All subset regression with leaps, bestglm, glmulti, and meifly

```
## Settings for RMarkdown
http://yihui.name/knitr/options#chunk_options
opts_chunk$set(comment = "", warning = FALSE, message = FALSE,
tidy = FALSE,
    echo = T, fig.width = 5, fig.height = 5)
options(width = 100, scipen = 5, digits = 5)

setwd("~/statistics/Rmedstats/")
```

Summary

- For linear regression, use **leaps**, which allows use of adjusted \(R^2 \) and Mallow Cp.
- For logistic regression, use **glmulti**.
- For Cox regression, use **glmulti**.
- This article about **glmulti** is a good summary of this topic (http://www.jstatsoft.org/v34/i12/paper).

Load and prepare dataset

http://www.umass.edu/statdata/statdata/data/lowbwt.txt

SOURCE: Hosmer and Lemeshow (2000) Applied Logistic Regression: Second Edition. These data are copyrighted by John Wiley & Sons Inc. and must be acknowledged and used accordingly. Data were collected at Baystate Medical Center, Springfield, Massachusetts during 1986.

```
library(gdata)
lbw <-
read.xls("http://www.umass.edu/statdata/statdata/data/lowbwt.xls")

names(lbw) <- tolower(names(lbw))

## Recoding
lbw <- within(lbw, {
    ## race relabeling
    race.cat <- factor(race, levels = 1:3, labels =
c("White","Black","Other"))

    ## ftv (frequency of visit) relabeling
    ftv.cat <- cut(ftv, breaks = c(-Inf, 0, 2, Inf), labels =
c("None","Normal","Many"))
    ftv.cat <- relevel(ftv.cat, ref = "Normal")

    ## ptl
    preterm <- factor(ptl >= 1, levels = c(F,T), labels =
c("0","1+"))
})
```

leaps (regression subset selection)

Regression subset selection including exhaustive search. This is only for linear regression.

Reference: http://www.statmethods.net/stats/regression.html

Perform all subset regression, and choose "nbest" model(s) for each number of predictors up to nvmax.

The result shows how it was performed.

```
library(leaps)
regsubsets.out <-
    regsubsets(bwt ~ age + lwt + race.cat + smoke + preterm +
ht + ui + ftv.cat,
               data = lbw,
               nbest = 1,       # 1 best model for each number
of predictors
               nvmax = NULL,    # NULL for no limit on number
of variables
               force.in = NULL, force.out = NULL,
               method = "exhaustive")
regsubsets.out
```

```
Subset selection object
Call: regsubsets.formula(bwt ~ age + lwt + race.cat + smoke +
preterm +
    ht + ui + ftv.cat, data = lbw, nbest = 1, nvmax = NULL,
force.in = NULL,
    force.out = NULL, method = "exhaustive")
10 Variables  (and intercept)
              Forced in Forced out
age               FALSE      FALSE
lwt               FALSE      FALSE
race.catBlack     FALSE      FALSE
race.catOther     FALSE      FALSE
smoke             FALSE      FALSE
preterm1+         FALSE      FALSE
ht                FALSE      FALSE
ui                FALSE      FALSE
ftv.catNone       FALSE      FALSE
ftv.catMany       FALSE      FALSE
1 subsets of each size up to 10
Selection Algorithm: exhaustive
```

Best model at each variable number

The best model in the 10-variable case includes all variables, as that is the only way to have 10 variables.

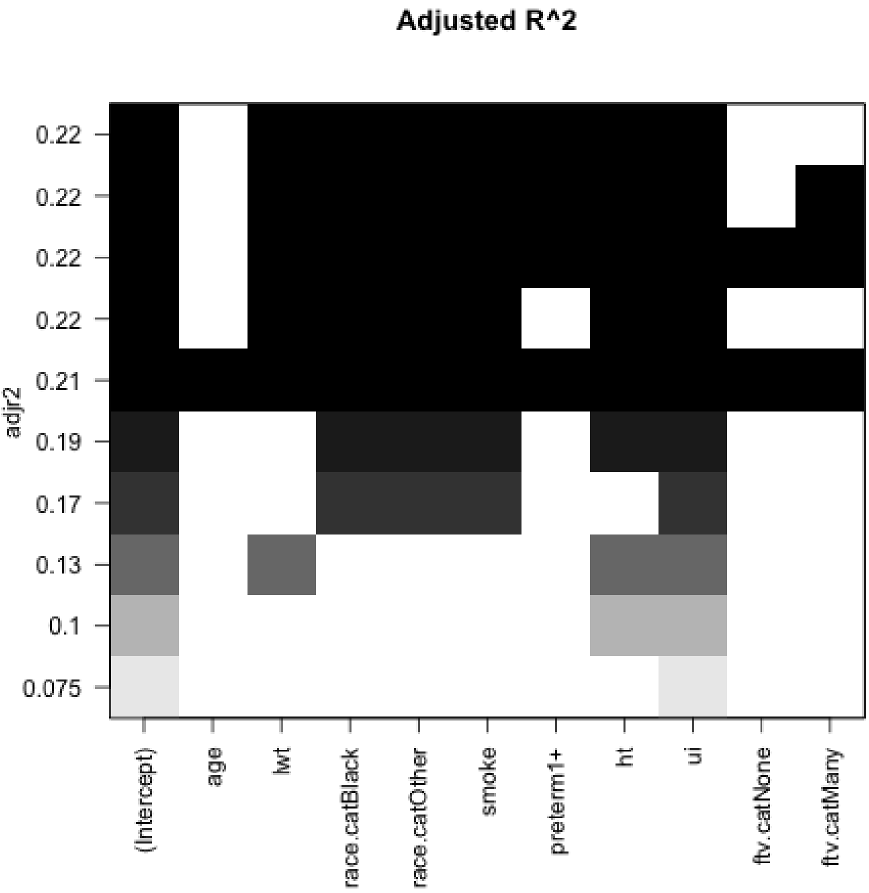
```
summary.out <- summary(regsubsets.out)
as.data.frame(summary.out$outmat)
```

```
          age lwt race.catBlack race.catOther smoke preterm1+
ht ui ftv.catNone ftv.catMany
1  ( 1 )
*
2  ( 1 )
*  *
3  ( 1 )        *
*  *
4  ( 1 )                      *             *     *
*
5  ( 1 )                      *             *     *
*  *
6  ( 1 )        *             *             *     *
*  *
7  ( 1 )        *             *             *     *         *
*  *
8  ( 1 )        *             *             *     *         *
*  *                       *
9  ( 1 )        *             *             *     *         *
*  *           *           *
10  ( 1 )   *   *             *             *     *         *
*  *           *           *
```

Graphical table of best subsets (plot.regsubsets)

By adjusted \(R^2 \), the best model includes lwt, race.cat, preterm, ht, and ui (variables that have black boxes at the higest Y-axis value).

```
## Adjusted R2
plot(regsubsets.out, scale = "adjr2", main = "Adjusted R^2")
```

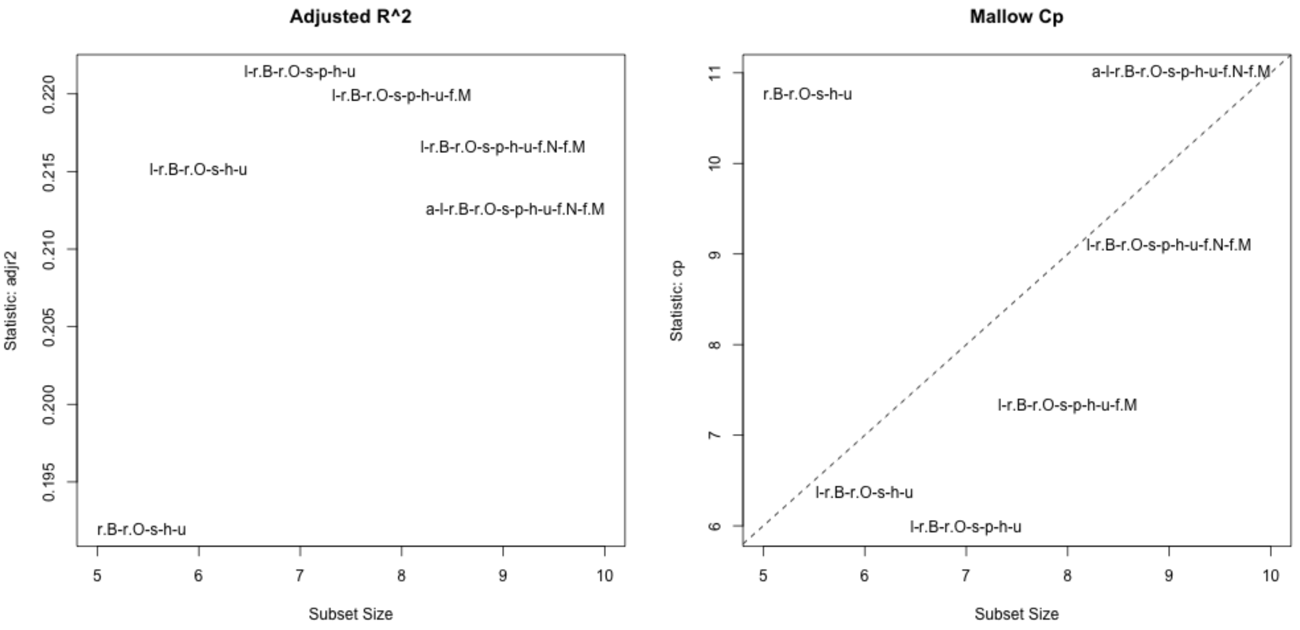


Plot Output from regsubsets Function in leaps package

This is just another way of presenting the same information for adjusted \(R^2 \). The model with 7 variables (counting dummy variables seprately) has the highest adjusted \(R^2 \).

Mallow Cp is used to decide on the number of predictors to include. The stopping rule is to start with the smallest model and gradually increase number of variables, and stop when Mallow Cp is approximately (number of regressors + 1, broken line) for the first time. In this case, the model with 6 regressors is the first one to achieve such a condition.

```
library(car)
layout(matrix(1:2, ncol = 2))
## Adjusted R2
res.legend <-
    subsets(regsubsets.out, statistic="adjr2", legend = FALSE,
min.size = 5, main = "Adjusted R^2")
## Mallow Cp
res.legend <-
    subsets(regsubsets.out, statistic="cp", legend = FALSE,
min.size = 5, main = "Mallow Cp")
abline(a = 1, b = 1, lty = 2)
```



```
res.legend
```

```
               Abbreviation
age                       a
lwt                       l
race.catBlack           r.B
race.catOther           r.O
smoke                     s
preterm1+                 p
ht                        h
ui                        u
ftv.catNone             f.N
ftv.catMany             f.M
```

See which model has the highest adjusted R2

The model with 7 variables (counting dummy variables separately) has the highest adjusted \(R^2 \). Variables marked with TRUE are the ones chosen.

```
which.max(summary.out$adjr2)
```

```
[1] 7
```

```
summary.out$which[7,]
```

```
  (Intercept)           age           lwt race.catBlack
race.catOther         smoke     preterm1+ 
         TRUE         FALSE          TRUE          TRUE
TRUE          TRUE          TRUE 
           ht            ui   ftv.catNone   ftv.catMany 
         TRUE          TRUE         FALSE         FALSE 
```

Do regression with the best model

Somehow I had to recreate the best model from the output above.

```
best.model <- lm(bwt ~ lwt + race.cat + smoke + preterm + ht +
ui, data = lbw)
summary(best.model)
```

```


Call:
lm(formula = bwt ~ lwt + race.cat + smoke + preterm + ht + ui, 
    data = lbw)

Residuals:
    Min      1Q  Median      3Q     Max 
-1886.4  -441.0    53.5   494.2  1620.9 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept)    2871.99     243.67   11.79   <2e-16 ***
lwt               4.04       1.67    2.42   0.0167 *  
race.catBlack  -466.32     145.13   -3.21   0.0016 ** 
race.catOther  -335.68     112.28   -2.99   0.0032 ** 
smoke          -323.57     104.96   -3.08   0.0024 ** 
preterm1+      -208.44     133.50   -1.56   0.1202    
ht             -573.69     198.96   -2.88   0.0044 ** 
ui             -489.96     135.93   -3.60   0.0004 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 643 on 181 degrees of freedom
Multiple R-squared: 0.25,   Adjusted R-squared: 0.221 
F-statistic: 8.64 on 7 and 181 DF,  p-value: 0.00000000396
```

bestglm (Best subset GLM)

Best subset glm using AIC, BIC, EBIC, BICq or Cross-Validation. For the normal case, the 'leaps' is used. Otherwise, a slower exhaustive search. The 'xtable' package is needed for vignette 'SimExperimentBICq.Rnw' accompanying this package.

References:

- http://cran.r-project.org/web/packages/bestglm/
- http://cran.r-project.org/web/packages/bestglm/vignettes/bestglm.pdf

Load bestglm

```
library(bestglm)
```

Reformat data

The outcome variable must be named y, no extraneous variables should be present in the dataset.

```
lbw.for.bestglm <- within(lbw, {
    id   <- NULL        # Delete
    low  <- NULL
    race <- NULL
    ptl  <- NULL
    ftv  <- NULL

    y    <- bwt         # bwt into y
    bwt  <- NULL        # Delete bwt
})

## Reorder variables
lbw.for.bestglm <-
    lbw.for.bestglm[,
c("age","lwt","race.cat","smoke","preterm","ht","ui","ftv.cat","y")]
```

Perform all-subset linear (gaussian) regression based on Akaike Information Criteria (AIC)

```
res.bestglm <-
    bestglm(Xy = lbw.for.bestglm,
            family = gaussian,
            IC = "AIC",                 # Information criteria
for
            method = "exhaustive")
```

```
Morgan-Tatar search since factors present with more than 2
levels.
```

```
## Show top 5 models
res.bestglm$BestModels
```

```
    age  lwt race.cat smoke preterm   ht   ui ftv.cat Criterion
1 FALSE TRUE     TRUE  TRUE    TRUE TRUE TRUE   FALSE    2450.2
2 FALSE TRUE     TRUE  TRUE   FALSE TRUE TRUE   FALSE    2450.7
3  TRUE TRUE     TRUE  TRUE    TRUE TRUE TRUE   FALSE    2452.1
4  TRUE TRUE     TRUE  TRUE   FALSE TRUE TRUE   FALSE    2452.5
5 FALSE TRUE     TRUE  TRUE    TRUE TRUE TRUE    TRUE    2453.3
```

Show result for the best model

The model identical to the one chosen by the adjusted \(R^2 \) was selected.

```
summary(res.bestglm$BestModel)
```

```
Call:
lm(formula = y ~ ., data = data.frame(Xy[, c(bestset[-1],
FALSE),
    drop = FALSE], y = y))

Residuals:
    Min      1Q  Median      3Q     Max
-1886.4  -441.0    53.5   494.2  1620.9

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)    2871.99     243.67   11.79   <2e-16 ***
lwt               4.04       1.67    2.42   0.0167 *
race.catBlack  -466.32     145.13   -3.21   0.0016 **
race.catOther  -335.68     112.28   -2.99   0.0032 **
smoke          -323.57     104.96   -3.08   0.0024 **
preterm1+      -208.44     133.50   -1.56   0.1202
ht             -573.69     198.96   -2.88   0.0044 **
ui             -489.96     135.93   -3.60   0.0004 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 643 on 181 degrees of freedom
Multiple R-squared: 0.25,   Adjusted R-squared: 0.221
F-statistic: 8.64 on 7 and 181 DF,  p-value: 0.00000000396
```

Logistic regression

Do the same, but as a logistic regression model. The resulting model was identical to the best linear model in this case.

```
## Prepare data
lbw.for.best.logistic <- within(lbw, {
    id   <- NULL        # Delete
    bwt  <- NULL
    race <- NULL
    ptl  <- NULL
    ftv  <- NULL

    y    <- low         # bwt into y
    low  <- NULL        # Delete bwt
})

## Reorder variables
lbw.for.best.logistic <-
    lbw.for.best.logistic[,
c("age","lwt","race.cat","smoke","preterm","ht","ui","ftv.cat","y")]


## Perform
res.best.logistic <-
    bestglm(Xy = lbw.for.best.logistic,
            family = binomial,          # binomial family for
logistic
            IC = "AIC",                 # Information criteria
for
            method = "exhaustive")
```

```
Morgan-Tatar search since family is non-gaussian.
Note: factors present with more than 2 levels.
```

```


## Show top 5 models
res.best.logistic$BestModels
```

```
    age  lwt race.cat smoke preterm   ht    ui ftv.cat
Criterion
1 FALSE TRUE     TRUE  TRUE    TRUE TRUE  TRUE   FALSE
211.85
2 FALSE TRUE     TRUE  TRUE    TRUE TRUE FALSE   FALSE
212.48
3  TRUE TRUE     TRUE  TRUE    TRUE TRUE  TRUE   FALSE
212.83
4  TRUE TRUE     TRUE  TRUE    TRUE TRUE FALSE   FALSE
213.15
5 FALSE TRUE     TRUE  TRUE    TRUE TRUE  TRUE    TRUE
214.37
```

```
## Show result for the best model: Same model was chosen
summary(res.best.logistic$BestModel)
```

```

Call:
glm(formula = y ~ ., family = family, data = Xi, weights = 
weights)

Deviance Residuals: 
   Min      1Q  Median      3Q     Max  
-1.731  -0.784  -0.514   0.954   2.198  

Coefficients:
              Estimate Std. Error z value Pr(>|z|)   
(Intercept)   -0.12533    0.96756   -0.13   0.8969   
lwt           -0.01592    0.00695   -2.29   0.0221 * 
race.catBlack  1.30086    0.52848    2.46   0.0138 * 
race.catOther  0.85441    0.44091    1.94   0.0526 . 
smoke          0.86658    0.40447    2.14   0.0322 * 
preterm1+      1.12886    0.45039    2.51   0.0122 * 
ht             1.86690    0.70737    2.64   0.0083 **
ui             0.75065    0.45882    1.64   0.1018   
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 234.67  on 188  degrees of freedom
Residual deviance: 197.85  on 181  degrees of freedom
AIC: 213.9

Number of Fisher Scoring iterations: 4
```

glmulti (Model selection and multimodel inference made easy)

Automated model selection and model-averaging. Provides a wrapper for glm and other functions, automatically generating all possible models (under constraints set by the user) with the specified response and explanatory variables, and finding the best models in terms of some Information Criterion (AIC, AICc or BIC). Can handle very large numbers of candidate models. Features a Genetic Algorithm to find the best models when an exhaustive screening of the candidates is not feasible.

References

- http://www.jstatsoft.org/v34/i12/paper
- http://cran.r-project.org/web/packages/glmulti/index.html

Load package

```
library(glmulti)
```

All-subset linear regression using lm() based on AIC

```
glmulti.lm.out <-
    glmulti(bwt ~ age + lwt + race.cat + smoke + preterm + ht +
ui + ftv.cat, data = lbw,
            level = 1,               # No interaction
considered
            method = "h",            # Exhaustive approach
            crit = "aic",            # AIC as criteria
            confsetsize = 5,         # Keep 5 best models
            plotty = F, report = F,  # No plot or interim
reports
            fitfunction = "lm")      # lm function

## Show 5 best models (Use @ instead of $ for an S4 object)
glmulti.lm.out@formulas
```

```
[[1]]
bwt ~ 1 + race.cat + preterm + lwt + smoke + ht + ui
<environment: 0x11a320bb8>

[[2]]
bwt ~ 1 + race.cat + lwt + smoke + ht + ui
<environment: 0x11a320bb8>

[[3]]
bwt ~ 1 + race.cat + preterm + age + lwt + smoke + ht + ui
<environment: 0x11a320bb8>

[[4]]
bwt ~ 1 + race.cat + age + lwt + smoke + ht + ui
<environment: 0x11a320bb8>

[[5]]
bwt ~ 1 + race.cat + preterm + ftv.cat + lwt + smoke + ht + ui
<environment: 0x11a320bb8>
```

```
## Show result for the best model
summary(glmulti.lm.out@objects[[1]])
```

```
Call:
fitfunc(formula = as.formula(x), data = data)

Residuals:
    Min      1Q  Median      3Q     Max 
-1886.4  -441.0    53.5   494.2  1620.9 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept)    2871.99     243.67   11.79   <2e-16 ***
race.catBlack  -466.32     145.13   -3.21   0.0016 ** 
race.catOther  -335.68     112.28   -2.99   0.0032 ** 
preterm1+      -208.44     133.50   -1.56   0.1202    
lwt               4.04       1.67    2.42   0.0167 *  
smoke          -323.57     104.96   -3.08   0.0024 ** 
ht             -573.69     198.96   -2.88   0.0044 ** 
ui             -489.96     135.93   -3.60   0.0004 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 643 on 181 degrees of freedom
Multiple R-squared: 0.25,   Adjusted R-squared: 0.221 
F-statistic: 8.64 on 7 and 181 DF,  p-value: 0.00000000396
```

All-subset logistic regression using glm() based on AIC

```
glmulti.logistic.out <-
    glmulti(low ~ age + lwt + race.cat + smoke + preterm + ht +
ui + ftv.cat, data = lbw,
            level = 1,               # No interaction
considered
            method = "h",            # Exhaustive approach
            crit = "aic",            # AIC as criteria
            confsetsize = 5,         # Keep 5 best models
            plotty = F, report = F,  # No plot or interim
reports
            fitfunction = "glm",     # glm function
            family = binomial)       # binomial family for
logistic regression

## Show 5 best models (Use @ instead of $ for an S4 object)
glmulti.logistic.out@formulas
```

```
[[1]]
low ~ 1 + race.cat + preterm + lwt + smoke + ht + ui
<environment: 0x11a2588e8>

[[2]]
low ~ 1 + race.cat + preterm + lwt + smoke + ht
<environment: 0x11a2588e8>

[[3]]
low ~ 1 + race.cat + preterm + age + lwt + smoke + ht + ui
<environment: 0x11a2588e8>

[[4]]
low ~ 1 + race.cat + preterm + age + lwt + smoke + ht
<environment: 0x11a2588e8>

[[5]]
low ~ 1 + race.cat + preterm + ftv.cat + lwt + smoke + ht + ui
<environment: 0x11a2588e8>
```

```
## Show result for the best model
summary(glmulti.logistic.out@objects[[1]])
```

```

Call:
fitfunc(formula = as.formula(x), family = ..1, data = data)

Deviance Residuals:
   Min      1Q  Median      3Q     Max
-1.731  -0.784  -0.514   0.954   2.198

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)   -0.12533    0.96756   -0.13   0.8969
race.catBlack  1.30086    0.52848    2.46   0.0138 *
race.catOther  0.85441    0.44091    1.94   0.0526 .
preterm1+      1.12886    0.45039    2.51   0.0122 *
lwt           -0.01592    0.00695   -2.29   0.0221 *
smoke          0.86658    0.40447    2.14   0.0322 *
ht             1.86690    0.70737    2.64   0.0083 **
ui             0.75065    0.45882    1.64   0.1018
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 234.67  on 188  degrees of freedom
Residual deviance: 197.85  on 181  degrees of freedom
AIC: 213.9

Number of Fisher Scoring iterations: 4
```

Load pbc survival data in survival package

```
library(survival)
pbc <- within(pbc, {
    status.dichotomous <- status > 1
    survival.vector    <- Surv(time, status.dichotomous)
})
```

All-subset Cox regression using coxph() based on AIC

```
glmulti.coxph.out <-
    glmulti(survival.vector ~ trt + age + sex + ascites +
hepato + spiders, data = pbc,
            level = 1,               # No interaction
considered
            method = "h",            # Exhaustive approach
            crit = "aic",            # AIC as criteria
            confsetsize = 5,         # Keep 5 best models
            plotty = F, report = F,  # No plot or interim
reports
            fitfunction = "coxph")   # coxph function

## Show 5 best models (Use @ instead of $ for an S4 object)
glmulti.coxph.out@formulas
```

```
[[1]]
survival.vector ~ 1 + age + ascites + hepato + spiders
<environment: 0x104baef88>

[[2]]
survival.vector ~ 1 + trt + age + ascites + hepato + spiders
<environment: 0x104baef88>

[[3]]
survival.vector ~ 1 + sex + age + ascites + hepato + spiders
<environment: 0x104baef88>

[[4]]
survival.vector ~ 1 + sex + trt + age + ascites + hepato +
spiders
<environment: 0x104baef88>

[[5]]
survival.vector ~ 1 + sex + ascites + hepato + spiders
<environment: 0x104baef88>
```

```
## Show result for the best model
summary(glmulti.coxph.out@objects[[1]])
```

```
Call:
fitfunc(formula = as.formula(x), data = data)

  n= 312, number of events= 125 
   (106 observations deleted due to missingness)

           coef exp(coef) se(coef)    z     Pr(>|z|)    
age     0.02620   1.02655  0.00864 3.03      0.00242 ** 
ascites 1.49687   4.46770  0.26109 5.73 0.0000000099 ***
hepato  0.84318   2.32375  0.21198 3.98 0.0000695829 ***
spiders 0.66601   1.94646  0.19530 3.41      0.00065 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

        exp(coef) exp(-coef) lower .95 upper .95
age          1.03      0.974      1.01      1.04
ascites      4.47      0.224      2.68      7.45
hepato       2.32      0.430      1.53      3.52
spiders      1.95      0.514      1.33      2.85

Concordance= 0.771  (se = 0.029 )
Rsquare= 0.27   (max possible= 0.983 )
Likelihood ratio test= 98.3  on 4 df,   p=0
Wald test            = 121  on 4 df,   p=0
Score (logrank) test = 156  on 4 df,   p=0
```

meifly (Interactive model exploration using GGobi)

Exploratory model analysis. Fit and graphical explore ensembles of linear models.

This function just conduct all-subset regression, thus it can handle coxph without problems, but users will have to do model comparison using the result object. Interaction terms cannot be handled, thus inclusion of interaction terms needs creation of product term beforehand.

References:

- http://cran.r-project.org/web/packages/meifly/index.html

Load meifly package

```
library(meifly)
```

Fit all subsets (main effects only)

For x, give a data.frame without the outcome variable.

```
fitall.out <- fitall(y = pbc$survival.vector,
                     x = pbc[,c("trt", "age", "sex",
"ascites","hepato","spiders")],
                     method="coxph")
```

Show the result

As expected, it is the same as the model chosen here (http://rpubs.com/kaz_yos/exhaustive).

```
## Extract AIC from each model
fitall.out.aic <- t(sapply(fitall.out, extractAIC))

## Create an order list of increasing AIC
final.out.order <- order(fitall.out.aic[,2])

## Show the result for the best model
fitall.out[final.out.order][1]
```

```
$`58`
Call:
coxph(formula = y ~ age + ascites + hepato + spiders, data =
data,
    model = FALSE)

          coef exp(coef) se(coef)    z            p
age     0.0262      1.03  0.00864 3.03 0.0024000000
ascites 1.4969      4.47  0.26109 5.73 0.0000000099
hepato  0.8432      2.32  0.21198 3.98 0.0000700000
spiders 0.6660      1.95  0.19530 3.41 0.0006500000

Likelihood ratio test=98.3  on 4 df, p=0  n= 312, number of
events= 125
   (106 observations deleted due to missingness)

attr(,"class")
[1] "ensemble"
```

Exhaustive model selection without using packages

See: http://rpubs.com/kaz_yos/exhaustive