Stat 428/528: Advanced Data Analysis 2

Chapters 3 and 10: Variable Selection and Model Building

Instructor: Yan Lu

Topics:

- Become familiar with model selection criteria
- Understand when/how to use selection algorithms such as stepwise and best subsets
- Understand how to validate a model

General multiple linear regression model

$$Y_i = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \cdots + \beta_{p-1} X_{i,p-1} + \epsilon_i,$$

- $i = 1, 2, \cdots, n$
- Y_i is the value of the response variable for the ith case
- $X_{i1}, X_{i2}, \cdots, X_{i,p-1}$ are known constants, X_{ik} is the value of the kth explanatory variable for the ith case
- $\beta_0, \beta_1, \cdots, \beta_{p-1}$ are parameters, $p-1$ predictors, p parameters
- $\epsilon_i \overset{iid}{\sim} N(0, \sigma^2)$

Multiple Regression

- Multiple–More than one predictor variable
- Y_i is the response variable
- $X_{i1}, X_{i2}, \cdots X_{i,p-1}$ are the $p-1$ explanatory variables for cases $i = 1$ to n
- Potential problem: These predictor variables are likely to be themselves correlated

- models with lots of predictors (especially interactions) are harder to interpret
- models with lots of predictors will tend to have larger confidence intervals for their estimates
- models with too many predictors can be "overfitted"–they account for the current data but are unlikley to generalize well to future data sets
- often we have more predictors than observations!
- often predictors are very closely related, and so have redundant information (collinearity, more on this later)

Problems: have a set of predictor variables, how do you select a subset of these that is in some way "best" for predicting the response?

- Subset size, how many explanatory variables should be used to construct the regression model
- Given the subset size, which variables should we choose?

Model selection

Often, philosophers are interested in big scientific theories, such as Copernicus's sun-centered solar system versus earth-centered solar system models, Darwin's theory of natural selection, Freud's theories about the subconscious, Relativity, etc. In statistics, our goals are usually more

modest, and often we are not looking for models that are literally true.

- We are usually quite happy with models that find relationships between variables that are approximately correct and that find trends in the data rather than exact relationships.
- A famous saying from the statistician George Box is

"All models are wrong, but some are useful"

—— Here usefulness might mean that we can make predictions that help us plan for the future, or that we can be convinced that certain variables are more important than others for understanding things like graduation rates.

Notations:

- $P-1$: total possible number of predictor variables
- $p-1$: number of predictor variables selected in a regression model, p is the number of parameters in the model.
- $p-1 \leq P-1$, $n > p$
- For any set of $p-1$ predictors, 2^{p-1} alternative models can be constructed, including the one with no X variables.

Criteria for Model Selection

1. R_p^2 or SSE_p Criterion
   - R_p^2 is the coefficient of Multiple Determination for model with $p-1$ predictors
   - $R_p^2 = 1 - SSE_p/SSTO$
   - Plot R_p^2 v.s p, R_p^2 will increase as $p-1$ increases.
   - The R_p^2 plot will tend to level off at some point. Take the model to be the one where there is no more "meaningful" increase in R_p^2.
   - A drawback to R^2 is that the addition of any variable to the model (significant or not) will increase R^2.

Example 1

Goal: Anthropologists conducted a study to determine the long-term effects of an environmental change on systolic blood pressure.
——They measured the blood pressure and several other characteristics of 39 Indians who migrated from a very primitive environment high in the Andes into the mainstream of Peruvian society at a lower altitude.
—— All of the Indians were males at least 21 years of age, and were born at a high altitude.

```
> fn.data <- "http://statacumen.com/teach/ADA2/
ADA2_notes_Ch02_indian.dat"
> indian <- read.table(fn.data, header=TRUE)
>indian$yrage <- indian$yrmig / indian$age
> indian
id age yrmig   wt   ht chin fore calf pulse sysbp diabp      y
1  21     1 71.0 1629  8.0  7.0 12.7    88   170    76 0.0476
2  22     6 56.5 1569  3.3  5.0  8.0    64   120    60 0.2727
3  24     5 56.0 1561  3.3  1.3  4.3    68   125    75 0.2083
4  24     1 61.0 1619  3.7  3.0  4.3    52   148   120 0.0416
5  25     1 65.0 1566  9.0 12.7 20.7    72   140    78 0.0400
6  27    19 62.0 1639  3.0  3.3  5.7    72   106    72 0.7037
7  28     5 53.0 1494  7.3  4.7  8.0    64   120    76 0.1785
8  28    25 53.0 1568  3.7  4.3  0.0    80   108    62 0.8928
9  31     6 65.0 1540 10.3  9.0 10.0    76   124    70 0.1935
10  32    13 57.0 1530  5.7  4.0  6.0    60   134    64 0.4062
```

```
> signif(i.cor$r[1, ], 3)
 sysbp     wt    ht   chin   fore   calf  pulse  yrage
 1.000  0.521 0.219  0.170  0.272  0.251  0.133 -0.276
```

```
> # The leaps package provides best subsets with other selecti
> library(leaps)
> # First, fit the full model
> lm.indian.full <- lm(sysbp ~ wt + ht + chin + fore + calf
+ pulse + yrage, data = indian)
> summary(lm.indian.full)

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 106.45766   53.91303   1.975 0.057277 .
wt            1.71095    0.38659   4.426 0.000111 ***
ht           -0.04533    0.03945  -1.149 0.259329
chin         -1.15725    0.84612  -1.368 0.181239
fore         -0.70183    1.34986  -0.520 0.606806
calf          0.10357    0.61170   0.169 0.866643
pulse         0.07485    0.19570   0.383 0.704699
yrage       -29.31810    7.86839  -3.726 0.000777 ***
---
```

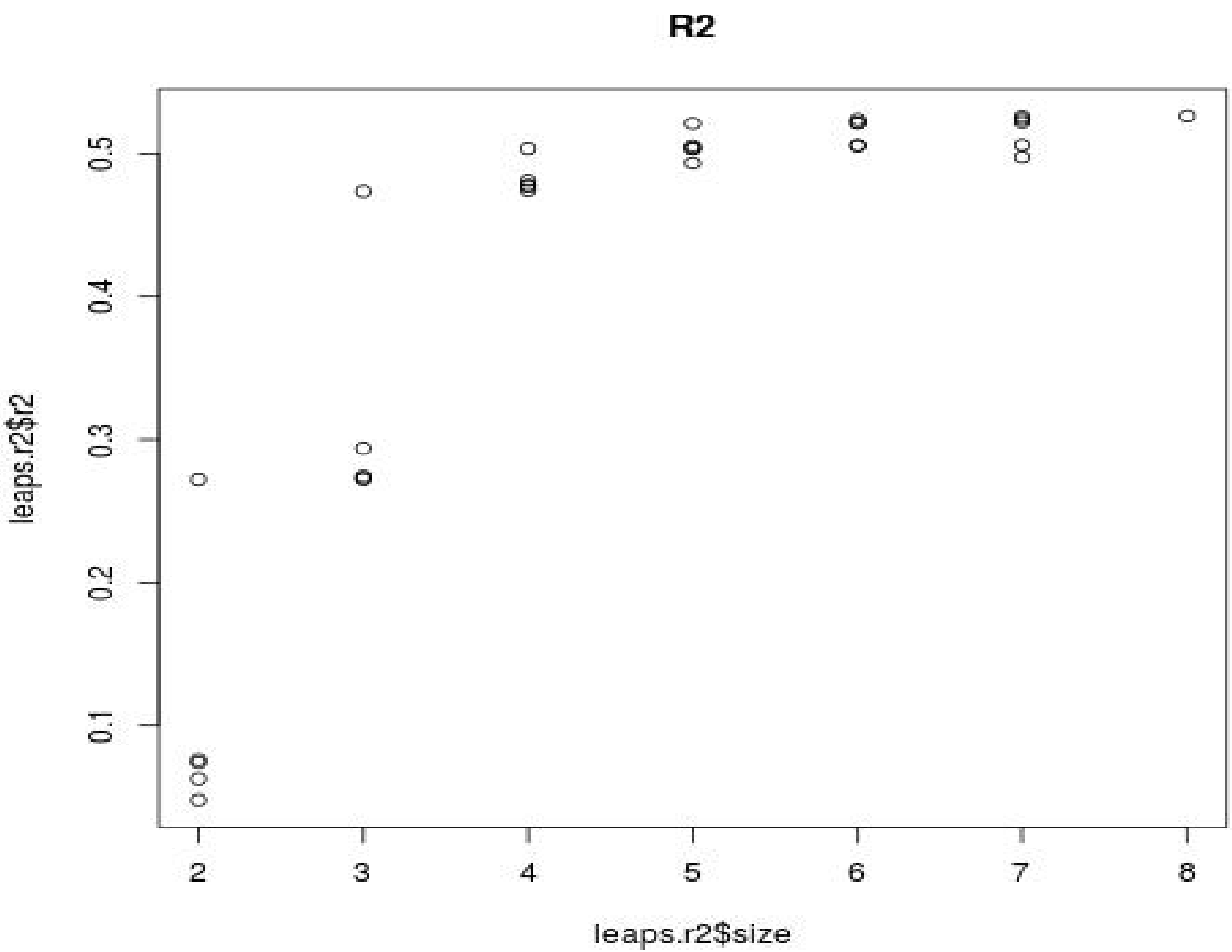
```
> # R^2 -- for each model size, report best subset of
 size 5
X.indian <- indian[,c(4:9,12)]
leaps.r2 <- leaps(x = X.indian, y = indian$sysbp
                  , method = 'r2'
                  ,nbest = 5, names =c("wt","ht", "chin",
                  "fore", "calf", "pulse","yrage"))
> leaps.r2
$which
     wt    ht  chin  fore  calf pulse yrage
1  TRUE FALSE FALSE FALSE FALSE FALSE FALSE
1 FALSE FALSE FALSE FALSE FALSE FALSE  TRUE
1 FALSE FALSE FALSE  TRUE FALSE FALSE FALSE
1 FALSE FALSE FALSE FALSE  TRUE FALSE FALSE
1 FALSE  TRUE FALSE FALSE FALSE FALSE FALSE
2  TRUE FALSE FALSE FALSE FALSE FALSE  TRUE
2  TRUE FALSE  TRUE FALSE FALSE FALSE FALSE
```

```
$label
[1] "(Intercept)" "wt"          "ht"          "chin"
 "fore"         "calf"        "pulse"       "yrage"

$label
[1] "(Intercept)" "wt"          "ht"          "chin"

$size
 [1] 2 2 2 2 2 3 3 3 3 3 4 4 4 4 4 5 5 5 5 5 6 6 6 6 6 7 7 7

$r2
 [1] 0.27182072 0.07625642 0.07413652 0.06289527 0.04801119
[13] 0.47773431 0.47674226 0.47400828 0.52071413 0.50513668
[25] 0.50517225 0.52547798 0.52367894 0.52178233 0.50572524
# plot model R^2 vs size of model
plot(leaps.r2$size, leaps.r2$r2, main = "R2")
```


R2
leaps.r2$r2
0.1
0.2
0.3
0.4
0.5
2
3
4
5
6
7
8
leaps.r2$size

```
> leaps.r2$which[order(-leaps.r2$r2)[1:5],]
    wt   ht chin  fore  calf pulse yrage
7 TRUE TRUE TRUE  TRUE  TRUE  TRUE  TRUE
6 TRUE TRUE TRUE  TRUE FALSE  TRUE  TRUE
6 TRUE TRUE TRUE  TRUE  TRUE FALSE  TRUE
5 TRUE TRUE TRUE  TRUE FALSE FALSE  TRUE
6 TRUE TRUE TRUE FALSE  TRUE  TRUE  TRUE
> leaps.r2$r2[order(-leaps.r2$r2)[1:5]]
[1] 0.5259164 0.5254780 0.5236789 0.5234360 0.5217823
```

All the five best models are with most of the variables.

```
# report the best model (indicate which terms are
in the model)
> best.model.r2 <- leaps.r2$which[which((leaps.r2$r2 ==
 max(leaps.r2$r2))),]
> # these are the variable names for the best model
> names(best.model.r2)[best.model.r2]
[1] "(Intercept)" "wt"          "ht"          "chin"
  "fore"        "calf"        "pulse"       "yrage"
>
```

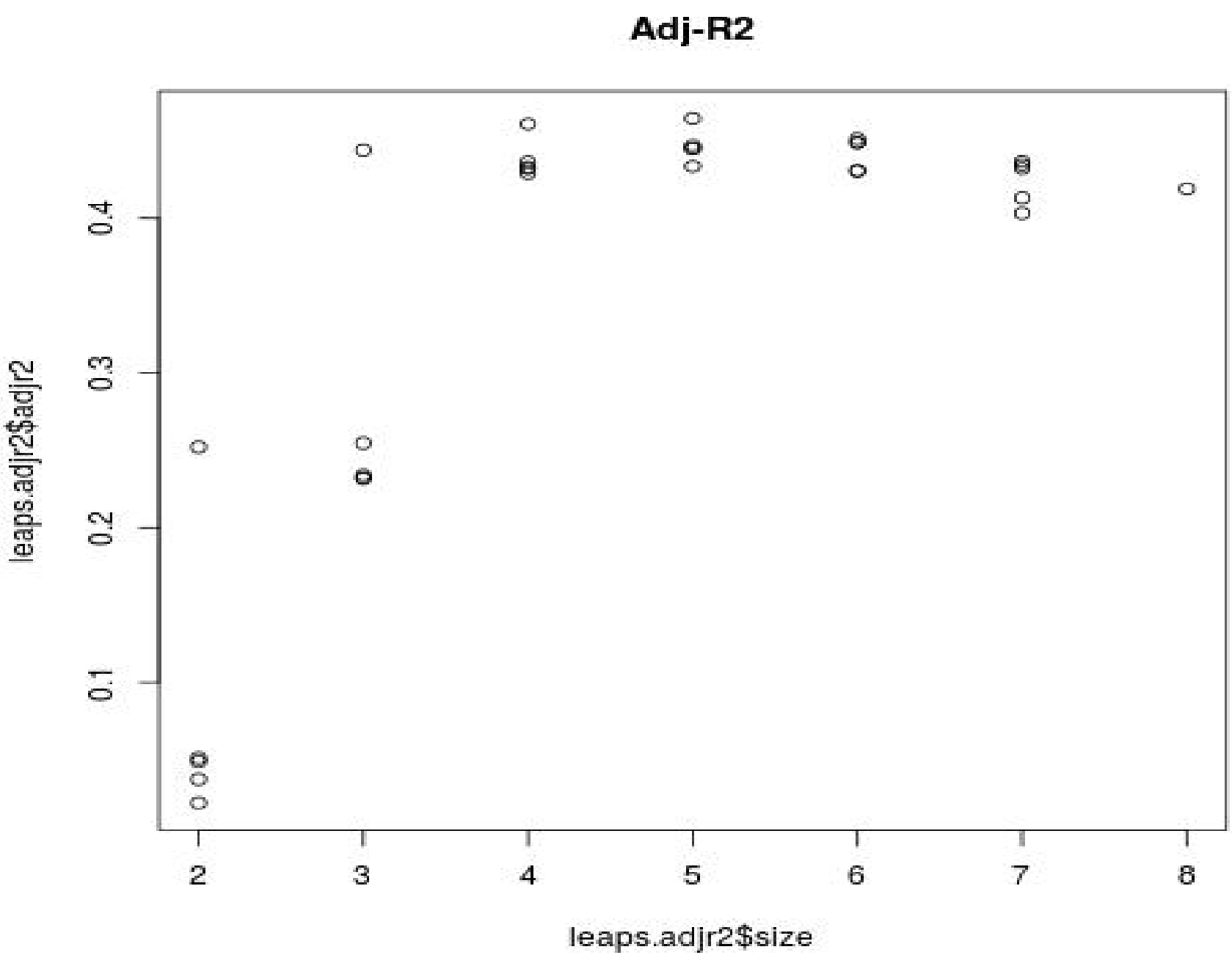
2. $R^2_{a,p}$ or MSE_p Criterion

$$R^2_{a,p} = 1 - \frac{SSE_p/(n-p)}{SSTO/(n-1)}$$

$$= 1 - \frac{MSE_p}{SSTO/(n-1)}$$

- $R^2_{a,p}$ increases if and only if MSE_p decreases. This is the same as using MSE.
- Select the subset with the largest $R^2_{a,p}$ or there is no meaningful increase in $R^2_{a,p}$

```
# adj-R^2 -- for each model size, report best
 subset of size 5
# adj-R^2 -- for each model size, report best subset of size 5
leaps.adjr2 <- leaps(x = X.indian, y = indian$sysbp
                     , method = 'adjr2'
                     , nbest = 5, names =c("wt","ht", "chin",
                      "fore", "calf", "pulse","yrage"))
# plot model R^2 vs size of model
plot(leaps.adjr2$size, leaps.adjr2$adjr2, main = "Adj-R2")
```


Adj-R2
leaps.adjr2$adjr2
0.1
0.2
0.3
0.4
2
3
4
5
6
7
8
leaps.adjr2$size

```
> leaps.adjr2$which[order(-leaps.adjr2$adjr2)[1:5],]
    wt    ht chin  fore  calf pulse yrage
4 TRUE  TRUE TRUE FALSE FALSE FALSE  TRUE
3 TRUE FALSE TRUE FALSE FALSE FALSE  TRUE
5 TRUE  TRUE TRUE  TRUE FALSE FALSE  TRUE
5 TRUE  TRUE TRUE FALSE FALSE  TRUE  TRUE
5 TRUE  TRUE TRUE FALSE  TRUE FALSE  TRUE
> leaps.adjr2$adjr2[order(-leaps.adjr2$adjr2)[1:5]]
[1] 0.4643276 0.4607546 0.4512293 0.4488217 0.4484703
```

```
# report the best model (indicate which
terms are in the model)
best.model.adjr2 <- leaps.adjr2$
which[which((leaps.adjr2$adjr2 == max(leaps.adjr2$adjr2))),]
> # these are the variable names for the best model
> names(best.model.adjr2)[best.model.adjr2]
[1] "wt"    "ht"    "chin"  "yrage"
>
```

Model with only "wt","chin" and "yrage" is with adjusted R^2 of 0.4643276 compared to the best model ("wt", "ht" "chin", "yrage") with adjusted R^2 of 0.4607. For simplicity, it is actually better to select the model with "wt","chin" and "yrage" according to adjusted R^2 criterion consider assumptions are all met.

3. Mallow's C_p criterion

- Mallow's criterion tries to find the model that minimizes

$$\frac{1}{\sigma^2}\sum_{i=1}^{n} E[(\hat{y}_i - E(y_i))^2]$$

- Mallows found an estimate for this criterion called C_p with

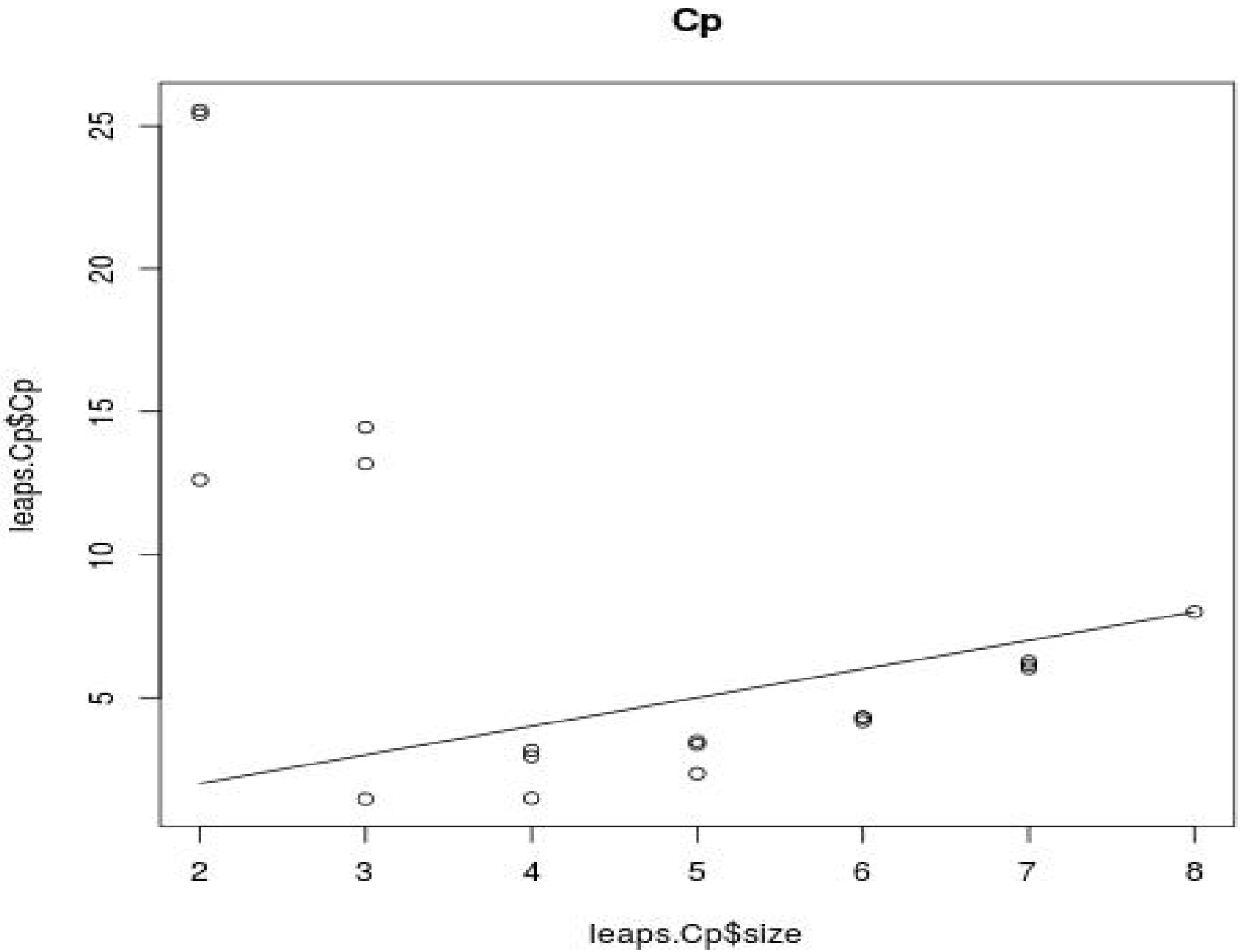
$$C_p = \frac{SSE_p}{MSE_{(Full)}} - (n - 2p).$$

The full model is good at prediction, but if there is multicollinearity, our interpretations of the parameter estimates may not makes sense. A subset model is good if there is not substantial bias in the predicted values (relative to the full model). The C_p criterion looks at the ratio of error SS for the model with p variables to the MSE of the full model, then adds a penalty for the number of variables. SSE_p is based on a specific choice of $p-1$ predictors; while $MSE_{(Full)}$ is based on the full set of variables.

- Adequately fitted model have $C_p \approx p$. Models with lack of fit have $C_p > p$. In considering possible models we would generally consider any subset with $C_p \leq p$.
- Select as the "best" subset, the one with the smallest C_p value.

```
# Cp -- for each model size, report best subset of size 3
leaps.Cp <- leaps(x = X.indian, y = indian$sysbp
                  , method = 'Cp'
                  , nbest = 3, names = c("wt","ht", "chin",
                  "fore", "calf", "pulse","yrage"))

# plot model R^2 vs size of model
plot(leaps.Cp$size, leaps.Cp$Cp, main = "Cp")
  lines(leaps.Cp$size, leaps.Cp$size)
   # adds the line for Cp = p
```


Cp
leaps.Cp$Cp
5
10
15
20
25
2
3
4
5
6
7
8
leaps.Cp$size

```
> leaps.Cp$which[order(leaps.Cp$Cp)[1:5],]
    wt    ht  chin  fore  calf pulse yrage
2 TRUE FALSE FALSE FALSE FALSE FALSE  TRUE
3 TRUE FALSE  TRUE FALSE FALSE FALSE  TRUE
4 TRUE  TRUE  TRUE FALSE FALSE FALSE  TRUE
3 TRUE FALSE FALSE  TRUE FALSE FALSE  TRUE
3 TRUE FALSE FALSE FALSE  TRUE FALSE  TRUE
> leaps.Cp$Cp[order(leaps.Cp$Cp)[1:5]]
[1] 1.453122 1.477132 2.340175 2.947060 3.150596
```

Model with "wt" and "yrage" is with cp value of $1.453122 < p = 3$, choose this model according to Cp criterion.

4. *AIC* and *BIC*

These criteria are motivated from information theory (AIC) and from Bayesian statistics. They are Criteria based on log(likelihood) plus a penalty for more complexity. We want to choose models that minimize AIC and BIC.

$$AIC = -2\ln(L) + 2p$$

$$BIC = -2\ln(L) + p[\ln(n)]$$

L is the maximized value of the likelihood function for the estimated model.

```
# best subset, returns results sorted by BIC
f.bestsubset <- function(form, dat, nbest = 5){
  library(leaps)
  bs <- regsubsets(form, data=dat, nvmax=30, nbest=nbest,
  method="exhaustive");
  bs2 <- cbind(summary(bs)$which
               , (rowSums(summary(bs)$which)-1)
                , summary(bs)$rss
                , summary(bs)$rsq
                , summary(bs)$adjr2
                , summary(bs)$cp
                , summary(bs)$bic);
  cn <- colnames(bs2);
    cn[(dim(bs2)[2]-5):dim(bs2)[2]]
     <- c("SIZE", "rss", "r2", "adjr2", "cp", "bic");
    colnames(bs2) <- cn;
  ind <- sort.int(summary(bs)$bic, index.return=TRUE);
   bs2 <- bs2[ind$ix,]; return(bs2);}
```

```
# perform on our model
i.best <- f.bestsubset(formula(sysbp ~ wt + ht + chin
+ fore + calf + pulse + yrage)
          , indian)
  op <- options(); # saving old options
  options(width=90) # setting command window
  output text width wider
i.best
  options(op); # reset (all) initial options
```

```
> i.best
  (Intercept) wt ht chin fore calf pulse yrage SIZE
2           1  1  0    0    0    0     0     1    2
3           1  1  0    1    0    0     0     1    3
3           1  1  0    0    1    0     0     1    3
3           1  1  0    0    0    1     0     1    3
3           1  1  1    0    0    0     0     1    3
  rss          r2       adjr2
  3441.363 0.47310778 0.44383599
3243.990 0.50332663 0.46075463
3390.815 0.48084699 0.43634816
3411.145 0.47773431 0.43296868
3417.624 0.47674226 0.43189159
          cp          bic
2  1.453122 -13.9989263
3  1.477132 -12.6388375
3  2.947060 -10.9124614
3  3.150596 -10.6793279
```

- R^2 select the model with all the 7 variables since R^2 always increase when there is variable added.
  according to the plot of R^2 v.s. size of the model, R^2 tend to increase and then level off when size is 4, i.e., 3 variables
- Adjusted R^2 select the model with "wt"ht", "chin" and "yrage" (0.9943862). For simplicity, model with "wt, "chin" and "yrage" performs well with adjusted R^2 of 0.9943487.
- Cp select model with "wt" and "yrage" with the smallest cp value of $1.453122 < p = 3$.
- BIC select model with "wt" and "yrage" with the smallest BIC value of -13.9989263.

For simplicity, model with "wt" and "yrage"may be preferred.

- Check model assumptions of this model
  All the assumptions seems not violated (chapter 2)
- We then decide this is the final model for use

Comments

- The different criteria will not always give the identical answer.
- The all subsets method is good for identifying a collection of possible models. One should not necessarily use the model that is declared "best" by any method.
- There might be several subsets that provide a good fit. The final selection of a model should involve residual analysis and knowledge of the subject matter.

Comments:

- As the number of predictors increases, the number of possible models blows up! We need clever computer algorithms to find the really good models.

  Two possible approaches:
- If $p-1$ is less than 30, use best subsets procedures
- If $p-1$ is greater than 30, use stepwise procedures: These are "greedy" algorithms that first find the best single term model. Given that term, add the next best term, and so on.

Stepwise Regression analysis

- A computationally available method for subset selection
- Evaluate the variables one at a time and look at a sequence of models
- Backwards elimination (start with full additive model)
- Forward elimination (start with intercept model)
- Stepwise methods (variables can be both added and deleted)

Backwards elimination

- Begins with the full model and sequentially eliminates from the model the least important variable. Importance of the variable is judged by the size t or F statistic or the p-value.
- $$F_i^* = \frac{MSR(x_i | x_1, \cdots, x_{p-1} \quad \text{except} \quad x_i)}{MSE(x_1, x_2, \cdots, x_{p-1})}, \quad \text{for} \quad i = 1, 2, \cdots, p-1.$$

  Find the smallest F_i^*, If the smallest $F_i^* < F - out$ (predetermined value), remove x_i; or find the largest p-value that is greater than the nominal level, remove that variable associated.
- After the variable with the smallest F statistic is dropped, the model is refitted and the F statistic is recalculated. Again, the variable with the smallest F statistic is dropped
- Process ends when all of F statistics are greater than some predetermined level (predetermined value can change depending on the step).

```
> # First, fit the full model
> lm.indian.full <- lm(sysbp ~ wt + ht + chin + fore + calf
+ pulse + yrage, data = indian)
> summary(lm.indian.full)

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 106.45766   53.91303   1.975 0.057277 .
wt            1.71095    0.38659   4.426 0.000111 ***
ht           -0.04533    0.03945  -1.149 0.259329
chin         -1.15725    0.84612  -1.368 0.181239
fore         -0.70183    1.34986  -0.520 0.606806
calf          0.10357    0.61170   0.169 0.866643
pulse         0.07485    0.19570   0.383 0.704699
yrage       -29.31810    7.86839  -3.726 0.000777 ***
---
Residual standard error: 9.994 on 31 degrees of freedom
Multiple R-squared:  0.5259,Adjusted R-squared:  0.4189
```

The least important variable in the full model, as judged by the p-value, is calf skin fold.

This variable, upon omission, reduces R^2 the least, or equivalently, increases the Residual SS the least.

The p-value of 0.87 exceeds the default 0.10 cut-off, so calf will be the first to be omitted from the model.

Below, we will continue in this way. After deleting calf , the six predictor model can be fitted. The least important predictor left is pulse . This variable is omitted from the model because the p-value for including it exceeds the 0.10 threshold.

```
> lm.indian2.red <- lm.indian.full;
> lm.indian2.red <- update(lm.indian2.red, ~ . - calf );
> summary(lm.indian2.red)
Call:
lm(formula = sysbp ~ wt + ht + chin + fore + pulse + yrage,
data = indian)
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 106.13739   53.05581   2.000 0.053993 .
wt            1.70900    0.38051   4.491 8.65e-05 ***
ht           -0.04478    0.03871  -1.157 0.256008
chin         -1.14165    0.82823  -1.378 0.177635
fore         -0.56731    1.07462  -0.528 0.601197
pulse         0.07103    0.19142   0.371 0.713018
yrage       -29.54000    7.63983  -3.867 0.000509 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1
```

This procedure is repeated until all predictors remain significant at a 0.10 significance level.

Next page is the final model according to this criterion. Only "wt" and "yrage" are left in the model.

```
Call:
lm(formula = sysbp ~ wt + yrage, data = indian)

Residuals:
     Min       1Q   Median       3Q      Max
-18.4330  -7.3070   0.8963   5.7275  23.9819

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  60.8959    14.2809   4.264 0.000138 ***
wt            1.2169     0.2337   5.207 7.97e-06 ***
yrage       -26.7672     7.2178  -3.708 0.000699 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 9.777 on 36 degrees of freedom
Multiple R-squared:  0.4731,Adjusted R-squared:  0.4438
F-statistic: 16.16 on 2 and 36 DF,  p-value: 9.795e-06
```

Use step function for model selection

```
## step() function specification
## The first two arguments of step(object, scope, ...) are
# object = a fitted model object.
# scope  = a formula giving the terms to be considered for
adding or dropping
## default is AIC
# for BIC, include k = log(nrow( [data.frame name] ))
# test="F" includes additional information
#          for parameter estimate tests that we're
familiar with
```

```
> lm.indian.backward.red.BIC <- step(lm.indian.full
+   , direction = "backward", test = "F",
k = log(nrow(indian)))

Start:  AIC=199.91
sysbp ~ wt + ht + chin + fore + calf + pulse + yrage

        Df Sum of Sq    RSS    AIC F value    Pr(>F)
- calf   1      2.86 3099.3 196.28  0.0287 0.8666427
- pulse  1     14.61 3111.1 196.43  0.1463 0.7046990
- fore   1     27.00 3123.4 196.59  0.2703 0.6068061
- ht     1    131.88 3228.3 197.88  1.3203 0.2593289
- chin   1    186.85 3283.3 198.53  1.8706 0.1812390
<none>               3096.4 199.91
- yrage  1   1386.76 4483.2 210.68 13.8835 0.0007773 ***
- wt     1   1956.49 5052.9 215.35 19.5874 0.0001105 ***
---
```

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

Step:  AIC=196.28
sysbp ~ wt + ht + chin + fore + pulse + yrage

         Df Sum of Sq    RSS    AIC F value    Pr(>F)
- pulse   1     13.34 3112.6 192.79  0.1377 0.7130185
- fore    1     26.99 3126.3 192.96  0.2787 0.6011969
- ht      1    129.56 3228.9 194.22  1.3377 0.2560083
- chin    1    184.03 3283.3 194.87  1.9000 0.1776352
<none>                3099.3 196.28
- yrage   1   1448.00 4547.3 207.57 14.9504 0.0005087 ***
- wt      1   1953.77 5053.1 211.69 20.1724 8.655e-05 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Step:  AIC=192.79
```

```
sysbp ~ wt + ht + chin + fore + yrage

        Df Sum of Sq    RSS    AIC F value    Pr(>F)
- fore   1     17.78 3130.4 189.35  0.1885  0.667013
- ht     1    131.12 3243.8 190.73  1.3902  0.246810
- chin   1    198.30 3310.9 191.53  2.1023  0.156514
<none>               3112.6 192.79
- yrage  1   1450.02 4562.7 204.04 15.3730  0.000421 ***
- wt     1   1983.51 5096.2 208.35 21.0290 6.219e-05 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Step:  AIC=189.35
sysbp ~ wt + ht + chin + yrage

        Df Sum of Sq    RSS    AIC F value    Pr(>F)
- ht     1    113.57 3244.0 187.07  1.2334 0.2745301
```

```
- chin    1     287.20 3417.6 189.11  3.1193 0.0863479 .
<none>                3130.4 189.35
- yrage   1    1445.52 4575.9 200.49 15.7000 0.0003607 ***
- wt      1    2263.64 5394.1 206.90 24.5857 1.945e-05 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Step:  AIC=187.07
sysbp ~ wt + chin + yrage

        Df Sum of Sq    RSS    AIC F value    Pr(>F)
- chin    1     197.37 3441.4 185.71  2.1295 0.1534065
<none>                3244.0 187.07
- yrage   1    1368.44 4612.4 197.14 14.7643 0.0004912 ***
- wt      1    2515.33 5759.3 205.80 27.1384 8.512e-06 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1
```

```
Step:  AIC=185.71
sysbp ~ wt + yrage

        Df Sum of Sq    RSS    AIC F value    Pr(>F)
<none>                3441.4 185.71
- yrage  1    1314.7 4756.1 194.67  13.753 0.0006991 ***
- wt     1    2592.0 6033.4 203.95  27.115 7.966e-06 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1
> summary(lm.indian.backward.red.BIC)

Call:
lm(formula = sysbp ~ wt + yrage, data = indian)

Residuals:
     Min       1Q   Median       3Q      Max
```

```
-18.4330  -7.3070   0.8963   5.7275  23.9819

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  60.8959    14.2809   4.264 0.000138 ***
wt            1.2169     0.2337   5.207 7.97e-06 ***
yrage       -26.7672     7.2178  -3.708 0.000699 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 9.777 on 36 degrees of freedom
Multiple R-squared:  0.4731,Adjusted R-squared:  0.4438
F-statistic: 16.16 on 2 and 36 DF,  p-value: 9.795e-06
```

Comments:

- The backward elimination procedure eliminates five variables from the full model, in the following order: calf skin fold (calf), pulse rate (pulse) , forearm skin fold (fore) , height (ht) , and chin skin fold (chin).
- The model selected by backward elimination includes two predictors: weight (wt) and fraction (yrage).
- As we progress from the full model to the selected model, R^2 decreases as follows: 0.53, 0.53, 0.52, 0.52, 0.50, and 0.47. The decrease is slight across this spectrum of models.
- Using a mechanical approach, we are led to a model with weight and years by age fraction as predictors of systolic blood pressure.
  —— we should closely examine this model.

Forward selection

- Begins with an initial model (could be intercept only) and adds variables to the model one at a time. Importance of the variable is judged by the size t or F statistic.
- $$F_k^* = \frac{MSR(x_k)}{MSE(x_k)}$$

  enter the variable with the largest F_k^* provided this $F_k^* > F - IN$ (predetermined value) or the corresponding P-value is less than a predetermined α
- One variable in the regression equation, say x_h. Compute all two variable regression equation between y and x_h and x_k for $k \neq h$, calculate

  $$F_k^* = \frac{MSR(x_k|x_h)}{MSE(x_k, x_h)},$$

  enter the variable with the largest F_k^* value provided this $F_k^* > F - IN$
- Procedure ends when none of the F statistic is greater than a predetermined level.

```
#forward selection
  # start with an empty model (just the intercept 1)
  lm.indian.empty <- lm(sysbp ~ 1, data = indian)
  # Forward selection, BIC with F-tests
  lm.indian.forward.red.AIC <- step(lm.indian.empty
  , sysbp ~ wt + ht + chin + fore + calf + pulse + yrage
    , direction = "forward", test = "F")

                                        Start:  AIC=201.71
sysbp ~ 1

        Df Sum of Sq    RSS    AIC F value    Pr(>F)
+ wt     1   1775.38 4756.1 191.34 13.8117 0.0006654 ***
+ yrage  1    498.06 6033.4 200.62  3.0544 0.0888139 .
+ fore   1    484.22 6047.2 200.71  2.9627 0.0935587 .
+ calf   1    410.80 6120.6 201.18  2.4833 0.1235725
<none>              6531.4 201.71
```

```
+ ht      1    313.58 6217.9 201.79  1.8660 0.1801796
+ chin    1    189.19 6342.2 202.57  1.1037 0.3002710
+ pulse   1    114.77 6416.7 203.02  0.6618 0.4211339
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Step:  AIC=191.34
sysbp ~ wt

        Df Sum of Sq    RSS    AIC F value    Pr(>F)
+ yrage  1   1314.69 3441.4 180.72 13.7530 0.0006991 ***
<none>               4756.1 191.34
+ chin   1    143.63 4612.4 192.15  1.1210 0.2967490
+ calf   1     16.67 4739.4 193.20  0.1267 0.7240063
+ pulse  1      6.11 4749.9 193.29  0.0463 0.8308792
+ ht     1      2.01 4754.0 193.32  0.0152 0.9024460
+ fore   1      1.16 4754.9 193.33  0.0088 0.9257371
```

```
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Step:  AIC=180.72
sysbp ~ wt + yrage

         Df Sum of Sq    RSS    AIC F value Pr(>F)
+ chin    1   197.372 3244.0 180.42  2.1295 0.1534
<none>                3441.4 180.72
+ fore    1    50.548 3390.8 182.15  0.5218 0.4749
+ calf    1    30.218 3411.1 182.38  0.3101 0.5812
+ ht      1    23.738 3417.6 182.45  0.2431 0.6251
+ pulse   1     5.882 3435.5 182.66  0.0599 0.8081

Step:  AIC=180.42
sysbp ~ wt + yrage + chin
```

```
         Df Sum of Sq    RSS    AIC F value Pr(>F)
<none>                3244.0 180.42
+ ht      1   113.565 3130.4 181.03  1.2334 0.2745
+ pulse   1    11.822 3232.2 182.28  0.1244 0.7265
+ fore    1     0.219 3243.8 182.42  0.0023 0.9620
+ calf    1     0.003 3244.0 182.42  0.0000 0.9959

> summary(lm.indian.forward.red.AIC)

Call:
lm(formula = sysbp ~ wt + yrage + chin, data = indian)

Residuals:
     Min       1Q   Median       3Q      Max
-16.6382  -6.6316   0.4521   6.3593  24.2086

Coefficients:
```

```
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  52.9092    15.0895   3.506 0.001266 **
wt            1.4407     0.2766   5.209 8.51e-06 ***
yrage       -27.3522     7.1185  -3.842 0.000491 ***
chin         -1.0135     0.6945  -1.459 0.153407
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 9.627 on 35 degrees of freedom
Multiple R-squared:  0.5033,Adjusted R-squared:  0.4608
F-statistic: 11.82 on 3 and 35 DF,  p-value: 1.684e-05
```

Stepwise methods

- Alternate between forward selection and backwards elimination
- Arrive at model by dropping a variable, check to see if any variable can be added to the model
- Arrive at a model by adding a variable, check to see if any variable can be dropped
- The value of the F statistic required for dropping a variable is allowed to be different from the value required for adding a variable
- Usually start with an initial model that contains only an intercept
- Stepwise methods gives the same result as forward selection if starting from an initial model; gives the same result as backward elimination if starting from a full model

Stepwise methods:

- Step 1: No variable in the regression equation, compute all one variable regression equation between y and $p-1$ predictors and calculate

$$F_k^* = \frac{MSR(x_k)}{MSE(x_k)}$$

enter the variable with the largest F_k^* provided this $F_k^* > F - IN$ (predetermined value) or the corresponding P-value is less than a predetermined α

- Step 2: 1 variable in the regression equation, say x_{k1}. Compute all two variable regression equation between y and x_{k1} and x_k for $k \neq k_1$, calculate

$$F_k^* = \frac{MSR(x_k|x_{k1})}{MSE(x_k, x_{k1})},$$

enter the variable with the largest F_k^* value provided this $F_k^* > F - IN$ (predetermined value) or the corresponding P-value is less than a predetermined α

- Step 3, two variables in regression equation, say x_{k1} and x_{k2}. Determine if any of the variables previously entered should be removed from the regression equation due to the addition of the latest variable.
  —Calculate

$$F_{k1}^{*} = \frac{MSR(x_{k1}|x_{k2})}{MSE(x_{k1}, x_{k2})}$$

  —If the F_{k1}^{*} falls below a predetermined value called F-out or the corresponding P-value is greater than a predetermined α, then x_{k1} is removed from the model

- Suppose there are $r-1$ variables in the regression equation, compute

$$F_k^* = \frac{MSR(x_k|x_{k1}, x_{k2}, \cdots, x_{k,r-1})}{MSE(x_k, x_{k1}, \cdots, x_{k,r-1})}$$

enter the variable with the largest F_k^* value provided $F_k^* > F - in$

—Suppose x_{kr} is added at the above step, compute

$$F_{ki}^* = \frac{MSR(x_{ki}|x_{k1}, \cdots, x_{kr} \quad \text{except} \quad x_{ki})}{MSE(x_{k1}, x_{k2}, \cdots, x_{kr})},$$

$$\text{for} \quad i = 1, 2, \cdots, r-1,$$

find the smallest F_{ki}^*, If the smallest $F_{ki}^* < F - out$, then remove x_{ki} from the equation.

- Go to next step to try to enter another variable, keep gong until no new variable can be entered.

```
# Stepwise (both) selection, BIC with F-tests, starting with
 intermediate model
 # (this is a purposefully chosen "opposite" model,
 #    from the forward and backward methods this model
 #    includes all the variables dropped and none kept)
 lm.indian.intermediate <- lm(sysbp ~ ht + fore + calf
  + pulse, data = indian)
 # option: trace = 0 does not print each step of the selectio
 lm.indian.both.red.BIC <- step(lm.indian.intermediate
             , sysbp ~ wt + ht + chin + fore + calf
+pulse + yrage
, direction = "both", test = "F", k = log(nrow(indian)),
 trace = 0)
 # the anova object provides a summary of
  the selection steps in order
 lm.indian.both.red.BIC$anova
 summary(lm.indian.both.red.BIC)
```

```
> summary(lm.indian.both.red.BIC)

Call:
lm(formula = sysbp ~ wt + yrage, data = indian)

Residuals:
     Min       1Q   Median       3Q      Max
-18.4330  -7.3070   0.8963   5.7275  23.9819

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  60.8959    14.2809   4.264 0.000138 ***
wt            1.2169     0.2337   5.207 7.97e-06 ***
yrage       -26.7672     7.2178  -3.708 0.000699 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 9.777 on 36 degrees of freedom
```

Model selection and case deletion

- Outliers tend to be cases with large residuals
  —-eliminating the largest residuals obviously makes the SSE and MSE smaller
- Variable selection methods tend to identify as good reduced models those with small MSEs
  —-Delete outliers if they are from recording errors (such as obvious typos), experimental accident (drop the tube) etc,.
  —-Usually after deleting outliers, new data will produce new outliers

Both variable selection and case deletion

- Cause the resulting model to appear better than it probably should
- Tend to give MSEs that are unrealistically small
- Prediction intervals are unrealistically narrow and test statistics are unrealistically large
- Test performed after variable selection or outlier deletion should be viewed as the greatest reasonable evidence against the null hypothesis, with the understanding that more appropriate tests would probably display a lower level of significance.

Model Selection Techniques Only Narrow the Field

Final choice of a model based on:

- p-values, residual plots, other diagnostics
- Parsimony (Occam's Razor): Simple models work best
- The sniff (giggle) test: does the model agree with expectations or theory? Do the signs make sense? Can you explain the results?
- Model validation studies

Model Validation

- The real test of a model or theory: How well does the model predict future observations?
- Problem with your model: the residuals are closer to the observations than they should be! So MSE is too small!!!!
  —-Why? Because picked the model that best predicts your data set. Your measure of predictive ability is biased.
- Optimism Principle: A model chosen by some selection process provides a more optimistic explanation of data used in its derivation than it does of other data that will arise in a similar fashion.

Getting an unbiased view

- Way 1: Collect n^* new observations and compute the mean squared prediction error:

$$\text{MSPR} = \frac{\sum_{i=1}^{n^*}(y_i - \hat{y}_i)^2}{n^*}$$

—y_i is the response variable in the ith validation case
—$\hat{y}_i$ is the predicted value for the ith validation case based on the model building data set
—n^* is the number of cases in the validation data set.

- Way 2: Cross-validation
—Keep n^* cases out of the data set (at random!).
—Base regression on the $n - n^*$ cases in the training set.
—Computer the MSPR for the n^* cases in the validation set (or test set).
—Usually $n^* \approx n/2$.

- Way 3: K-fold cross-validation (sample size n is small)
  — Break data into K roughly equal parts.
  —Of the K subsamples, a single subsample is retained as the validation data for testing the model, and the remaining $K - 1$ subsamples are used as training data.
  —The cross-validation process is then repeated K times (the folds), with each of the K subsamples used exactly once as the validation data.
  —The K results from the folds can then be averaged to produce a single estimation.
  —When $K = n$, the K-fold cross-validation estimate is identical to leave one out cross-validation.

Transformations:

If the residuals show a problem with

- lack of fit (having the wrong model for the mean)
- heteroscedasticity
- nonnormality

Try y transformation or x transformation or both

- y transformation is more common
- only works when $y_{\max}/y_{\min}$ is reasonably large
- choose a transformation to stabilize variance
- log or square transformations can solve many problems

Table: Variance stabilizing transformations

Data	Distribution	Mean, Variance Relationship	Transformation
Count	Poisson	$\mu_h \propto \sigma_h^2$	$\sqrt{y_h}$
Amount	Gamma	$\mu_h \propto \sigma_h$	$\log(y_h)$
Proportion	Binomial/N	$\mu_h(1-\mu_h)/N \propto \sigma_h^2$	$\sin^{-1}(\sqrt{y_h})$

Figure: Circle of Transformations

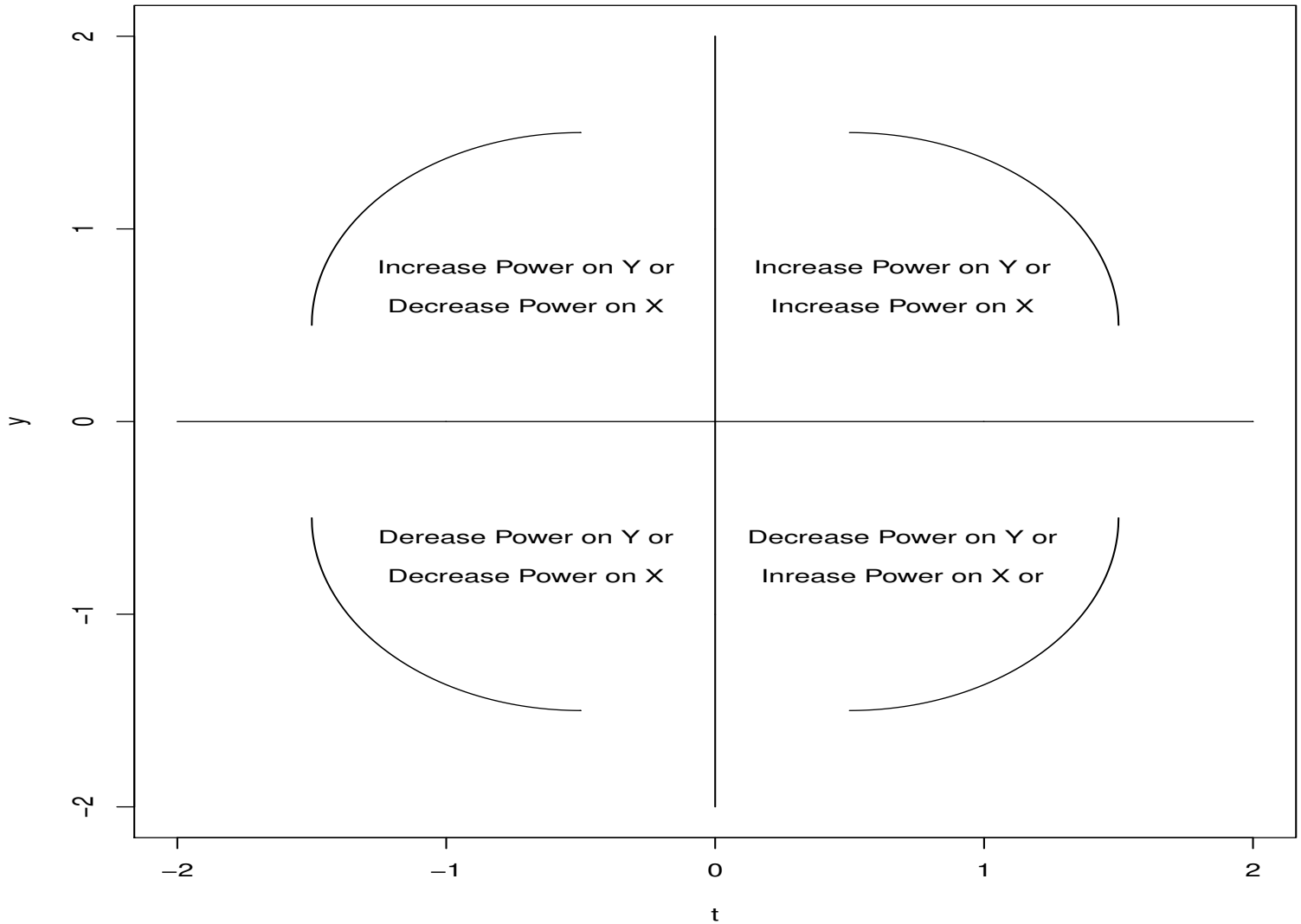


Figure: Curved x, y plot ($y = \cos x$ in the first quadrant. According to figure **??**, need to increase power of both x and y

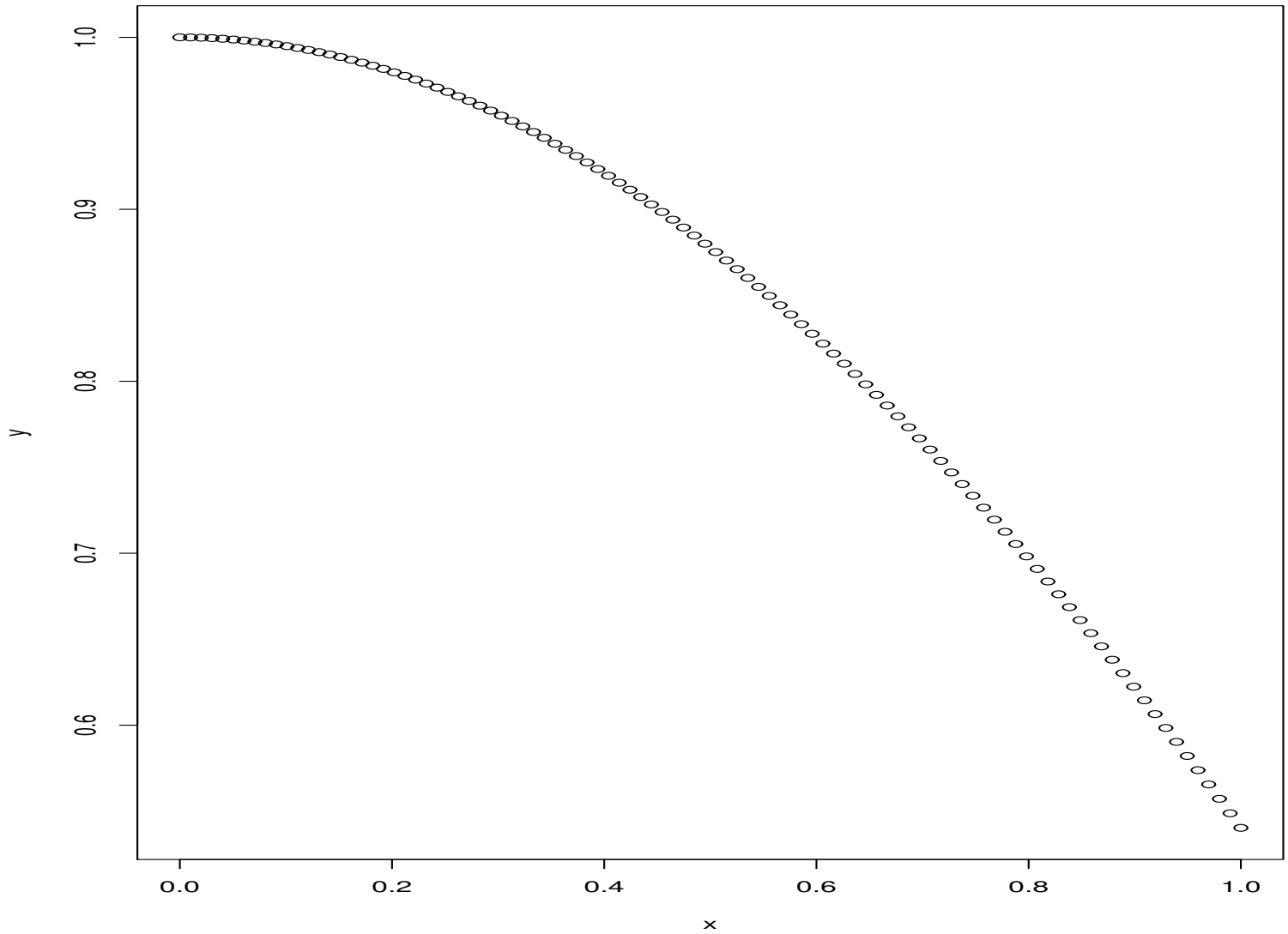
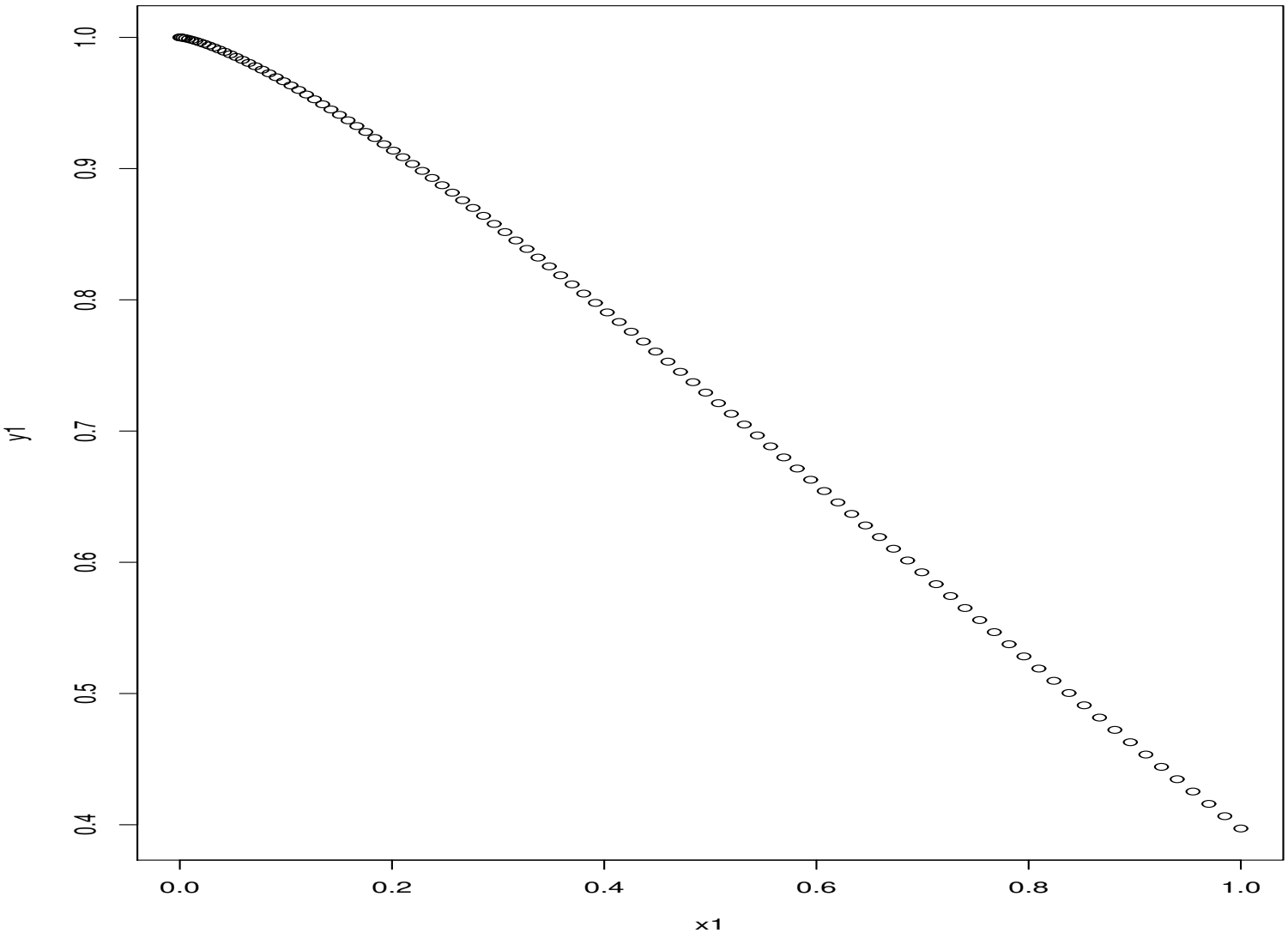


Figure: Plot of $x^{1.5}, y^{1.5}$. $y = \cos x$ in the first quadrant. After transformation $x^* = x^{1.5}$ and $y^* = y^{1.5}$, the curve is much straighter than the one in Figure **??**



Power Transformation:

- If the residuals appear to be normal with constant variance, and the relationship is linear, then go ahead with the regression model. No transformation is needed.
- Transformation is used to deal with model violations. Commonly used transformation is the power transformation (Box-Cox transformation)

$$y^* = \begin{cases} y^\lambda & \lambda \neq 0 \\ \ln y & \text{if} \quad \lambda = 0. \end{cases}$$

$$x^* = \begin{cases} x^\lambda & \lambda \neq 0 \\ \ln x & \text{if} \quad \lambda = 0. \end{cases}$$

- If the residuals appear to be normal with constant variance, but the relationship is non-linear, try transforming the X's to make it a straight line. The transformation on Y may materially change the shape of the distribution of the error terms from the normal distribution and may also lead to substantially differing error term variances.
- If the residuals are not randomly scattered around zero, but have trends. Try transforming Y .
- If you choose a transformation, you need to go back and do all the diagnostics all over again.

- Box and Cox (1964) developed a method to suggest an appropriate transformation of the response variable y, so that, the transformed y is appropriate for the simple linear regression model. The transformation are power transformation. The method selects the λ power to minimize the SSE of the regression

$$y^{\lambda} = \beta_0 + \beta_1 x + \varepsilon$$

and use maximum likelihood to estimate λ. The method runs the regression for a range of transformations between -2 and $+2$, pick the one that minimizes $SSE(\lambda)$. Eventually you would probably suggest the same transformation by eye.

- What transformation to use? Figure **??** gives some general reference.
- Not all scatter plots can be straightened by a power transformation
- Box-Cox suggests a transformation but there is no guarantee it will solve all our problems. We still have to check residuals, assumptions, etc.
- There may be a number of transformations that adequately "straighten" a scatterplot. Pick the transformation that is most interpretable (or the simplest).
- If variable ranges over several orders of magnitude, natural logs transformation usually work; often needed for economic data
- $1/Y$ often makes intuitive sense: If Y is customers per hour, $1/Y$ is hours per customer.
- Square root may make sense if you are measuring areas (square-feet, etc).
- If $Y = 0$ for some observations, cannot do $1/Y$ or $logY$; just add a constant k to all of the Ys first

Example: use boxcox to do transformation

Recall that

```
 lm.sysbp.yrage.wt <- lm(sysbp ~ yrage + wt, data = indian)
library(MASS)
boxcox(lm.sysbp.yrage.wt, lambda = seq(-5, 5, length = 10),
plotit = TRUE)
```

Since $\lambda = 1$ is within the 95% CI of log-likelihood, no need to do transformation

Figure: Box-Cox transformation