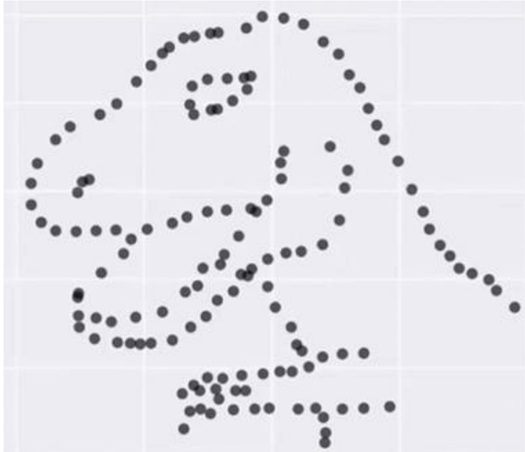
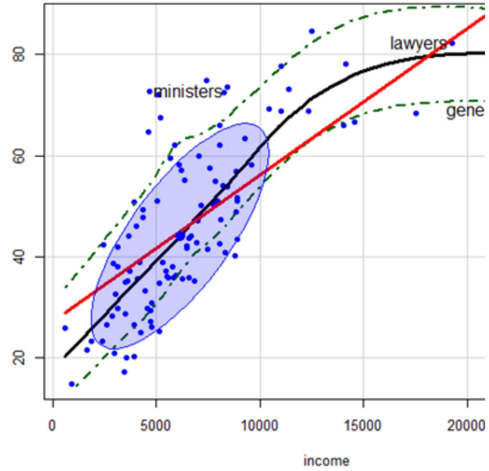
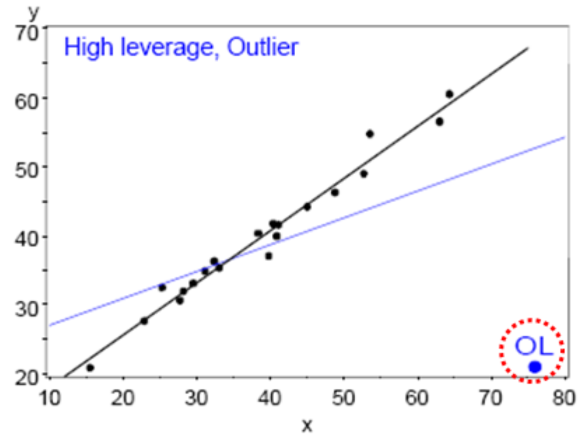




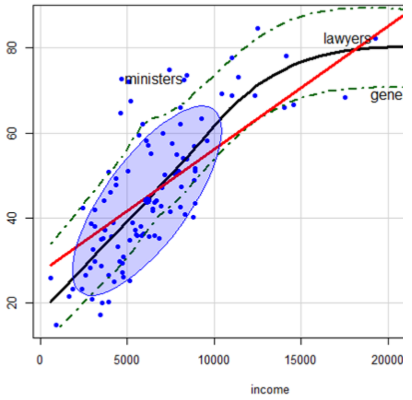


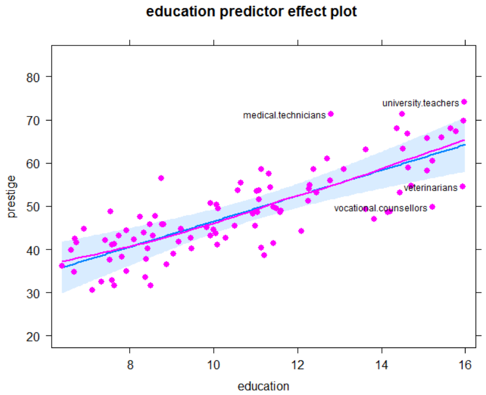
Visualizing Linear Models: An R Bag of Tricks Session 1: Getting Started

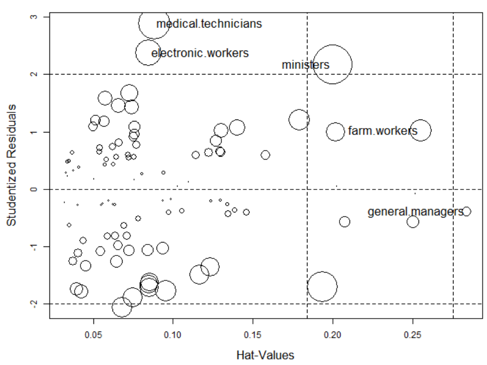
Michael Friendly
SCS Short Course
Oct-Nov, 2022
https://friendly.github.io/VisMLM-course/

Today's topics

- What you need for this course
- Why plot your data?
- Linear models review
- Data plots: really see your data
- Model (effect) plots: see net effects of predictors
- Diagnostic plots: what's wrong with my model?







What you need

- R, version >=3.6 [R 4.1 is current]
  - Download from https://cran.r-project.org/
- RStudio IDE, highly recommended
  - https://www.rstudio.com/products/rstudio/
- R packages: see course web page
  - car
  - effects
  - heplots
  - candisc
  - visreg

R script to install packages:
https://friendly.github.io/VisMLM-course/R/install-vismlm-pkgs.R

Why plot your data?

Getting information from a table is like extracting sunlight from a cucumber. --- Farquhar & Farquhar, 1891

Information that is imperfectly acquired, is generally as imperfectly retained; and a man who has carefully investigated a printed table, finds, when done, that he has only a very faint and partial idea of what he has read; and that like a figure imprinted on sand, is soon totally erased and defaced.
--- William Playfair, *The Commercial and Political Atlas* (p. 3), 1786

Cucumbers

Table 7
Stevens et al. 2006, table 2: Determinants of authoritarian aggression

Variable	Coefficient (Standard Error)
Constant	.41 (.93)
Countries	
Argentina	1.31 (.33)**B,M
Chile	.93 (.32)**B,M
Colombia	1.46 (.32)**B,M
Mexico	.07 (.32)A,CH,CO,V
Venezuela	.96 (.37)**B,M
Threat	
Retrospective egocentric economic perceptions	.20 (.13)
Prospective egocentric economic perceptions	.22 (.12)#
Retrospective sociotropic economic perceptions	−.21 (.12)#
Prospective sociotropic economic perceptions	−.32 (.12)*
Ideological distance from president	−.27 (.07)**
Ideology	
Ideology	.23 (.07)**
Individual Differences	
Age	.00 (.01)
Female	−.03 (.21)
Education	.13 (.14)
Academic Sector	.15 (.29)
Business Sector	.31 (.25)
Government Sector	−.10 (.27)
R^2	.15
Adjusted R^2	.12
N	500

Results of a one model for authoritarian aggression

The information is overwhelmed by footnotes & significance **stars**

**p < .01, *p < .05, #p < .10 (twotailed)

[A]Coefficient is significantly different from Argentina's at p < .05;

[B]Coefficient is significantly different from Brazil's at p < .05;

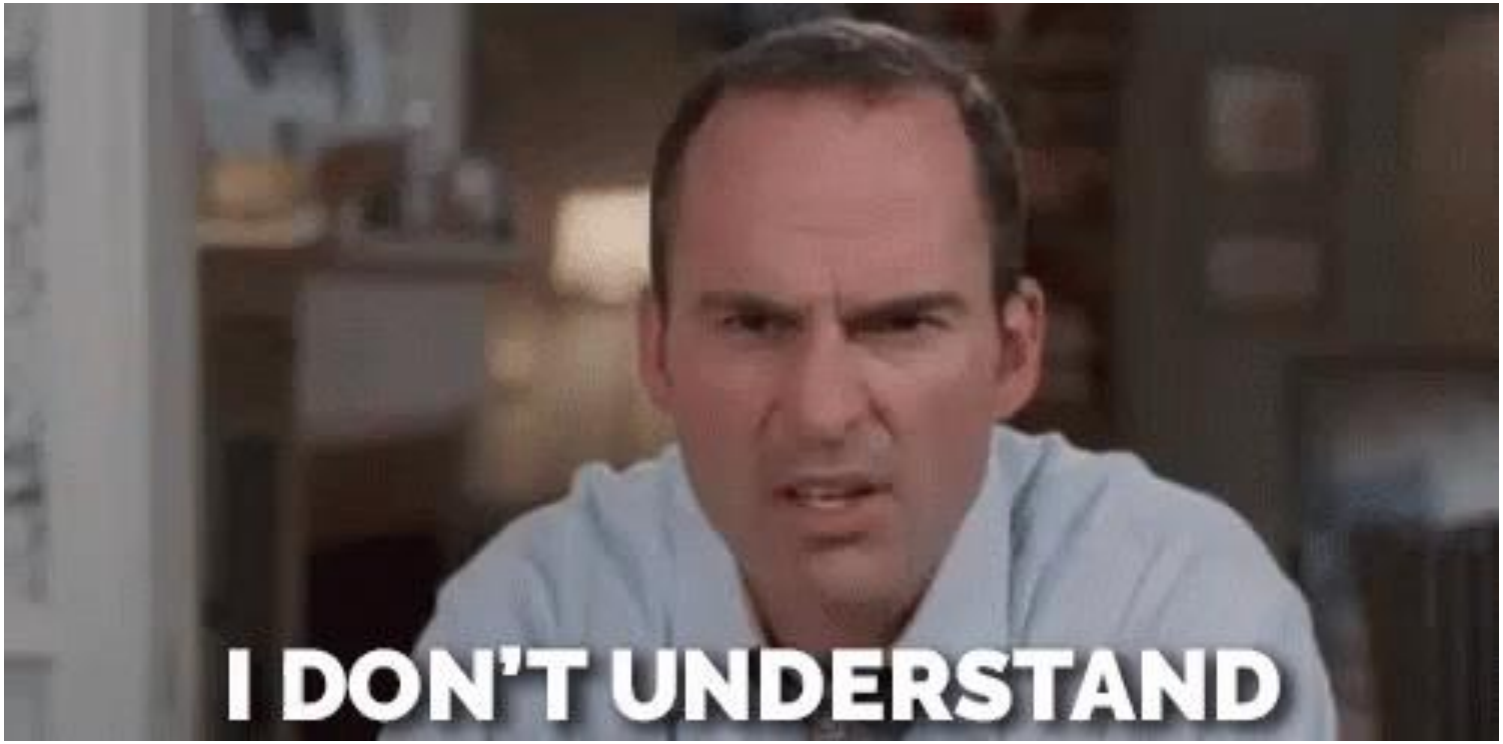
[CH]Coefficient is significantly different from Chile's at p < .05;

[CO]Coefficient is significantly different from Colombia's at p < .05;

[M]Coefficient is significantly different from Mexico's at p < .05;

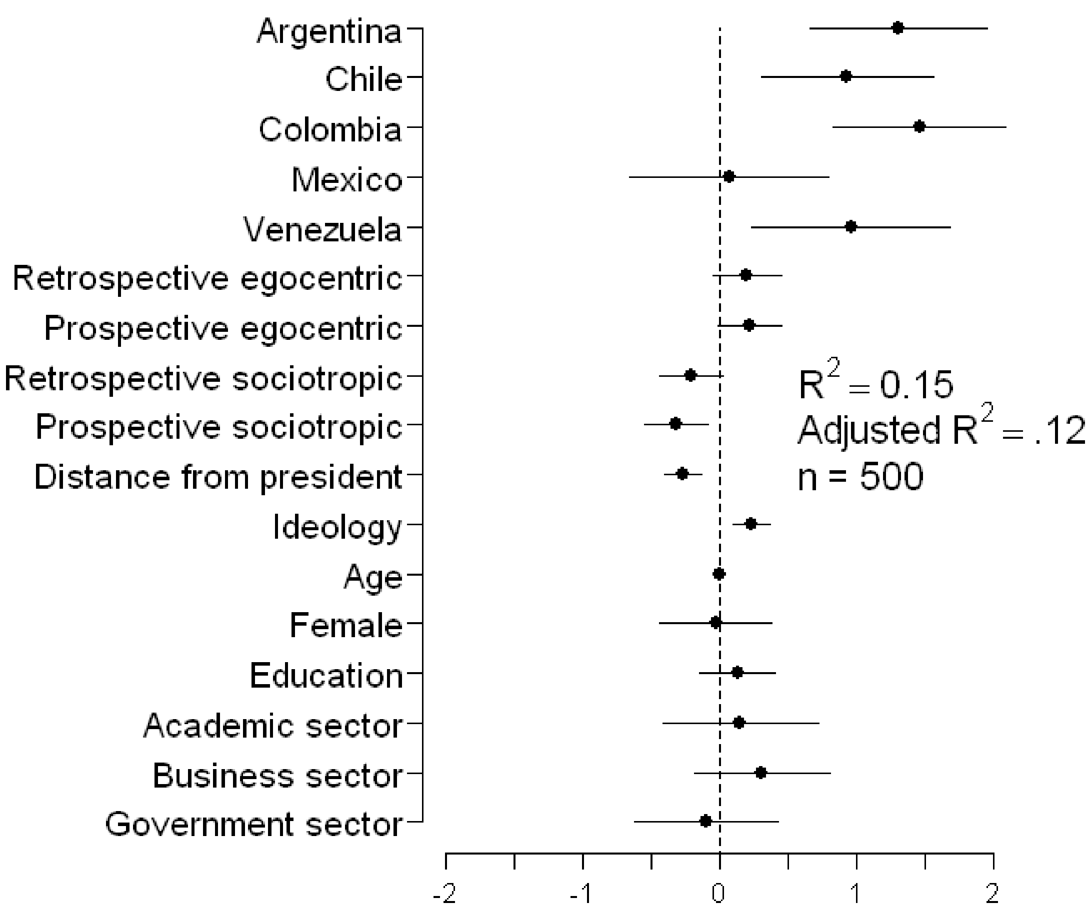
[V]Coefficient is significantly different from Venezuela's at p < .05.

What's wrong with this picture?



Sunlight

coefplot(model)



Why didn't they say this in the first place?

NB: This is a presentation graph equivalent of the table

Shows standardized coefficient with 95% CI

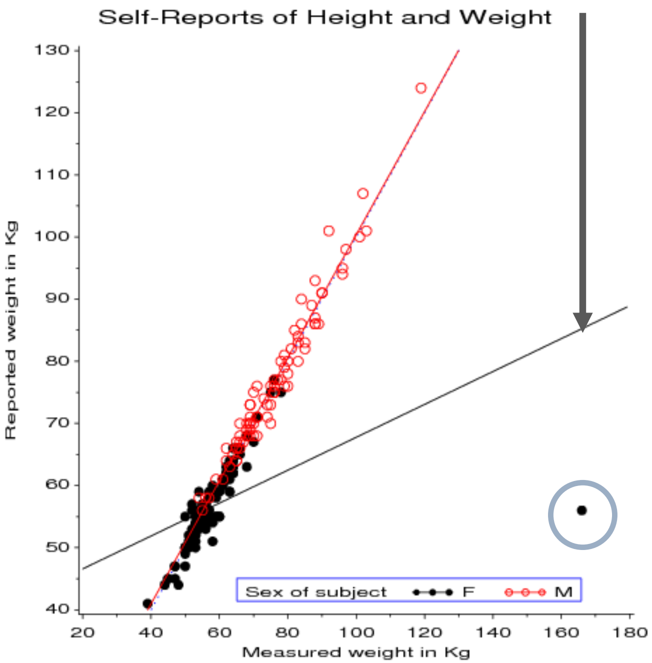
Factors (Country, sector) are shown relative to the baseline category

Run, don’t walk toward the sunlight

Graphs can give enlightenment

The greatest value of a picture is when it forces us to notice what we never expected to see.
-- John W. Tukey

Effect of one rotten point on regression



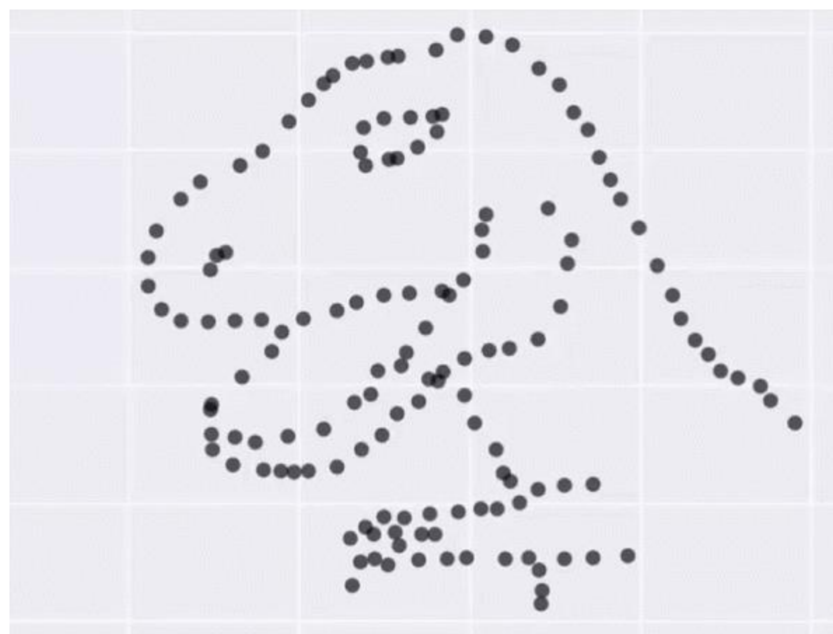
Dangers of numbers-only output

Student: You said to run descriptives and compute the correlation. What next?

Consultant: Did you plot your data?

```
X Mean: 54.26
Y Mean: 47.83
X SD  : 16.76
Y SD  : 26.93
Corr. : -0.06
```

With exactly the same stats, the data could be *any* of these plots



See how this in done in R: https://cran.r-project.org/web/packages/datasauRus/

Sometimes, don't need numbers at all

COVID transmission risk ~ Occupancy * Ventilation * Activity * Mask? * Contact.time

A complex 5-way table, whose message is clearly shown w/o numbers

A semi-graphic table shows the patterns in the data

There are 1+ unusual cells here. Can you see them?

From: N.R. Jones et-al (2020). Two metres or one: what is the evidence for physical distancing in covid-19? *BMJ* 2020;370:m3223, *doi:* https://doi.org/10.1136/bmj.m3223

If you do need tables– make them pretty

Several R packages make it easier to construct informative & pretty semi-graphic tables

Presentation graph

Perhaps too cute!

Distribution of variables shown

Flipper lengths (mm) of the famous penguins of Palmer Station, Antarctica.

		Female		Male	
Species	Distribution	Avg.	Std. Dev.	Avg.	Std. Dev.
		188	5.6	192	6.6
		192	5.8	200	6.0
		213	3.9	222	5.7

Artwork by @allison_horst

produced using modelsummary::datasummary, https://vincentarelbundock.github.io/modelsummary/articles/datasummary.html

Visual table ideas: Heatmap shading

Heatmap shading: Shade the background of each cell according to some criterion

The trends in the US and Canada are made obvious

NB: Table rows are sorted by Jan. value, lending coherence

Background shading ~ value:
US & Canada are made to stand out.

Tech note: use white text on a darker background

Unemployment rate in selected countries

January-August 2020, sorted by the unemployment rate in January.

country	Jan ▲	Feb	Mar	Apr	May	Jun	Jul	Aug
Japan	2.4%	2.4%	2.5%	2.6%	2.9%	2.8%	2.9%	3.0%
Netherlands	3.0%	2.9%	2.9%	3.4%	3.6%	4.3%	4.5%	4.6%
Germany	3.4%	3.6%	3.8%	4.0%	4.2%	4.3%	4.4%	4.4%
Mexico	3.6%	3.6%	3.2%	4.8%	4.3%	5.4%	5.2%	5.0%
US	**3.6%**	**3.5%**	**4.4%**	**14.7%**	**13.3%**	**11.1%**	**10.2%**	**8.4%**
South Korea	4.0%	3.3%	3.8%	3.8%	4.5%	4.3%	4.2%	3.2%
Denmark	4.9%	4.9%	4.8%	4.9%	5.5%	6.0%	6.3%	6.1%
Belgium	5.1%	5.0%	5.0%	5.1%	5.0%	5.0%	5.0%	5.1%
Australia	5.3%	5.1%	5.2%	6.4%	7.1%	7.4%	7.5%	6.8%
Canada	**5.5%**	**5.6%**	**7.8%**	**13.0%**	**13.7%**	**12.3%**	**10.9%**	**10.2%**
Finland	6.8%	6.9%	7.0%	7.3%	7.5%	7.8%	8.0%	8.1%

Source: OECD • Get the data • Created with Datawrapper

Visual table ideas: Sparklines

Sparklines: Mini graphics inserted into table cells or text

From: https://www.pluralsight.com/guides/tableau-playbook-sparklines

Linear models

- Model:

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_p x_{ip} + \varepsilon_i$$

- Xs: quantitative predictors, factors, interactions, …
- Assumptions:
  - **Linearity**: Predictors (possibly transformed) are linearly related to the outcome, **y**. [This just means linear in the **parameters**.]
  - **Specification**: No important predictors have been omitted; only important ones included. [This is often key & overlooked.]
  - The "holy trinity":
    - **Independence**: the errors are uncorrelated
    - **Homogeneity of variance**: $Var(\varepsilon_i) = \sigma^2$ = constant
    - **Normality**: ε_i have a normal distribution

$$\varepsilon_i \sim_{iid} \mathcal{N}(0, \sigma^2)$$

The General Linear Model

- "linear" models can include:
  - transformed predictors: $\sqrt{age}$, log(*income*)
  - polynomial terms: age^2, age^3, `poly(age, n)`
  - categorical "factors", coded as dummy (0/1) variables
    - treated (Yes/No), Gender (M/F/non-binary)
  - interactions: effects of x_1 vary over levels of x_2
    - treated × age, treated × sex, (2 way)
    - treated × age × sex (3 way)
- Linear model means linear in the parameters (β_i),

$$y = \beta_0 + \beta_1 \text{age} + \beta_2 \text{age}^2 + \beta_3 \log(\text{income}) + \beta_4 (\text{sex}=\text{"F"}) + \beta_5 \text{age} \times (\text{sex}=\text{"F"}) + \epsilon$$

- In R, all handled by `lm(y ~ …)`

Fitting linear models in R: lm()

- In R, **`lm()`** for everything
  - Regression models (X1, ... quantitative)

```
lm(y ~ X1, data=dat)                    # simple linear regression
lm(y ~ X1+X2+X3, data=dat)              # multiple linear regression
lm(y ~ (X1+X2+X3)^2, data=dat)          # all two-way interactions
lm(log(y) ~ poly(X,3), data=dat)        # arbitrary transformations
```

  - ANOVA/ANCOVA models (A, B, ... factors)

```
lm(y ~ A)                   # one way ANOVA
lm(y ~ A*B)                 # two way: A + B + A:B
lm(y ~ X + A)               # one way ANCOVA
lm(y ~ (A+B+C)^2)           # 3-way ANOVA: A, B, C, A:B, A:C, B:C
```

Fitting linear models in R: lm()

- Multivariate models: **`lm()`** with 2+ y vars
  - Multivariate regression

```
lm(cbind(y1, y2) ~ X1 + X2 + X3)                # std MMreg: all linear
lm(cbind(y1, y2) ~ poly(X1,2) + poly(X2,2))     # response surface
```

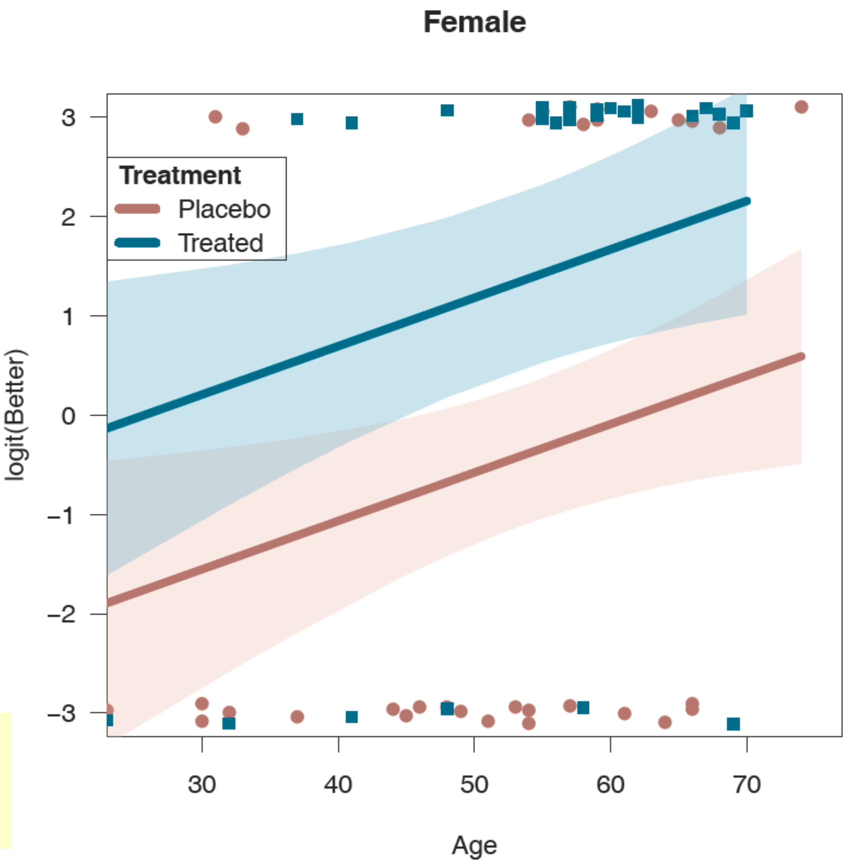
  - MANOVA/MANCOVA models

```
lm(cbind(y1, y2, y3) ~ A * B)       # 2-way MANOVA: A + B + A:B
lm(cbind(y1, y2, y3) ~ X + A)       # MANCOVA (equal slopes)
lm(cbind(y1, y2) ~ X + A + X:A)     # heterogeneous slopes
```

Generalized Linear Models: glm()

Transformations of y & other error distributions

- $y \in$ (0/1): lived/died; success/fail; …
- logit (log odds) model:
  - $\text{logit}(y) = \log \frac{\Pr(y=1)}{\Pr(y=0)}$
  - linear logit model: $\text{logit}(y) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots$



glm(better ~ age + treat, family=binomial,
data=Arthritis)

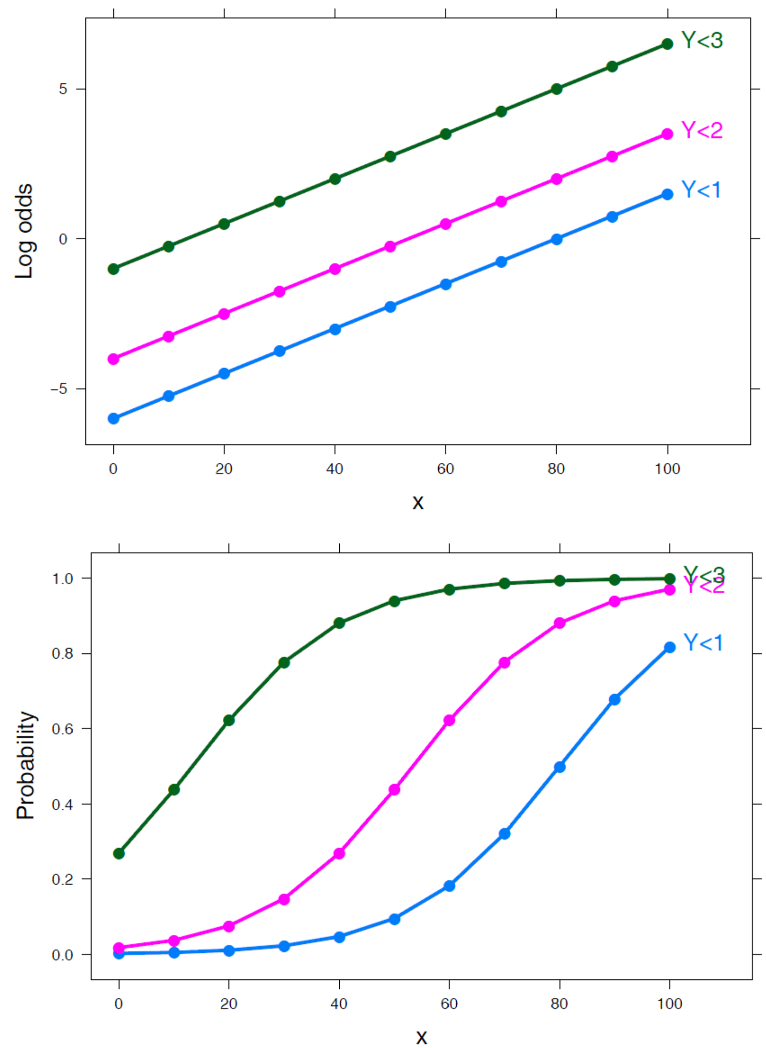
Generalized Linear Models

Ordinal responses

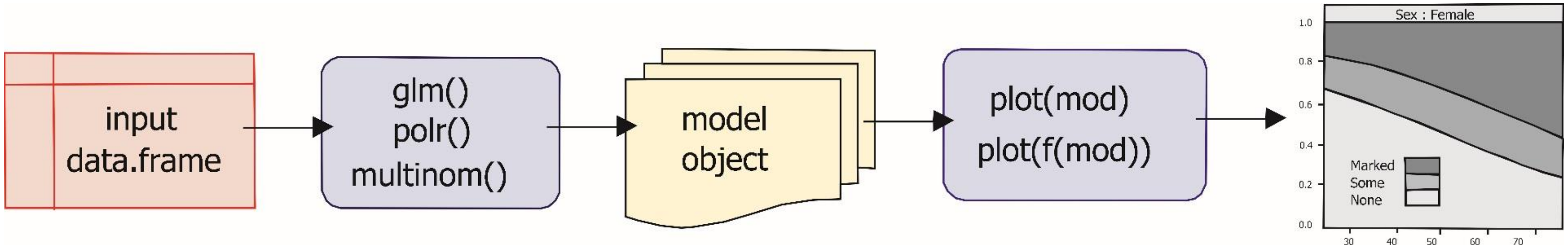
- Improved ∈ ("None" < "Some" < "Marked")
- Models: Proportional odds, generalized logits, …

```
library(MASS)
polr(Improved ~ Sex + Treat + Age,
     data=Arthritis)

library(nnet)
multinom(Improved ~ Sex + Treat + Age,
         data=Arthritis)
```

The linear modeling framework remains the same!



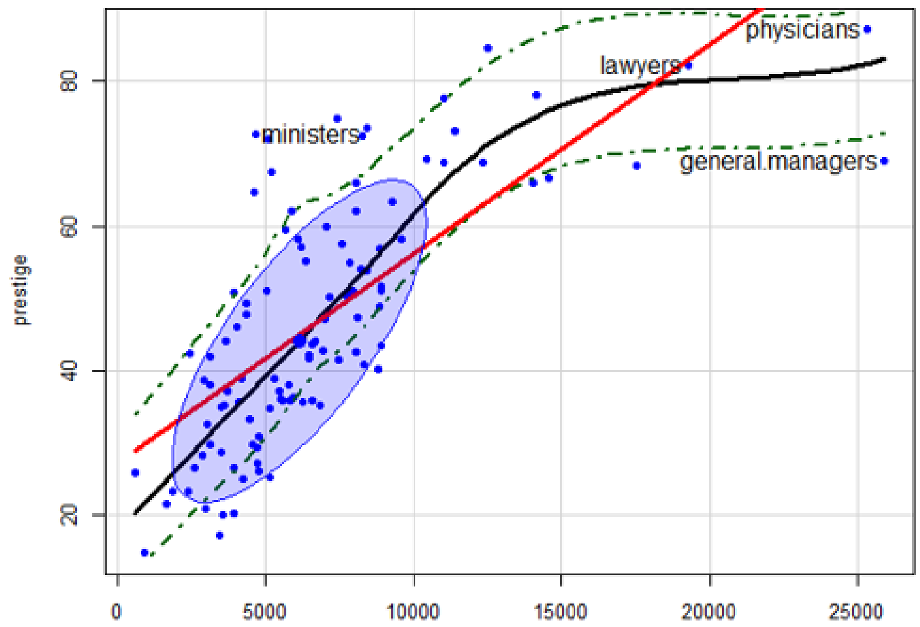
Model-based methods: Overview

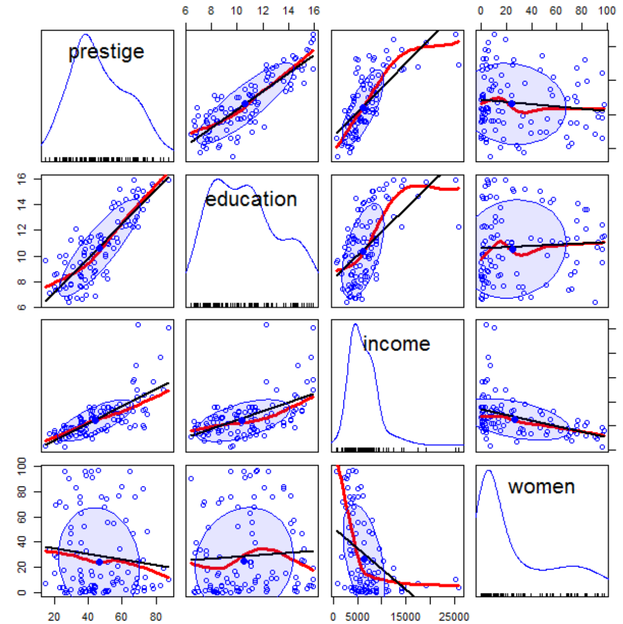


- models in R are specified by a symbolic model formula, applied to a data.frame
  - mod<-lm(prestige ~ income + educ, data=Prestige)
  - mod<-glm(better ~ age + sex + treat, data=Arthritis, family=binomial)
  - mod<-MASS:polr(improved ~ age + sex + treat, data=Arthritis)
- result (`mod`) is a "model object", of class "`lm`", "`glm`", …
- method functions:
  - plot(mod), plot(f(mod)), …
  - summary(mod), coef(mod), predict(mod), …

Plots for linear models

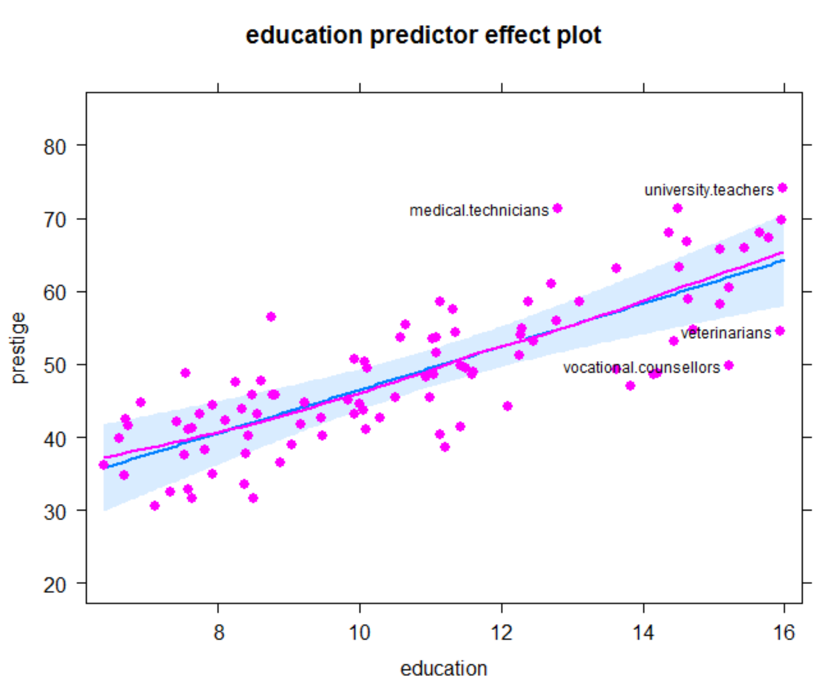
- Data plots:
  - plot response (y) vs. predictors, with smooth summaries
  - scatterplot matrix --- all pairs

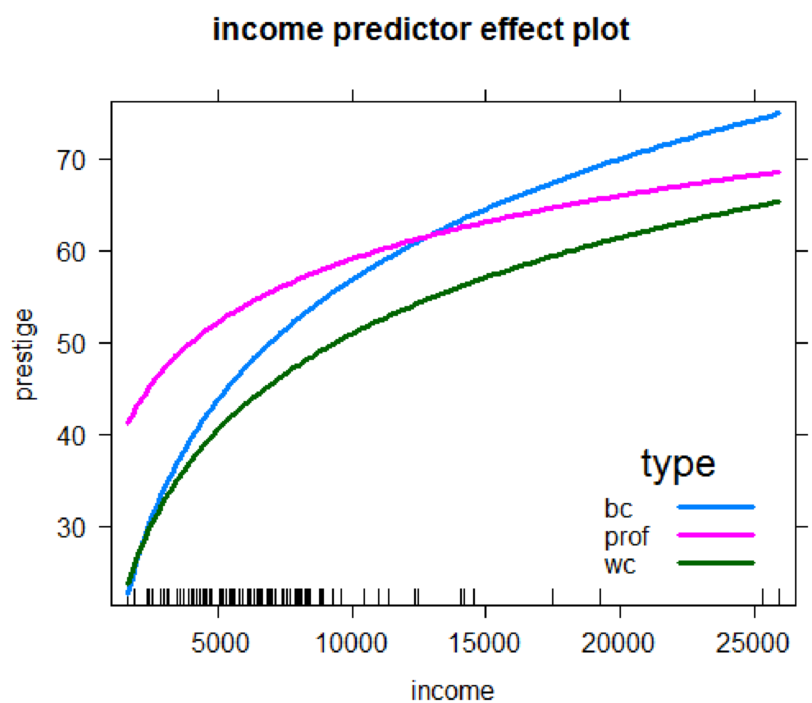




Plots for linear models

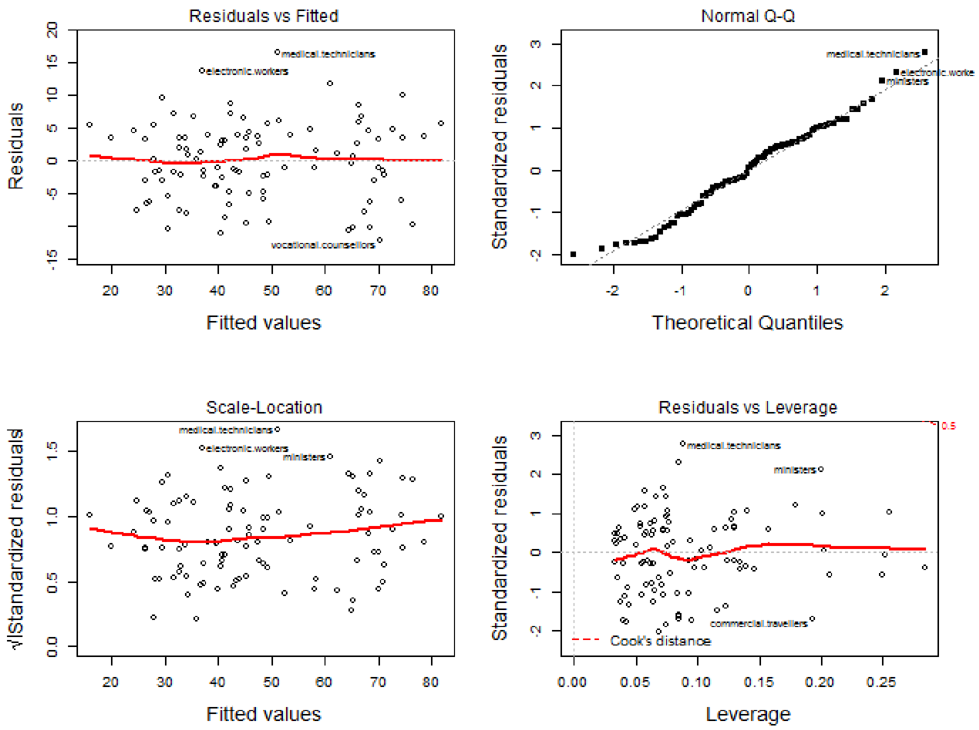
- Model (effect) plots
  - plot predicted response ($\hat{y}$) vs. predictors, controlling for variables not shown.





Plots for linear models

- Diagnostic plots
  - N QQ plot: normality of residuals? outliers?
  - Influence plots: leverage & outliers
  - Spread-level plots (non-constant variance?)



R packages

- `car`
  - Enhanced scatterplots
  - Diagnostic plots
- `effects`
  - Plot fitted effects of one predictor, controlling all others
- `visreg`
  - similar to effect plots, simpler syntax
- Both `effects` & `visreg` handle nearly all formula-based models
  - lm(), glm(), gam(), rlm, nlme(), …

Occupational Prestige data

- Data on prestige of 102 occupations and
  - average education (years)
  - average income ($)
  - % women
  - type (Blue Collar, Professional, White Collar)

```
> car::some(Prestige, 6)
                    education income women prestige census type
architects              15.44  14163  2.69     78.1   2141 prof
physicians              15.96  25308 10.56     87.2   3111 prof
commercial.artists      11.09   6197 21.03     57.2   3314 prof
tellers.cashiers        10.64   2448 91.76     42.3   4133   wc
bakers                   7.54   4199 33.30     38.9   8213   bc
aircraft.workers         8.78   6573  5.78     43.7   8515   bc
```

Follow along

The R script (prestige-ex.R) for this example is linked on the course page. Download and open in R Studio to follow along.

- **Examples:**
  - Prestige data prestige-ex.R || prestige-ex.html
  - Penguin data penguins-lm-ex.R || penguins-lm-ex.html

The script was run with `knitr` (ctrl+shift+K) in R Studio to create the HTML output (prestige-ex.html)

The **Code** button there allows you to download the R code and comments

Linear models example: Occupational Prestige data Code ▾ Download Rmd

Michael Friendly

(These show a simple way to turn R scripts into finished documents)

Informative scatterplots

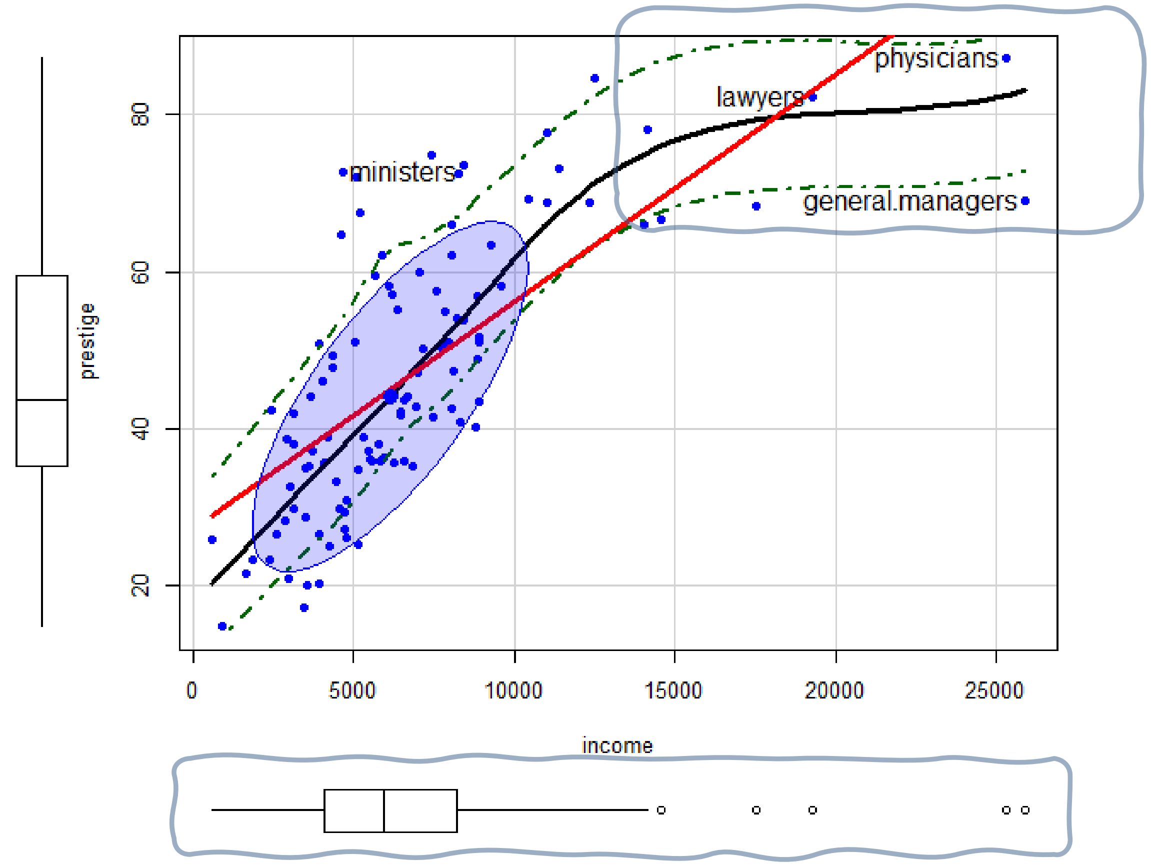
Scatterplots are most useful when enhanced with annotations & statistical summaries

Data ellipse and regression line show the linear model, prestige ~ income

Point labels show possible outliers

Smoothed (loess) curve and CI show the trend

Boxplots show marginal distributions



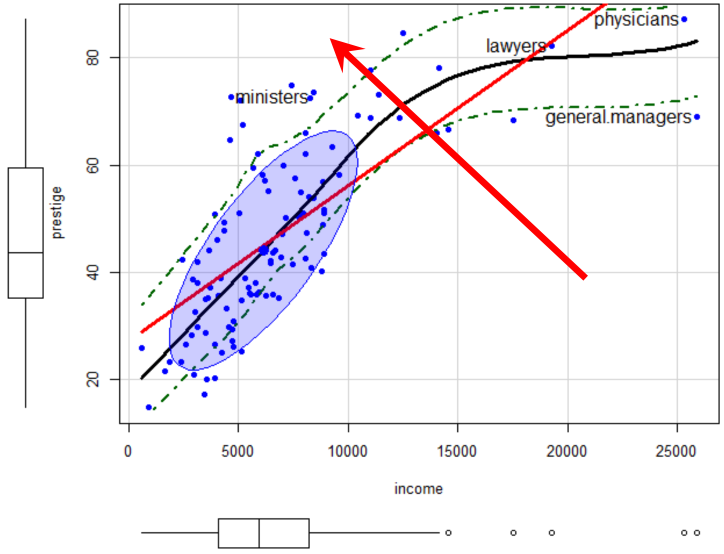
Informative scatterplots

`car::scatterplot()` provides all these enhancements

```
scatterplot(prestige ~ income, data=Prestige,
            pch = 16,
            regLine = list(col = "red", lwd=3),
            smooth = list(smoother=loessLine,
                          lty.smooth = 1, col.smooth = "black",
                          lwd.smooth=3, col.var = "darkgreen"),
            ellipse = list(levels = 0.68),
            id = list(n=4, col="black", cex=1.2))
```

Skewed distribution of income & non-linear relation suggest need for a transformation

Arrow rule: move on the scale of powers in direction of the bulge
e.g.: x → sqrt(income) or log(income)

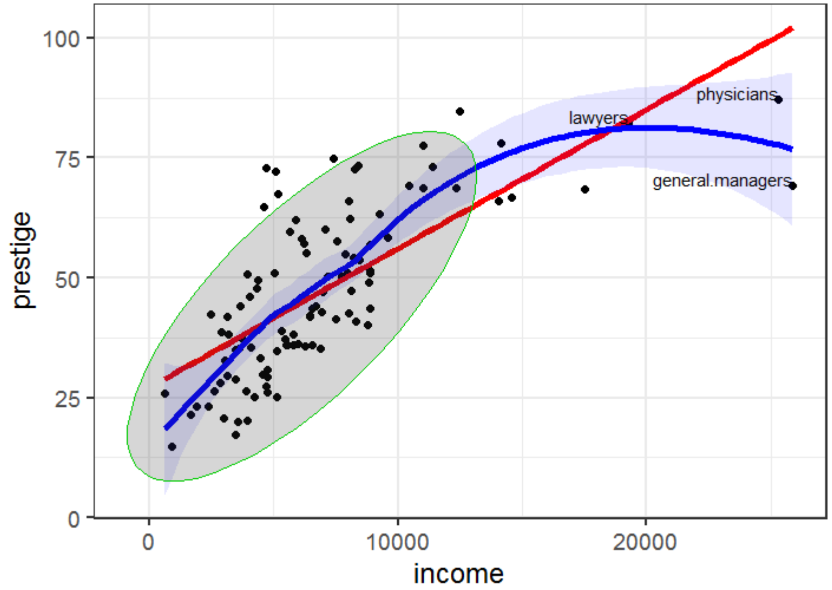


The same, with ggplot2

```
ggplot(data=Prestige,
     aes(x = income, y = prestige)) +
  geom_point(size=2) +
  geom_smooth(method = "lm", color = "red", se=FALSE, size=2) +
  geom_smooth(method = "loess", color = "blue", size = 2, fill="blue", alpha=0.1) +
  stat_ellipse(geom = "polygon", alpha = 0.2, color = "green3") +
  geom_text(aes(label=ifelse(income>18000,                    # select points to label (kludge!)
                             as.character(row.names(Prestige)),'')),
                hjust=1, vjust=0) +
  theme_bw(base_size = 18)
```

You can do the same with ggplot2, but it takes more steps to duplicate that plot

Each layer needs a geom_ or stat_



Try log(income)

```
scatterplot(prestige ~ income, data=Prestige,
            log = "x",       # plot on log scale
            pch = 16,
            regLine = list(col = "red", lwd=3),
            … )
```

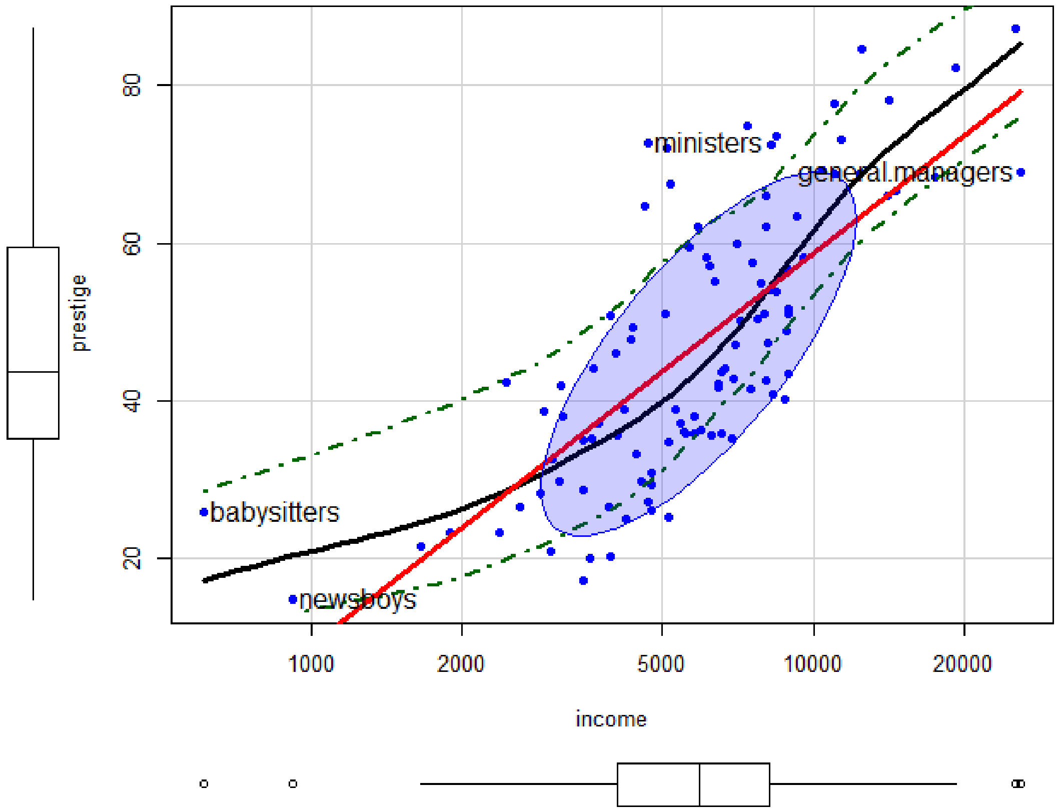
Income now ~ symmetric

Relation closer to linear

log(income): interpret as effect of a multiple
E.g., using log2(income)

```
> coef(mod_log)
 (Intercept) log2(income)
    -139.9       14.9
```

2 * income -> prestige ↑ 14.9



Stratify by type?

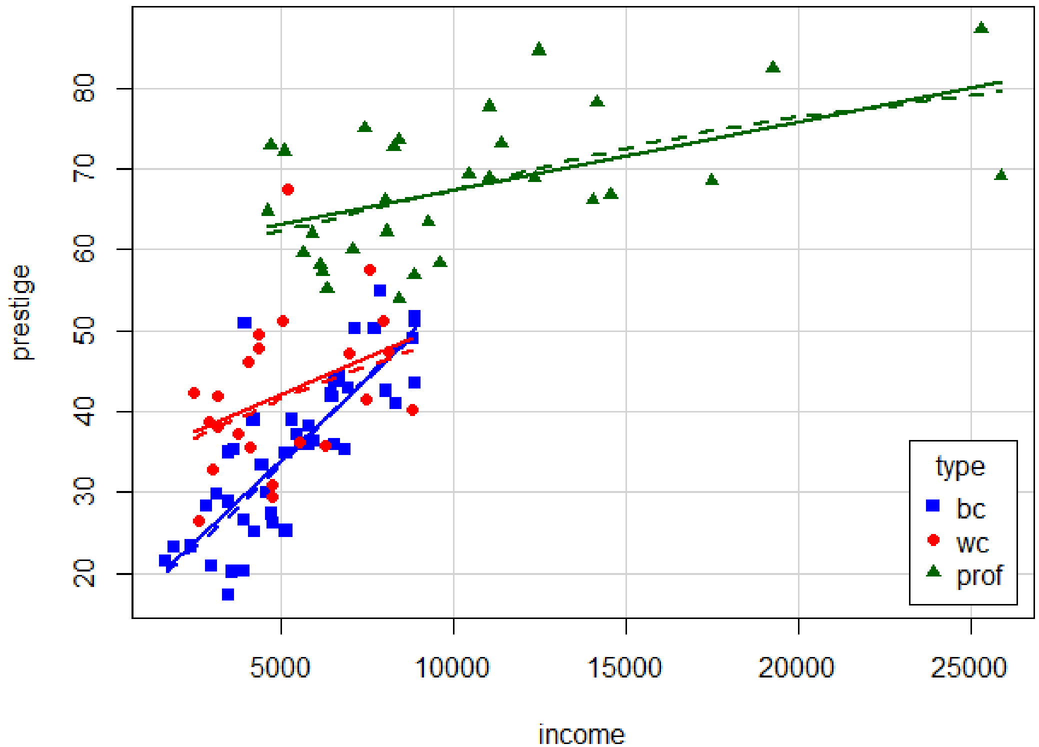
```
scatterplot(prestige ~ income | type, data=Prestige,
      col = c("blue", "red", "darkgreen"),
      pch = 15:17,
      legend = list(coords="bottomright"),
      smooth=list(smoother=loessLine, var=FALSE, span=1, lwd=4))
```

Formula: | type → "given type"

Different slopes: interaction of income * type

Provides another explanation of the non-linear relation

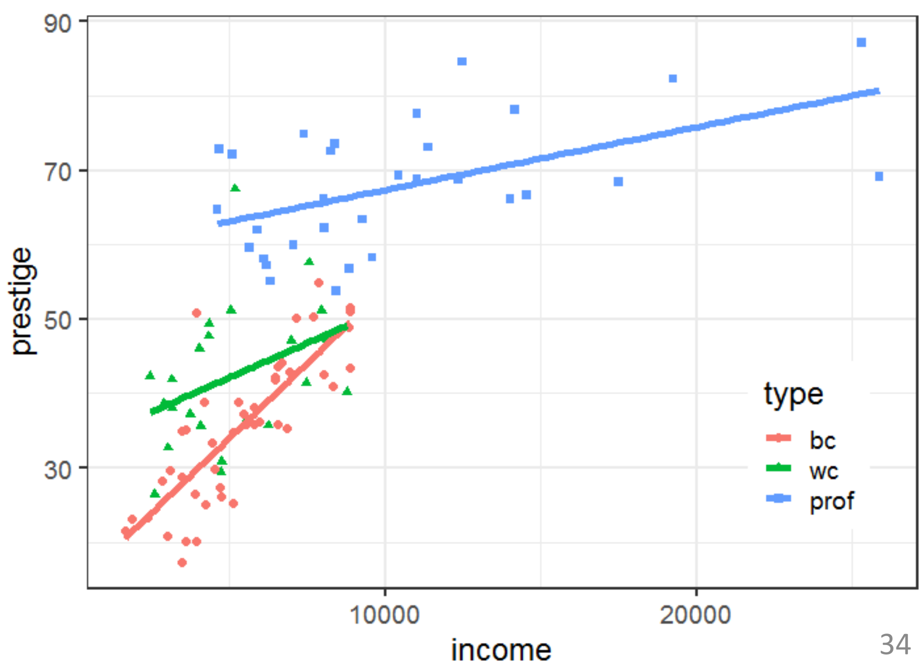
This may be a new finding!



ggplot2 version

```
ggplot(data=subset(Prestige, !is.na(type)),
      aes(x = income, y = prestige, color = type, shape=type)) +
  geom_point(size=2) +
  geom_smooth(method = "lm", se=FALSE, size=2) +
  theme_bw(base_size = 18) +
  theme(legend.position = c(0.87, 0.25))
```

Setting the color and shape aesthetics give different symbols and regression lines for each group



Scatterplot matrix

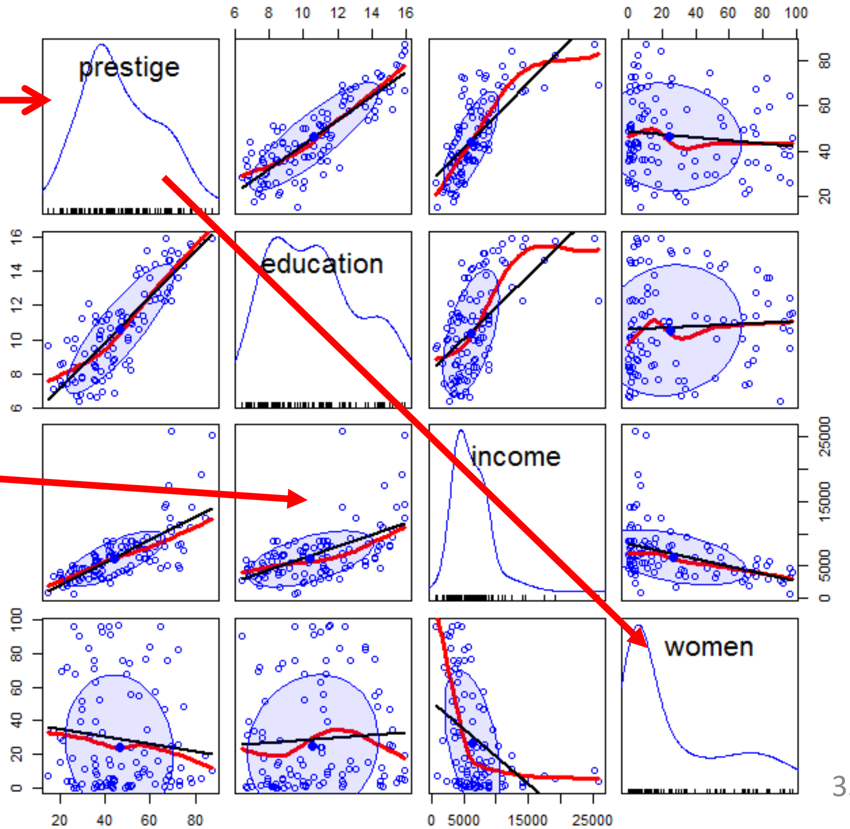
```
scatterplotMatrix(~ prestige + education + income + women ,
      data=Prestige,
       regLine = list(method=lm, lty=1, lwd=2, col="black"),
       smooth=list(smoother=loessLine, spread=FALSE,
              lty.smooth=1, lwd.smooth=3, col.smooth="red"),
       ellipse=list(levels=0.68, fill.alpha=0.1))
```

prestige vs. all predictors

diagonal: univariate distributions

- income: + skewed
- %women: bimodal

off-diagonal: relations among predictors



Fit a simple model

```
> mod0 <- lm(prestige ~ education + income + women,
+            data=Prestige)
> summary(mod0)

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) -6.7943342  3.2390886  -2.098   0.0385 *
education    4.1866373  0.3887013  10.771  < 2e-16 ***
income       0.0013136  0.0002778   4.729 7.58e-06 ***
women       -0.0089052  0.0304071  -0.293   0.7702
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Multiple R-squared:  0.7982,  Adjusted R-squared:  0.792
F-statistic: 129.2 on 3 and 98 DF,  p-value: < 2.2e-16
```

Fits very well

But this ignores:

- nonlinear relation with income: should use log(income)
- occupation type
- possible interaction of income*type

Fit a more complex model

```
> mod1 <- lm(prestige ~ education + women +
+              log(income)*type, data=Prestige)
> summary(mod1)


Coefficients:
                       Estimate Std. Error t value Pr(>|t|)
(Intercept)          -152.20589   23.24988  -6.547 3.54e-09 ***
education               2.92817    0.58828   4.978 3.08e-06 ***
women                   0.08829    0.03234   2.730  0.00761 **
log(income)            18.98191    2.82853   6.711 1.67e-09 ***
typeprof               85.26415   30.45819   2.799  0.00626 **
typewc                 29.41334   36.50749   0.806  0.42255
log(income):typeprof   -9.01239    3.41020  -2.643  0.00970 **
log(income):typewc     -3.83343    4.26034  -0.900  0.37063
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Multiple R-squared:  0.8751,  Adjusted R-squared:  0.8654
F-statistic: 90.07 on 7 and 90 DF,  p-value: < 2.2e-16
```

add interaction of log income by type

Fits even better!

But how to understand?

Coefs for `type` compare mean "wc" and "prof" to "bc"
Coefs for `log(income)*type` compare "wc" and "prof" slopes with that of "bc"

Coefficient plots

Plots of coefficients with CI often more informative than tables, but care is needed

Compare 3 models:

```
mod0 <- lm(prestige ~ education + income + women, data=Prestige)
mod1 <- lm(prestige ~ education + women + income + type, data=Prestige)
mod2 <- lm(prestige ~ education + women + income * type, data=Prestige)
```

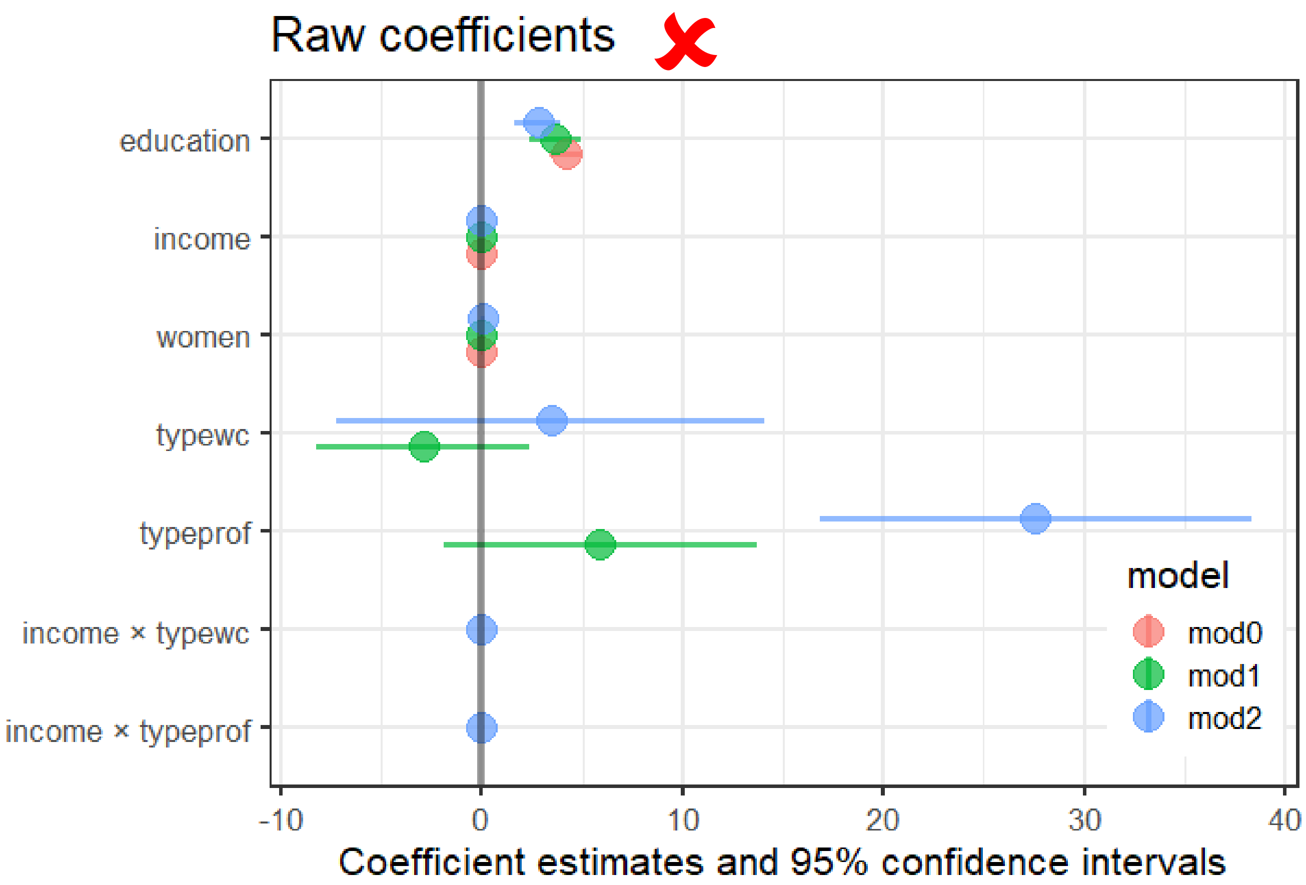
The `modelsummary` package gives reasonably nice plots. Feed it a list of models:

```
library(modelsummary)
models <- list("mod0" = mod0, "mod1" = mod1, "mod2" = mod2)
modelplot(models,
          coef_omit="Intercept",
          size=1.3, alpha=0.7) +
    labs(title="Raw coefficients")
```

Coefficient plots

Raw **b** coefficients are on different scales, so are not comparable

Effect of income appears to be NS



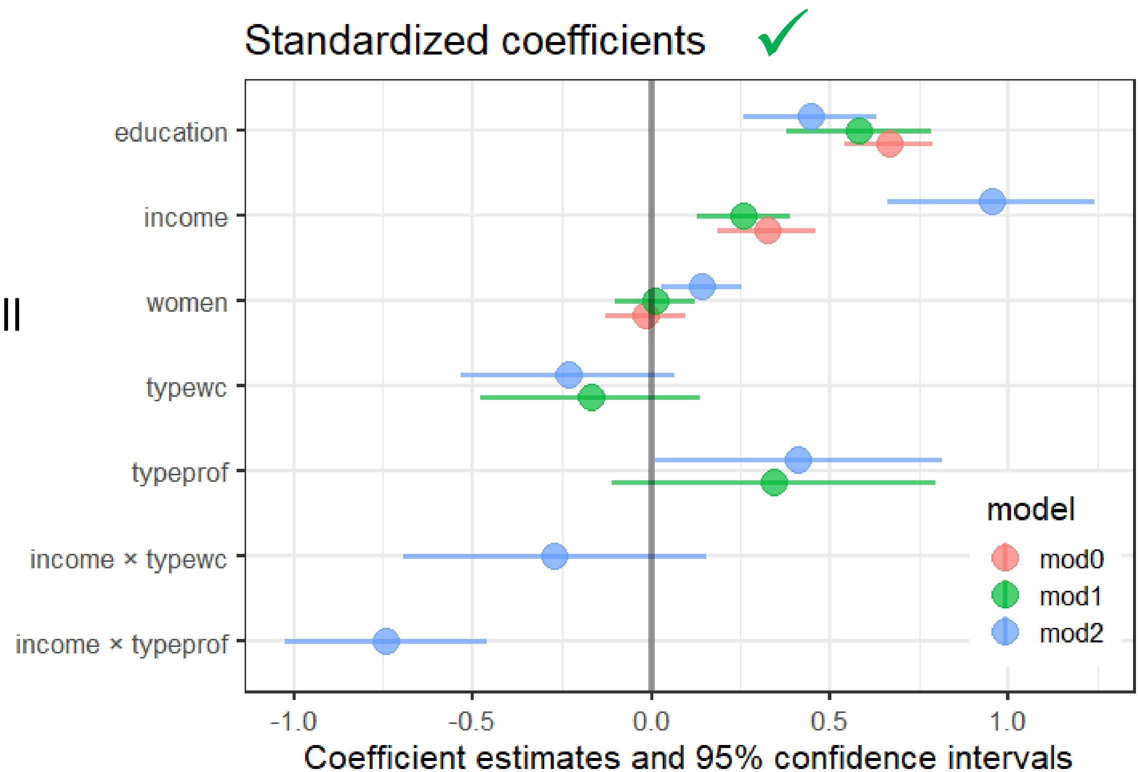
Instead, plot the standardized **β** coefficients.

- Get these by scaling the variables to mean=0, stddev=1
- Re-fit the models to the standardized data

```
Prestige_std <- Prestige |> mutate(across(where(is.numeric), scale))

mod0_std <- lm(prestige ~ education + income + women, data=Prestige_std)
mod1_std <- lm(prestige ~ education + women + income + type, data=Prestige_std)
mod2_std <- lm(prestige ~ education + women + income * type, data=Prestige_std)
```

This reflects the results shown in tabular output

Effect of income is signif. in all 3 models

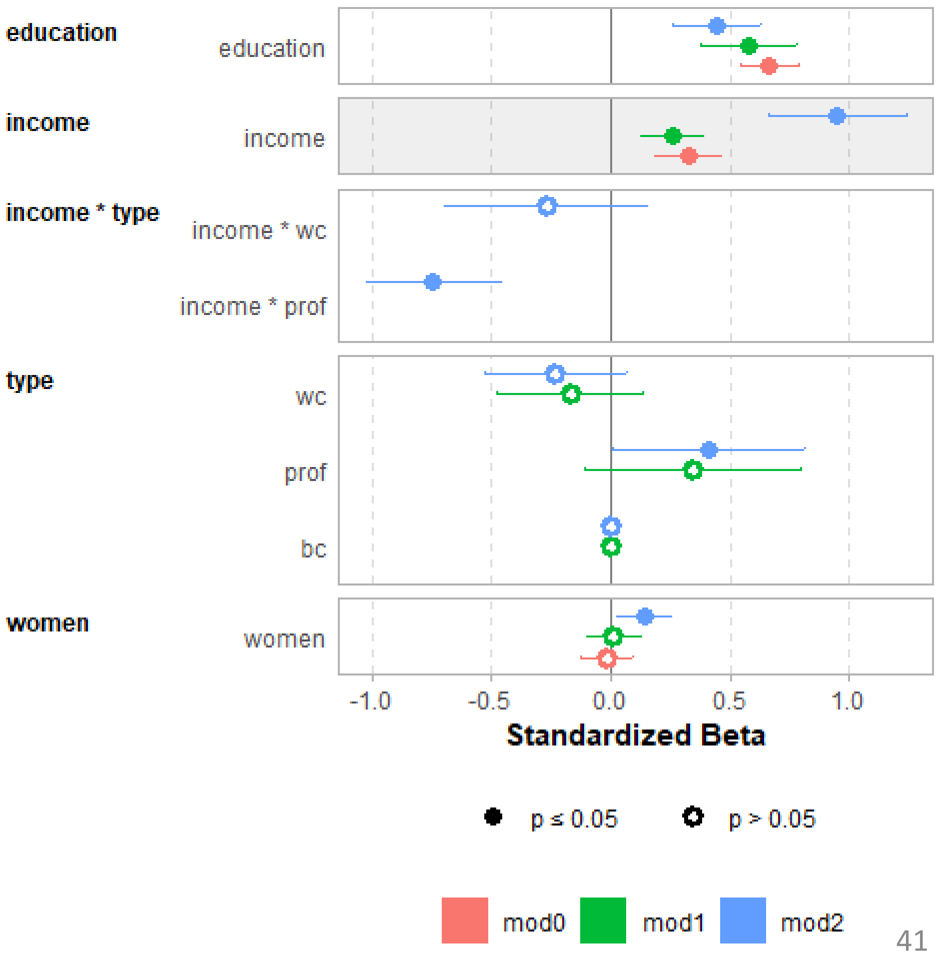


GGally::ggcoef_*() plots

The GGally package provides ggcoef_plot() and ggcoef_compare() for pretty plots
It uses the broom package to extract information from models

```
models <- list("mod0" = mod0_std,
        "mod1" = mod1_std,
        "mod2" = mod2_std)
ggcoef_compare(models) +
    xlab("Standardized Beta")
```

The reference category is shown for factors, facilitating interpretation



Model (effect) plots

- We'd like to see plots of the predicted value ($\hat{y}$) of the response against predictors (x_j)
  - Ordinary plot of y vs. x_j doesn't allow for other correlations
  - $\rightarrow$ Must control (adjust) for other predictors (x_{-j}) not shown in a given plot
- Effect plots
  - Variables not shown (x_{-j}) are averaged over.
  - Slopes of lines reflect the partial coefficient in the model
  - Partial residuals can be shown also

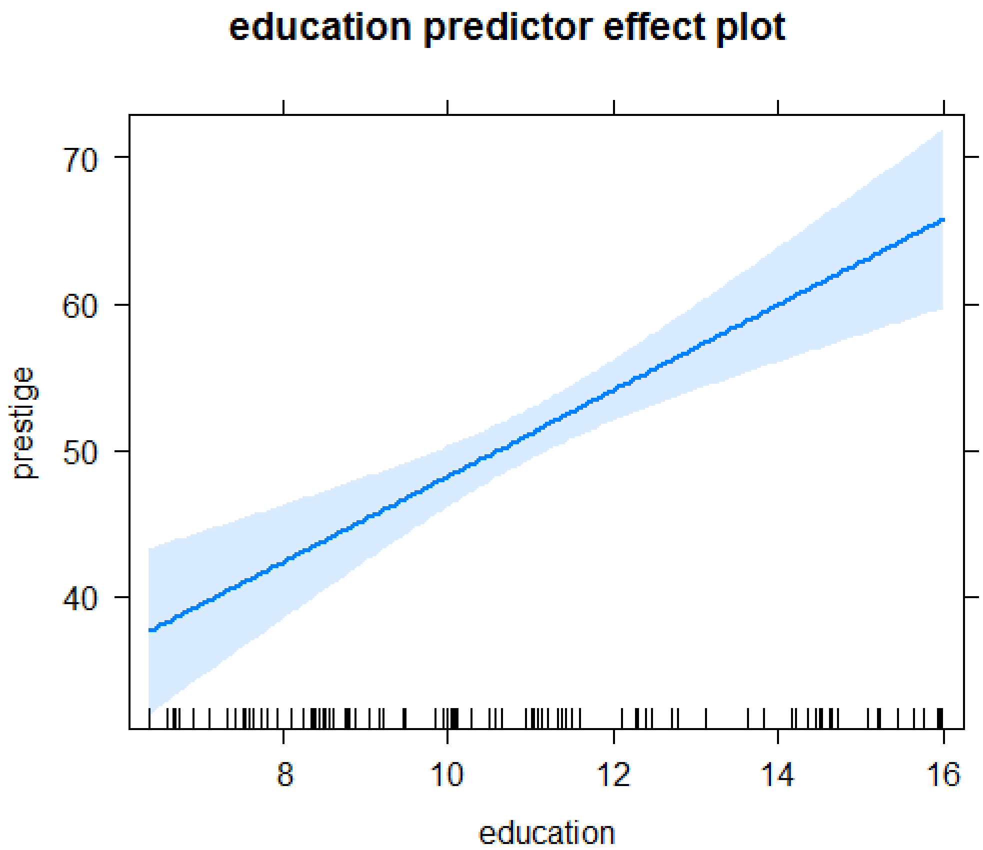
For details, see `vignette("predictor-effects-gallery", package="effects)`

Model (effect) plots: education

```
library("effects")
mod1.e1 <- predictorEffect("education", mod1)
plot(mod1.e1)
```

This graph shows the partial slope for education, controlling for all others

For each ↑ year in education, fitted prestige ↑2.93 points, (other predictors held fixed)

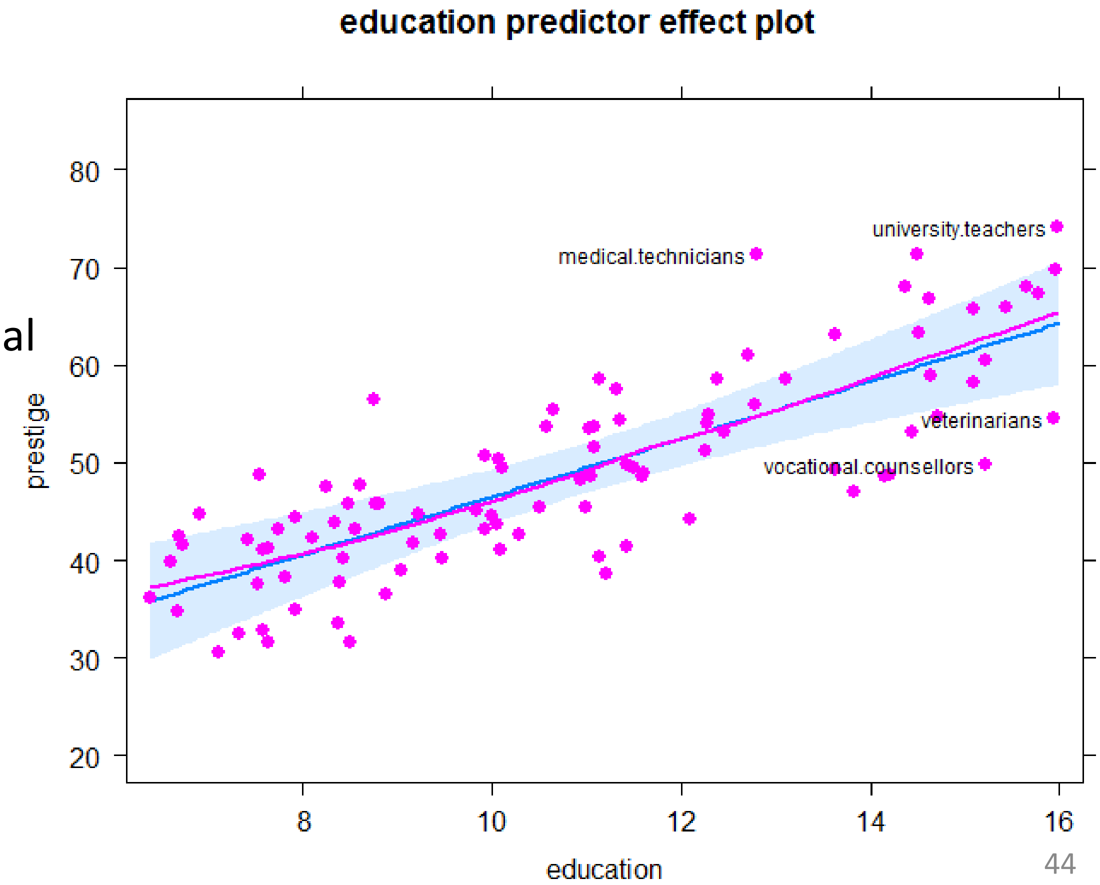


Model (effect) plots

```
mod1.e1a <- predictorEffect("education", mod1, residuals=TRUE)
plot(mod1.e1a,
     residuals.pch=16, id=list(n=4, col="black"))
```

Partial residuals show the residual of prestige controlling for other predictors

Unusual points here would signal undue influence



Model (effect) plots: women

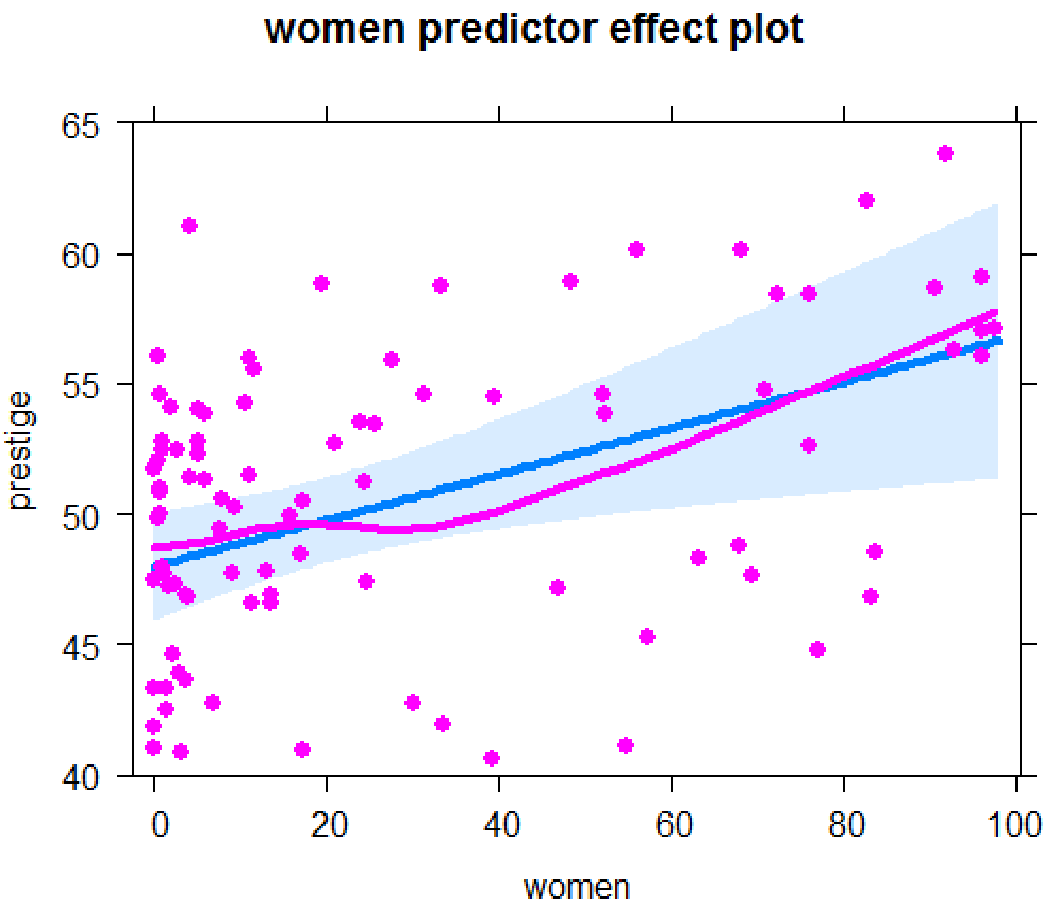
```
mod1.e2 <- predictorEffect("women", mod1, residuals=TRUE)
plot(mod1.e2, ylim=c(40, 65), lwd=4,
     residuals.pch=16)
```

Surprise!

Prestige of occupations ↑ with % women (controlling for other variables)

Another 10% women ↑ prestige by 0.88 points

How to interpret this?



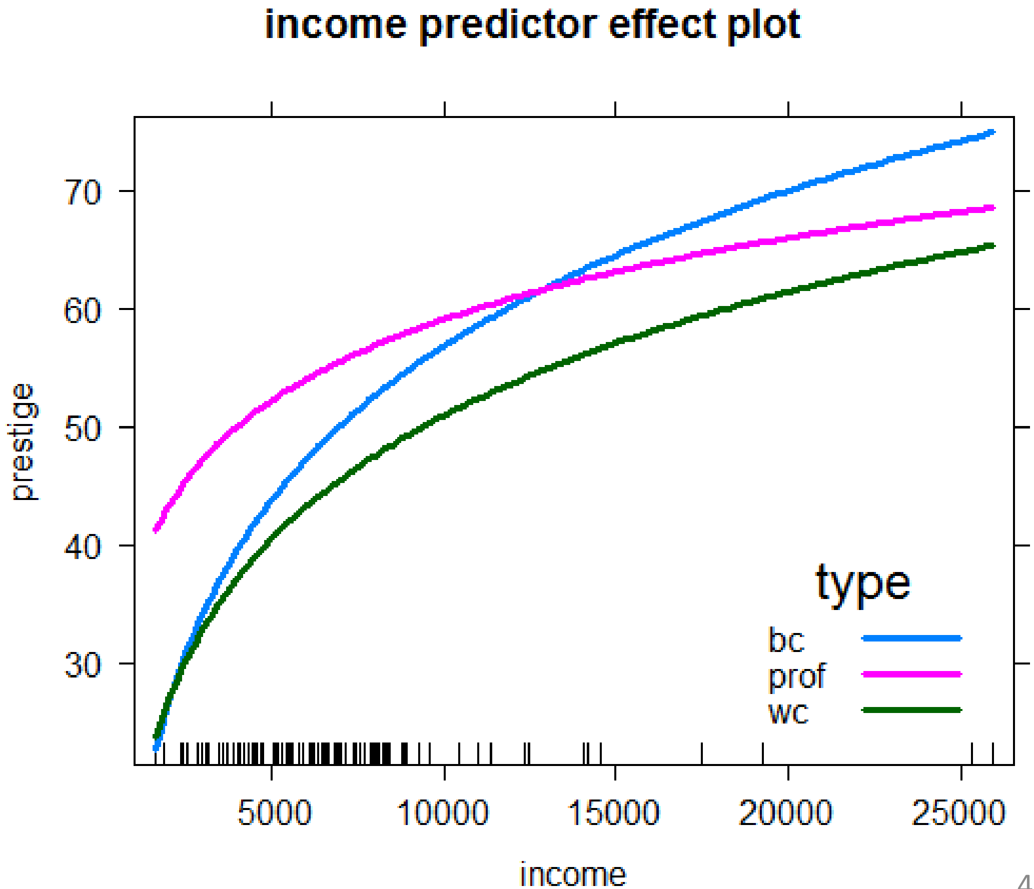
Model (effect) plots: income

```
plot(predictorEffect("income", mod1),
     lines=list(multiline=TRUE, lwd=3),
     key.args = list(x=.7, y=.35))
```

Income interacts with type in the model

The plot is curved because log(income) is in the model

Curvature reflects marginal effect of income for each occupation type



visreg plots: Air quality data

Daily air quality measurements in New York, May - Sep 1973

How does Ozone concentration vary with solar radiation, wind speed & temperature?

```
> head(airquality)
  Ozone Solar.R Wind Temp Month Day
1    41     190  7.4   67     5   1
2    36     118  8.0   72     5   2
3    12     149 12.6   74     5   3
4    18     313 11.5   62     5   4
5    NA      NA 14.3   56     5   5
6    28      NA 14.9   66     5   6
```

see: https://pbreheny.github.io/visreg/ for examples & details

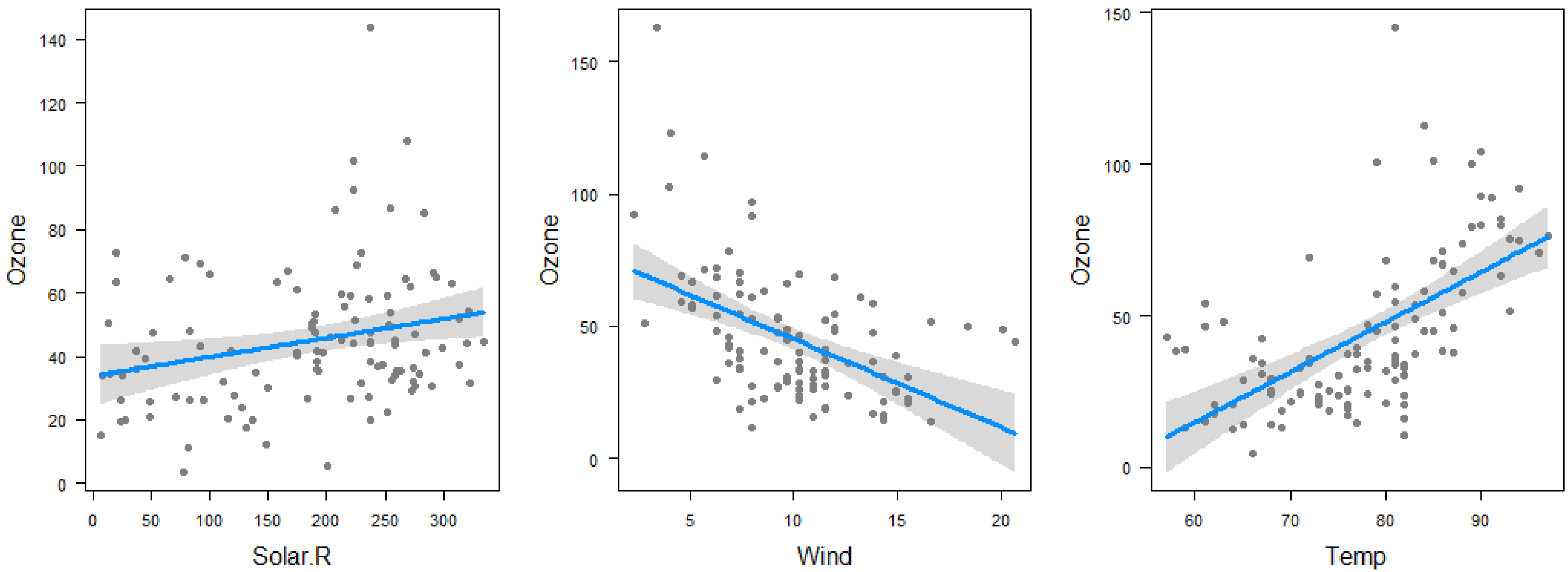
Air quality: main effects model

```
> fit1 <- lm(Ozone ~ Solar.R + Wind + Temp, data=airquality)
> summary(fit1)

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -64.3421    23.0547   -2.79   0.0062 **
Solar.R       0.0598     0.0232    2.58   0.0112 *
Wind         -3.3336     0.6544   -5.09  1.5e-06 ***
Temp          1.6521     0.2535    6.52  2.4e-09 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 21.18 on 107 degrees of freedom
  (42 observations deleted due to missingness)
Multiple R-squared:  0.6059,    Adjusted R-squared:  0.5948
F-statistic: 54.83 on 3 and 107 DF,  p-value: < 2.2e-16
```

visreg conditional plots

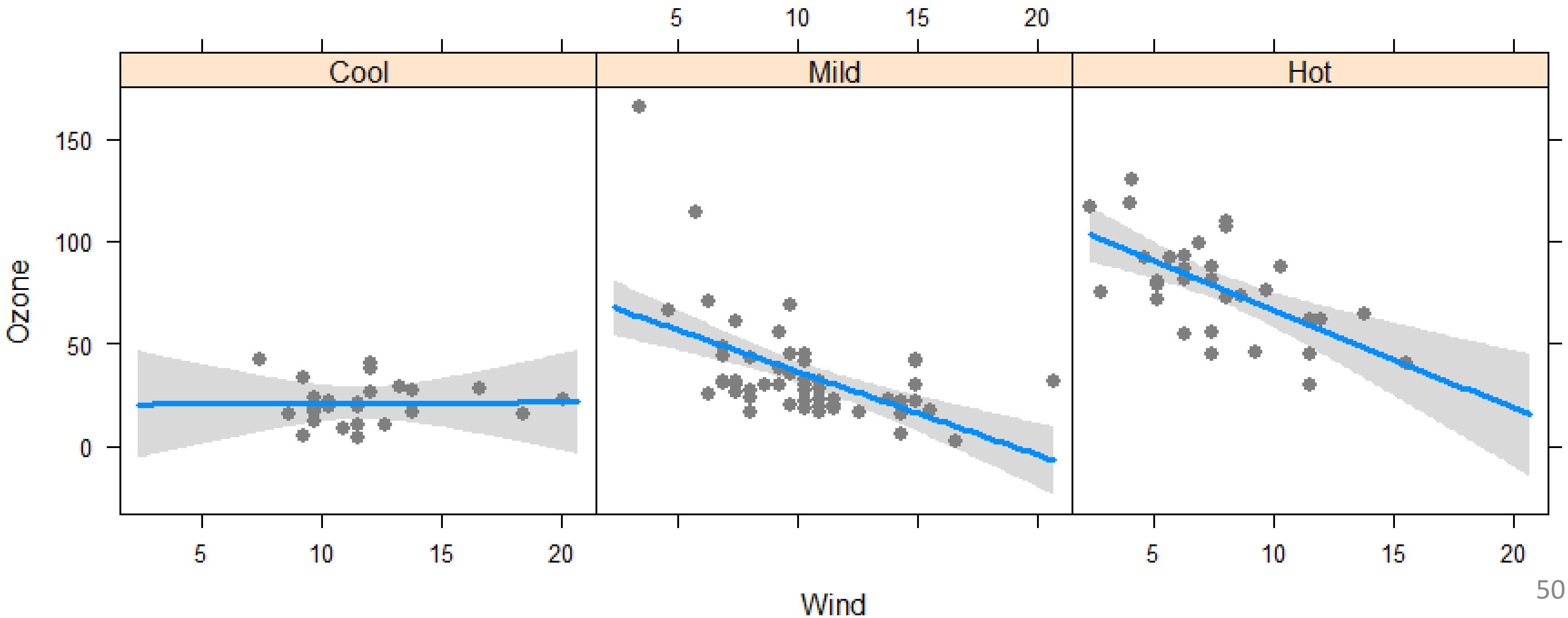
```
visreg(fit1, "Solar.R“)
visreg(fit1, "Wind“)
visreg(fit1, "Temp")
```

model summary =
- predicted values (line) +
- confidence band (uncertainty) +
- partial residuals (objections)



Factor variables & interactions

```
                                        # cut Temp into three ordered levels of equal range
airquality$Heat <- cut(airquality$Temp, 3,
                       labels=c("Cool","Mild","Hot"))

                                        # fit model with interaction  of Wind * Heat
fit2 <- lm(Ozone ~ Solar.R + Wind*Heat, data=airquality)
visreg(fit2, "Wind", by="Heat", layout=c(3,1), points=list(cex=1))
```



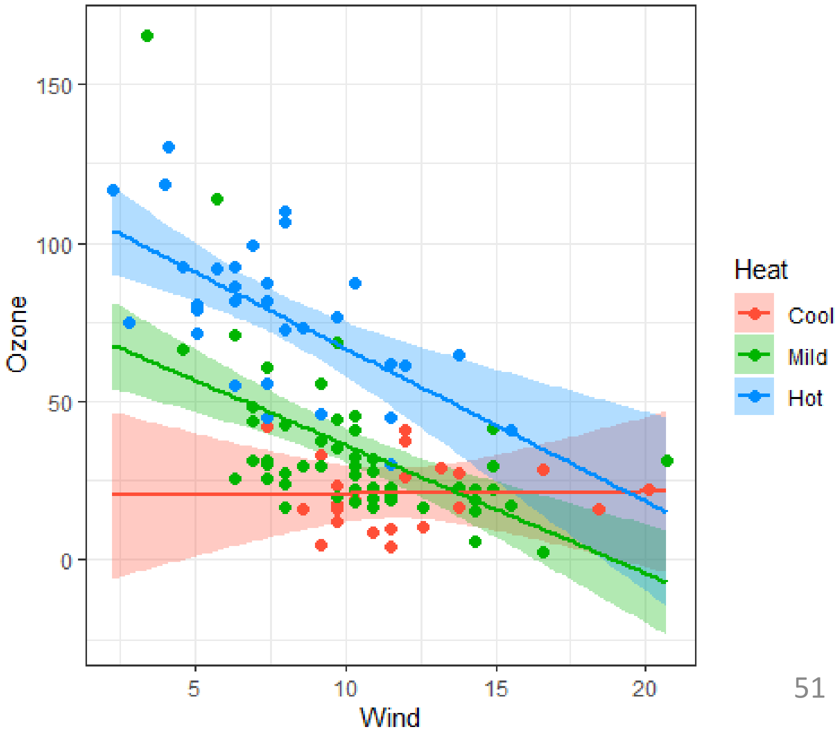
Factor variables & interactions

```
visreg(fit2, "Wind", by="Heat",
       overlay=TRUE,
            gg=TRUE,
       points=list(size=2)) +
  theme_bw()
```

overlay=TRUE → superpose panels
gg=TRUE → uses ggplot

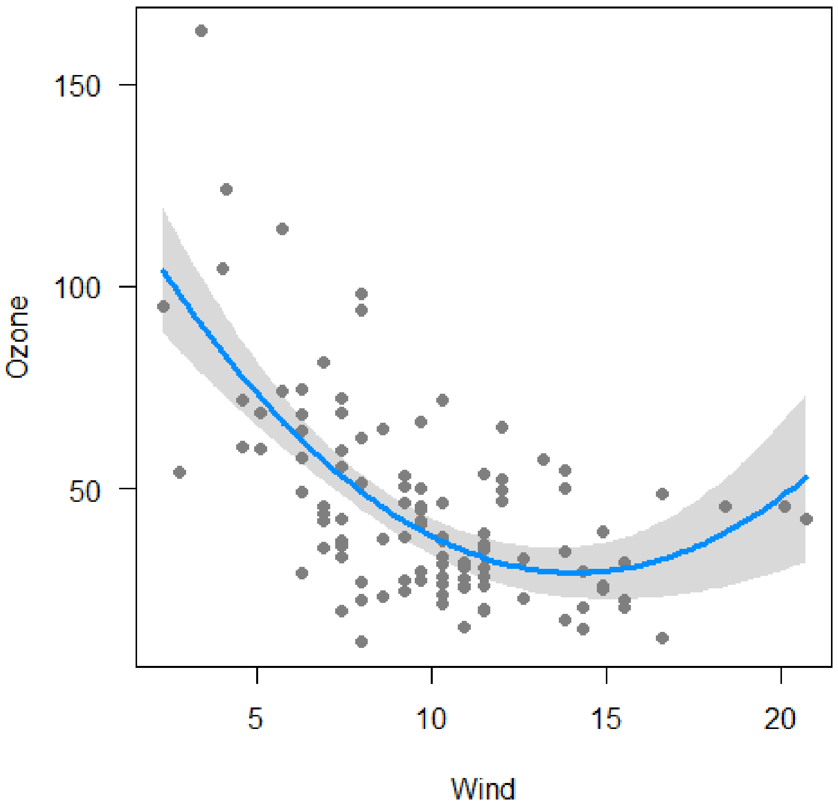
This allows slope for Wind to vary with Heat e.g., Wind has no effect when Cool

This model still assumes linear effects of Heat & Wind

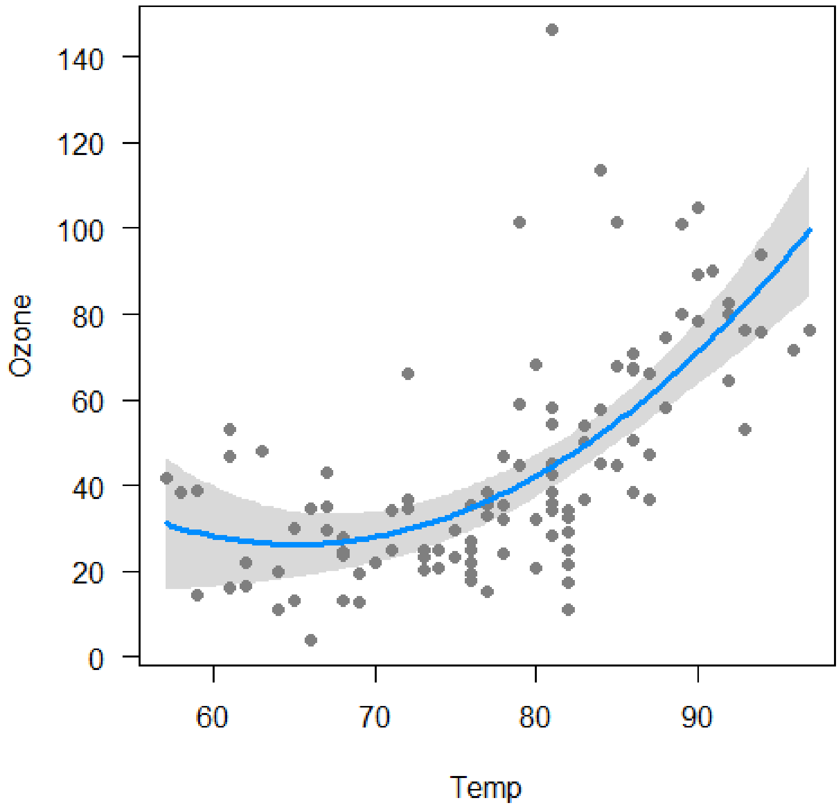


Non-linear effects

```
fit <- lm(Ozone ~ Solar.R + poly(Wind,2) +
          Temp, data=airquality)
visreg(fit, "Wind“)
```



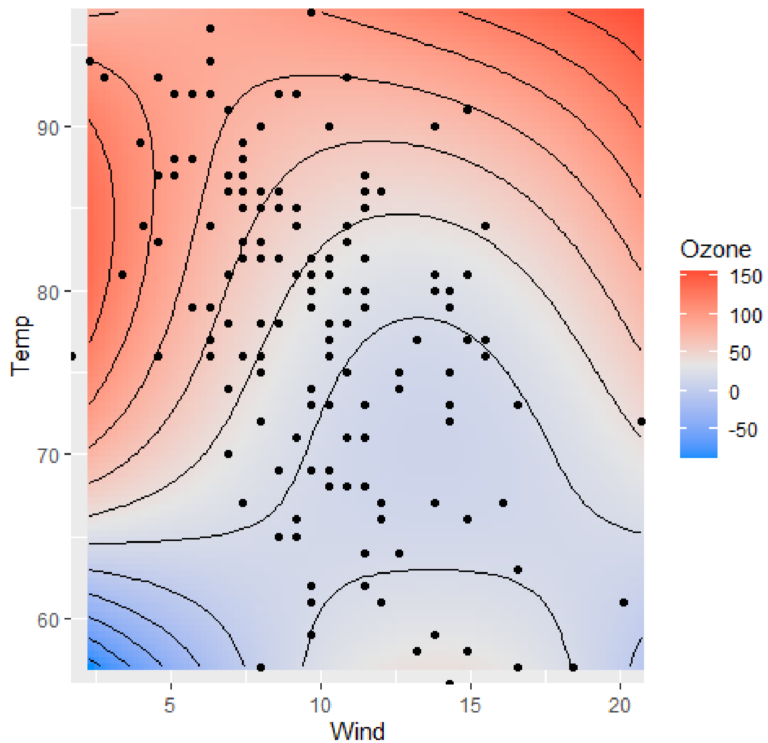

```
fit <- lm(Ozone ~ Solar.R + Wind +
          poly(Temp,2), data=airquality)
visreg(fit, "Temp")
```



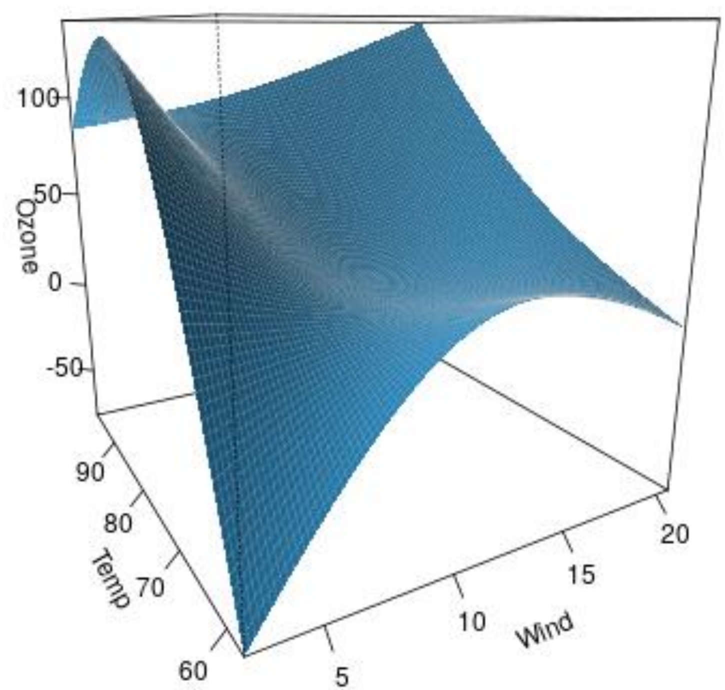
Response surface models (visreg2d)

```
# Fit quadratics in both Wind & Temp and interaction Wind * Temp
fitp <- lm(Ozone ~ Solar.R + poly(Wind,2) * poly(Temp,2), data=airquality)
```

```
visreg2d(fitp, "Wind", "Temp", plot.type="gg") +
    geom_contour(aes(z=z), color="black")
```



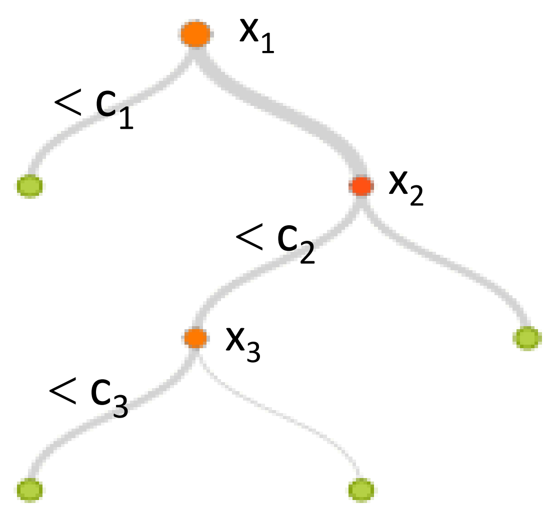

```
visreg2d(fitp, "Wind", "Temp", plot.type="persp“ )
```



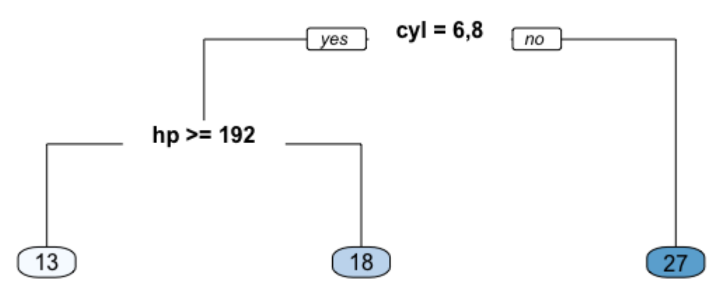
Regression trees

Regression trees are a non-parametric alternative to linear models

- Essential ideas:
  - Find predictor and split value which minimizes SSE
  - fitted value in each subgroup = mean
  - repeat, recursively, splitting by next best predictor
- Large literature
  - cost, complexity tradeoff
  - pruning methods
  - boosting, cross-validation
  - tree averaging



e.g.: mpg ~ cyl + hp



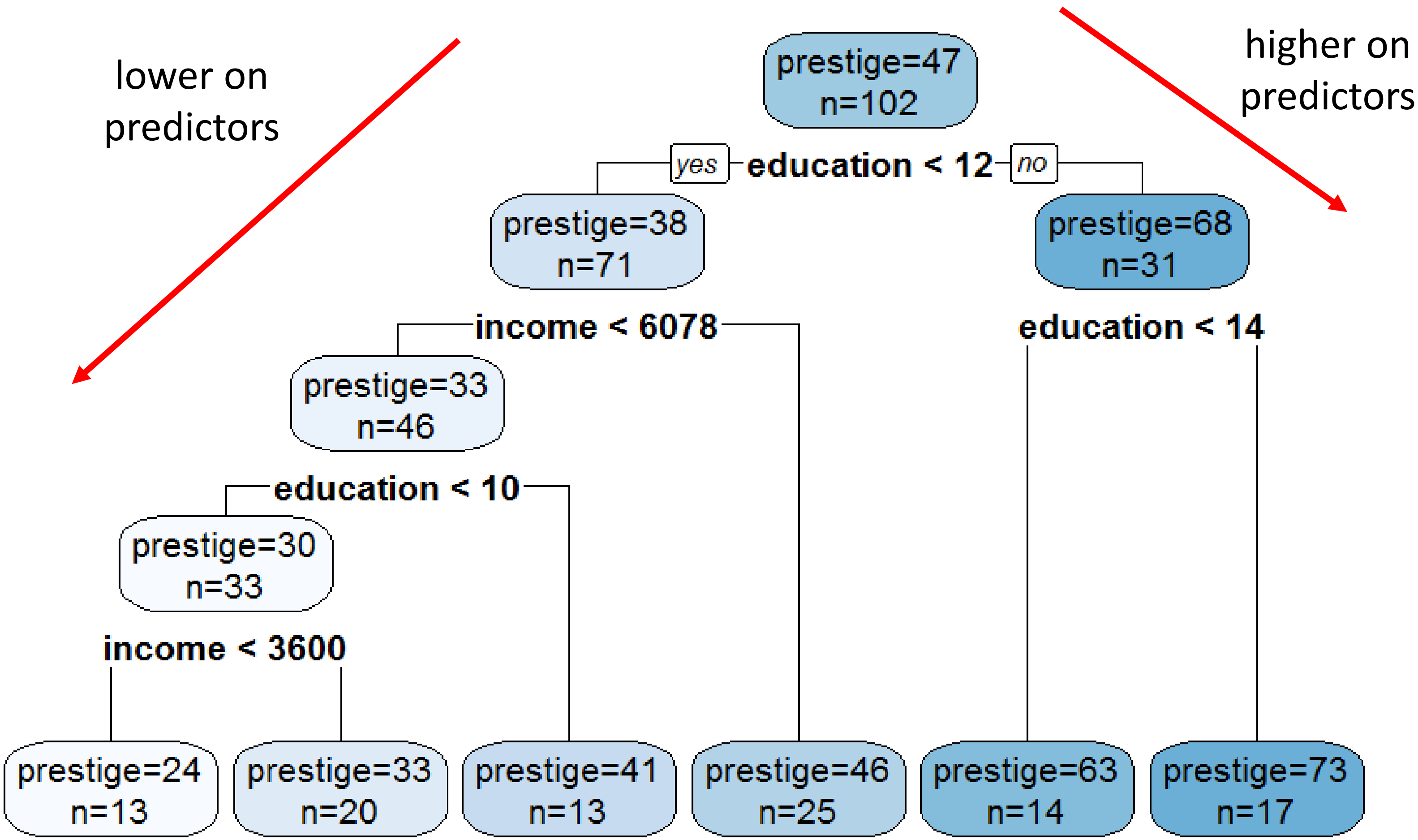
Prestige data: rpart tree

```
> library(rpart)            # calculating regression trees
> library(rpart.plot)       # plotting regression trees

> rmod <- rpart(prestige ~ education + income + women + type,
          data=Prestige,
          method = "anova")

> rpart.rules(rmod)         # print prediction rules
 prestige
    24 when education <  10         & income <  3600
    33 when education <  10         & income is 3600 to 6078
    41 when education is 10 to 12 & income <  6078
    46 when education <  12         & income >=  6078
    63 when education is 12 to 14
    73 when education >=      14
```

Only education & income are involved in this simple model.
Other controls allow setting classification details

Prestige data: rpart tree

```
rpart.plot(rmod, prefix="prestige=")
```



Diagnostic plots

- The linear model, $\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}$ assumes:
  - Residuals, ε_i are normally distributed, $\varepsilon_i \sim N(0,\sigma^2)$
  - (Normality not required for **X**s)
  - Constant variance, $Var(\varepsilon_i) = \sigma^2$
  - Observations y_i are statistically independent
- Violations $\rightarrow$ inferences may not be valid
- A variety of plots can diagnose all these problems
- Other methods (boxCox, boxTidwell) diagnose the need for transformations of **y** or **X**s.

The “regression quartet”

In R, plotting a **`lm`** model object → the “regression quartet” of plots

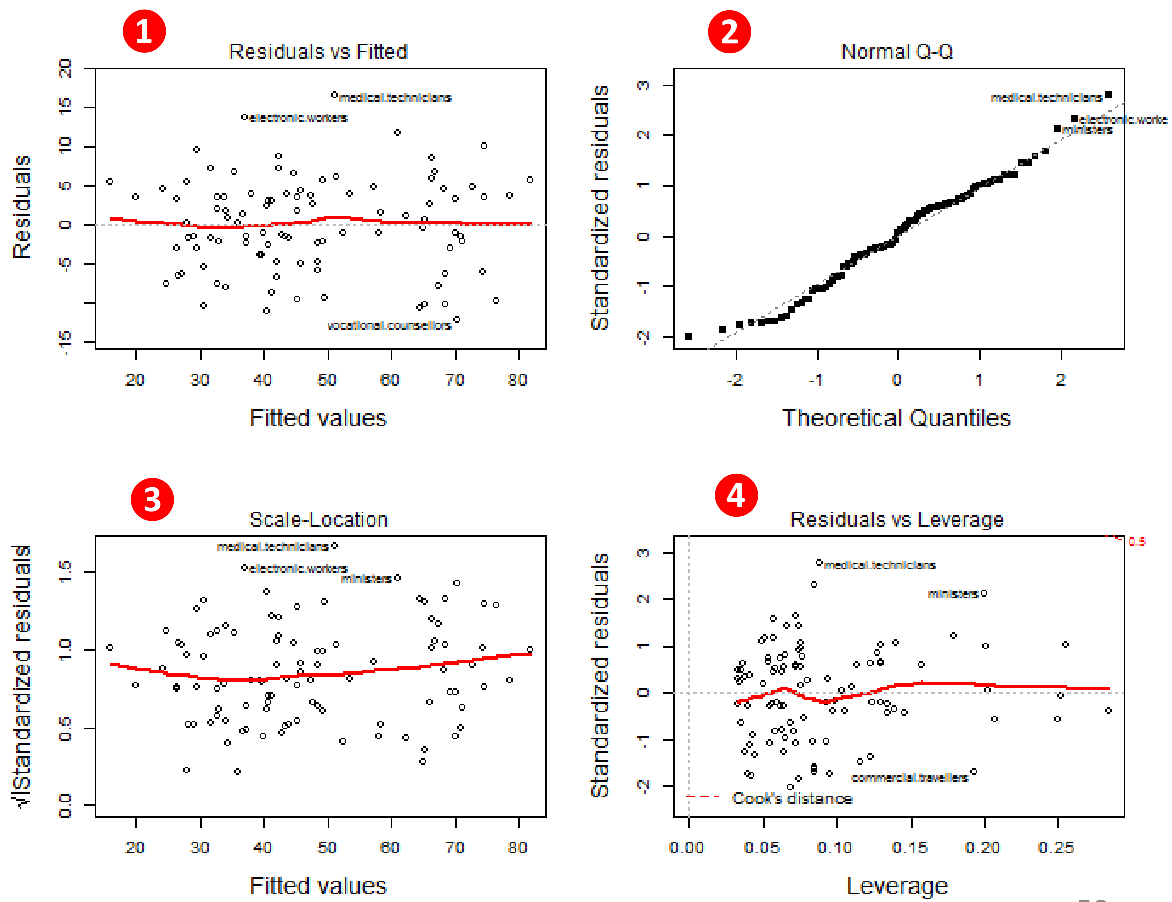
```
plot(mod1, lwd=2, cex.lab=1.4)
```

1. Residuals: should be flat vs. fitted values ✓

2. Q-Q plot: should follow the 45° line ✓

3. Scale-location: should be flat if constant variance ✓

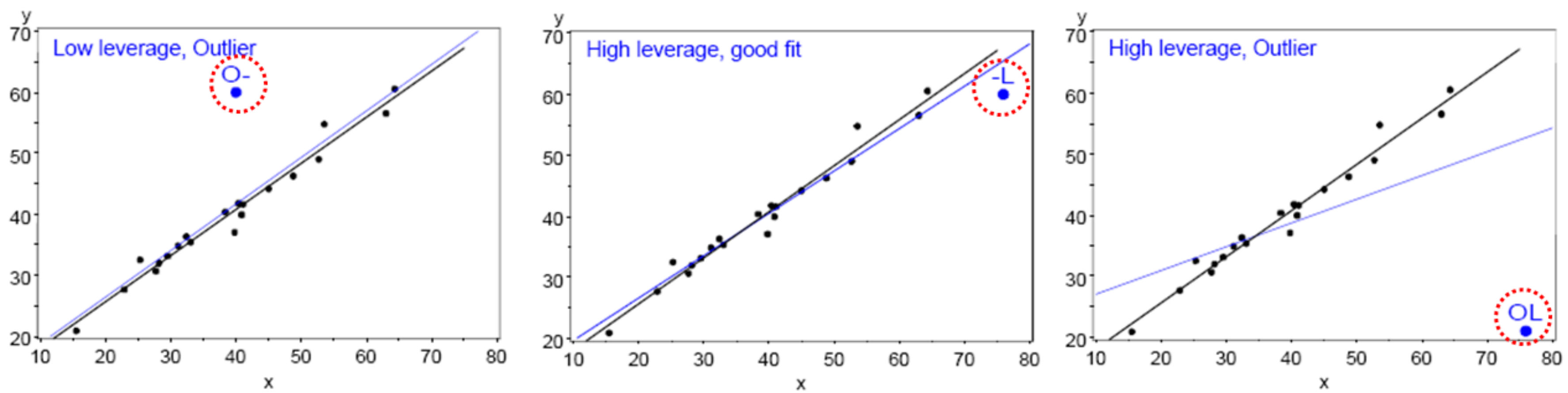
4. Resids vs. leverage: can show influential observations



Unusual data: Leverage & Influence

- "Unusual" observations can have dramatic effects on least-squares estimates in linear models
- Three archetypal cases:
  - Typical X (low leverage), bad fit -- Not much harm
  - Unusual X (high leverage), good fit -- Not much harm
  - Unusual X (high leverage), bad fit -- BAD, BAD, BAD
- Influential observations: unusual in *both* X & Y
- Heuristic formula:

$$\text{Influence} = \text{X leverage} \times \text{Y residual}$$



Influence plots

Influence (Cook's D) measures impact of individual obs. on coefficients, fitted values

Influence ~ Residual (y -ŷ) × Hat-value $(X - \bar{X})^2$

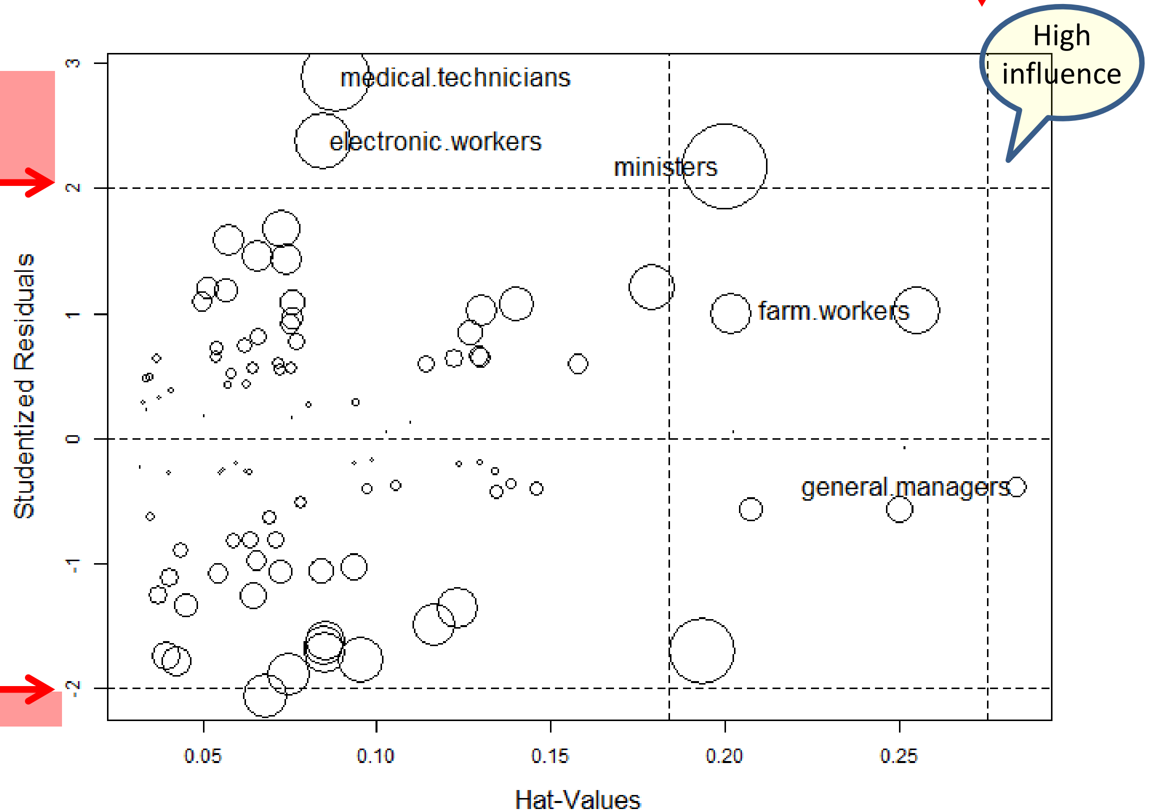
High leverage

Bubble size ~ influence

Bad fit

```
influencePlot(mod1)
```

Bad fit

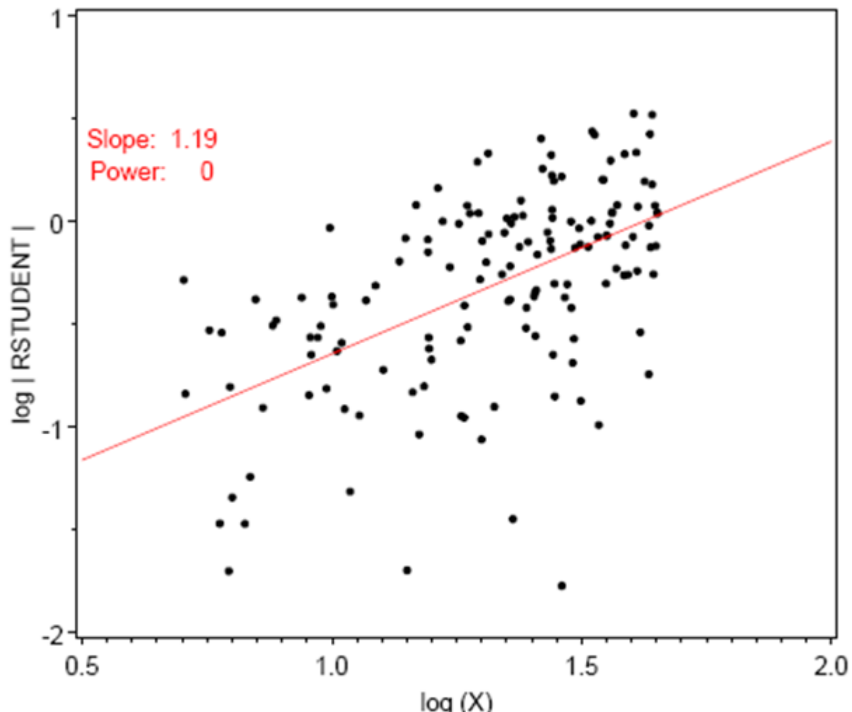


Spread-level plots

- To diagnose non-constant variance, plot:
  - log |Std. residual| vs. log (x)
  - log (IQR) vs log (median)   [for grouped data]
- If ≈ linear w/ slope b, transform $y \rightarrow y^{(1-b)}$

Artificial data, generated so σ ~ x

- b ≈ 1 → power = 0
- → analyze log(y)



Spread-level plot: baseball data

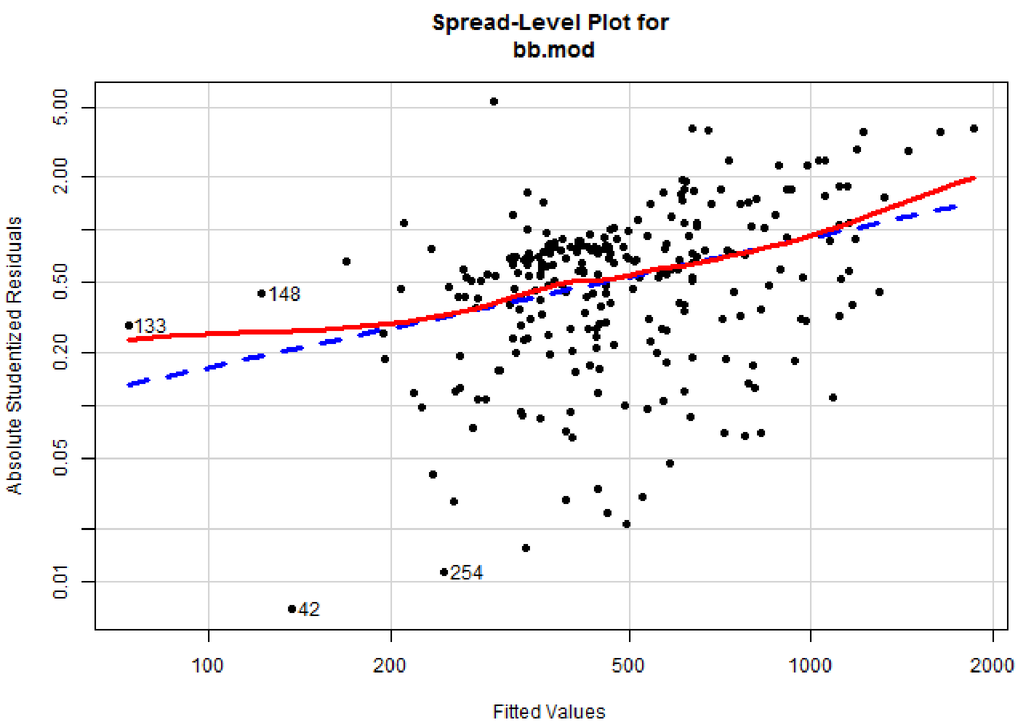
Data on salary and batter performance from 1987 season

```
data("Baseball", package="vcd")
bb.mod <- lm(sal87 ~ years + hits + runs + homeruns, data=Baseball)
spreadLevelPlot(bb.mod, pch=16, lwd=3,
                id=list(n=2))

##  Suggested power transformation: 0.2609
```

slope = .74 $\rightarrow$ p = .26

i.e., y $\rightarrow$ log(y) or $y^{1/4}$

NB: both axes plotted on log scale



Box Cox transformation

- Box & Cox proposed to transform y to a power, $y \rightarrow y^{(\lambda)}$ to minimize the residual SS (or maximize the likelihood)
  - Makes $y^{(\lambda)}$ more nearly normal
  - Makes $y^{(\lambda)}$ more nearly linear in with X

Formula for $y^{(\lambda)}$

- $y^{(0)}$: $\log_e(y)$
- $\lambda < 0$: flip sign to keep same order

$$y_i^{(\lambda)} = \begin{cases} \dfrac{y_i^\lambda - 1}{\lambda} & \text{if } \lambda \neq 0, \\ \ln(y_i) & \text{if } \lambda = 0, \end{cases}$$

Power(p)	Transformation	Name
2	Y^2	Square
1	Y (No transformation)	Original Data
½	√ Y	Square root
"0"	log Y or $\log_{10}$ (Y)	Logarithm
-½	-1 / √ Y	Reciprocal Root
-1	-1 / Y	Reciprocal
-2	-1 / Y^2	Reciprocal Square

Example: Cars93 data

How does gas mileage (MPG.city) depend on vehicle weight?

```
> cars.mod <- lm(MPG.city ~ Weight, Cars93)
> coef(cars.mod)
(Intercept)      Weight
   47.04835    -0.00803
```

Relationship clearly non-linear

Tukey arrow rule: transform Y (or X) as arrow thru the curve bulges

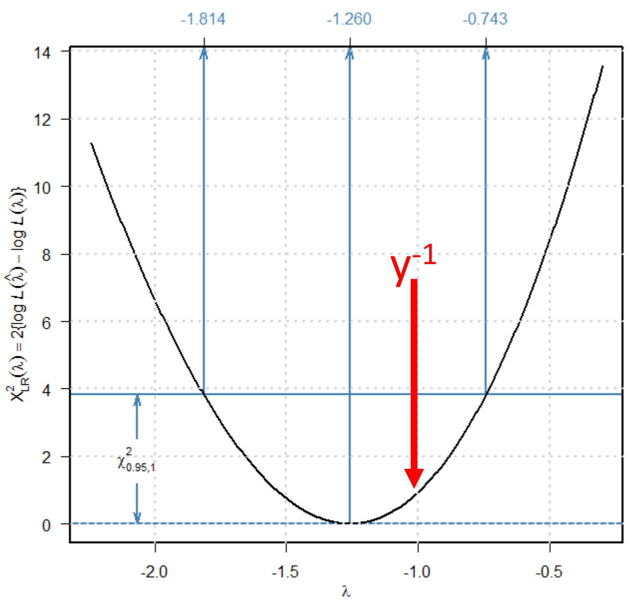
y $\rightarrow \sqrt{y}$, log(y), 1/y

x $\rightarrow \sqrt{x}$, log(x), 1/x

MASSextra package: box_cox()

```
> library(MASSExtra)
> box_cox(cars.mod)      # plot log likelihood vs. lambda
> lamba(cars.mod)
[1] -1.26
```

The plot of – log(L) ~ RSS shows the minimum & CI



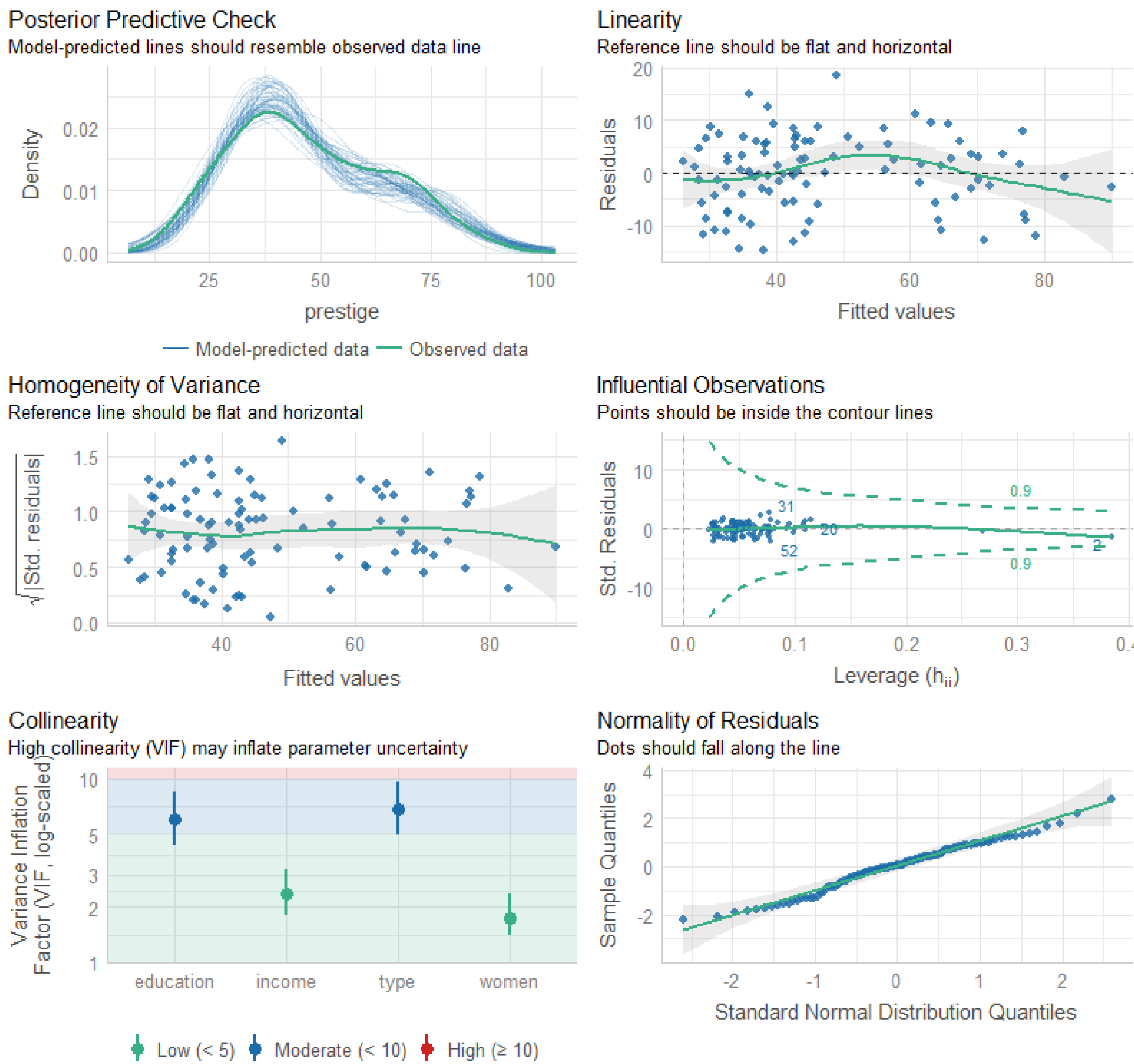
plot (bc(MPG.city, lamba(cars.mod))

easystats: performance package

```
library(performance)
check_model(mod0)
```

This package gives all the standard plots plus some others

Captions indicate what should be seen for a good model



easystats: report package

The report package generates a textual description of the results of a fitted model

```
library(report); report(mod0)
```

We fitted a linear model (estimated using OLS) to predict prestige with education (formula: prestige ~ education + income + women + type). The model explains a statistically significant and substantial proportion of variance (R2 = 0.83, F(5, 92) = 93.07, p < .001, adj. R2 = 0.83). The model's intercept, corresponding to education = 0, is at 0.18 (95% CI [-13.81, 14.18], t(92) = 0.03, p = 0.979). Within this model:

- The effect of **education** is statistically significant and positive (beta = 3.66, 95% CI [2.38, 4.95], t(92) = 5.67, p < .001; Std. beta = 0.59, 95% CI [0.38, 0.80])

- The effect of **income** is statistically significant and positive (beta = 1.04e-03, 95% CI [5.22e-04, 1.56e-03], t(92) = 3.98, p < .001; Std. beta = 0.26, 95% CI [0.13, 0.39])

- The effect of **women** is statistically non-significant and positive (beta = 6.44e-03, 95% CI [-0.05, 0.07], t(92) = 0.21, p = 0.832; Std. beta = 0.01, 95% CI [-0.10, 0.12])

- The effect of **type [linear]** is statistically non-significant and positive (beta = 4.18, 95% CI [-1.35, 9.71], t(92) = 1.50, p = 0.137; Std. beta = 0.24, 95% CI [-0.08, 0.57])

- The effect of **type [quadratic]** is statistically significant and positive (beta = 4.79, 95% CI [1.72, 7.86], t(92) = 3.10, p = 0.003; Std. beta = 0.28, 95% CI [0.10, 0.46])

Summary

- Tables are for look-up; graphs can give insight
- "Linear" models include so much more than ANOVA & regression
- Data plots are more effective when enhanced
  - data ellipses $\rightarrow$ strength & precision of correlation
  - regression lines and smoothed curves
  - point identification $\rightarrow$ noteworthy observations
- Effect plots show informative views of models
  - Visualize conditional effects, holding others constant
- Diagnostic plots can reveal:
  - violations of assumptions,
  - influential observations,
  - need for transformations.