Multiple Linear Regression in R

Fitting Models to Data Not Data to Models

Model Fitting Series - With Applications in R

Jesse Ghashti

October 16, 2025

Centre for Scholarly Communication
The University of British Columbia | Okanagan Campus | Syilx Okanagan Nation Territory

Welcome, feel free to get ready while we wait to start...

Install Required Packages

```r
packages <- c("dplyr", "mgcv", "ggplot2",
              "gratia", "daggity", "leaps", "car")
toInstall <- packages[!(packages %in%
                        installed.packages()[,"Package"])]
if(length(toInstall)) install.packages(toInstall)

library('dplyr')   # for data wrangling
library('mgcv')    # for modeling
library('ggplot2') # for fancy plots
library('daggity') # for DAGs
library('gratia')  # ggplot-based graphics
library('leaps')   # for best subset regression
library('car')     # for variance inflation factors
```

Workshop Series Overview

Session	Topic	Date/Time
1	Simple Linear Regression	Oct 7, 9:00 AM
2	Fitting Linear Models in R	Oct 8, 10:30 AM
3	**Multiple Linear Regression in R**	**Oct 16, 4:00 PM**
4	Interaction Terms & Hierarchical Linear Models	Oct 21, 11:00 AM
5	Generalized Linear Models	Oct 23, 4:00 PM
6	Generalized Additive Models (GAMs)	Oct 28, 11:00 AM
7	Interpreting & Predicting from GAMs	Oct 29, 10:30 AM
8	Hierarchical GAMs	Nov 4, 12:00 PM
9	Penalized Models	Nov 18, 11:00 AM
10	Survival Models	Nov 25, 11:00 AM
11	Nonparametric Models	Dec 2, 11:00 AM

New Here?

<————————————-

New to R? Check out the Fundamentals of R series!

GitHub code and slides for today's workshop (and previous workshops)

————————————>

Alternatively, code/slides available at the bottom of
`https://csc-ubc-okanagan.github.io/workshops/`

Last Time (Workshop 2) — Quick Recap

Key Concepts

- Built a diagnostics function to check all 5 assumptions
- Fitted models to real datasets: `ChickWeight`, `prostate`, `state.x77`
- Saw how outliers can dominate inference, skew predictions.

Today: Multiple Linear Regression in R

Today we will...

- Extend from simple ($Y = \beta_0 + \beta_1 x + \epsilon$) to multiple regression
- Use DAGs (Directed Acyclic Graphs) to visualize relationships
- Interpret coefficients when predictors are correlated
- Understand R^2_{adj} and model selection
- Work with `state.x77` data: Life expectancy modeling

Today we require...

```r
library('dplyr')    # for data wrangling
library('mgcv')     # for modeling
library('ggplot2') # for fancy plots
library('gratia')  # for ggplot-based model graphics
library('dagitty') # for drawing Directed Acyclical Graphs (DAGs)
library('leaps')    # for best subset regression
library('car')      # for variantion inflation factors
```

Multiple Linear Regression: The Model

From Simple to Multiple

Simple Linear Regression:

$$Y_i = \beta_0 + \beta_1 x_i + \epsilon_i$$

Multiple Linear Regression:

$$Y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_p x_{ip} + \epsilon_i$$

where:

- Y_i is the response for observation i
- x_{ij} is the value of predictor j for observation i
- β_j is the coefficient for predictor j
- $\epsilon_i \sim N(0, \sigma^2)$ (same assumptions as before)

Least Squares Estimates: SLR vs MLR

Simple Linear Regression

Model: $Y = \beta_0 + \beta_1 x + \epsilon$ Estimates:

$$\hat{\beta}_1 = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(Y_i - \bar{Y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2}$$

$$\hat{\beta}_0 = \bar{Y} - \hat{\beta}_1 \bar{x}$$

Multiple Linear Regression

Model: $Y = \beta_0 + \beta_1 x_1 + \cdots + \beta_p x_p + \epsilon$

Matrix form: $\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}$ Estimates:

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y},$$

where $\mathbf{X}$ is the model matrix.

Estimates for both are solved via least squares, but MLR requires matrix derivatives. In MLR, $\hat{\beta}_j$ is the effect of x_j *holding all other predictors constant*, while in SLR, $\hat{\beta}_1$ is the total effect of x on Y

Data Preparation: US States (1970s)

Preparing state.x77 data

```
states <- state.x77 %>%
  as.data.frame() %>%
  mutate(State = rownames(.)) %>% # state.x77 has state names as row names
  `rownames<-`(NULL) %>% # drop rownames
  relocate(State, .before = 1) %>% # make the States column the first one
  rename(Life_exp = `Life Exp`,
         Murder_1e5 = Murder,
         HS_grad_perc = `HS Grad`) %>%
  as_tibble()
```

Variables include Life expectancy, Income (per capita, 1974), Murder rate (per 100k, 1976), High school graduation %

Directed Acyclic Graphs (DAGs)

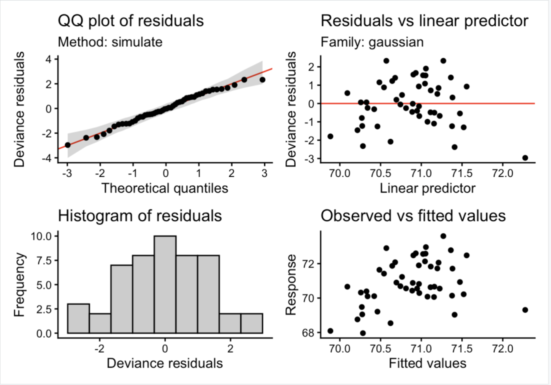
What are DAGs?

- Visual representation of causal relationships
- Arrows show direction of causation
- Help identify confounders
- Guide model specification

```r
dag_1 <- dagitty('Income -> Life_exp')
plot(dag_1)
```

Model 1: Income → Life Expectancy

```r
m_1 <- gam(Life_exp ~ Income,
           family = gaussian(),
           data = states,
           method = 'ML')
appraise(m_1, method = 'sim',
         n_simulate = 1e3)
```



Model 1: Income → Life Expectancy

```r
summary(m_1) # relatively low R^2_adj
```

Acceptable fit but relatively low R^2_{adj} — suggests we might be missing important predictors

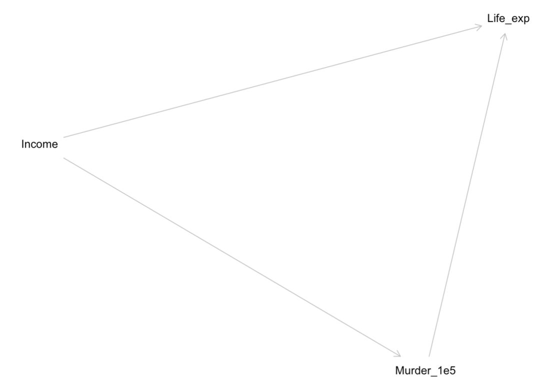
```
Formula:
Life_exp ~ Income

Parametric coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 6.758e+01  1.328e+00  50.906   <2e-16 ***
Income      7.433e-04  2.965e-04   2.507   0.0156 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1


R-sq.(adj) =  0.0974   Deviance explained = 11.6%
-ML = 82.089  Scale est. = 1.6266    n = 50
```

Model 2: Adding Murder Rate

DAG with two predictors

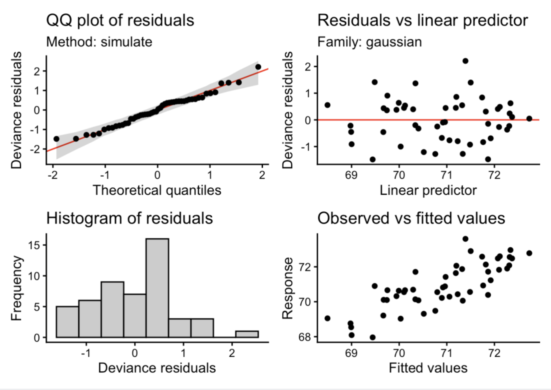
```
dag_2 <- dagitty(
  'Income -> Life_exp
   Murder_1e5 -> Life_exp
   Income -> Murder_1e5')
plot(dag_2)
```

Income affects life expectancy both directly AND indirectly through murder rate



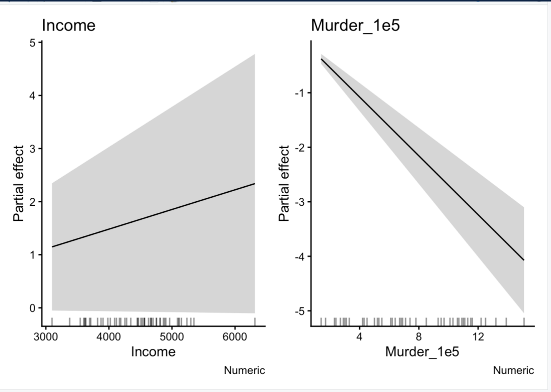
Model 2: Results I

```r
m_2 <- gam(Life_exp ~ Income + Murder_1e5,
           family = gaussian(),
           data = states,
           method = 'ML')
appraise(m_2, method = 'sim',
n_simulate = 1e3)
```



Model 2: Results II

```r
draw(m_2, parametric = TRUE)
```



Model 2: Results III

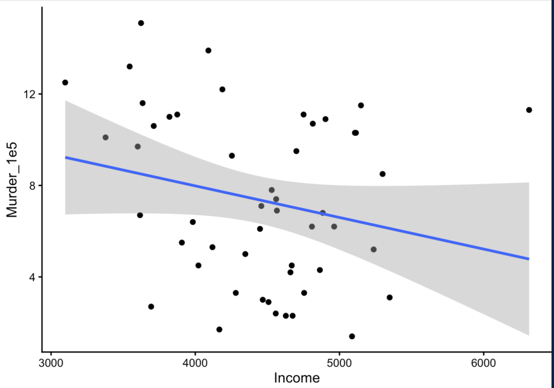
```r
summary(m_2) # relatively good R^2_adj
# the p-value for Income went up. Why?
```

```
Formula:
Life_exp ~ Income + Murder_1e5

Parametric coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 71.2255815  0.9673952  73.626  < 2e-16 ***
Income       0.0003705  0.0001973   1.878   0.0666 .
Murder_1e5  -0.2697594  0.0328408  -8.214 1.22e-10 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1


R-sq.(adj) =  0.622   Deviance explained = 63.7%
-ML = 59.834  Scale est. = 0.68205   n = 50
```

Understanding Multicollinearity

```r
# murder rate and income are correlated,
# but that's ok (see the DAG)
ggplot(states, aes(Income, Murder_1e5)) +
  geom_point() +
  geom_smooth(method = 'lm',
              formula = y ~ x)
```

When predictors are correlated, coefficient interpretation changes: each β represents the effect **holding other variables constant**

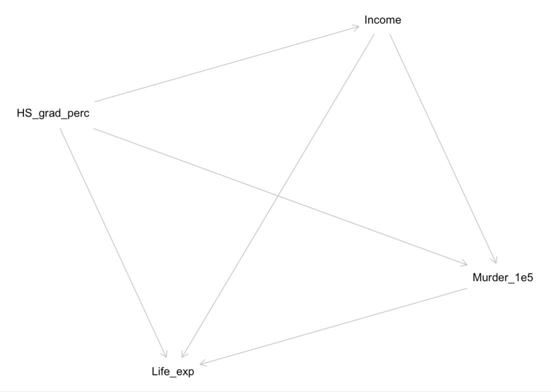


Model 3: Adding High School Graduation Rate

Complex DAG

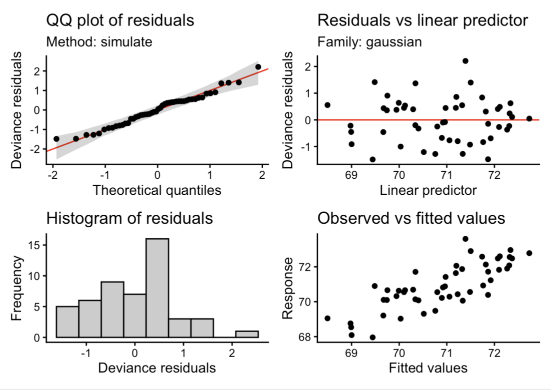
```
dag_3 <- dagitty(
  'Income -> Life_exp
   Income -> Murder_1e5
   Murder_1e5 -> Life_exp
   HS_grad_perc -> Life_exp
   HS_grad_perc -> Murder_1e5
   HS_grad_perc -> Income')
plot(dag_3)
```

Notice that HS graduation affects all other variables — a potential confounder!



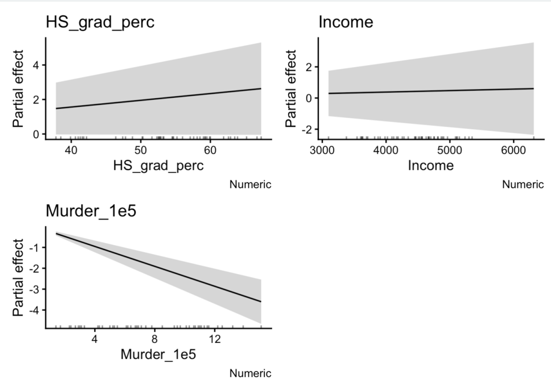
Model 3: Results I

```r
m_3 <- gam(Life_exp ~ Income + Murder_1e5
           + HS_grad_perc,
           family = gaussian(),
           data = states,
           method = 'ML')
appraise(m_3, method = 'sim',
n_simulate = 1e3)
```



Model 3: Results II

```r
draw(m_3, parametric = TRUE)
```



Model 3: Results III

```r
summary(m_3) # R^2_adj did not improve much
```

Adding more predictors doesn't always improve the model!

```
Formula:
Life_exp ~ Income + Murder_1e5 + HS_grad_perc

Parametric coefficients:
               Estimate Std. Error t value Pr(>|t|)
(Intercept)   7.014e+01  1.096e+00  63.979  < 2e-16 ***
Income        9.526e-05  2.393e-04   0.398   0.6924
Murder_1e5   -2.386e-01  3.581e-02  -6.664 2.92e-08 ***
HS_grad_perc  3.906e-02  2.030e-02   1.924   0.0605 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1


R-sq.(adj) =  0.642   Deviance explained = 66.4%
-ML = 57.898  Scale est. = 0.64496    n = 50
```

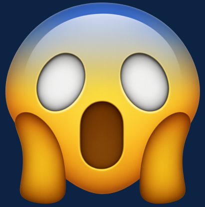
Quick Sidenote on Model Matrix $\mathbf{X}$

Recall that

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y},$$

where $\mathbf{X}$ is an $n \times (p+1)$ matrix, n is the number of observation in your data and p is the number of predictors (the $+1$ corresponds to an extra column for the intercept term β_0.

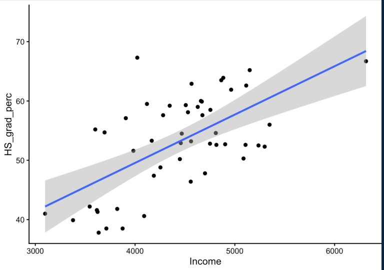
```r
model_mat <- cbind(rep(1,nrow(states)),
                   states$Income,
                   states$Murder_1e5,
                   states$HS_grad_perc)

XTX_inv <- solve(t(model_mat)%*%model_mat)
Betas <- XTX_inv%*%t(model_mat)%*%states$Life_exp
Betas # Same estimates of previous slide!
```

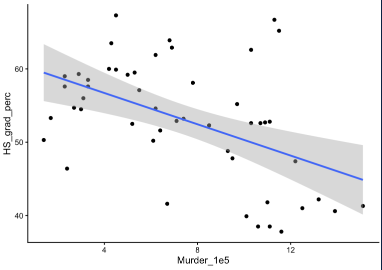


Correlation Structure

```r
# % grad and income are correlated
ggplot(states, aes(Income, HS_grad_perc)) +
  geom_point() +
  geom_smooth(method = 'lm', formula = y ~ x)
```

```r
# same for % grad and murder rate
ggplot(states, aes(Murder_1e5, HS_grad_perc)) +
  geom_point() +
  geom_smooth(method = 'lm', formula = y ~ x)
```





Caution High correlations between predictors make coefficients unstable and harder to interpret

Variance Inflation Factor (VIF)

VIFs

The VIF is given by

$$\text{VIF}_i = \frac{1}{1 - R_i^2},$$

which is used for each independent variable i in a multiple regression model. Here, R_i^2 is the R-squared value obtained from regressing that variable against all the other independent variables in the model.

```
m_3.1 <- lm(Life_exp~Income + Murder_1e5 + HS_grad_perc,
            data = states)
vif(m_3.1)
m_3.2 <- lm(Life_exp~.-State,
             data = states)
vif(m_3.2)
```

A VIF ≥ 10 indicates severe multicollinearity, consider removing that variable, whereas $5 \leq$ VIF < 10 should be investigated.

```
> vif(m_3.1)
      Income   Murder_1e5 HS_grad_perc
    1.642581     1.327381     2.041819
> vif(m_3.2)
  Population       Income   Illiteracy   Murder_1e5 HS_grad_perc
    1.499915     1.992680     4.403151     2.616472     3.134887
       Frost         Area
    2.358206     1.789764
>
```

Interpreting Multiple Regression Coefficients

Key Concepts

- **Partial effects:** Each β_j represents the effect of x_j *holding all other predictors constant*
- R^2_{adj}: Adjusts for number of predictors: $R^2_{adj} = 1 - \frac{(1-R^2)(n-1)}{n-p-1}$
- **Multicollinearity:** When predictors are correlated:
  - Standard errors increase
  - Coefficients become unstable
  - Individual p-values may be misleading

Statistical significance $\neq$ Causal effect; use DAGs to think about causality.

Model Selection: Do's and Don'ts

Good Reasons to Add Variables

- Theory/domain knowledge suggests causality
- Substantial improvement in R^2_{adj}
- Reduces omitted variable bias
- Improves predictions on new data

Bad Reasons to Add Variables

- Just to increase R^2 (not adjusted)
- "Fishing" for significance
- Without theoretical justification
- When it creates severe multicollinearity

Next time: We'll explore interaction terms — when the effect of one variable depends on another.

A Handy Little Function

```r
# part of the 'leaps' package
m_full <- regsubsets(Life_exp~.-State,
                     data = states,
                     nvmax = 5,
                     really.big = T)
summary(m_full)
best_ss <- gam(Life_exp~Murder_1e5,
               data = states,
               method = "ML")
summary(best_ss)
```

Well, turns out possibly the most obvious model is the "best" according to best subset regression

```
1 subsets of each size up to 5
Selection Algorithm: exhaustive
         Population Income Illiteracy Murder_1e5 HS_grad_perc Frost Area
1  ( 1 ) " "        " "    " "        "*"        " "          " "   " "
2  ( 1 ) " "        " "    " "        "*"        "*"          " "   " "
3  ( 1 ) " "        " "    " "        "*"        "*"          "*"   " "
4  ( 1 ) "*"        " "    " "        "*"        "*"          "*"   " "
5  ( 1 ) "*"        "*"    " "        "*"        "*"          "*"   " "
> best_ss <- gam(Life_exp~Murder_1e5, data = states, method = "ML")
> summary(best_ss)

Family: gaussian
Link function: identity

Formula:
Life_exp ~ Murder_1e5

Parametric coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 72.97356    0.26997  270.30  < 2e-16 ***
Murder_1e5  -0.28395    0.03279   -8.66 2.26e-11 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1


R-sq.(adj) =  0.602   Deviance explained =   61%
-ML = 61.642  Scale est. = 0.71794    n = 50
```

Key Takeaways

- Multiple regression extends simple regression: $Y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \epsilon$
- DAGs help visualize and reason about causal relationships
- Correlated predictors complicate interpretation but are often unavoidable
- R^2_{adj} penalizes model complexity — more predictors isn't always better
- Each coefficient represents a *partial* effect, holding others constant
- Think causally, not just statistically!

You Try: Practice Multiple Linear Regression

Exercise: Choose a dataset and fit a MLR model

```
# Option 1: mtcars - Motor Trend Car Road Tests
?mtcars  # Predict mpg by finding a good combination of predictors

# Option 2: airquality - New York Air Quality
?airquality  # Predict Ozone by finding a good combination of predictors

# Option 3: swiss - Swiss Fertility and Socioeconomic Data
?swiss  # Predict Fertility by finding a good combination of predictors

# Option 4: attitude - Chatterjee-Price Attitude Data
?attitude  # Predict rating by finding a good combination of predictors

# 1. Draw a DAG for your chosen variables
# 2. Fit the model using gam()
# 3. Check diagnostics with appraise()
# 4. Interpret coefficients and R^2_adj
# 5. Try adding/removing predictors - does it improve the model?
```

What's Next?

Additional Questions?
Book an Appointment!

Next Workshop

Interaction Terms & Hierarchical Linear Models

October 21, 11:00 AM

– When effects depend on context
– Random effects
– Mixed models