Type::Type: a factor with levels "Small", "Sporty", "Compact", "Midsize", "Large" and "Van".
Min.Price::Minimum Price (in ¥\$1,000):: price for a basic version.
Price::Midrange Price (in ¥\$1,000):: average of Min.Price and Max.Price.
Max.Price::Maximum Price (in ¥\$1,000):: price for "a premium version".
MPG.city::City MPG (miles per US gallon by EPA rating).
MPG.highway::Highway MPG.
AirBags::Air Bags standard. Factor:: none, driver only, or driver & passenger.
DriveTrain::Drive train type:: rear wheel, front wheel or 4WD; (factor).
Cylinders::Number of cylinders (missing for Mazda RX-7, which has a rotary engine).
EngineSize::Engine size (litres).
Horsepower::Horsepower (maximum).
RPM::RPM (revs per minute at maximum horsepower).
Rev.per.mile::Engine revolutions per mile (in highest gear).
Man.trans.avail::Is a manual transmission version available? (yes or no, Factor).
Fuel.tank.capacity::Fuel tank capacity (US gallons).
Passengers::Passenger capacity (persons)
Length::Length (inches).
Wheelbase::Wheelbase (inches).
Width::Width (inches).
Turn.circle::U-turn space (feet).
Rear.seat.room::Rear seat room (inches) (missing for 2-seater vehicles).
Luggage.room::Luggage capacity (cubic feet) (missing for vans).
Weight::Weight (pounds).
Origin::Of non-USA or USA company origins? (factor).
Make::Combination of Manufacturer and Model (character).

相関係数と散布図

```
# Pick up numeric variables
cars93n=Cars93[,c(8, 5, 12, 13, 14, 19, 20, 21, 25)]
colnames(cars93n)[colnames(cars93n)=="MPG.highway"]="MPGh"
attach(cars93n)
```

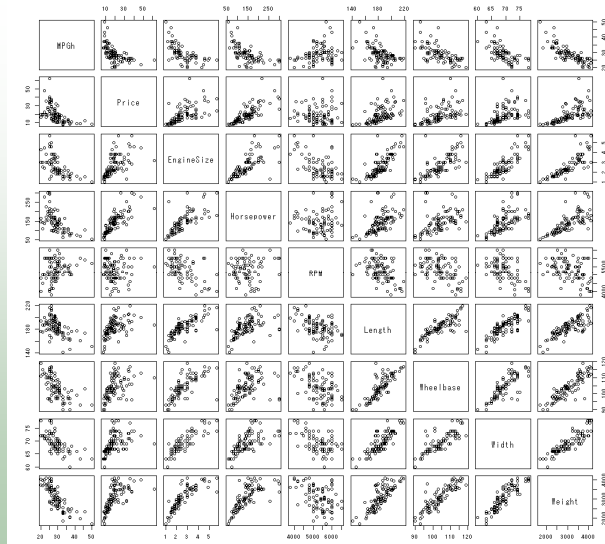
```
# correlation matrix
R=cor(cars93n)
```

```
# scatter plot
pairs(cars93n)
```

相関係数行列

	MPGh	Price	EngineSize	Horsepower	RPM	Length	Wheelbase	Width	Weight
MPGh	1.00	-0.56	-0.63	-0.62	0.31	-0.54	-0.62	-0.64	-0.81
Price	-0.56	1.00	0.60	0.79	-0.00	0.50	0.50	0.46	0.65
EngineSize	-0.63	0.60	1.00	0.73	-0.55	0.78	0.73	0.87	0.85
Horsepower	-0.62	0.79	0.73	1.00	0.04	0.55	0.49	0.64	0.74
RPM	0.31	-0.00	-0.55	0.04	1.00	-0.44	-0.47	-0.54	-0.43
Length	-0.54	0.50	0.78	0.55	-0.44	1.00	0.82	0.82	0.81
Wheelbase	-0.62	0.50	0.73	0.49	-0.47	0.82	1.00	0.81	0.87
Width	-0.64	0.46	0.87	0.64	-0.54	0.82	0.81	1.00	0.87
Weight	-0.81	0.65	0.85	0.74	-0.43	0.81	0.87	0.87	1.00

散布図行列



相関係数行列の並べ替え

```
R=cor(cars93n)
vecR=R[t(lower.tri(R))]
names(vecR)=outer(rownames(R),rownames(R),paste,sep="-")[t(lower.tri(R))]
od=order(abs(vecR),decreasing=1)
vecR=vecR[od]
vecR=matrix(vecR,,1, dimnames=list(names(vecR),NULL))
```

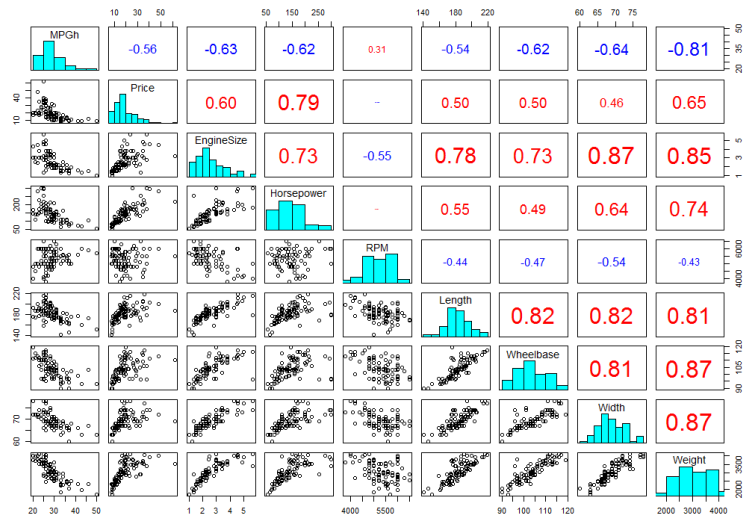
相関係数行列並べ替え

Width-Weight	0. 874960542	EngineSize-RPM	-0. 547897805
Wheelbase-Weight	0. 871895314	MPGh-Length	-0. 542897411
EngineSize-Width	0. 867110168	RPM-Width	-0. 539721132
EngineSize-Weight	0. 845075335	Price-Length	0. 503628440
Length-Wheelbase	0. 823650435	Price-Wheelbase	0. 500864163
Length-Width	0. 822147852	Horsepower-Wheelbase	0. 486854213
MPGh-Weight	-0. 810658057	RPM-Wheelbase	-0. 467812289
Wheelbase-Width	0. 807213365	Price-Width	0. 456027866
Length-Weight	0. 806274301	RPM-Length	-0. 441249316
Price-Horsepower	0. 788217578	RPM-Weight	-0. 427931473
EngineSize-Length	0. 780283106	MPGh-RPM	0. 313468728
Horsepower-Weight	0. 738797516	Horsepower-RPM	0. 036688212
EngineSize-Wheelbase	0. 732484169	Price-RPM	-0. 004954931
EngineSize-Horsepower	0. 732119730		
Price-Weight	0. 647179005		
Horsepower-Width	0. 644413421		
MPGh-Width	-0. 640359217		
MPGh-EngineSize	-0. 626794553		
MPGh-Horsepower	-0. 619043685		
MPGh-Wheelbase	-0. 615384250		
Price-EngineSize	0. 597425392		
MPGh-Price	-0. 560680362		
Horsepower-Length	0. 550864666		

散布図行列 + α

```
# scatter plot matrix with histograms and correlation coefficients
pairs(cars93n, upper.panel=panel.cor, diag.panel=panel.hist, cex.col=1)
```

散布図行列 + α



回帰モデル

$$\begin{aligned}
 \hat{y}_i &= \mathcal{E}(y_i | \mathbf{x}_i) \\
 &= f(\mathbf{x}_i) \\
 &= \beta_0 + \beta_1 x_{i1} \\
 &= \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} \\
 &\vdots \\
 &= \sum_{j=0}^{q+1} \beta_j x_{ij}
 \end{aligned}$$

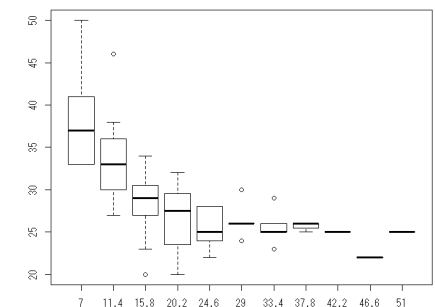
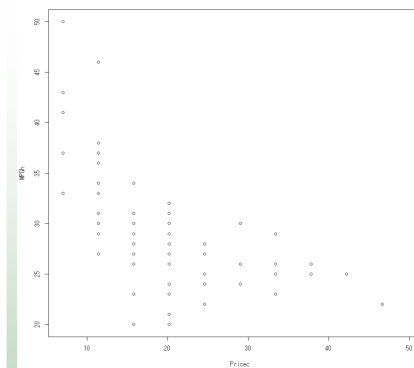
離散化した説明変数ごとの平均

Price と Weight を適当に離散化したものを Pricec と Weightc とする。

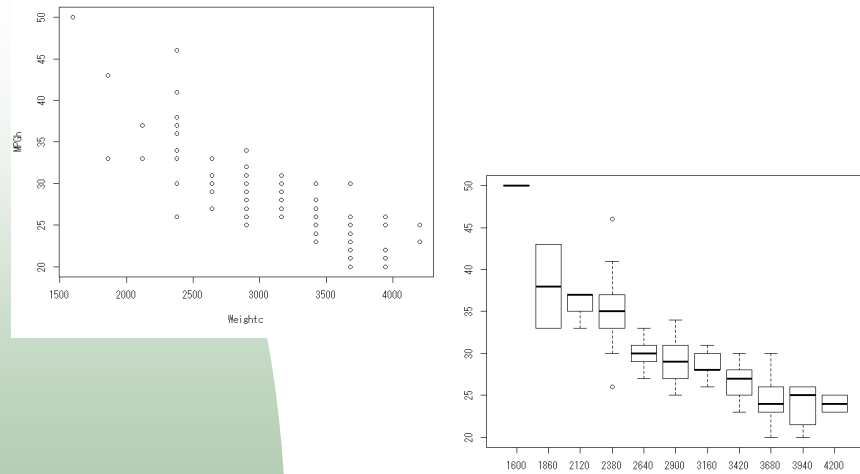
```
plot( Pricec, MPGh )
boxplot( MPGh ~ Pricec )
```

```
plot( Weightc, MPGh )
boxplot( MPGh ~ Weightc )
```

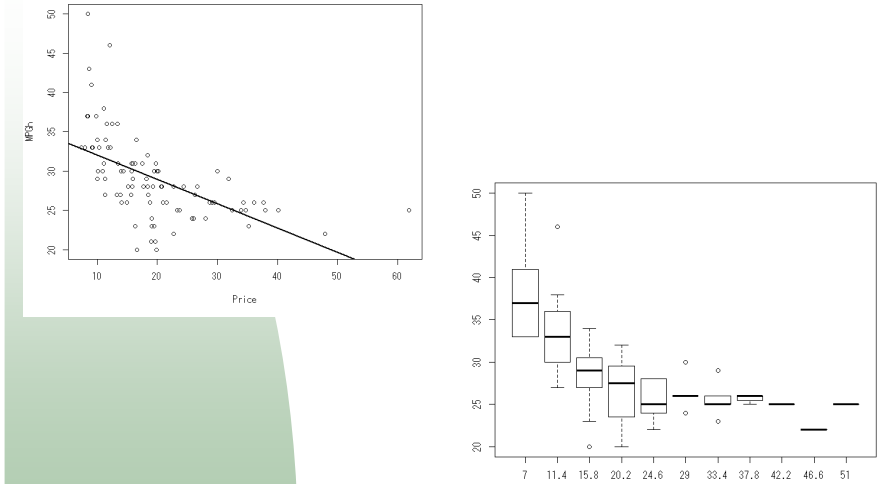
MPGh ~ Price



MPGh ~ Weight



MPGh ~ Price

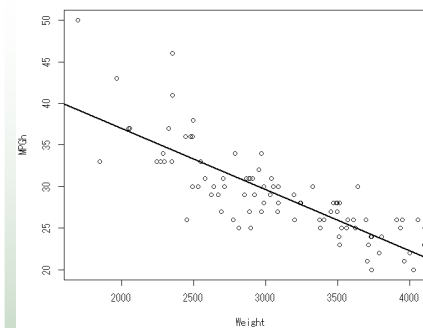


単回帰直線

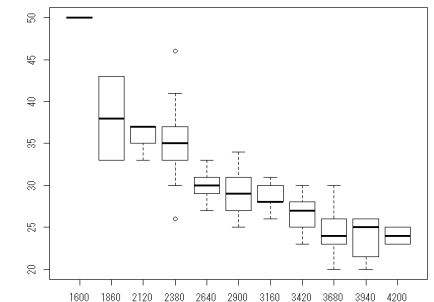
```
plot( Price, MPGh )  
abline( lm(MPGh ~ Price), lty=1, lwd=2 )
```

```
plot( Weight, MPGh )  
abline( lm(MPGh ~ Weight), lty=1, lwd=2 )
```

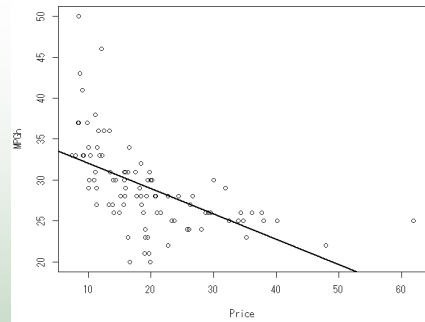
MPGh ~ Weight



```
lm(formula = MPGh ~ Weight)  
Coefficients:  
(Intercept)  Weight  
51.601365  -0.007327
```



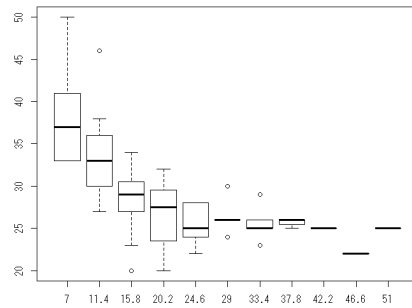
MPGh ~ Price



lm(formula = MPGh ~ Price)

Coefficients:

(Intercept) Price
35.1239 -0.3095



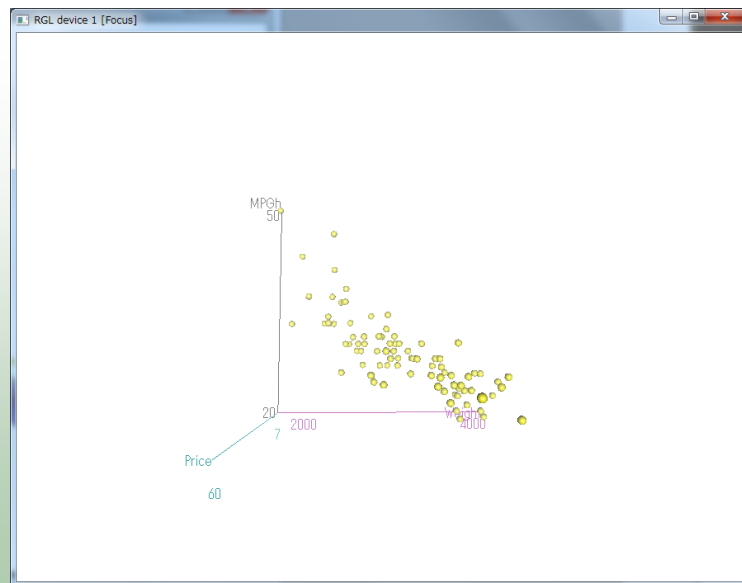
重回帰分析

■ 3変数のプロット

```
library(Rcmdr)
scatter3d(Weight, MPGh, Price)
scatter3d(Weight, MPGh, Price, surface=0)
```

```
library(rgl)
plot3d(Weight, Price, MPGh)
reslm=lm(MPGh~Price+Weight)
b=reslm$coef
planes3d( b["Price"], b["Weight"],-1,b["Intercept"], alpha=1)
```

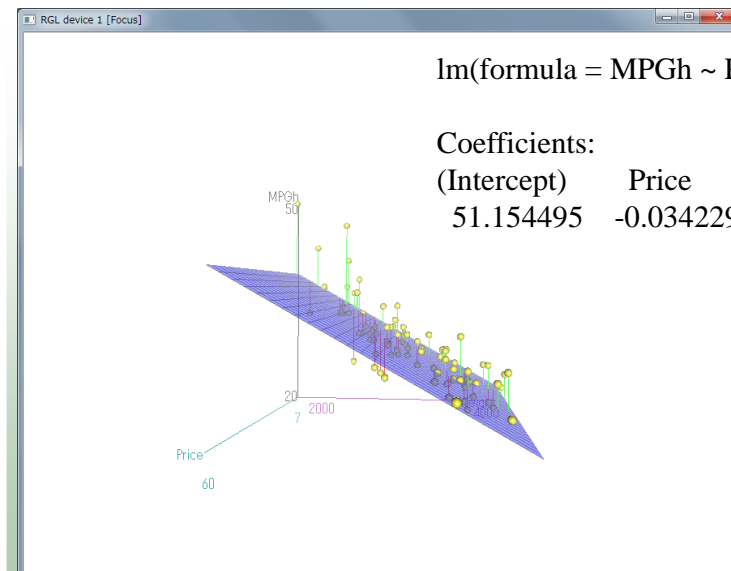
MPGh ~ Price + Weight



lm(formula = MPGh ~ Price + Weight)

Coefficients:

(Intercept) Price Weight
51.154495 -0.034229 -0.006964



変数選択

```
library(leaps)
```

```
resss=regsubsets(MPGh ~ EngineSize + Horsepower + Length  
+ Price + RPM + Weight + Wheelbase + Width  
, data=Cars93, nbest=3, nvmax=3)
```

```
plot(resss)
```

いろいろなモデルを作って見て、当てはまりの良さを比較する。

AIC or BIC

