A Three Day Course in Applied Regression Analysis

[2] Multiple Linear Regression, Parameter Inference, Residual Analysis and Transformations

Matt Taddy, University of Chicago Booth School of Business

faculty.chicagobooth.edu/matt.taddy/teaching

The MLR Model

In general, the MLR model is same as always,
but with more covariates.

$$Y|X_1 \ldots X_d \overset{ind}{\sim} N(\beta_0 + \beta_1 X_1 \ldots + \beta_d X_d, \sigma^2)$$

Recall the key assumptions of our linear regression model:

1. The conditional mean of Y is linear in the X_j variables.
2. The additive errors (deviations from line)
   - are normally distributed
   - independent from each other
   - identically distributed (i.e., they have constant variance)

The MLR Model

Our interpretation of regression coefficients can be extended from the simple single covariate regression case:

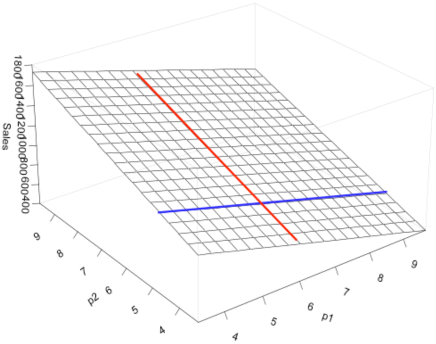
$$\beta_j = \frac{\partial \mathbb{E}[Y | X_1, \ldots, X_d]}{\partial X_j}$$

Holding all other variables constant, β_j is the average change in Y per unit change in X_j.

The MLR Model

If $d = 2$, we can plot the regression surface in 3D.

Consider sales of a product as predicted by price of this product (P1) and the price of a competing product (P2).



Sales = 1 – 1.0*P1 + 1.1*P2

hold P2 fixed and vary P1

hold P1 fixed and vary P2

Everything measured on log-scale, of course.

The Data and Least Squares

The data in multiple regression is a set of points with values for output Y and for each input variable.

Data: Y_i and $\mathbf{x}_i = [X_{1i}, X_{2i}, \ldots, X_{di}]$, for $i = 1, \ldots, n$.

Or, as a data array (i.e., `data.frame`),

$$\text{Data} = \begin{bmatrix} Y_1 & X_{11} & X_{21} & \ldots & X_{d1} \\ Y_2 & X_{12} & X_{22} & \ldots & X_{d2} \\ & & \vdots & & \\ Y_n & X_{1n} & X_{2n} & \ldots & X_{dn} \end{bmatrix}$$

The Data and Least Squares

$$Y = \beta_0 + \beta_1 X_1 \ldots + \beta_d X_d + \varepsilon, \quad \varepsilon \sim N(0, \sigma^2)$$

How do we estimate the MLR model parameters?

The principle of Least Squares is exactly the same as before:

- Define the fitted values
- Find the best fitting plane by minimizing the sum of squared residuals.

The Data and Least Squares

Fitted Values: $\hat{Y}_i = b_0 + b_1 X_{1i} + b_2 X_{2i} \ldots + b_d X_{di}$.

Residuals: $e_i = Y_i - \hat{Y}_i$.

Standard Error: $s = \sqrt{\frac{\sum_{i=1}^n e_i^2}{n-p}}$, where $p = d + 1$.

Least Squares: Find $b_0, b_1, b_2, \ldots, b_d$ to minimize s^2.

The Data and Least Squares

What are the LS coefficient values? Say that

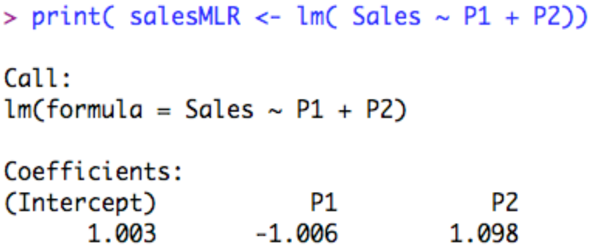
$$\mathbf{Y} = \begin{bmatrix} Y_1 \\ \vdots \\ Y_n \end{bmatrix} \quad \hat{\mathbf{X}} = \begin{bmatrix} 1 & X_{11} & X_{12} & \dots & X_{1d} \\ & & \vdots & & \\ 1 & X_{n1} & X_{n2} & \dots & X_{nd} \end{bmatrix}$$

Then the estimates are $[b_0 \cdots b_d]' = \mathbf{b} = (\hat{\mathbf{X}}'\hat{\mathbf{X}})^{-1}\hat{\mathbf{X}}'\mathbf{Y}$.

Same intuition as for SLR: **b** captures the covariance between X_j and Y ($\hat{\mathbf{X}}'\mathbf{Y}$), normalized by input sum of squares ($\hat{\mathbf{X}}'\hat{\mathbf{X}}$).

The Data and Least Squares

Obtaining these estimates in R is very easy:



```
> print( salesMLR <- lm( Sales ~ P1 + P2))

Call:
lm(formula = Sales ~ P1 + P2)

Coefficients:
(Intercept)           P1           P2
      1.003       -1.006        1.098
```

Fitted Model: $Sales_i = b_0 + b_1 P1_i + b_2 P2_i + e_i$.

Multiple vs Simple Linear Regression

Basic concepts and techniques translate directly from SLR.

- Individual parameter inference and estimation is the same, conditional on the rest of the model.
- ANOVA and sums of squares are exactly the same.
- Assumptions about the errors ε are unchanged.

The hardest part would be moving to matrix algebra to translate all of our equations.

Luckily, software does all that for you.

Residual Standard Error

First off, the calculation for $s^2 = \mathrm{var}(e)$ is exactly the same:

$$s^2 = \frac{\sum_{i=1}^{n} (Y_i - \hat{Y})^2}{n - p}$$

- $\hat{Y}_i = b_0 + \sum b_j X_{ji}$ and $p = d + 1$.
- The residual standard error is $\hat{\sigma} = s = \sqrt{s^2}$.

Residuals in MLR

As in the SLR model, the residuals in multiple regression are purged of any relationship to the independent variables.

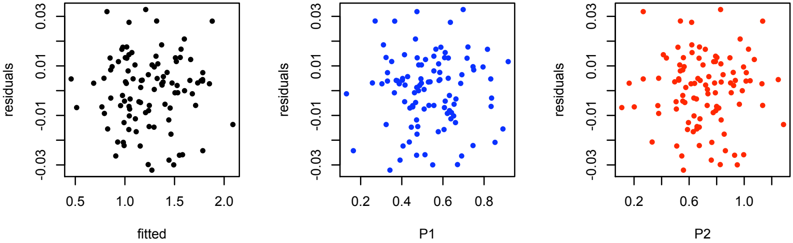
We decompose Y into the part predicted by X and the part due to idiosyncratic error.

$$Y = \hat{Y} + e$$

$$\text{corr}(X_j, e) = 0 \quad \text{corr}(\hat{Y}, e) = 0$$

Regression Diagnostics for MLR

Consider the residuals from the Sales data:



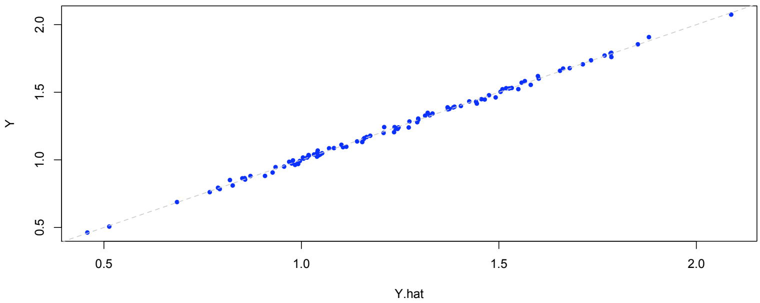
We use the same residual diagnostics (scatterplots, QQ, etc).

- Look at residuals (raw or student) against $\hat{Y}$ to see overall fit.
- Compare e or r against individual X's to identify problems.

Regression Diagnostics for MLR

Another informative plot for MLR problems is to look at Y (true values) against $\hat{Y}$ (fitted values).

Fitted vs True Response for Sales data



If things are working, these values should form a nice straight line.

[Regression is all about finding a few great scatterplots!]

Residuals and the Model Assumptions

Inference and prediction relies on this model being true!

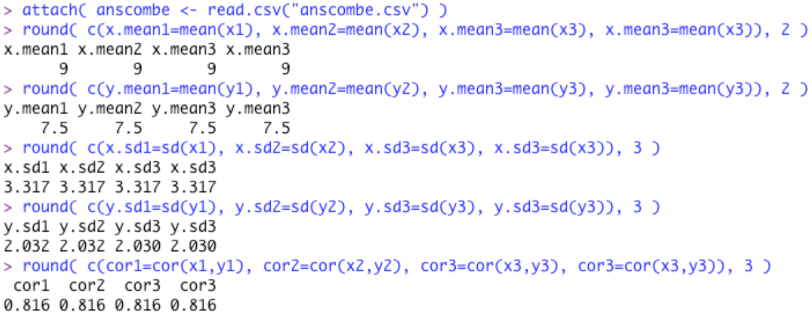
If the model assumptions do not hold, then all bets are off:

- prediction can be systematically biased
- standard errors, intervals, and t-tests are wrong

We will focus on using graphical methods (plots!) to detect violations of the model assumptions.

Example Model Violations

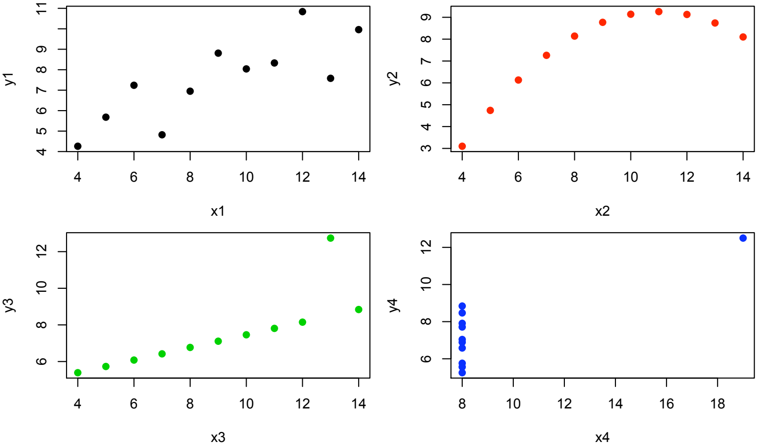
Anscombe's quartet comprises four datasets that have similar statistical properties...



```
> attach( anscombe <- read.csv("anscombe.csv") )
> round( c(x.mean1=mean(x1), x.mean2=mean(x2), x.mean3=mean(x3), x.mean3=mean(x3)), 2 )
x.mean1 x.mean2 x.mean3 x.mean3 
      9       9       9       9 
> round( c(y.mean1=mean(y1), y.mean2=mean(y2), y.mean3=mean(y3), y.mean3=mean(y3)), 2 )
y.mean1 y.mean2 y.mean3 y.mean3 
    7.5     7.5     7.5     7.5 
> round( c(x.sd1=sd(x1), x.sd2=sd(x2), x.sd3=sd(x3), x.sd3=sd(x3)), 3 )
x.sd1 x.sd2 x.sd3 x.sd3 
3.317 3.317 3.317 3.317 
> round( c(y.sd1=sd(y1), y.sd2=sd(y2), y.sd3=sd(y3), y.sd3=sd(y3)), 3 )
y.sd1 y.sd2 y.sd3 y.sd3 
2.032 2.032 2.030 2.030 
> round( c(cor1=cor(x1,y1), cor2=cor(x2,y2), cor3=cor(x3,y3), cor3=cor(x3,y3)), 3 )
 cor1  cor2  cor3  cor3 
0.816 0.816 0.816 0.816 
```

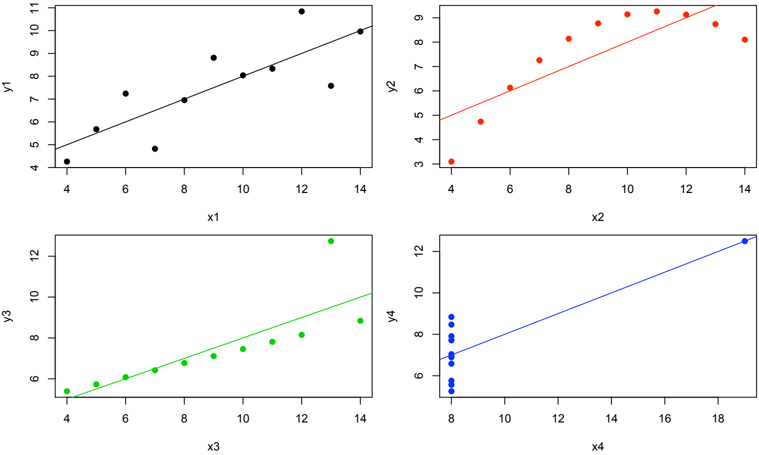
Example Model Violations

...but vary considerably when graphed.



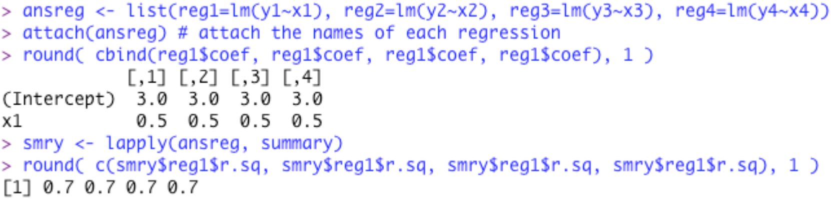
Example Model Violations

Similarly, let's consider linear regression for each dataset.



Example Model Violations

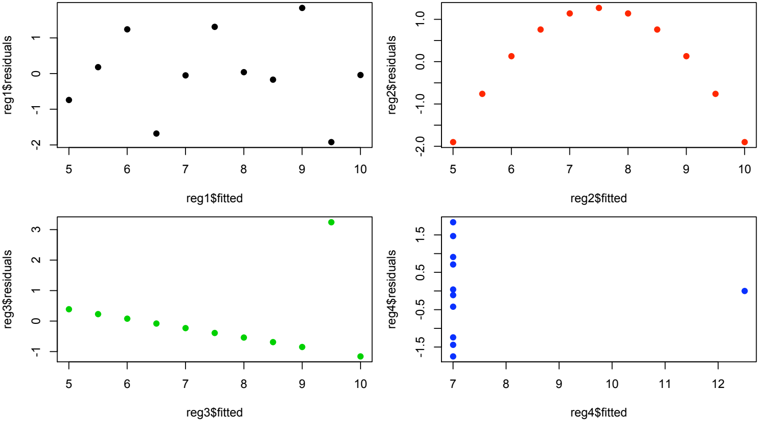
The regression lines and R^2 values are the same...



```
> ansreg <- list(reg1=lm(y1~x1), reg2=lm(y2~x2), reg3=lm(y3~x3), reg4=lm(y4~x4))
> attach(ansreg) # attach the names of each regression
> round( cbind(reg1$coef, reg1$coef, reg1$coef, reg1$coef), 1 )
            [,1] [,2] [,3] [,4]
(Intercept)  3.0  3.0  3.0  3.0
x1           0.5  0.5  0.5  0.5
> smry <- lapply(ansreg, summary)
> round( c(smry$reg1$r.sq, smry$reg1$r.sq, smry$reg1$r.sq, smry$reg1$r.sq), 1 )
[1] 0.7 0.7 0.7 0.7
```

Example Model Violations

...but the residuals (plotted against $\hat{Y}$) look totally different.



Plotting e vs $\hat{Y}$ is your #1 tool for finding fit problems.

Residuals and the Model Assumptions

Recall that the linear regression model assumes

$$Y_i = \mathbf{X}_i'\boldsymbol{\beta} + \varepsilon_i, \quad \textit{where} \;\; \varepsilon_i \overset{iid}{\sim} \mathrm{N}(0, \sigma^2)$$

Our goal is to determine if the “true” residuals are *iid* normal and unrelated to **X**. If the model assumptions are true, then the residuals must be just “white noise”:

1. Each ε_i has the same variance (σ^2).
2. Each ε_i has the same mean (0).
3. All of the ε_i have the same normal distribution.

Residuals and the Model Assumptions

Since the true ε_i residuals are unknown, we must look instead at the least squares estimated residuals.

We estimate $Y_i = \mathbf{X}_i'\mathbf{b} + e_i$, such that the sample least squares regression residuals are $e_i = Y_i - \hat{Y}_i$.

What should the e_i residuals look like if the model is true?

Residuals and the Model Assumptions

Focus on SLR: if the model is true, it turns out that

$$e_i \sim N(0, \sigma^2[1 - h_i]), \quad h_i = \frac{1}{n} + \frac{(X_i - \bar{X})^2}{\sum_{j=1}^{n}(X_j - \bar{X})^2}.$$

The h_i term is referred to as the i^{th} observation's leverage:

- It is that point's share of the data $(1/n)$ plus its proportional contribution to variability in X.
- As $n \to \infty$, $h_i \to 0$ and residuals become $\varepsilon_i \sim N(0, \sigma^2)$.

Understanding Leverage

The h_i leverage term measures sensitivity of the estimated least squares regression line to changes in Y_i.

The term "leverage" provides a mechanical intuition:

- The farther you are from a pivot joint, the more torque you have pulling on a lever.

Rob McCulloch has a nice online illustration of leverage:

`www.rob-mcculloch.org/teachingApplets/Leverage`

Outliers do more damage if they have high leverage!

Standardized Residuals

Since $e_i \sim N(0, \sigma^2[1 - h_i])$, we know that

$$\frac{e_i}{\sigma\sqrt{1 - h_i}} \sim N(0, 1).$$

These transformed e_i's are called the Standardized Residuals.

They all have the same distribution if the SLR model is true.

They are almost (close enough) independent ($\overset{iid}{\sim} N(0, 1)$).

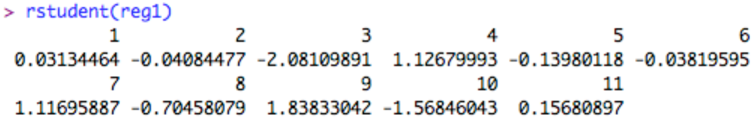
Standardized Residuals

As always, we don't know $\mathrm{sd}(\varepsilon) = \sigma$.

We thus define a standard Studentized Residual as

$$r_i = \frac{e_i}{s\sqrt{1 - h_i}} \sim t_{n-p-1}(0, 1)$$

These are easy to get in R with the `rstudent()` function.



```
> rstudent(reg1)
          1           2           3           4           5           6
 0.03134464 -0.04084477 -2.08109891  1.12679993 -0.13980118 -0.03819595
          7           8           9          10          11
 1.11695887 -0.70458079  1.83833042 -1.56846043  0.15680897
```

How to Deal with Outliers

Since the studentized residuals are distributed $t_{n-p-1}(0,1)$, we should be curious about any r_i outside of about $[-3,3]$.

When should you delete outliers?

Only when you have a really good reason!

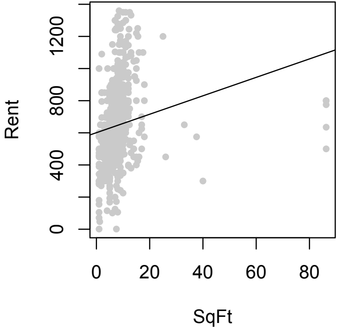
There is nothing wrong with running regression with and without potential outliers to see whether results are significantly impacted.

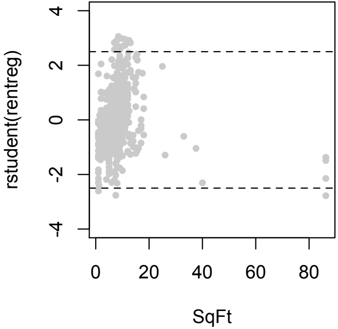
Any time outliers are dropped the reasons for removing observations should be clearly noted.

Outliers, Leverage, and Residuals

Warning: Unfortunately, outliers with high leverage are hard to catch through r_i (since the line is pulled towards them).

Recall your *Rent* vs *SqFt* example in homework 1:





Plots of r_i or e_i against $\hat{Y}_i$ or X_i are still your best diagnostic!

Normality and the Studentized Residuals

A more subtle issue is normality of the distribution on ε.

We can look at the residuals to judge normality if n is big enough (say > 20; less than that makes it too hard to call).

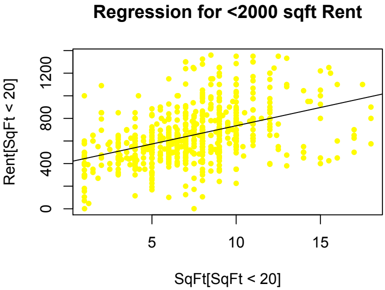
In particular, if we have decent size n, we want the shape of the studentized residual distribution to "look" like $N(0,1)$.

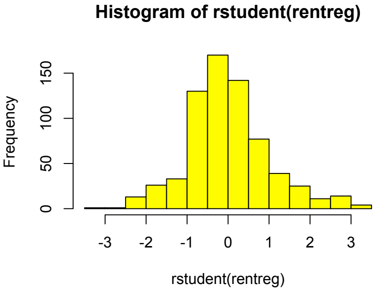
The most obvious tactic is to look at a histogram of r_i.

Normality and the Studentized Residuals

For example, consider the residuals from a regression of *Rent* on *SqFt* which ignores houses with $\geq$ 2000 sqft.

```
> rentreg <- lm(Rent[SqFt<20]   SqFt[SqFt<20])
> plot( SqFt[SqFt<20], Rent[SqFt<20], pch=20, col=7)
> abline(rentreg)
> hist(rstudent(rentreg), col=7)
```





Assessing Normality via Q-Q plots

More precise diagnostics are provided by Normal Q-Q plots.

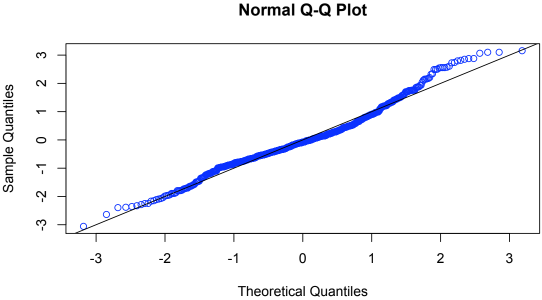
Q-Q stands for quantile-quantile: plot the sample quantiles (e.g. 10^{th} percentile, etc) against true percentiles from a $N(0,1)$ distribution (e.g. -1.96 is the true 2.5% quantile).

If $r_i \sim N(0,1)$ these quantiles should be equal (lie on a line).

Assessing Normality via Q-Q plots

R has a function for normal Q-Q plots:

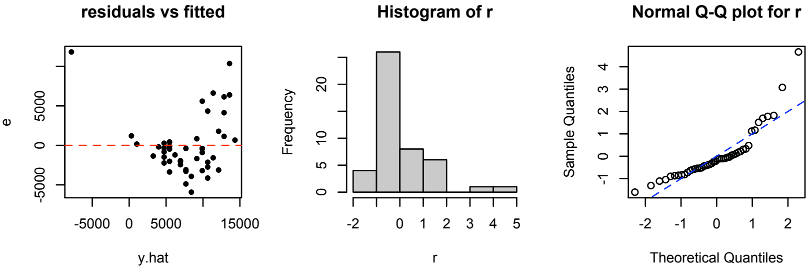
```
> qqnorm(rstudent(rentreg), col=4)
> abline(a=0, b=1)
```



It is good to add the line $Y = X$ to see where points should be.

Residual plots example: pickup regression

Consider our pickup data regression of price onto years:

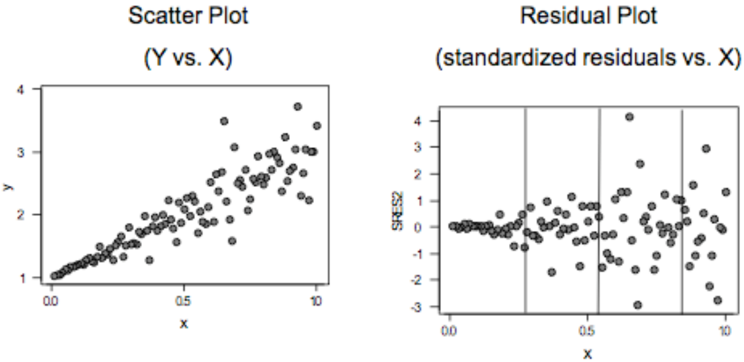


The plots tell us that:

- Data are more curved than straight (i.e. line doesn't fit).
- Residuals are skewed to the right.
- There is a huge positive e_i for an old "classic" truck.

Nonconstant Variance

If you get a trumpet shape, you have nonconstant variance.



Scatter Plot

(Y vs. X)

Residual Plot

(standardized residuals vs. X)

This violates our assumption that all ε_i have the same σ^2.

Variance Stabilizing Transformations

This is one of the most common model violations; luckily, it is usually fixable by transforming the response (Y) variable.

log(Y) is the most common variance stabilizing transform.

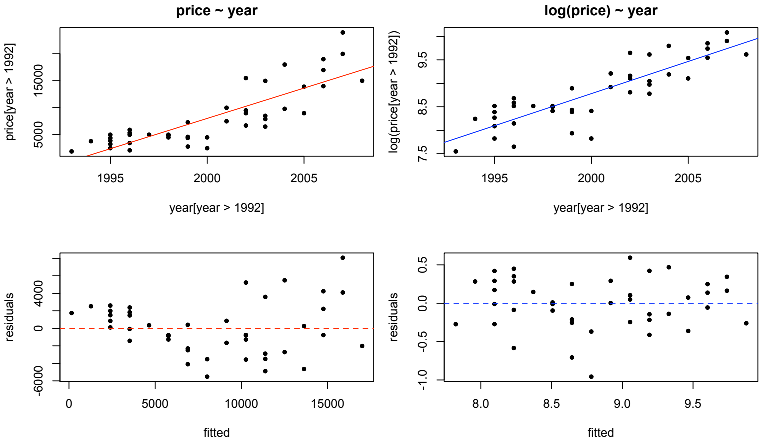
- If Y has only positive values (e.g. sales) or is a count (e.g. # of customers), take log(Y) (always natural log).

$\sqrt{Y}$ is another useful transformation; also, consider looking at Y/X or dividing by some other factor.

In general, think about in what scale you expect linearity.

Log Transform Example: the Pickup Data

Reconsider the regression of truck *price* onto *year*, after removing trucks older than 15 years (`truck[year>1992,]`).



Variance Stabilizing Transformations

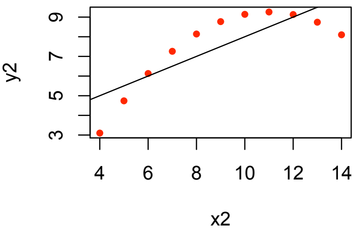
Warning: Be careful when interpreting the transformed model.

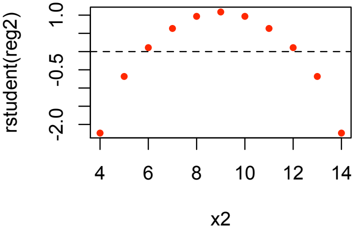
If $\mathbb{E}[\log(Y)] = b_0 + b_1 X$, then $\mathbb{E}[Y] \approx e^{b_0} e^{b_1 X}$.
We have a multiplicative model now!

Also, you can not compare R^2 values for regressions corresponding to different transformations of the response (look to residuals to see which model is better).

Nonlinear Residual Patterns

Consider regression residuals for the second Anscombe dataset:





Things are not good! It appears that we do not have a linear mean function; that is $\mathbb{E}[Y] \neq \beta_0 + \beta_1 X$.

Polynomial Regression

Even though we are limited to a linear mean, it is possible to get nonlinear regression by transforming the X variable.

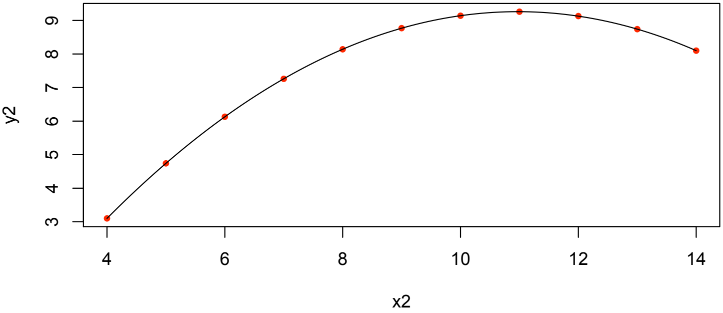
In general, we can add powers of X to get polynomial regression: $\mathbb{E}[Y] = \beta_0 + \beta_1 X + \beta_2 X^2 \ldots + \beta_m X^m$

You can fit any mean function if m is big enough.
Usually, $m = 2$ does the trick.

Polynomial Regression

Try $\mathbb{E}[Y] = \beta_0 + \beta_1 X + \beta_2 X^2$ for Anscombe's 2nd dataset:

```
x2squared <- x2ˆ2
NL <- lm(y2 ∼ x2 + x2squared)
xgrid <- seq(4,14,length=100)
ygrid <- NL$coef[1] + NL$coef[2]*xgrid + NL$coef[3]*xgridˆ2
plot(x2, y2, col=2, pch=20);  lines(xgrid, ygrid)
```



Testing for Nonlinearity

To see if you need more nonlinearity, try the regression which includes the next polynomial term, and see if it is significant.

For example, to see if you need a quadratic term , fit the model then run the regression $\mathbb{E}[Y] = \beta_0 + \beta_1 X + \beta_2 X^2$.

If your test implies $\beta_2 \neq 0$, you need X^2 in your model.

Note: p-values are calculated "given the other β's are nonzero"; i.e., conditional on X being in the model. More detail on this later.

More on Transformations: the log-log Model

The other common covariate transform is $\log(X)$.

In practice, this is often used in conjunction with a $\log(Y)$ response transformation.

The log-log model is $\log(Y) = \beta_0 + \beta_1 \log(X) + \varepsilon$.

It is super useful, and has some special properties...

The log-log Model

Recall that log is always the natural log, with base $e = 2.7...$ and that $\log(ab) = \log(a) + \log(b)$ and $\log(a^b) = b\log(a)$.

Consider the multiplicative model $\mathbb{E}[Y|X] = AX^B$.
Take logs of both sides to get

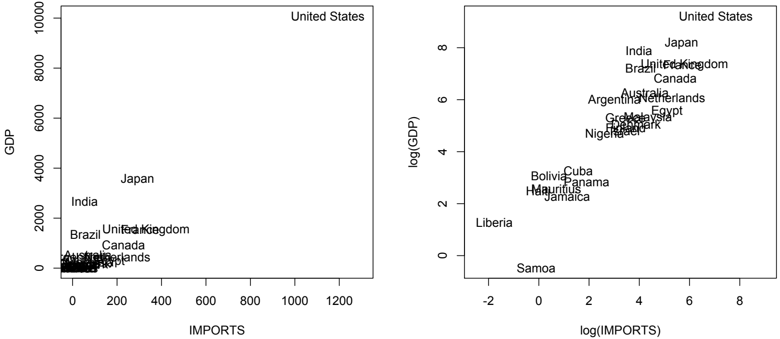
$$\log(\mathbb{E}[Y|X]) = \log(A) + \log(X^B) = \log(A) + B\log(X)$$

The log-log model is appropriate whenever things are linearly related on a multiplicative, or percentage, scale.

The log-log Model

Consider a country's *GDP* as a function of *IMPORTS*:

- Since trade multiplies, we might expect to see %*GDP* to increase with %*IMPORTS*.



Elasticity and the log-log Model

In a log-log model, the slope β_1 is sometimes called elasticity.

The elasticity is (roughly) % change in Y per 1% change in X.

$$\beta_1 \approx \frac{d\%Y}{d\%X}$$

For example, economists often assume that GDP has import elasticity of 1. Indeed,

```
Call:   lm(formula = log(GDP) ∼ log(IMPORTS))
Coefficients:
 (Intercept)  log(IMPORTS)
    1.8915        0.9693
```

Price Elasticity

In marketing, the slope coefficient β_1 in the regression $\log(\mathit{sales}) = \beta_0 + \beta_1 \log(\mathit{price}) + \varepsilon$ is called price elasticity.

This is the % change in *sales* per 1% change in *price*.

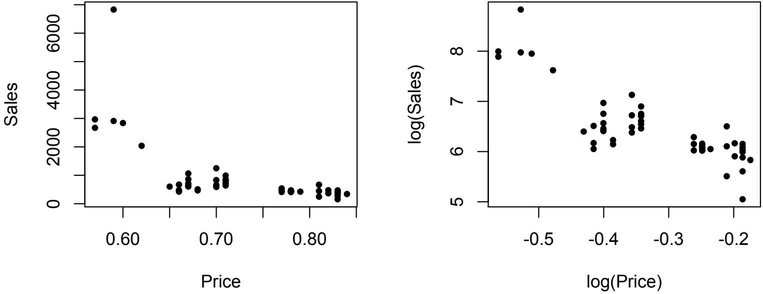
The model implies that $\mathbb{E}[\mathit{sales}] = A * \mathit{price}^{\beta_1}$ such that β_1 is the constant rate of change.

Economists have “demand elasticity” curves, which are just more general and harder to measure.

Price Elasticity Example

We have Nielson SCANTRACK data on supermarket sales of a canned food brand produced by Consolidated Foods Inc.

```
> attach( confood <- read.csv("confood.csv") )
> plot(Price,Sales, pch=20)
> plot(log(Price), log(Sales), pch=20)
```

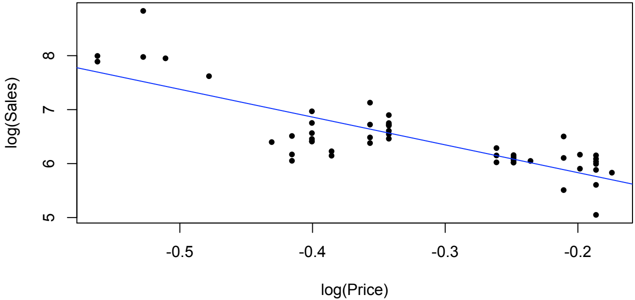


Price Elasticity Example

Run the regression to determine price elasticity:

```
Call:   lm(formula = log(Sales) ∼ log(Price))

Coefficients:
 (Intercept)  log(Price)
    4.803       -5.148
```



Sales decrease by about 5% for every 1% price increase.

Summary of Transformations

Use plots of residuals *vs* X or $\hat{Y}$ to determine the next step.

Log transform is your best friend ($\log(X)$, $\log(Y)$, or both).

Add polynomial terms (e.g. X^2) to get nonlinear regression.

Be careful to get the interpretation correct after transforming.

Inference for Coefficients

As before in SLR, the LS linear coefficients are random (different for each sample) and correlated with each other.

The LS estimators are unbiased: $\mathbb{E}[b_j] = \beta_j$ for $j = 0, \ldots, d$.

In particular, the sampling distribution for **b** is a multivariate normal, with mean $\boldsymbol{\beta} = [\beta_0 \cdots \beta_d]'$ and covariance matrix $\mathbf{S_b}$.

$$\mathbf{b} \sim \mathrm{N}(\boldsymbol{\beta}, \mathbf{S_b})$$

Coefficient Covariance Matrix

$\mathrm{var}(\mathbf{b})$: the $p \times p$ covariance matrix for random vector **b** is

$$\mathbf{S_b} = \begin{bmatrix} \mathrm{var}(b_0) & \mathrm{cov}(b_0, b_1) & & & \\ \mathrm{cov}(b_1, b_0) & \mathrm{var}(b_1) & & & \\ & & \ddots & & \\ & & & \mathrm{var}(b_{d-1}) & \mathrm{cov}(b_{d-1}, b_d) \\ & & & \mathrm{cov}(b_d, b_{d-1}) & \mathrm{var}(b_d) \end{bmatrix}$$

$\Rightarrow$ Standard errors are the square root of the diagonal of $\mathbf{S_b}$.

Coefficient Standard Errors

The coefficent covariance matrix is “easy” to calculate

$$\mathbf{S_b} = s^2 \left(\hat{\mathbf{X}}' \hat{\mathbf{X}} \right)^{-1}$$

Variance decreases with n and $\mathrm{var}(\mathbf{X})$, and increases with s^2.

Any regression summary will provide all the standard errors.

Forecasting in MLR

Prediction follows exactly the same methodology as in SLR.

For new data $\mathbf{x}_f = [X_{1,f} \cdots X_{d,f}]'$,

- $\mathbb{E}[Y_f | \mathbf{x}_f] = \hat{Y}_f = b_0 + b_1 X_{1f} + \ldots b_d X_{df}$
- $\mathrm{var}[Y_f | \mathbf{x}_f] = \mathrm{var}(\hat{Y}_f) + \mathrm{var}(e_f) = s_{fit}^2 + s^2 = s_{pred}^2.$

With $\hat{\mathbf{X}}$ our design matrix (slide 9) and $\hat{\mathbf{x}}_f = [1, X_{1,f} \cdots X_{d,f}]'$

$$s_{fit}^2 = s^2 * \hat{\mathbf{x}}_f'(\hat{\mathbf{X}}'\hat{\mathbf{X}})^{-1}\hat{\mathbf{x}}_f$$

A $(1 - \alpha)$ level prediction interval is still $\hat{Y}_f \pm t_{\alpha/2, n-p} s_{pred}$.

Forecasting in MLR

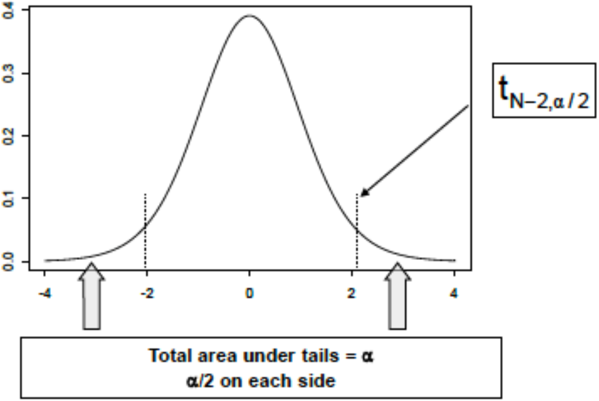
The syntax in R is also exactly the same as before:

```
> predict( salesMLR, data.frame(P1=1, P2=1),
           interval="prediction", level=0.95)
    fit      lwr      upr
1 1.094661 1.064015 1.125306
```

Testing and Confidence Intervals

Suppose Z_{n-p} is distributed $t_{n-p}(0, 1)$. A centered interval is

$$\mathrm{P}(-t_{n-p,\alpha/2} < Z_{n-p} < t_{n-p,\alpha/2}) = 1 - \alpha$$



$t_{N-2,\alpha/2}$

Total area under tails = α
α/2 on each side

Confidence Intervals

Since $b_j \sim t_{n-p}(\beta_j, s_{b_j})$,

$$\begin{aligned} 1-\alpha &= \mathrm{P}(-t_{n-p,\alpha/2} < \frac{b_j-\beta_j}{s_{b_j}} < t_{n-p,\alpha/2}) \\ &= \mathrm{P}(b_j - t_{n-p,\alpha/2} s_{b_j} < \beta_j < b_j + t_{n-p,\alpha/2} s_{b_j}) \end{aligned}$$

Thus $(1-\alpha)$*100% of the time, β_j is within the Confidence Interval: $b_j \pm t_{n-p,\alpha/2} s_{b_j}$

Testing

Similarly, suppose that assuming $b_j \sim t_{n-p}(\beta_j, s_{b_j})$ for our sample b_j leads to (recall $Z_{n-p} \sim t_{n-p}(0,1)$)

$$\mathrm{P}\left(Z_{n-p} < -\left|\frac{b_j - \beta_j}{s_{b_j}}\right|\right) + \mathrm{P}\left(Z_{n-p} > \left|\frac{b_j - \beta_j}{s_{b_j}}\right|\right) = \varphi.$$

Then the "p-value" is $\varphi = 2\mathrm{P}(Z_{n-p} > |b_j - \beta_j|/s_{b_j})$.

You do this calculation for $\beta_j = \beta_j^0$, an assumed null/safe value, and only reject β_j^0 if φ is too small (e.g., $\varphi < 1/20$).

In regression, $\beta_j^0 = 0$ almost always.

Testing

Suppose that we are interested in the slope parameter, β_1.

For example, is there any evidence in the data to support the existence of a relationship between X and Y?

We can rephrase this in terms of competing hypotheses.

$H_0 : \beta_1 = 0$. Null/safe; implies "no effect" and we ignore X.

$H_1 : \beta_1 \neq 0$. Alternative; leads us to our best guess $\beta_1 = b_1$.

Hypothesis Testing

If we want statistical support for a certain claim about the data, we want that claim to be the alternative hypothesis.

Our hypothesis test will either reject or not reject the null hypothesis (the default if our claim is not true).

If the hypothesis test rejects the null hypothesis, we have statistical support for our claim!

Hypothesis Testing

We use b_j for our test about β_j.

- Reject H_0 when b_j is far from β_j^0 (usually 0).
- Assume H_0 when b_j is close to β_j^0.

An obvious tactic is to look at the difference $b_j - \beta_j^0$.

But this measure doesn't take into account the uncertainty in estimating b_j: What we really care about is how many standard deviations b_j is away from β_j^0.

Hypothesis Testing

The *t*-statistic for this test is

$$z_{b_j} = \frac{b_j - \beta_j^0}{s_{b_j}} = \frac{b_j}{s_{b_j}} \text{ } \mathit{for} \text{ } \beta_j^0 = 0.$$

If H_0 is true, this should be distributed $z_{b_j} \sim t_{n-p}(0, 1)$.

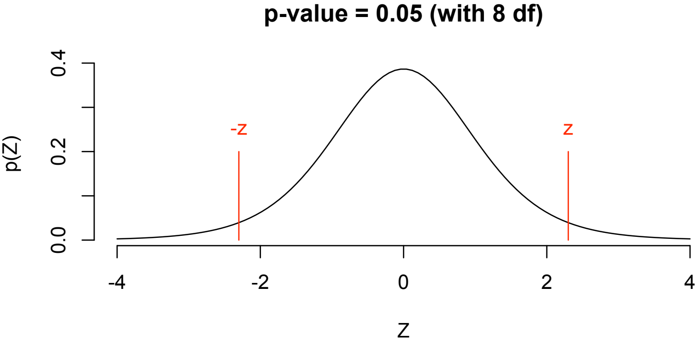
- Small $|z_{b_j}|$ leaves us happy with the null β_j^0.
- Large $|z_{b_j}|$ (i.e., $>$ about 2) should get us worried!

Hypothesis Testing

We assess the size of z_{b_j} with the p-value :

$$\varphi = \mathrm{P}(|Z_{n-p}| > |z_{b_j}|) = 2\mathrm{P}(Z_{n-p} > |z_{b_j}|)$$

(once again, $Z_{n-p} \sim t_{n-p}(0, 1)$).



Hypothesis Testing

The p-value is the probability, assuming that the null hypothesis is true, of seeing something more extreme (further from the null) than what we have observed.

You can think of $1 - \varphi$ (inverse p-value) as a measure of distance between the data and the null hypothesis. In other words, $1 - \varphi$ is the strength of evidence against the null.

Inference for Individual Coefficients

MLR NOTE:

Intervals and testing via b_j & s_{b_j} are one-at-a-time procedures:

- You are evaluating the j^{th} coefficient conditional on the other X's being in the model, but regardless of the values you've estimated for the other b's.

Multicollinearity

Multicollinearity refers to strong linear dependence between some of the covariates in a multiple regression model.

The usual marginal effect interpretation is lost:

- Change in one X variable leads to change in others.

Coefficient standard errors will be large, such that multicollinearity leads to large uncertainty about b_j's.

Multicollinearity

Suppose that you regress Y onto X_1 and $X_2 = 10 * X_1$.

Then $\mathbb{E}[Y] = \beta_0 + \beta_1 X_1 + \beta_2 X_2 = \beta_0 + \beta_1 X_1 + \beta_2 (10 X_1)$ and the marginal effect of X_1 on Y is

$$\frac{\partial \mathbb{E}[Y|X_1, X_2]}{\partial X_1} = \beta_1 + 10\beta_2$$

X_1 and X_2 do not act independently!

Multicollinearity

Multicollinearity is not a big problem in and of itself,
you just need to know that it is there.

If you recognize multicollinearity:

- Understand that the β_j are not true marginal effects.
- Consider dropping variables to get a more simple model (use the partial F-test!).
- Expect to see big standard errors on your coefficients (i.e., your coefficient estimates are unstable).

Variable Interaction

So far we have considered the impact of each independent variable in a additive way.

We can extend this notion by the inclusion of multiplicative effects through interaction terms.

$$Y_i = \beta_0 + \beta_1 X_{1i} + \beta_2 X_{2i} + \beta_3 (X_{1i} X_{2i}) + \ldots + \varepsilon$$

$$\frac{\partial \mathbb{E}[Y|X_1, X_2]}{\partial X_1} = \beta_1 + \beta_3 X_2$$

College GPA and Age

Consider the connection between college and MBA grades:
A model to predict Booth GPA from college GPA could be

$$GPA^{MBA} = \beta_0 + \beta_1 GPA^{Bach} + \varepsilon$$

```
> summary( lm(MBAGPA ~ BachGPA) )
            Estimate   Std.Error   t value   Pr(>|t|)
  BachGPA   0.26269     0.09244     2.842    0.00607 **
```

For every 1 point increase in college GPA, your expected GPA at Booth increases by about .26 points.

College GPA and Age

However, this model assumes that the marginal effect of College GPA is the same for any age.

It seems that how you did in college should have less effect on your MBA GPA as you get older (farther from college).

We can account for this intuition with an interaction term:

$$GPA^{MBA} = \beta_0 + \beta_1 GPA^{Bach} + \beta_2 (Age \times GPA^{Bach}) + \varepsilon$$

Now, the college effect is $\frac{\partial \mathbb{E}[GPA^{MBA} | GPA^{Bach} \quad Age]}{\partial GPA^{Bach}} = \beta_1 + \beta_2 Age$.

Depends on Age!

College GPA and Age

Fitting interactions in R is easy:

`lm(Y ~ X1*X2)` fits $\mathbb{E}[Y] = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_1 X_2$.

Here, we want the interaction but do not want to include the main effect of age (your age shouldn't matter individually).

```
> summary( lm(MBAGPA ~ BachGPA*Age - Age) )
             Estimate   Std.Error   t value  Pr(>|t|)
   BachGPA    0.455750    0.103026    4.424   4.07e-05 ***
BachGPA:Age -0.009377    0.002786   -3.366   0.00132 **
```

College GPA and Age

Without the interaction term

- Marginal effect of College GPA is $b_1 = 0.26$.

With the interaction term:

- Marginal effect is $b_1 + b_2 Age = 0.46 - 0.0094 Age$.

Age	Marginal Effect
25	0.22
30	0.17
35	0.13
40	0.08

Glossary and Equations

- $\hat{Y}_i = b_0 + b_1 X_i$ is the ith fitted value.
- $e_i = Y_i - \hat{Y}_i$ is the ith residual.
- s: standard error of regression residuals ($\approx \sigma = \sigma_\varepsilon$).

$$s^2 = \frac{1}{n-2} \sum e_i^2$$

- s_{b_j}: standard error of regression coefficients.

$$s_{b_1} = \sqrt{\frac{s^2}{(n-1)s_x^2}} \qquad s_{b_0} = s\sqrt{\frac{1}{n} + \frac{\bar{X}^2}{(n-1)s_x^2}}$$

Glossary and Equations

- α is the significance level (prob of type 1 error).
- $t_{n-p,\alpha/2}$ is the value such that for $Z_{n-p} \sim t_{n-p}(0,1)$,

$$\mathrm{P}(Z_{n-p} > t_{n-p,\alpha/2}) = \mathrm{P}(Z_{n-p} < -t_{n-p,\alpha/2}) = \alpha/2.$$

- $z_{b_j} \sim t_{n-p}(0,1)$ is the standardized coefficient t-value:

$$z_{b_j} = \frac{b_j - \beta_j^0}{s_{b_j}} \quad (= b_j/s_{b_j} \; \textit{most often})$$

- The $(1-\alpha) * 100\%$ for β_j is $b_j \pm t_{n-p,\alpha/2} s_{b_j}$.
- $\varphi = 2\mathrm{P}(Z_{n-p} > z_{b_j})$ is the coefficient p-value.

Glossary and Equations

- $\hat{Y}_f = b_0 + X_f b_1$ is a forecast prediction.

$$\mathrm{sd}(\hat{Y}_f) = s_{fit} = s\sqrt{\frac{1}{n} + \frac{(X_f - \bar{X})^2}{(n-1)s_x^2}}$$

- Forecast residual is $e_f = Y_f - \hat{Y}_f$ and $\mathrm{var}(e_f) = s^2 + s_{fit}^2$. That is, the predictive standard error is

$$s_{pred} = s\ \sqrt{1 + \frac{1}{n} + \frac{(X_f - \bar{X})^2}{(n-1)s_x^2}}.$$

and $\hat{Y}_f \pm t_{n-p,\alpha/2} s_{pred}$ is the $(1-\alpha)100\%$ PI at X_f.

Glossary and Equations

Leverage is $h_i = \frac{1}{n} + \frac{(X_i - \bar{X})^2}{s_x^2}$

Studentized residuals are $r_i = \frac{e_i}{s_{-i}\sqrt{1-h_i}} \sim t_{n-p-1}(0,1)$

Elasticity is the slope in a log-log model: $\beta_1 \approx \frac{d\%Y}{d\%X}$.

Glossary and Equations

MLR updates to the LS equations:

- $\mathbf{b} = (\hat{\mathbf{X}}'\hat{\mathbf{X}})^{-1}\hat{\mathbf{X}}'\mathbf{Y}$
- $\mathrm{var}(\mathbf{b}) = \mathbf{S_b} = s^2 \left(\hat{\mathbf{X}}'\hat{\mathbf{X}}\right)^{-1}$
- $s^2_{fit} = s^2 * \hat{\mathbf{x}}'_f(\hat{\mathbf{X}}'\hat{\mathbf{X}})^{-1}\hat{\mathbf{x}}_f$
- $R^2 = SSR/SST = \mathrm{cor}^2(\hat{Y}, Y) = r^2_{\hat{y}y}$