```
errorcolor##
Error
in
library("readr"):
there
is
no
package
called
'readr'
{
{
}
}
errorcolor##
Error
in
library("gapminder"):
there
is
no
package
called
'gapminder'
```

POLS/CS&SS 503: Advanced Quantitative Political Methodology

TRANSFORMATIONS

May 5, 2015

Jeffrey B. Arnold

UNIVERSITY of WASHINGTON
DEPARTMENT OF POLITICAL SCIENCE

CENTER for STATISTICS and the SOCIAL SCIENCES

Overview

Residuals and Misspecification

Example

data(”gapminder”)

`## Warning in data(”gapminder”): data set ’gapminder’ not found` ggplot(data = filter(gapminder, year == 2007), aes(x = gdpPercap, y = lifeExp)) +
geom$_{p}oint() + geom_{s}mooth(method = "lm", se = FALSE) + theme_{l}ocal()$

`## Error in filter_(.data, .dots = lazyeval::lazy_dots(...)): object ’gapminder’ not found`

Residuals and Misspecification

Example

data("gapminder")

`## Warning in data("gapminder"): data set 'gapminder' not found` ggplot(data = augment(lm(lifeExp gdpPercap, data = filter(gapminder, year == 2007))), aes(x = gdpPercap, y = .resid)) + geom$_point() + geom_hline(yintercept = 0) + theme_local()$

`## Error in ggplot(data = augment(lm(lifeExp ~ gdpPercap, data = filter(gapminder, : could not find function "augment"`

Residuals and Misspecification

Example

data("gapminder")

`## Warning in data("gapminder"): data set 'gapminder' not found` ggplot(data = filter(gapminder, year == 2007), aes(x = log(gdpPercap), y = lifeExp)) + geom$_point() + geom_smooth(method = "lm", se = FALSE) + theme_local()$

`## Error in filter_(.data, .dots = lazyeval::lazy_dots(...)): object 'gapminder' not found`

Residuals and Misspecification

Example

data(”gapminder”)

`## Warning in data(”gapminder”): data set ’gapminder’ not found` ggplot(data = augment(lm(lifeExp log(gdpPercap), data = filter(gapminder, year == 2007))), aes(x = log.gdpPercap., y = .resid)) + geom$_point() + geom_hline(yintercept = 0) + theme_local()$

`## Error in ggplot(data = augment(lm(lifeExp ~ log(gdpPercap), data = filter(gapminder, : could not find function ”augment”`

Interpreting Logarithms

How would you interpret the following?

- GDP per $\text{cap}_i = \alpha + \beta \log(\text{school})_i$
- $\log$ GDP per $\text{cap}_i = \alpha + \beta(\text{school})_i$
- $\log$ GDP per $\text{cap}_i = \alpha + \beta \log(\text{school})_i$

Linearizing Functions

Can you linearize these with logarithms?

Exponential

$$y_i = e^{\beta_0 + \beta_1 x_1 + \beta_2 x_2 \epsilon_i}$$

Gravity Equation

$$\text{trade}_{ij} = \frac{\alpha \text{GDP}_i^{\beta_1} \text{GDP}_j^{\beta_2}}{\delta d_{ij}^{\gamma}}$$

Cobb-Douglas

$$y = \alpha (x^{\delta \gamma} x^{(1-\delta)})^{\gamma}$$

CES Production Function

$$y = \alpha (\delta x^{\rho} + (1-\delta) x^{\rho})^{\gamma/\rho}$$

Interpretating Logarithms

Why use natural log for regression

- Note: $\log(1+r) \approx r$ when r small
- $$\log(x) - \log(x(1+r)) = \log(1+r) \approx r = \%\Delta x/100$$
- Only holds for natural logarithm

Converting between bases

To convert $\log_e$ to $\log_{10}$

$$\log_{10}(x) = \frac{\log_e(x)}{\log_e(10)}$$

Box-Cox Family of Transformations

.dat <- expand.grid(lambda = c(-2, -1, -0.5, 0, 0.5, 1, 2, 3))

group$_by(lambda)do(data_frame(x=seq(0.01,4,by=0.01)),y= car::bcPowe$

ggplot(.dat, aes(x = x, y = y, colour = factor(lambda))) +

geom$_line()+theme_local()+scale_colour_discrete("lambda")+scale_ycontin$

Error in eval(expr, envir, enclos): could not find function "theme_local"

Plot for $\lambda = 0.25, 0.5, 0, 2, 4, 8$ for $x = (0, 4]$

Box-Cox Family of Transforms

$$\begin{cases} f(x,\lambda) = \frac{x^\lambda - 1}{\lambda} & \text{if } \lambda \neq 0 \\ f(x,\lambda) = \log x & \text{if } \lambda = 0 \end{cases}$$

- Requires $x > 0$. If negative, use $x + c$ (some problems), or Yeo-Johnson
- Can solve for λ to transform x as close to wrt. Normal skew.
- **car** function: `powerTransform|, bcTransform|.`
- `In regression: If know` λ `can transform` y `or` x

Linear Transformations of Regression

$$(y_i + a)/b = \alpha + \beta(x_i + d)/e + \epsilon_i$$

$$(y_i + \bar{y})/s_y = \alpha + \beta(x_i + \bar{x})/s_x + \epsilon_i$$

Standardized Coefficients / Regressors

$$y = \alpha + \beta 0 + \beta_1 \frac{x_i - \bar{x}}{\text{SD}\left(x\right)} + \epsilon_i$$

- Can be useful for default interpretation (controversial)
- Bad for skewed variables, binary variables? But about same as comparing $X + \text{SD}\, X$ post-estimation.
- Transform regressors, not functions of regressors.
- Gelman: Continuous: divide by 2 SD; Binary: center at mean.
- No need for them for default interpretation. With computational power, simulations better.
- Very important to standardize X in machine learning applications, or anywhere with complicated optimization problems.