Multiple Regression - Model Selection

Duncan's Occupational Prestige Data

```
head(Duncan, 3)
```

```
##            type income education prestige occupation
## accountant prof     62        86       82 accountant
## pilot      prof     72        76       83      pilot
## architect  prof     75        92       90  architect
```

References:

- Fox, J. and Weisberg, S. (2011) An R Companion to Applied Regression, Second Edition, Sage.
- Duncan, O. D. (1961) A socioeconomic index for all occupations. In Reiss, A. J., Jr. (Ed.) Occupations and Social Status. Free Press [Table VI-1].

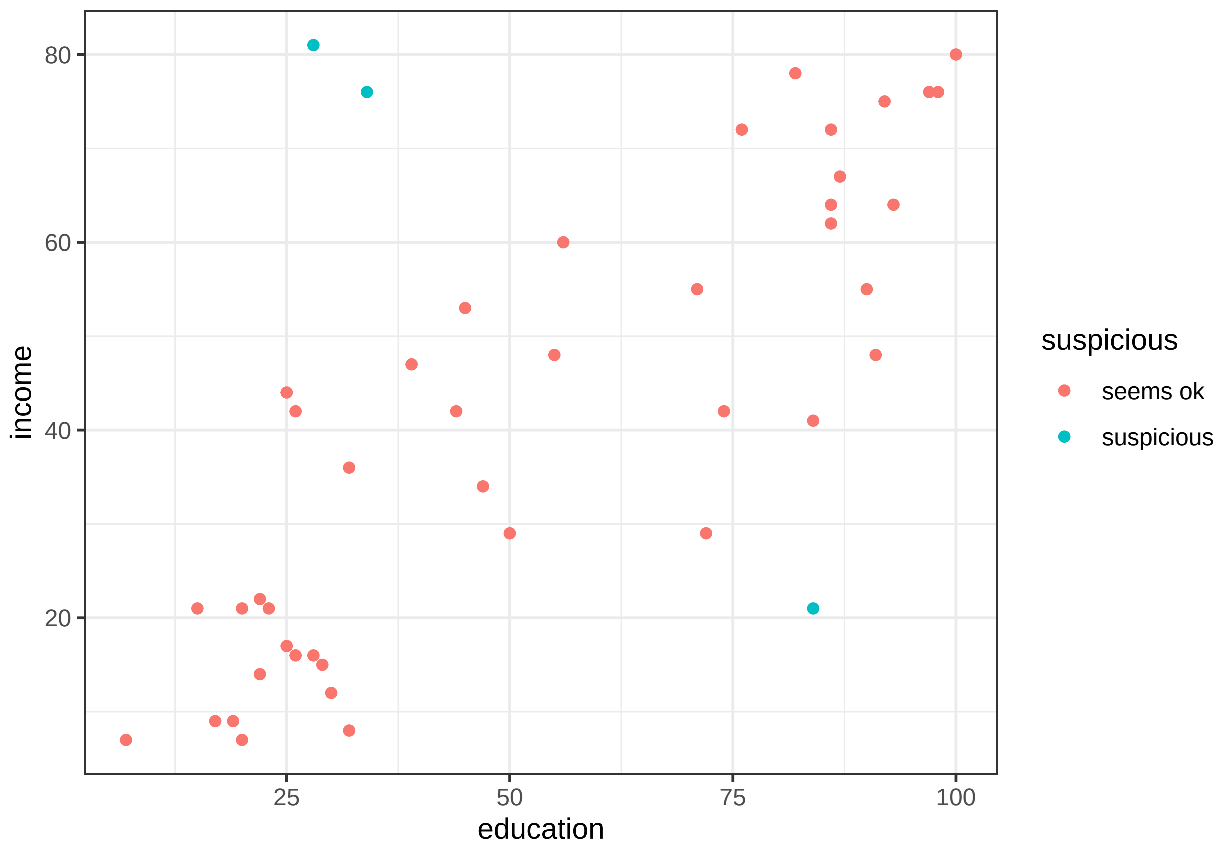
Where we left off:

```
obs_to_investigate <- c(6, 16, 27)
```

```
Duncan[obs_to_investigate, ]
```

```
##             type income education prestige  occupation
## minister    prof     21        84       87    minister
## conductor     wc     76        34       38   conductor
## RR.engineer   bc     81        28       67 RR.engineer
```

```
Duncan <- Duncan %>%
  mutate(
    suspicious = ifelse(row_number() %in% obs_to_investigate, "suspicious", "seems ok")
  )
ggplot(data = Duncan, mapping = aes(x = education, y = income, color = suspicious)) +
  geom_point() +
  theme_bw()
```




```
Duncan_minus_suspicious <- Duncan[-obs_to_investigate, ]
lm_fit_without_suspicious <- lm(prestige ~ income + education + type, data = Duncan_minus_suspicious)
# summary(lm_fit_without_suspicious)
```

```
Duncan_minus_minister <- Duncan[-6, ]
lm_fit_without_minister <- lm(prestige ~ income + education + type, data = Duncan_minus_minister)
# summary(lm_fit_without_minister)
```

Schwarz's Bayesian Information Criterion (BIC)

- Used for model selection
- Takes a measure of lack of fit of a model (here, the residual sum of squares, SSRes) and adds a penalty for the number of terms in the model:

$$BIC = n \times \log\left(\frac{SSRes}{n}\right) + \log(n) \times (p+1)$$

- Subsets that produce smaller BIC values are better; within 2-3 points implies roughly similar performance

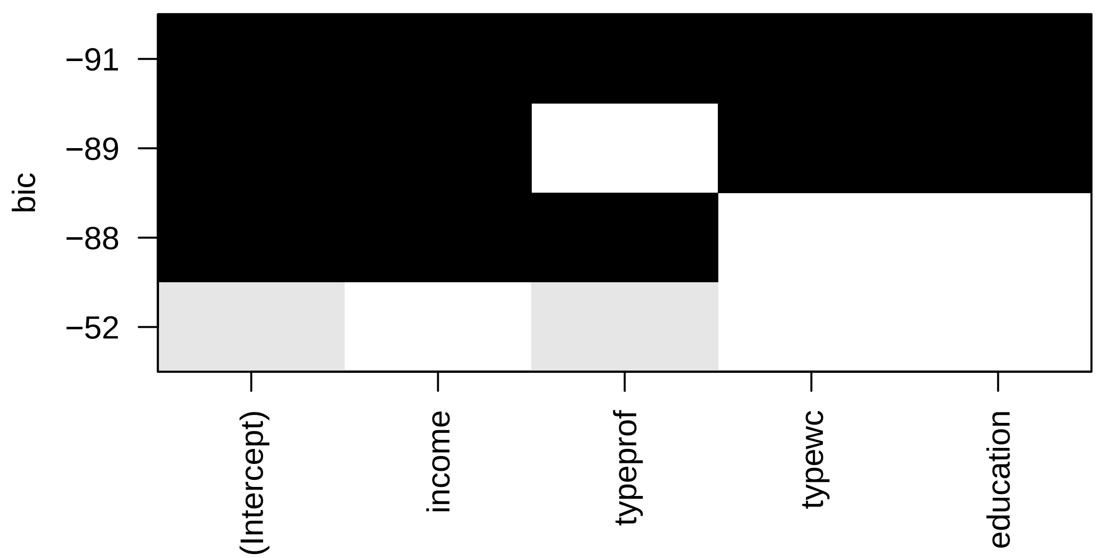
All subsets regression

- Involves fitting all possible subset models and identifying the ones with "best fit" as those that best satisfy some model-fitting criteria (here we are going to use BIC)
- Avoids problems with sequential variable selection techniques (i.e. forward selection, backward elimination, stepwise regression), which tend to select models with too many variables if the set contains unimportant ones

```
library(leaps)
```

```
## Warning: package 'leaps' was built under R version 3.6.3
```

```
candidate_models1 <- regsubsets(prestige ~ income + type + education, data=Duncan)
plot(candidate_models1)
```




```r
summary(candidate_models1)
```

```
## Subset selection object
## Call: regsubsets.formula(prestige ~ income + type + education, data = Duncan)
## 4 Variables  (and intercept)
##           Forced in Forced out
## income        FALSE      FALSE
## typeprof      FALSE      FALSE
## typewc        FALSE      FALSE
## education     FALSE      FALSE
## 1 subsets of each size up to 4
## Selection Algorithm: exhaustive
##          income typeprof typewc education
## 1  ( 1 ) " "    "*"      " "    " "
## 2  ( 1 ) "*"    "*"      " "    " "
## 3  ( 1 ) "*"    " "      "*"    "*"
## 4  ( 1 ) "*"    "*"      "*"    "*"
```

```r
str(summary(candidate_models1))
```

```
## List of 8
##  $ which : logi [1:4, 1:5] TRUE TRUE TRUE TRUE FALSE TRUE ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : chr [1:4] "1" "2" "3" "4"
```

```
##    .. ..$ : chr [1:5] "(Intercept)" "income" "typeprof" "typewc" ...
##  $ rsq    : num [1:4] 0.737 0.891 0.901 0.913
##  $ rss    : num [1:4] 11500 4743 4337 3798
##  $ adjr2 : num [1:4] 0.731 0.886 0.893 0.904
##  $ cp     : num [1:4] 80.12 10.95 8.67 5
##  $ bic    : num [1:4] -52.4 -88.5 -88.7 -90.9
##  $ outmat: chr [1:4, 1:4] " " "*" "*" "*" ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : chr [1:4] "1  ( 1 )" "2  ( 1 )" "3  ( 1 )" "4  ( 1 )"
##   .. ..$ : chr [1:4] "income" "typeprof" "typewc" "education"
##  $ obj    :List of 28
##   ..$ np       : int 5
##   ..$ nrbar    : int 10
##   ..$ d        : num [1:5] 45 10.8 14126.3 2.65 11891.84
##   ..$ rbar     : num [1:10] 0.4 52.556 0.133 41.867 47.963 ...
##   ..$ thetab   : num [1:5] 47.689 54.593 0.487 -6.5 0.598
##   ..$ first    : int 2
##   ..$ last     : int 5
##   ..$ vorder   : int [1:5] 1 3 5 4 2
##   ..$ tol      : num [1:5] 3.35e-09 3.65e-09 3.64e-07 2.16e-09 2.21e-07
##   ..$ rss      : num [1:5] 43688 11500 8156 8044 3798
##   ..$ bound    : num [1:5] 43688 11500 4743 4337 3798
##   ..$ nvmax    : int 5
##   ..$ ress     : num [1:5, 1] 43688 11500 4743 4337 3798
##   ..$ ir       : int 5
##   ..$ nbest    : int 1
##   ..$ lopt     : int [1:15, 1] 1 1 3 1 2 3 1 2 5 4 ...
##   ..$ il       : int 15
##   ..$ ier      : int 0
##   ..$ xnames   : chr [1:5] "(Intercept)" "income" "typeprof" "typewc" ...
##   ..$ method   : chr "exhaustive"
##   ..$ force.in : Named logi [1:5] TRUE FALSE FALSE FALSE FALSE
##   .. ..- attr(*, "names")= chr [1:5] "" "income" "typeprof" "typewc" ...
##   ..$ force.out: Named logi [1:5] FALSE FALSE FALSE FALSE FALSE
##   .. ..- attr(*, "names")= chr [1:5] "" "income" "typeprof" "typewc" ...
##   ..$ sserr    : num 3798
##   ..$ intercept: logi TRUE
##   ..$ lindep   : logi [1:5] FALSE FALSE FALSE FALSE FALSE
##   ..$ nullrss  : num 43688
##   ..$ nn       : int 45
##   ..$ call     : language regsubsets.formula(prestige ~ income + type + education, data = Duncan)
##   ..- attr(*, "class")= chr "regsubsets"
##  - attr(*, "class")= chr "summary.regsubsets"
```

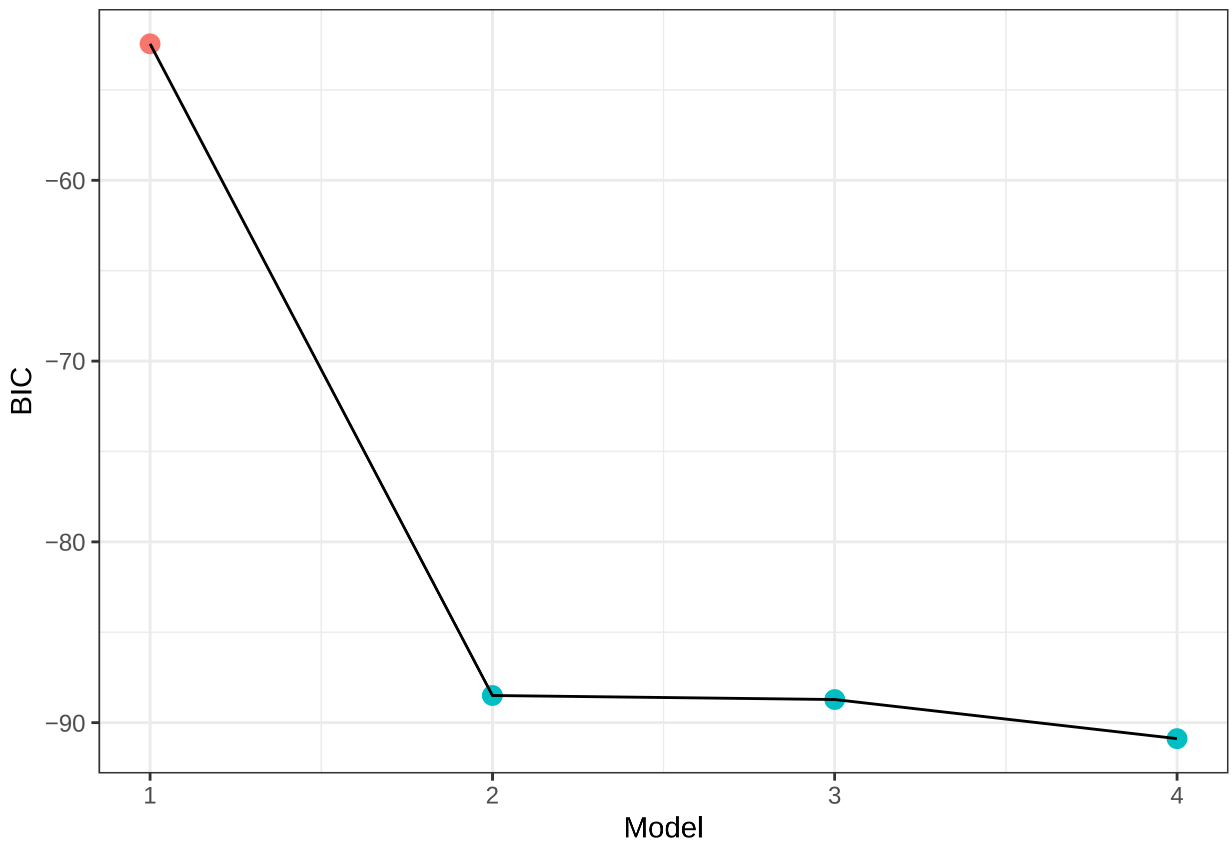
```
summary(candidate_models1)$bic
```

```
## [1] -52.44958 -88.49874 -88.72119 -90.88381
```

```
vis_bic1 <- data.frame(Model=1:4, BIC=summary(candidate_models1)$bic)

ggplot(data=vis_bic1, aes(x=Model, y=BIC)) +
  geom_point(aes(color=BIC < - 88), size=3) +
  geom_line() +
```

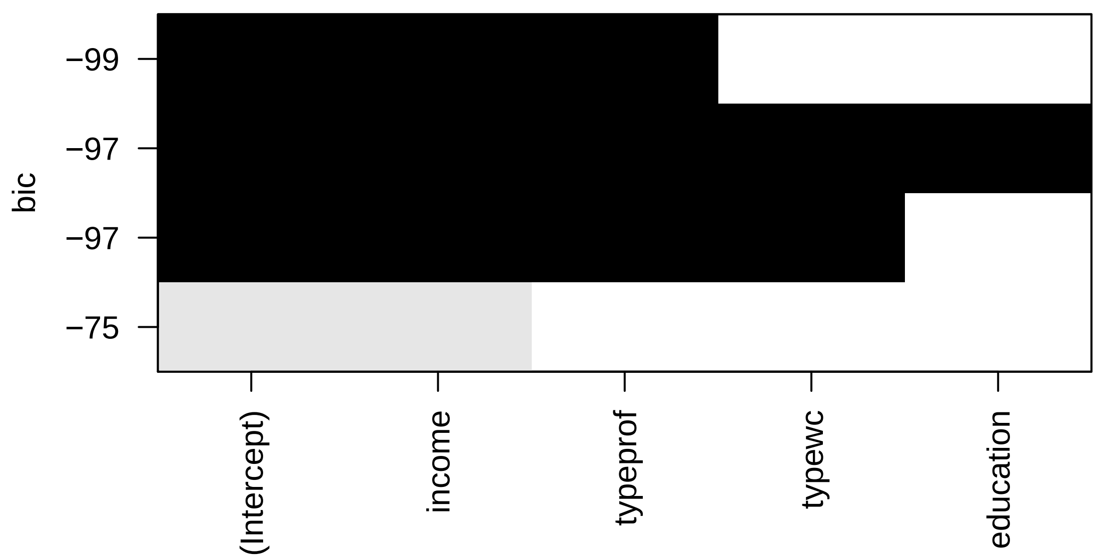
```r
  theme_bw() +
  theme(legend.position = "none")
```



Model 2, Model 3, and Model 4 have roughly similar performance.

- Model 2: income and typeprof (versus type_notprof) (BIC= -88.50)
- Model 3: income, typewc (versus type_notwc), education (BIC= -88.72)
- Model 4: income, typeprof, typewc, education (BIC= -90.88)

```r
candidate_models2 <- regsubsets(prestige ~ income +type + education, data=Duncan_minus_suspicious)
plot(candidate_models2)
```




```r
summary(candidate_models2)
```

```
## Subset selection object
## Call: regsubsets.formula(prestige ~ income + type + education, data = Duncan_minus_suspicious)
## 4 Variables  (and intercept)
##           Forced in Forced out
## income        FALSE      FALSE
## typeprof      FALSE      FALSE
## typewc        FALSE      FALSE
## education     FALSE      FALSE
## 1 subsets of each size up to 4
## Selection Algorithm: exhaustive
##          income typeprof typewc education
## 1  ( 1 ) "*"    " "      " "    " "
## 2  ( 1 ) "*"    "*"      " "    " "
## 3  ( 1 ) "*"    "*"      "*"    " "
## 4  ( 1 ) "*"    "*"      "*"    "*"
```

```r
# str(summary(candidate_models2))

summary(candidate_models2)$bic
```

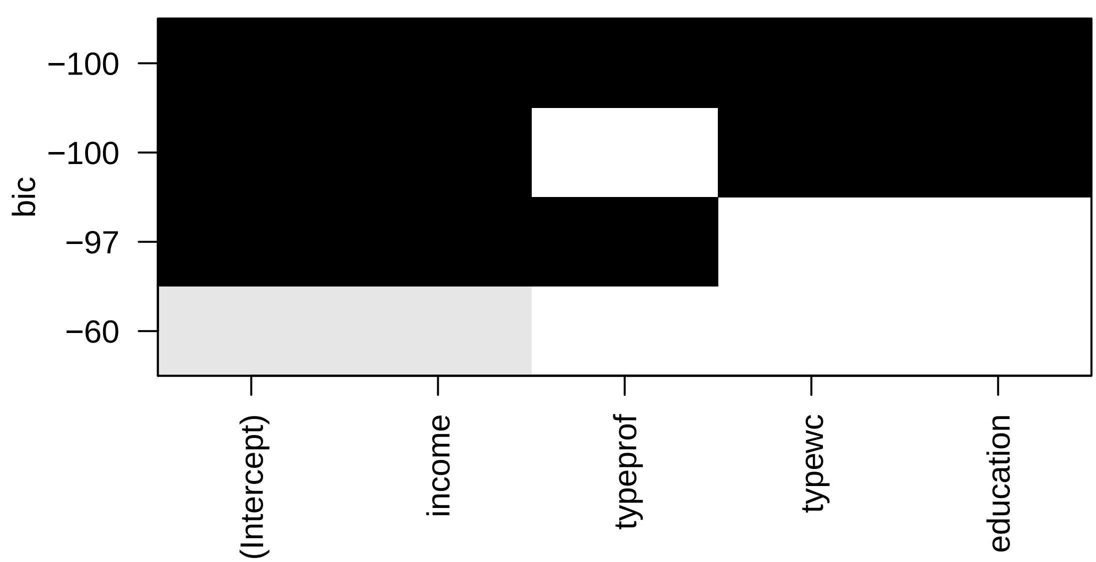
```
## [1] -74.73726 -99.45390 -97.28478 -97.28789
```

```r
vis_bic1 <- data.frame(Model=1:4, BIC=summary(candidate_models2)$bic)
```

Model 2, Model 3, and Model 4 have roughly similar performance.

- Model 2: income and typeprof (versus type_notprof) (BIC= -99.45)
- Model 3: income, typeprof, typewc (BIC=-97.28)
- Model 4: income, typeprof, typewc, education (-97.29)

```r
candidate_models3 <- regsubsets(prestige ~ income +type + education, data=Duncan_minus_minister)
plot(candidate_models3)
```




```r
summary(candidate_models3)
```

```
## Subset selection object
## Call: regsubsets.formula(prestige ~ income + type + education, data = Duncan_minus_minister)
## 4 Variables  (and intercept)
##           Forced in Forced out
## income        FALSE      FALSE
## typeprof      FALSE      FALSE
## typewc        FALSE      FALSE
## education     FALSE      FALSE
## 1 subsets of each size up to 4
```

```
## Selection Algorithm: exhaustive
##          income typeprof typewc education
## 1  ( 1 ) "*"    " "      " "    " "
## 2  ( 1 ) "*"    "*"      " "    " "
## 3  ( 1 ) "*"    " "      "*"    "*"
## 4  ( 1 ) "*"    "*"      "*"    "*"
```

```r
# str(summary(candidate_models3))

summary(candidate_models3)$bic
```

```
## [1]  -60.11700  -97.28833  -99.59875 -100.97645
```

Model 2, Model 3, and Model 4 have roughly similar performance.

- Model 2: income, typeprof (versus type_notprof) (BIC= -97.29)
- Model 3: income, typewc (versus type_notwc), education (BIC= -99.60)
- Model 4: income, typeprof, typewc, education (BIC= -100.98)

Consistency across analyses

```r
Duncan <- Duncan %>%
  mutate(
    type_reduced_prof = ifelse(type %in% c("wc", "bc"), "other", "prof"),
    type_reduced_wc = ifelse(type %in% c("prof", "bc"), "other", "wc")
  )

fit1a <- lm(prestige ~ income + type_reduced_prof, data=Duncan)
summary(fit1a)
```

```
##
## Call:
## lm(formula = prestige ~ income + type_reduced_prof, data = Duncan)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -23.836  -6.374  -0.124   3.769  31.666
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)            6.73045    3.20828   2.098    0.042 *
## income                 0.64294    0.08312   7.735 1.31e-09 ***
## type_reduced_profprof 35.10210    4.09938   8.563 9.31e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10.63 on 42 degrees of freedom
## Multiple R-squared:  0.8914, Adjusted R-squared:  0.8863
## F-statistic: 172.4 on 2 and 42 DF,  p-value: < 2.2e-16
```

```r
fit2a <- lm(prestige ~ income + type_reduced_wc, data=Duncan)
summary(fit2a)
```

```
##
## Call:
## lm(formula = prestige ~ income + type_reduced_wc, data = Duncan)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -32.064  -3.965  -1.032   7.551  59.645
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)         3.6460     4.6243   0.788  0.43486
## income              1.1290     0.0964  11.712 8.18e-15 ***
## type_reduced_wcwc -24.1815     6.8518  -3.529  0.00103 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 15.46 on 42 degrees of freedom
## Multiple R-squared:  0.7701, Adjusted R-squared:  0.7591
## F-statistic: 70.34 on 2 and 42 DF,  p-value: 3.914e-14
```

```r
fit3a <- lm(prestige ~ income + type + education, data=Duncan)
summary(fit3a)
```

```
##
## Call:
## lm(formula = prestige ~ income + type + education, data = Duncan)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -14.890  -5.740  -1.754   5.442  28.972
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -0.18503    3.71377  -0.050  0.96051
## income        0.59755    0.08936   6.687 5.12e-08 ***
## typeprof     16.65751    6.99301   2.382  0.02206 *
## typewc      -14.66113    6.10877  -2.400  0.02114 *
## education     0.34532    0.11361   3.040  0.00416 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.744 on 40 degrees of freedom
## Multiple R-squared:  0.9131, Adjusted R-squared:  0.9044
## F-statistic:   105 on 4 and 40 DF,  p-value: < 2.2e-16
```

```r
Duncan_minus_suspicious <- Duncan_minus_suspicious %>%
  mutate(
    type_reduced_prof = ifelse(type %in% c("wc", "bc"), "other", "prof"),
    type_reduced_wc = ifelse(type %in% c("prof", "bc"), "other", "wc")
```

```r
  )

fit1b <- lm(prestige ~ income + type_reduced_prof, data=Duncan_minus_suspicious)
summary(fit1b)
```

```
##
## Call:
## lm(formula = prestige ~ income + type_reduced_prof, data = Duncan_minus_suspicious)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -18.2775  -6.1224  -0.0996   4.6799  24.9029
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)            2.41453    2.89732   0.833     0.41
## income                 0.82451    0.08937   9.226 2.38e-11 ***
## type_reduced_profprof 26.23336    4.26755   6.147 3.23e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.749 on 39 degrees of freedom
## Multiple R-squared:  0.9283, Adjusted R-squared:  0.9246
## F-statistic: 252.4 on 2 and 39 DF,  p-value: < 2.2e-16
```

```r
fit2b <- lm(prestige ~ income + type_reduced_wc, data=Duncan_minus_suspicious)
summary(fit2b)
```

```
##
## Call:
## lm(formula = prestige ~ income + type_reduced_wc, data = Duncan_minus_suspicious)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -31.490  -5.688   0.611   7.437  23.594
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)         -2.62792    3.28478  -0.800 0.428539
## income               1.26470    0.06979  18.123  < 2e-16 ***
## type_reduced_wcwc -18.64260    5.05091  -3.691 0.000682 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10.57 on 39 degrees of freedom
## Multiple R-squared:  0.8953, Adjusted R-squared:   0.89
## F-statistic: 166.8 on 2 and 39 DF,  p-value: < 2.2e-16
```

```r
fit3b <- lm(prestige ~ income + type + education, data=Duncan_minus_suspicious)
summary(fit3b)
```

```
##
```

```
## Call:
## lm(formula = prestige ~ income + type + education, data = Duncan_minus_suspicious)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -18.0415  -5.3802  -0.6189   5.0992  23.2906
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -1.1053     3.2745  -0.338   0.7376
## income        0.7733     0.1171   6.607 9.53e-08 ***
## typeprof     15.2512     6.4123   2.378   0.0227 *
## typewc      -12.3622     5.9478  -2.078   0.0447 *
## education     0.2180     0.1174   1.857   0.0714 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.432 on 37 degrees of freedom
## Multiple R-squared:  0.9368, Adjusted R-squared:   0.93
## F-statistic: 137.1 on 4 and 37 DF,  p-value: < 2.2e-16
```

```r
Duncan_minus_minister <- Duncan_minus_minister %>%
  mutate(
    type_reduced_prof = ifelse(type %in% c("wc", "bc"), "other", "prof"),
    type_reduced_wc = ifelse(type %in% c("prof", "bc"), "other", "wc")
  )

fit1c <- lm(prestige ~ income + type_reduced_prof, data=Duncan_minus_minister)
summary(fit1c)
```

```
##
## Call:
## lm(formula = prestige ~ income + type_reduced_prof, data = Duncan_minus_minister)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -21.7057  -5.4561   0.2744   4.2892  26.5674
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)            4.08674    2.90531   1.407    0.167
## income                 0.73183    0.07683   9.526 5.99e-12 ***
## type_reduced_profprof 30.34045    3.82257   7.937 8.10e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.324 on 41 degrees of freedom
## Multiple R-squared:  0.9153, Adjusted R-squared:  0.9112
## F-statistic: 221.6 on 2 and 41 DF,  p-value: < 2.2e-16
```

```r
fit2c <- lm(prestige ~ income + type_reduced_wc, data=Duncan_minus_minister)
summary(fit2c)
```

```
##
## Call:
## lm(formula = prestige ~ income + type_reduced_wc, data = Duncan_minus_minister)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -30.5399  -4.3093   0.0511   7.6437  27.8159
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)         0.14288    3.78568   0.038 0.970077
## income              1.17612    0.07809  15.061  < 2e-16 ***
## type_reduced_wcwc -23.06628    5.51298  -4.184 0.000147 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.43 on 41 degrees of freedom
## Multiple R-squared:  0.8495, Adjusted R-squared:  0.8422
## F-statistic: 115.7 on 2 and 41 DF,  p-value: < 2.2e-16
```

```r
fit3c <- lm(prestige ~ income + type + education, data=Duncan_minus_minister)
summary(fit3c)
```

```
##
## Call:
## lm(formula = prestige ~ income + type + education, data = Duncan_minus_minister)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -17.0521  -6.4105  -0.7819   4.6552  23.5212
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -1.62984    3.22841  -0.505  0.61651
## income        0.71813    0.08332   8.619 1.44e-10 ***
## typeprof     13.43111    6.09592   2.203  0.03355 *
## typewc      -15.87744    5.28357  -3.005  0.00462 **
## education     0.28924    0.09917   2.917  0.00584 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.413 on 39 degrees of freedom
## Multiple R-squared:  0.9344, Adjusted R-squared:  0.9277
## F-statistic:   139 on 4 and 39 DF,  p-value: < 2.2e-16
```

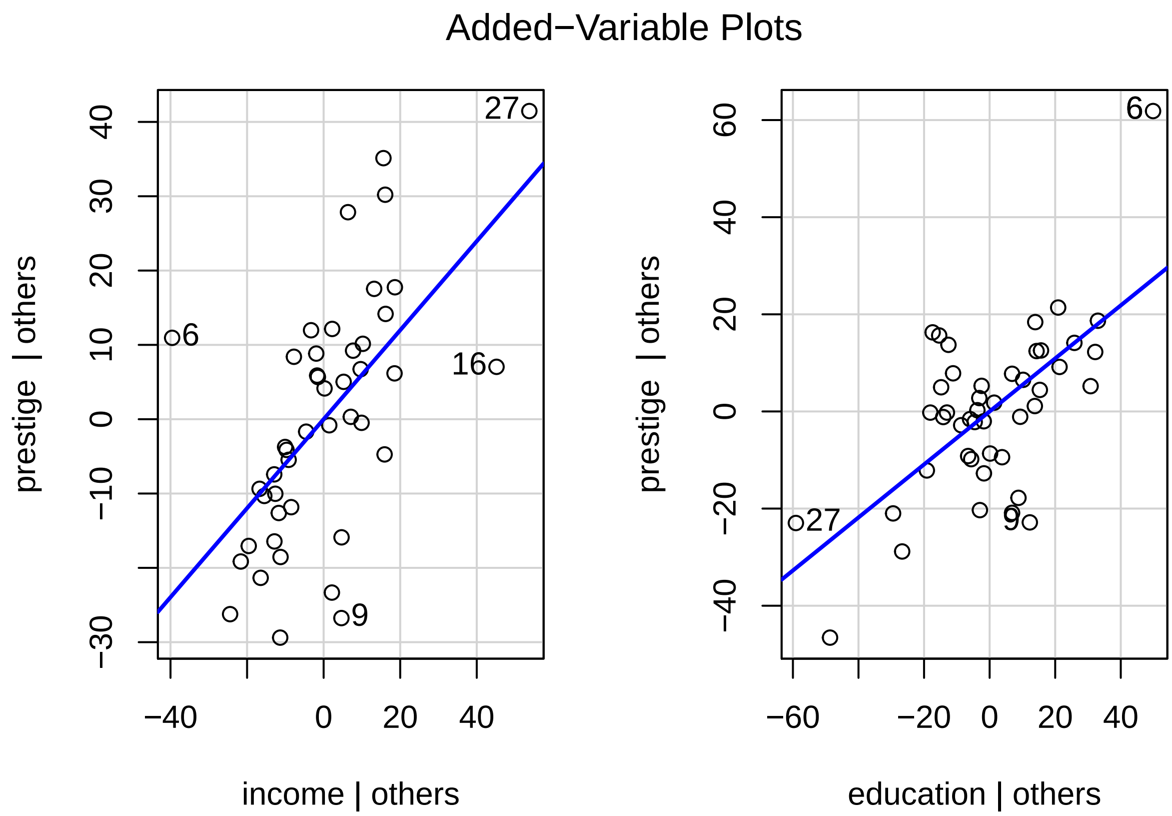
Added variable plots

- Also called partial regression plots
- Used to examine the effect of adding another explanatory variable to a model that already has one or more explanatory variables
- Strong linear relationship indicates that adding variable will likely be of value
- Can be used to identify influential points (6, 16, and 17 were already identified through other diagnostics)

Generating with avPlots (from R package car)

```
fit_prestige <- lm(prestige ~ income + education, data=Duncan)

avPlots(fit_prestige)
```



Generating by hand

```
fit_prestige1 <- lm(prestige ~ education, data=Duncan)
fit_inc <- lm(education ~ income, data=Duncan)

fit_prestige2 <- lm(prestige ~ income, data=Duncan)
```

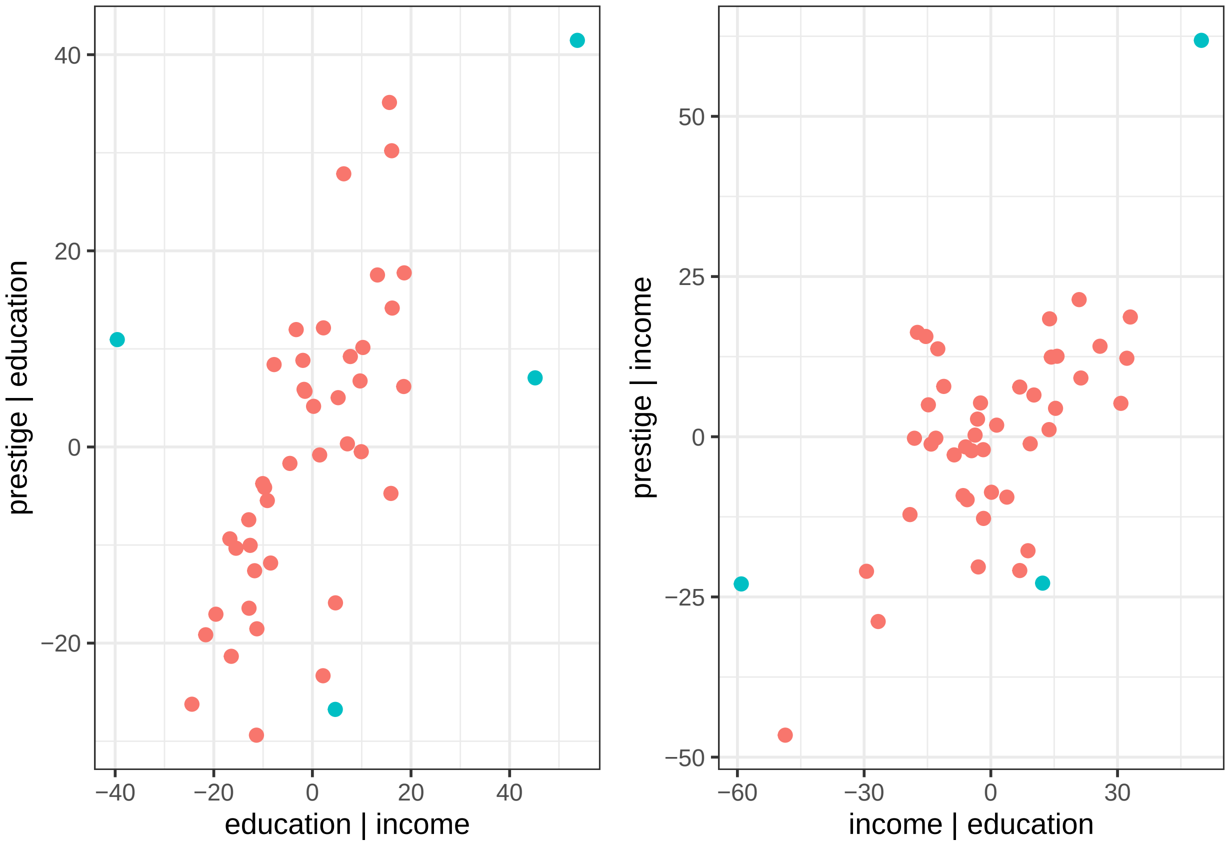
```r
fit_edu <- lm(income ~ education, data=Duncan)

Duncan <- Duncan %>%
  mutate(
    resid_edu = residuals(fit_inc),
    resid_inc = residuals(fit_edu),
    resid_prestige1 = residuals(fit_prestige1),
    resid_prestige2 = residuals(fit_prestige2),
    id = 1:nrow(Duncan)
  )

p1 <- ggplot(data=Duncan, aes(x=resid_inc, y=resid_prestige1)) +
        geom_point(aes(color=id %in% c(6, 9, 16, 27)), size=2) +
        theme_bw() +
        theme(legend.position="none") +
        ylab("prestige | education") +
        xlab("education | income")

p2 <- ggplot(data=Duncan, aes(x=resid_edu, y=resid_prestige2)) +
        geom_point(aes(color=id %in% c(6, 9, 27)), size=2) +
        theme_bw() +
        theme(legend.position="none") +
        ylab("prestige | income") +
        xlab("income | education")

grid.arrange(p1, p2, nrow=1)
```



prestige | education
education | income
40
20
0
−20
−40
−20
0
20
40
prestige | income
income | education
50
25
0
−25
−50
−60
−30
0
30