Regression Analysis using R

Dr. Kapil Kumar
Assistant Professor, Department of Statistics
Central University of Haryana, Mahendergarh, India
kapilstats@gmail.com

National Workshop on Econometrics for Research in Economics (EFRIE 2022), 25–29 April 2022
Department of Economics, Central University of Haryana, Mahendergarh
28 April 2022 · Lecture 2 of 2

Plan of the session

1. The linear model and its assumptions
2. Simple regression: price and printer sales
3. Diagnostics: a trend that fits too well
4. Multiple regression and multicollinearity
5. Testing and checking: two more case studies
6. The workflow, and take-aways

By the end you can fit a regression with `lm()`; read coefficients, standard errors, t, F and R^2; build confidence and prediction intervals; test groups of coefficients; and check linearity, constant variance, autocorrelation, normality, multicollinearity and influence.

Where we left off

Lecture 1 gave us vectors and data frames; reading and checking data; our own functions; loops and `apply()`; and a simulation showing that an OLS slope is centred on the true value with SD $\sigma/\sqrt{\sum(x_i - \bar{x})^2}$.

It ended with a question: savings fall as the share of young people rises, but by how much, and how sure can we be?

```
lcs <- LifeCycleSavings
cor(lcs$sr, lcs$pop15)
coef(lm(sr ~ pop15, data = lcs))

## [1] -0.4555
## (Intercept)       pop15
##      17.497      -0.223
```

Today: what `lm()` estimates, how to test it and how to check the assumptions behind the test. We return to these savings data at the end.

PART 1

The linear model

What regression is for

- One variable of interest, the response y (sales, consumption, wages), and p regressors $x_1, \ldots, x_p$ that may explain it
- Three uses:
  - **describe** a relationship: marginal effects, elasticities
  - **predict** y for new values of x
  - **test** a hypothesis, e.g. that price does not affect demand

Response	Regressors
quantity sold	own price, rival's price
consumption	disposable income
log wage	schooling, experience
savings ratio	age structure, growth

Association is not causation
Regression measures how y moves with x. A causal reading needs an argument that nothing omitted drives both.

The general linear regression model

$$y_i = \beta_0 + \beta_1 x_{1i} + \cdots + \beta_p x_{pi} + \varepsilon_i, \quad i = 1, \ldots, n \qquad \Longleftrightarrow \qquad \mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$$

- β_j: the change in $E(y)$ when x_j rises by one unit, *holding the other regressors fixed*
- ε_i: everything else that moves y_i
- $\mathbf{X}$ is the $n \times (p+1)$ matrix of regressors with a first column of 1s
- $p = 1$: simple linear regression

Linear in the parameters not necessarily in the variables. These are linear models:
$\ln Q = \beta_0 + \beta_1 \ln P + \varepsilon$ (constant elasticity)
$y = \beta_0 + \beta_1 t + \beta_2 t^2 + \varepsilon$ (quadratic trend)

Ordinary least squares

Minimise $S(\boldsymbol{\beta}) = \sum_{i=1}^{n} (y_i - \mathbf{x}_i'\boldsymbol{\beta})^2$:

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y}, \quad \hat{\mathbf{y}} = \mathbf{X}\hat{\boldsymbol{\beta}}, \quad \mathbf{e} = \mathbf{y} - \hat{\mathbf{y}}$$

$$\hat{\sigma}^2 = \frac{\mathbf{e}'\mathbf{e}}{n-p-1}, \quad \widehat{\mathrm{Var}}(\hat{\boldsymbol{\beta}}) = \hat{\sigma}^2(\mathbf{X}'\mathbf{X})^{-1}$$

Simple regression: $\hat{\beta}_1 = S_{xy}/S_{xx}$, $\hat{\beta}_0 = \bar{y} - \hat{\beta}_1\bar{x}$

$\mathrm{SST} = \sum (y_i - \bar{y})^2$, $\mathrm{SSE} = \sum e_i^2$

$R^2 = 1 - \mathrm{SSE}/\mathrm{SST}$

$t_j = \hat{\beta}_j / \mathrm{SE}(\hat{\beta}_j) \sim t_{n-p-1}$ under $H_0 : \beta_j = 0$

`lm()` computes all of these; we shall also compute $\hat{\boldsymbol{\beta}}$ by matrix algebra once, to see that nothing is hidden.

Assumptions of the classical linear model

Assumption	If it fails	Check in R
Linear form, $E(\varepsilon_i \mid \mathbf{X}) = 0$	biased estimates	residual plot
Constant error variance	wrong SEs, inefficient OLS	`bptest()`
Uncorrelated errors	SEs too small	`dwtest()`
No exact collinearity ($\mathbf{X}$ full rank)	no unique estimates	`vif()`
Normal errors	small-sample t, F inexact	`shapiro.test()`

Gauss–Markov Under the first four, OLS is the best linear unbiased estimator (BLUE); normality adds exact t and F tests.

Correction Regressors need *not* be uncorrelated, as the original slide said; only an *exact* linear relation is ruled out.

Packages used in this session

- Base R (**stats**): `lm()`, `summary()`, `confint()`, `predict()`, `anova()`, diagnostic plots
- **lmtest**: `bptest()` (Breusch–Pagan), `dwtest()` (Durbin–Watson)
- **car**: `vif()` (variance inflation factors)

```r
# install once
# install.packages(c("lmtest", "car"))
library(lmtest)
library(car)

col_main <- "#1F5FA8"          # graph colours
col_alt  <- "#B45309"
options(show.signif.stars = FALSE) # no stars
```

PART 2

Simple regression

Case study 1: printer price and sales

```
printers <- read.csv("data/SellingPrinters.csv")
dim(printers)
head(printers, 3)
sapply(printers[, -1], range)
```

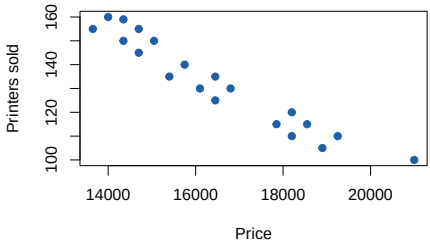
```
## [1] 20  3
##    Store Sold Price
## 1 Store1  100 21000
## 2 Store2  150 15050
## 3 Store3  130 16800
##      Sold Price
## [1,]  100 13650
## [2,]  160 21000
```

Data Illustrative data from the original lecture: units of a new printer sold last month (`Sold`) and its price (`Price`) at 20 stores; currency not stated.

The question How do sales respond to price, and which price maximises revenue?

Look first: scatter plot, covariance, correlation

```r
cov(printers$Price, printers$Sold)
cor(printers$Price, printers$Sold)
par(mar = c(4, 4, 1, 1))
plot(Sold ~ Price, data = printers, pch = 19,
     col = col_main, ylab = "Printers sold")

## [1] -37734
## [1] -0.9626
```



Correction The size of a covariance depends on the units; only its sign is informative.

Reading it $r = -0.963$: a strong, negative and roughly linear relation, with no outliers. A straight line is a fair first model.

Fitting the model with `lm()`

```
fit <- lm(Sold ~ Price, data = printers)
coef(fit)
class(fit)
names(fit)

## (Intercept)       Price
##  280.675900   -0.009007
## [1] "lm"
##  [1] "coefficients"  "residuals"     "effects"
##  [4] "rank"          "fitted.values" "assign"
##  [7] "qr"            "df.residual"   "xlevels"
## [10] "call"          "terms"         "model"
```

$$Sold_i = \beta_0 + \beta_1 Price_i + \varepsilon_i$$

- Formula `y ~ x`; `data =` names the data frame, so write `Sold ~ Price`, not `printers$Sold`
- The result has class `"lm"`: a list of 12 components (the original slide called its class `list`)

The regression output

```
s <- summary(fit)
coef(s)                                  # estimates, SEs, t values, p-values
c(R2 = s$r.squared, adj_R2 = s$adj.r.squared, sigma = s$sigma)
s$fstatistic                             # F on (1, 18) df
```

```
##               Estimate Std. Error t value  Pr(>|t|)
## (Intercept) 280.675900  9.9254848   28.28 2.275e-16
## Price        -0.009007  0.0005977  -15.07 1.195e-11
##     R2 adj_R2  sigma
## 0.9265 0.9225 5.3329
## value numdf dendf
## 227.1   1.0  18.0
```

`summary(fit)` prints all of this at once; here we take it apart, piece by piece.

Reading the output

- **Slope**: each 1,000 rise in price goes with about 9 fewer printers sold
- **Intercept** (280.7): sales at a price of zero, far outside the observed prices (13,650–21,000); it fixes the line, it has no economic meaning
- t **= -15.07**, $p = 1.2 \times 10^{-11}$: a slope this far from 0 would be very unlikely if price had no linear effect
- R^2 **= 0.927**: price accounts for 93% of the variation in sales across stores
- F **= 227.1** $= t^2$: with one regressor the F test and the t test agree

What a p-value is not It is the probability, computed *assuming H_0 is true*, of a result at least as extreme as the one observed. It is not the probability that H_0 is true, and "not significant" does not mean "accept H_0".

$\hat{\sigma} = 5.33$: a typical store is about 5 printers away from the line.

Intervals, elasticity and the best price

```r
confint(fit)                              # 95% intervals
b0 <- coef(fit)[["(Intercept)"]]
b1 <- coef(fit)[["Price"]]
b1 * mean(printers$Price) / mean(printers$Sold)
-b0 / (2 * b1)                            # revenue peak
```

```
##                 2.5 %     97.5 %
## (Intercept) 259.82323 301.528570
## Price        -0.01026  -0.007751
## [1] -1.123
## [1] 15581
```

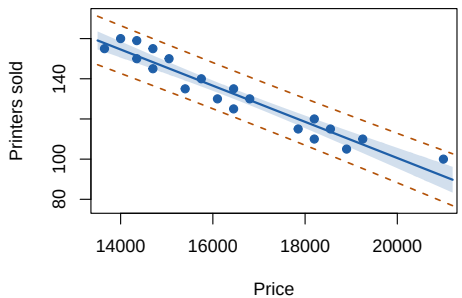
Elasticity at the means:
$\eta = \hat{\beta}_1 \, \bar{x}/\bar{y}$
Revenue $R(p) = p(\beta_0 + \beta_1 p)$
peaks at $p^* = -\beta_0/(2\beta_1)$

Reading it 1,000 more on the price costs 7.8 to 10.3 sales (95% CI). $\eta = -1.12$: elastic. Revenue peaks near 15,581, inside the data.

Prediction: the mean versus a single store

```
new <- data.frame(Price = c(15000, 17500))
predict(fit, new, interval = "confidence")
predict(fit, new, interval = "prediction")

##      fit   lwr   upr
## 1 145.6 142.5 148.7
## 2 123.1 120.2 125.9
##      fit   lwr   upr
## 1 145.6 133.9 157.2
## 2 123.1 111.5 134.6
```

Reading it At 15,000: *mean* sales 142.5–148.7, *one* store 133.9–157.2.



Shaded: 95% confidence band for the mean; dashed: 95% prediction band.

Extractor functions and the ANOVA table

Function	Returns
`coef()`	estimates
`fitted()`	fitted values
`residuals()`	residuals
`confint()`	confidence intervals
`vcov()`	covariance matrix
`predict()`	predictions
`anova()`	F tests
`deviance()`	SSE
`AIC()`	information criterion
`plot()`	diagnostic plots

```
anova(fit)

## Analysis of Variance Table
##
## Response: Sold
##           Df Sum Sq Mean Sq F value  Pr(>F)
## Price      1   6457    6457     227 1.2e-11
## Residuals 18    512      28
```

Reading it Total variation splits into a part explained by price (6457) and a residual part (512): $R^2 = 6457/6969$.

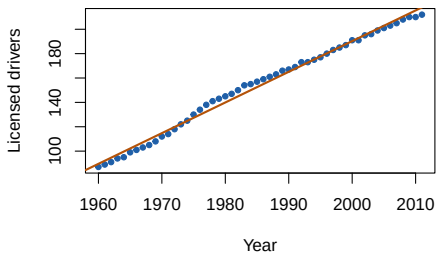
PART 3

Diagnostics

Case study 2: a trend that fits too well

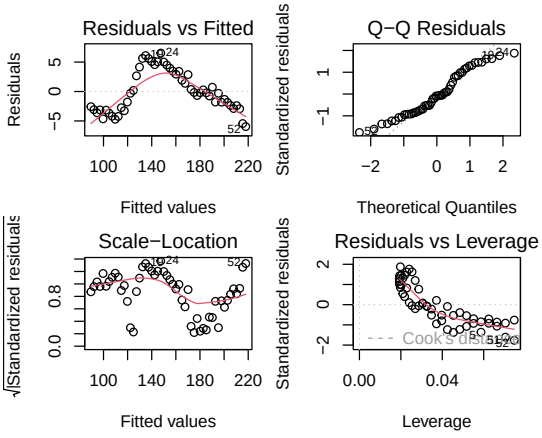
```
drivers <- read.csv("data/LicensedDrivers.csv")
trend <- lm(LicensedDrivers ~ Year, data = drivers)
coef(summary(trend))
summary(trend)$r.squared

##              Estimate Std. Error t value  Pr(>|t|)
## (Intercept) -4844.307   64.44641  -75.17 4.257e-53
## Year            2.517    0.03246   77.56 9.027e-54
## [1] 0.9918
```

Illustrative annual series, 1960–2011, from the original lecture; source and units not stated.



$R^2 = 0.9918$ and a slope of 2.52 a year. Is the model therefore right? Look at the residuals.

The four standard diagnostic plots




```r
par(mfrow = c(2, 2),
    mar = c(4, 4, 2, 1))
plot(trend)
```

What each panel checks

Residuals vs fitted: linearity
Normal Q-Q: normality
Scale-location: constant variance
Leverage plot: influence
Numbered points: most extreme rows

Reading the diagnostic plots

Plot	A good model shows	Here
Residuals vs fitted	flat cloud around 0 (linearity)	an arch: the trend is curved
Normal Q-Q	points on the line (normality)	close, with light tails
Scale-location	flat band (constant variance)	a wave following the arch
Residuals vs leverage	no point beyond Cook's 0.5 (influence)	no influential year

Lesson $R^2 = 0.992$ did not reveal the curvature; the residual plot did. A high R^2 is common for any trending series and is not evidence that the model is adequate.

Autocorrelated errors: the Durbin–Watson test

```
dw <- dwtest(trend)
round(dw$statistic, 3); dw$p.value
e <- residuals(trend)
cor(e[-1], e[-length(e)])        # lag-1 r

##    DW
## 0.105
## [1] 9.292e-30
## [1] 0.946
```

The test needs a natural (time) order; for the 20 stores of case study 1 it would be meaningless.

$$d = \frac{\sum_{t=2}^{n}(e_t - e_{t-1})^2}{\sum_{t=1}^{n} e_t^2} \approx 2(1 - \hat{\rho})$$

$d \approx 2$: no autocorrelation; $d \to 0$: strong positive autocorrelation

Reading it $d = 0.11$, lag-1 $r = 0.95$: each year's error carries into the next, so the SEs are far too small.

A quadratic trend

```
drivers$t <- drivers$Year - 1960        # centred
quad <- lm(LicensedDrivers ~ t + I(t^2), drivers)
coef(summary(quad))
anova(trend, quad)[2, c("F", "Pr(>F)")]
dwtest(quad)$statistic

##             Estimate Std. Error t value  Pr(>|t|)
## (Intercept) 83.52209   0.786298  106.22 1.364e-59
## t            3.24404   0.071300   45.50 9.520e-42
## I(t^2)      -0.01425   0.001352  -10.54 3.416e-14
##     F  Pr(>F)
## 2 111 3.4e-14
##     DW
## 0.3239
```

- `I()` protects the arithmetic: `I(t^2)` is the square of t
- `anova(small, big)` compares nested models

Reading it The negative t^2 term is significant ($F = 111.1$): growth is slowing. But $d = 0.32$ still: model the changes, add a lagged y, or use Newey–West SEs.

Why centre the year? A first look at VIF

```
raw <- lm(LicensedDrivers ~ Year + I(Year^2),
          data = drivers)
vif(raw)
vif(quad)
cor(drivers$Year, drivers$Year^2)
all.equal(fitted(raw), fitted(quad))

##      Year I(Year^2)
##     87606     87606
##      t I(t^2)
##  15.45  15.45
## [1] 1
## [1] TRUE
```

$$\text{VIF}_j = \frac{1}{1 - R_j^2}$$

R_j^2: from regressing x_j on the other regressors

Reading it Year and Year^2 are almost perfectly correlated. Centring cuts the VIFs to 15.5 with *identical* fits. $\sqrt{\text{VIF}}$ is the SE inflation factor.

PART 4

Multiple regression

Case study 3: Hald's cement data

```
cement <- read.csv("data/CementData.csv")
names(cement) <- c("x1", "x2", "x3", "x4",
                   "heat")
head(cement, 4)
rowSums(cement[, 1:4])      # % add to ~100

##   x1 x2 x3 x4  heat
## 1  7 26  6 60  78.5
## 2  1 29 15 52  74.3
## 3 11 56  8 20 104.3
## 4 11 31  8 47  87.6
##  [1] 99 97 95 97 98 97 97 98 96 98 98 98 98
```

Data The classic textbook data of Hald (1952), also in Montgomery, Peck and Vining: 13 batches of cement; `x1`–`x4` are the percentages by weight of four clinker compounds, `heat` the heat evolved (calories per gram) as the cement hardens.

Because the four percentages add to between 95 and 99, each is almost determined by the other three.

Least squares by matrix algebra

```
X <- cbind(1, as.matrix(cement[, 1:4]))
y <- cement$heat
beta <- solve(t(X) %*% X) %*% t(X) %*% y
full <- lm(heat ~ x1 + x2 + x3 + x4, data = cement)
round(cbind(matrix = drop(beta), lm = coef(full)), 4)

##     matrix      lm
##    62.4054 62.4054
## x1  1.5511  1.5511
## x2  0.5102  0.5102
## x3  0.1019  0.1019
## x4 -0.1441 -0.1441
```

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y}$$

- `cbind(1, ...)` adds the column of 1s for the intercept
- `%*%` is matrix product, `t()` transpose, `solve()` inverse
- `lm()` gives the same numbers, using a numerically safer QR decomposition

A puzzle: a significant model without a significant regressor

```
cs <- summary(full); coef(cs)
f <- cs$fstatistic                    # F, df1, df2
pval <- pf(f[[1]], f[[2]], f[[3]], lower.tail = FALSE)
round(c(R2 = cs$r.squared, F = f[[1]]), 3); pval

##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  62.4054    70.0710  0.8906  0.39913
## x1            1.5511     0.7448  2.0827  0.07082
## x2            0.5102     0.7238  0.7049  0.50090
## x3            0.1019     0.7547  0.1350  0.89592
## x4           -0.1441     0.7091 -0.2032  0.84407
##      R2       F
##   0.982 111.479
## [1] 0.0000004756
```

Reading it Jointly the four compounds explain 98.2% of the variation in heat, and the overall F test is decisive ($p = 4.8 \times 10^{-7}$). Yet no single coefficient is significant at 5%, and the SEs are as large as the estimates.

This is the signature of multicollinearity.

Diagnosing multicollinearity

```
round(cor(cement[, 1:4]), 2)
round(vif(full), 1)

##       x1    x2    x3    x4
## x1  1.00  0.23 -0.82 -0.25
## x2  0.23  1.00 -0.14 -0.97
## x3 -0.82 -0.14  1.00  0.03
## x4 -0.25 -0.97  0.03  1.00
##    x1    x2    x3    x4
##  38.5 254.4  46.9 282.5
```

Reading it `x2` and `x4` correlate at -0.97. All VIFs exceed 10, the usual warning level; for `x4` collinearity inflates the standard error about 17-fold.

What it does No bias, no harm to predictions within the data; but separate effects become imprecise.

A simpler model and the partial F test

```
reduced <- lm(heat ~ x1 + x2, data = cement)
coef(summary(reduced))
vif(reduced)
anova(reduced, full)[2, c("F", "Pr(>F)")]

##             Estimate Std. Error t value  Pr(>|t|)
## (Intercept)  52.5773    2.28617   23.00 5.457e-10
## x1            1.4683    0.12130   12.10 2.692e-07
## x2            0.6623    0.04585   14.44 5.029e-08
##    x1    x2
## 1.055 1.055
##      F Pr(>F)
## 2 0.84   0.47
```

$$F = \frac{(\mathrm{SSE}_R - \mathrm{SSE}_F)/q}{\mathrm{SSE}_F/(n-p-1)} \sim F_{q,\,n-p-1} \text{ under } H_0$$

q coefficients dropped; R reduced, F full model.

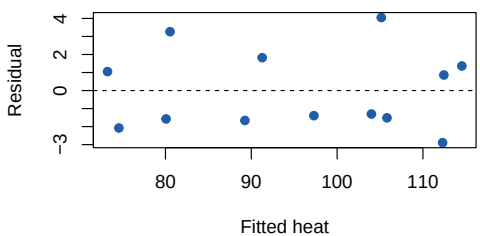
Reading it Dropping `x3`, `x4` costs nothing ($p = 0.47$); the two slopes are precise, VIFs near 1, adjusted R^2 0.9744 vs 0.9736.

R^2 by hand, and the residuals

```r
e <- residuals(reduced)
sst <- sum((y - mean(y))^2)
1 - sum(e^2) / sst
summary(reduced)$r.squared

## [1] 0.9787
## [1] 0.9787
```

$R^2 = 1 - \mathrm{SSE}/\mathrm{SST}$ reproduces `summary()`. With 13 batches, residual plots reveal only gross problems; none is visible here.

```r
par(mar = c(4, 4, 1, 1))
plot(fitted(reduced), e, pch = 19,
     col = col_main, xlab = "Fitted heat",
     ylab = "Residual")
abline(h = 0, lty = 2)
```



PART 5

Testing and checking

Case study 4: escaping hydrocarbons

```
hc <- read.csv("data/EscapingHydrocarbons.csv")
names(hc) <- c("TankTemp", "PetrolTemp", "TankPres", "PetrolPres", "HC")
round(cor(hc), 2)
```

```
##            TankTemp PetrolTemp TankPres PetrolPres   HC
## TankTemp       1.00       0.77     0.96       0.93 0.83
## PetrolTemp     0.77       1.00     0.78       0.84 0.91
## TankPres       0.96       0.78     1.00       0.99 0.87
## PetrolPres     0.93       0.84     0.99       1.00 0.92
## HC             0.83       0.91     0.87       0.92 1.00
```

Data The classic "sniffer" data (Weisberg, *Applied Linear Regression*), 32 fillings: hydrocarbon vapour escaping (HC) against tank and petrol temperature and pressure.

The full model

```
hfit <- lm(HC ~ ., data = hc)      # . = all others
coef(summary(hfit))
summary(hfit)$r.squared

##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.01502    1.86131  0.5453 0.590007
## TankTemp    -0.02861    0.09060 -0.3158 0.754608
## PetrolTemp   0.21582    0.06772  3.1870 0.003615
## TankPres    -4.32005    2.85097 -1.5153 0.141319
## PetrolPres   8.97489    2.77263  3.2370 0.003190
## [1] 0.9261
```

Reading it $R^2 = 0.926$. Petrol temperature and pressure are significant. Tank pressure has a *negative* sign although it correlates 0.87 with HC: it cannot be separated from petrol pressure ($r = 0.985$).

Which regressors can go? VIF and partial F tests

```
round(vif(hfit), 1)
h2 <- update(hfit, . ~ . - TankTemp)  # refit
anova(h2, hfit)[2, c("F", "Pr(>F)")]

##   TankTemp PetrolTemp   TankPres PetrolPres
##       13.0        4.7       71.3       61.9
##     F Pr(>F)
## 2 0.1   0.75
```

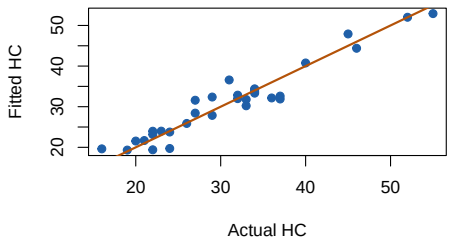
Correction With VIF 13, a large p means imprecision, not irrelevance: test a drop, don't just delete.

```
h3 <- lm(HC ~ PetrolTemp + PetrolPres,
         data = hc)
anova(h3, hfit)[2, c("F", "Pr(>F)")]

##      F Pr(>F)
## 2 2.49    0.1
```

Reading it Drop tank temperature: $p = 0.75$; drop both tank variables: $p = 0.102$, so the two-variable model is adequate at 5%.

Actual against fitted values

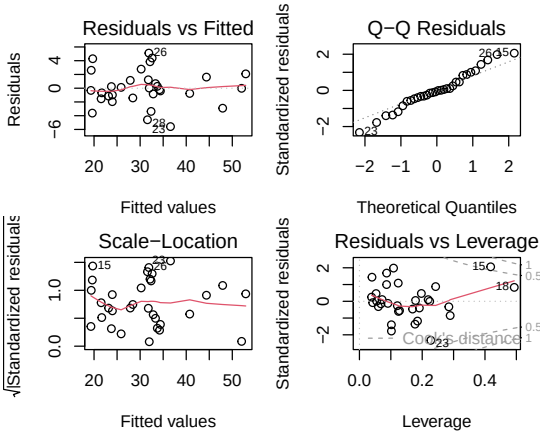
```
par(mar = c(4, 4, 1, 1))
plot(hc$HC, fitted(hfit), pch = 19,
     col = col_main, xlab = "Actual HC",
     ylab = "Fitted HC")
abline(0, 1, col = col_alt, lwd = 2)
```



- Points on the 45-degree line are fitted exactly
- Vertical distance from the line is the residual

Reading it A close fit across the range. Whether the assumptions hold is the next question.

Diagnostic plots for the full model




```r
par(mfrow = c(2, 2),
    mar = c(4, 4, 2, 1))
plot(hfit)
```

Reading it No curvature in the residuals, and the Q-Q points follow the line. The spread looks roughly even, but a formal test is more sensitive (next slide). Observation 15 lies beyond Cook's distance 0.5.

Formal tests: variance, normality, influence

```
bp <- bptest(hfit)                # Breusch-Pagan
c(bp$statistic, p = bp$p.value)
sw <- shapiro.test(residuals(hfit))
c(sw$statistic, p = sw$p.value)
cd <- cooks.distance(hfit)
round(sort(cd, decreasing = TRUE)[1:3], 3)

##       BP     p.BP
## 11.33843  0.02301
##      W      p
## 0.9785 0.7539
##    15    23    18
## 0.609 0.319 0.134
```

Reading it BP $p = 0.023$: the variance changes with the regressors, so the usual SEs are doubtful. SW $p = 0.75$: normality is fine. Observation 15 has the largest Cook's distance (0.61): check it.

Remedies robust (HC) SEs, weighted least squares, or a transformed HC.

Back to the savings data: a multiple regression

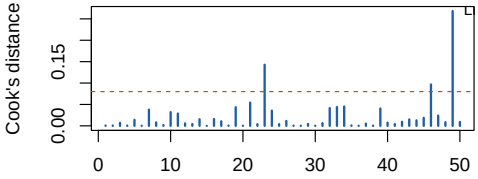
```
sav <- lm(sr ~ pop15 + pop75 + dpi + ddpi,
          data = lcs)
coef(summary(sav))
round(vif(sav), 2)
```

```
##               Estimate Std. Error t value  Pr(>|t|)
## (Intercept) 28.5660865  7.3545161  3.8842 0.0003338
## pop15       -0.4611931  0.1446422 -3.1885 0.0026030
## pop75       -1.6914977  1.0835989 -1.5610 0.1255298
## dpi         -0.0003369  0.0009311 -0.3618 0.7191732
## ddpi         0.4096949  0.1961971  2.0882 0.0424711
## pop15 pop75   dpi   ddpi
##  5.94  6.63  2.88  1.07
```

Reading it Other things equal, one point more of population under 15 goes with a savings ratio 0.46 points lower; one point faster income growth with 0.41 higher. $R^2 = 0.34$; VIFs below 10. (Simple slope of pop15: -0.22.)

Influence: one country can move a coefficient

```
cds <- cooks.distance(sav)
round(sort(cds, decreasing = TRUE)[1:3], 3)
4 / nrow(lcs)                    # rule of thumb
sav2 <- update(sav,
          subset = rownames(lcs) != "Libya")
coef(summary(sav2))["ddpi", ]

##  Libya  Japan Zambia
##  0.268  0.143  0.097
## [1] 0.08
##   Estimate Std. Error    t value   Pr(>|t|)
##    0.61028    0.26878    2.27057    0.02812
```

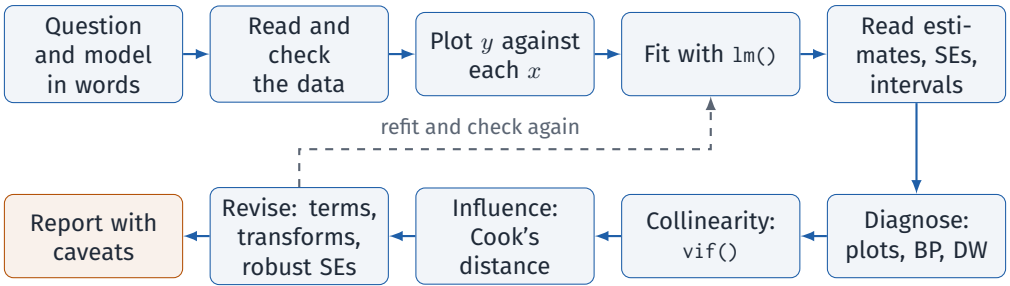


Reading it Libya (income growth 16.71%, the fastest grower in Lecture 1's list) has Cook's distance 0.27. Without it the growth effect rises from 0.41 to 0.61. Report both.

PART 6

Workflow

The regression workflow



Key take-aways

1. **Plot before fitting**: the scatter plot told us a line was reasonable for the printers, and the residual plot that it was not for the trend.
2. **Read every number**: a slope with its interval, t and p correctly interpreted, R^2 as a share of variation, F for groups of coefficients.
3. **Translate into economics**: elasticities, revenue-maximising prices, ceteris paribus effects.
4. **A high R^2 proves nothing**; check residuals, autocorrelation (DW) and heteroscedasticity (BP).
5. **Multicollinearity** inflates SEs and can flip signs: use VIFs and partial F tests, not single p-values, to simplify a model.
6. **Influential observations** deserve a sentence in the report, not silent deletion.

References

- Montgomery, D. C., Peck, E. A., and Vining, G. G. (2021). *Introduction to Linear Regression Analysis*, 6th Ed., Wiley, Hoboken.
- Fox, J., and Weisberg, S. (2019). *An R Companion to Applied Regression*, 3rd Ed., Sage, Thousand Oaks.
- Faraway, J. J. (2014). *Linear Models with R*, 2nd Ed., CRC Press, Boca Raton.
- Weisberg, S. (2014). *Applied Linear Regression*, 4th Ed., Wiley, Hoboken.
- Breusch, T. S., and Pagan, A. R. (1979). A simple test for heteroscedasticity and random coefficient variation. *Econometrica*, 47(5), 1287–1294.
- Durbin, J., and Watson, G. S. (1950). Testing for serial correlation in least squares regression I. *Biometrika*, 37(3/4), 409–428.
- Shapiro, S. S., and Wilk, M. B. (1965). An analysis of variance test for normality (complete samples). *Biometrika*, 52(3/4), 591–611.
- R Core Team (2022). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna. `https://www.R-project.org`

Thank you

Questions and discussion