Class 4

BUS 696

Prof. Jonathan Hersh

BUS 696: Class 4 Outline

1. Pset 2 solutions
2. Problem Sets
    - Must submit code output (regression tables and plots) along with code for full credit
    - Must compile Rmarkdown documents
3. Questions?

BUS 696: Class 4 Outline

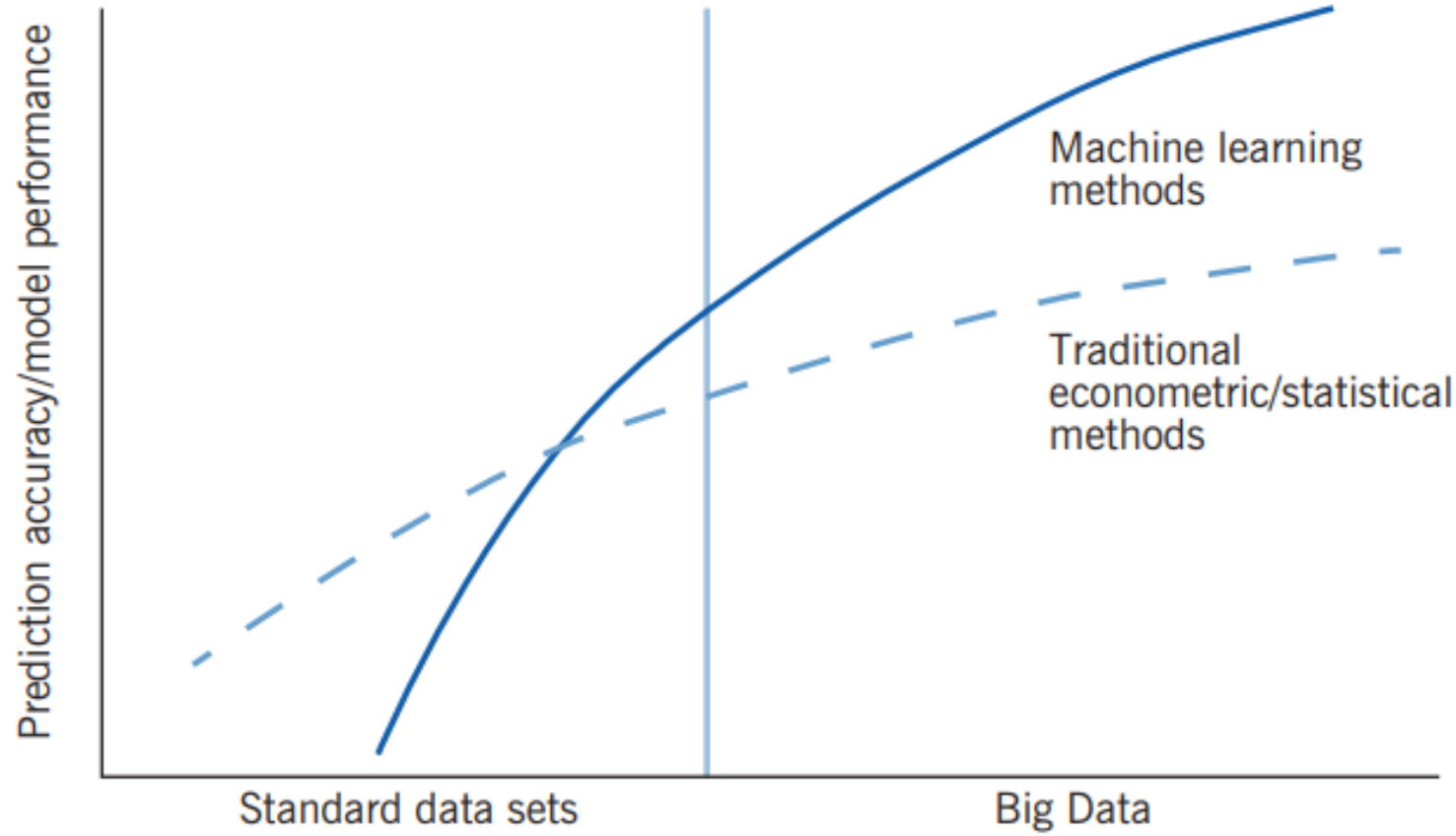
1. Everything Correlates:

   https://twitter.com/businessinsider/status/1173794704506400773?s=20
2. HBR AI Article
3. Transforming X and Y variables (Feature Transformation)
4. Dummy Variable Interpretation (Qualitative Predictors)
5. Interpreting and Generating Nonlinearity in X variables (Squared Terms)
6. Generating Predictions and Residuals from Fitted Models
7. Potential Problems (Regression Diagnostics)

Everything Correlates

https://twitter.com/businessinsider/status/1173794704506400773?s=20

The use of machine learning techniques for Big Data analytics



Source: Author's own compilation.

IZA World of Labor

Insights You Need from
Harvard Business Review

ARTIFICIAL INTELLIGENCE

THE BUSINESS OF ARTIFICIAL INTELLIGENCE

WHAT IT CAN – AND CANNOT – DO FOR YOUR ORGANIZATION

BY ERIK BRYNJOLFSSON AND ANDREW MCAFEE

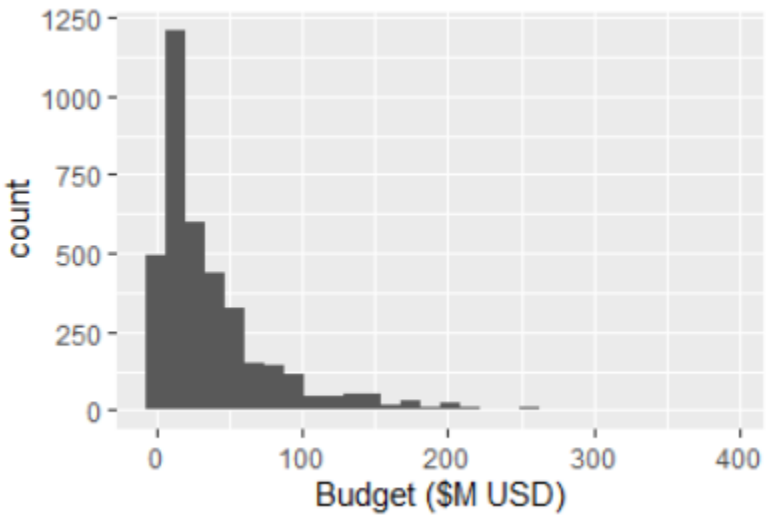
For more than 250 years the fundamental drivers of economic growth have been technological innovations. The most important of these are what economists call general-purpose technologies — a category that includes the steam engine, electricity, and the internal combustion engine. Each one catalyzed waves of complementary innovations and opportunities. The internal combustion engine, for example, gave rise to cars, trucks, airplanes, chain saws, and lawnmowers, along with big-box retailers, shopping centers, cross-docking warehouses, new supply chains, and, when you think about it, suburbs. Companies as diverse as Walmart, UPS, and Uber found ways to leverage the technology to create profitable new business models.

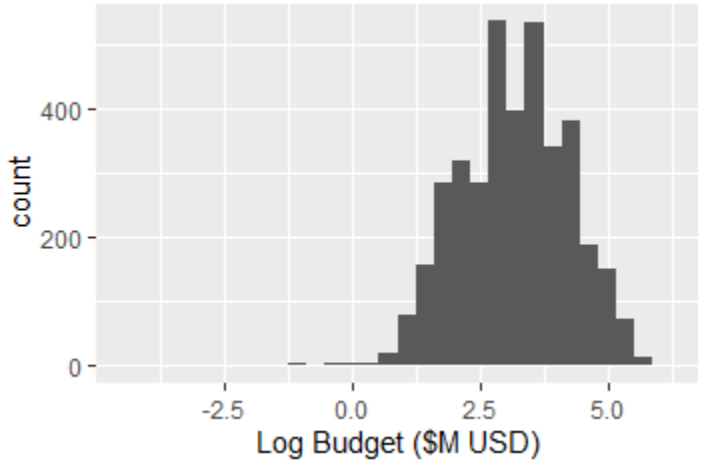
Interpreting Regression Output

```
> summary(mod1)

Call:
lm(formula = grossM ~ budgetM, data = movies_train)

Residuals:
    Min      1Q  Median      3Q     Max
-420.97  -44.31  -22.97   15.22  696.74

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 56.650257   1.293979  43.780  < 2e-16 ***
budgetM      0.030005   0.005506   5.449 5.43e-08 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 72.32 on 3274 degrees of freedom
  (506 observations deleted due to missingness)
Multiple R-squared:  0.008988,  Adjusted R-squared:  0.008685
F-statistic: 29.69 on 1 and 3274 DF,  p-value: 5.434e-08
```

Transforming X Variables: (Feature Selection)

```
> mod5 <- lm(grossM ~ budgetM + imdb_score, data = movies_train)
> summary(mod5)

Call:
lm(formula = grossM ~ budgetM + imdb_score, data = movies_train)

Residuals:
    Min      1Q  Median      3Q     Max
-386.78  -28.32   -9.26   17.12  490.30

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -76.44018    5.71581  -13.37   <2e-16 ***
budgetM       0.99252    0.02114   46.94   <2e-16 ***
imdb_score   14.10381    0.87362   16.14   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 54.3 on 3440 degrees of freedom
  (339 observations deleted due to missingness)
Multiple R-squared:  0.4271,    Adjusted R-squared:  0.4268
F-statistic:  1282 on 2 and 3440 DF,  p-value: < 2.2e-16
```





Log Budget VS Budget?

```
Call:
lm(formula = grossM ~ logbudgetM + imdb_score, data = movies_train)

Residuals:
    Min      1Q  Median      3Q     Max
-139.93  -34.90  -10.99   20.53  600.21

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -170.2519     6.9800  -24.39   <2e-16 ***
logbudgetM    36.0096     0.9792   36.77   <2e-16 ***
imdb_score    16.9175     0.9464   17.88   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 58.92 on 3440 degrees of freedom
  (339 observations deleted due to missingness)
Multiple R-squared:  0.3253,    Adjusted R-squared:  0.3249
F-statistic: 829.4 on 2 and 3440 DF,  p-value: < 2.2e-16
```

So Which Variable to Use??!

Factors to Consider

1. Adjusted R-Squared (higher better)

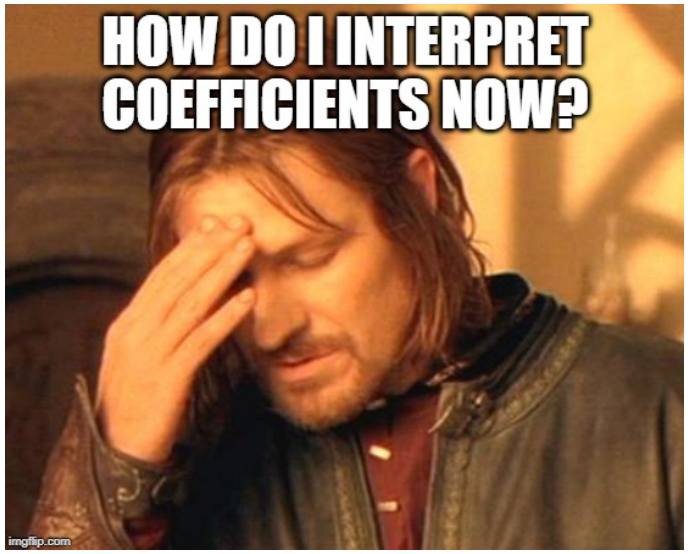
$$R^2_{adj} = 1 - (1 - R^2)\frac{n-1}{n-p-1}$$

P = number of variables

2. Out of sample error (in test set)
3. Intrepretability of Variable

Transforming Outcome Variable: Log Transform

```
Call:
lm(formula = log(grossM) ~ logbudgetM + imdb_score, data = movies_train)

Residuals:
     Min       1Q   Median       3Q      Max
-10.6300  -0.4906   0.2706   0.9121   5.4784

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -1.53709    0.18928  -8.121 6.39e-16 ***
logbudgetM   0.93739    0.02655  35.301  < 2e-16 ***
imdb_score   0.24203    0.02566   9.431  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.598 on 3440 degrees of freedom
  (339 observations deleted due to missingness)
Multiple R-squared:  0.2786,    Adjusted R-squared:  0.2782
F-statistic: 664.3 on 2 and 3440 DF,  p-value: < 2.2e-16
```

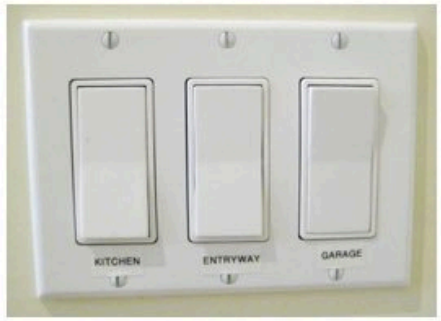


Log-log coefficients (logbudgetM) have a special interpretation

- Can be interpreted as elasticities
- E.g. 1% change in log(X) leads to a coefficient % change in Y

Factor: Switches, Continuous: Sliders

$$gross_i = \beta_0 + \beta_1 x_{budget} + \beta_2 \cdot x_{imdb\_score} + \epsilon_i$$



$$gross_i$$
$$= \beta_0 + \beta_1 x_{StevenSpielberg} + \beta_2 \cdot x_{MichaelBay} + \epsilon_i$$

Source: https://twitter.com/andrewheiss/status/1171084259660107777?s=20

Factor: Switches, Continuous: Sliders

$$gross_i = \beta_0 + \beta_1 x_{budget} + \beta_2 \cdot x_{imdb\_score} + \epsilon_i$$

$$gross_i = \beta_0 + \beta_1 x_{budget} + \beta_2 \cdot x_{imdb_{score}} + \beta_3 \cdot MichaelBay + \beta_4 \cdot StevenSpielberg + \epsilon_i$$

Source: https://twitter.com/andrewheiss/status/1171084259660107777?s=20

“Base” Levels and Factors: Data Frame

Intercept	Y	x1	x2
1	0.4	1	“A”
1	-0.5	2	“B”
1	-0.3	3	“B”
1	0.1	4	“A”
1	-0.8	5	“C”

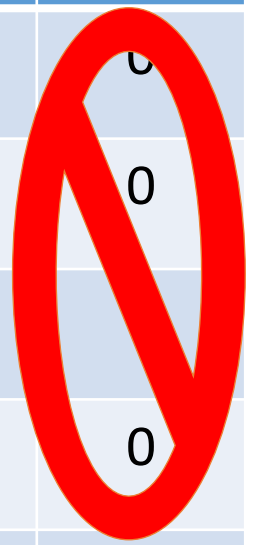
model.matrix(): From Factors to Dummy Variables

Intercept	Y	x1	x2_A	X2_B
1	0.4	1	1	0
1	-0.5	2	0	1
1	-0.3	3	0	1
1	0.1	4	1	0
1	-0.8	5	0	0

x2
“A”
“B”
“B”
“A”
“C”

Why Does Every Factor Level Not Get Its Own Dummy Variable?

Intercept	Y	x1	x2_A	X2_B	X2_C
1	0.4	1	1	0	0
1	-0.5	2	0	1	0
1	-0.3	3	0	1	0
1	0.1	4	1	0	0
1	-0.8	5	0	0	1



x2
“A”
“B”
“B”
“A”
“C”

Why? Because estimates are computed as

$$\beta = (X^T X)^{-1} X^T Y$$

Linear algebra requires $(X^T X)^{-1}$ to be full column rank i.e. each column of X must be linearly independent.

Intercept + X2_A + X_B + X2_C only “span” 3 dimensions

Including Qualitative Predictors (Dummy Variables)

```
> DF <- data.frame(y = rnorm(5),
+                  x1 = 1:5,
+                  x2 = c("A","B","B","A","C"))
> DF
           y x1 x2
1 -0.3475685  1  A
2 -0.5332828  2  B
3  1.4417620  3  B
4 -2.0573151  4  A
5 -0.7681746  5  C
```

```
> model.matrix(y ~ x1 + x2, DF)
  (Intercept) x1 x2B x2C
1           1  1   0   0
2           1  2   1   0
3           1  3   1   0
4           1  4   0   0
5           1  5   0   1
```

- Every qualitative predictor (factor) has been transformed into a column of binary information corresponding to that factor level

Excluded Base Level of Factor Becomes Base Level for Interpretation

		β_{x1}	β_{x2_A}	β_{x2_B}
Intercept	**Y**	**x1**	**x2_A**	**X2_B**
1	0.4	1	1	0
1	-0.5	2	0	1
1	-0.3	3	0	1
1	0.1	4	1	0
1	-0.8	5	0	0

Interpreting Coefficients:

- β_{x1}: We estimate y will increase by β_{x1} for each unit increase in x1
- β_{x2_A}: We estimate y will increase by β_{x2_A} *if it is of category A* relative to category C
- β_{x2_B}: We estimate y will increase by β_{x2_B} *if it is of category B* relative to category C

Interpreting Gender Dummy Variable Coefficients

$x_i = 1$, if female

$x_i = 0$, if male

$y_i =$ credit card balance

$$y_i = \beta_0 + \beta_1 \cdot x_i + \epsilon_i = \begin{cases} \beta_0 + \beta_1 \cdot x_i + \epsilon_i & i = female \\ \beta_0 + \epsilon_i & i = male \end{cases}$$

	Coefficient	Std. error	t-statistic	p-value
`Intercept`	509.80	33.13	15.389	< 0.0001
`gender[Female]`	19.73	46.05	0.429	0.6690

TABLE 3.7. *Least squares coefficient estimates associated with the regression of* `balance` *onto* `gender` *in the* `Credit` *data set. The linear model is given in (3.27). That is, gender is encoded as a dummy variable, as in (3.26).*

Interpreting Binary/Dummy Coefficients With Movie Data

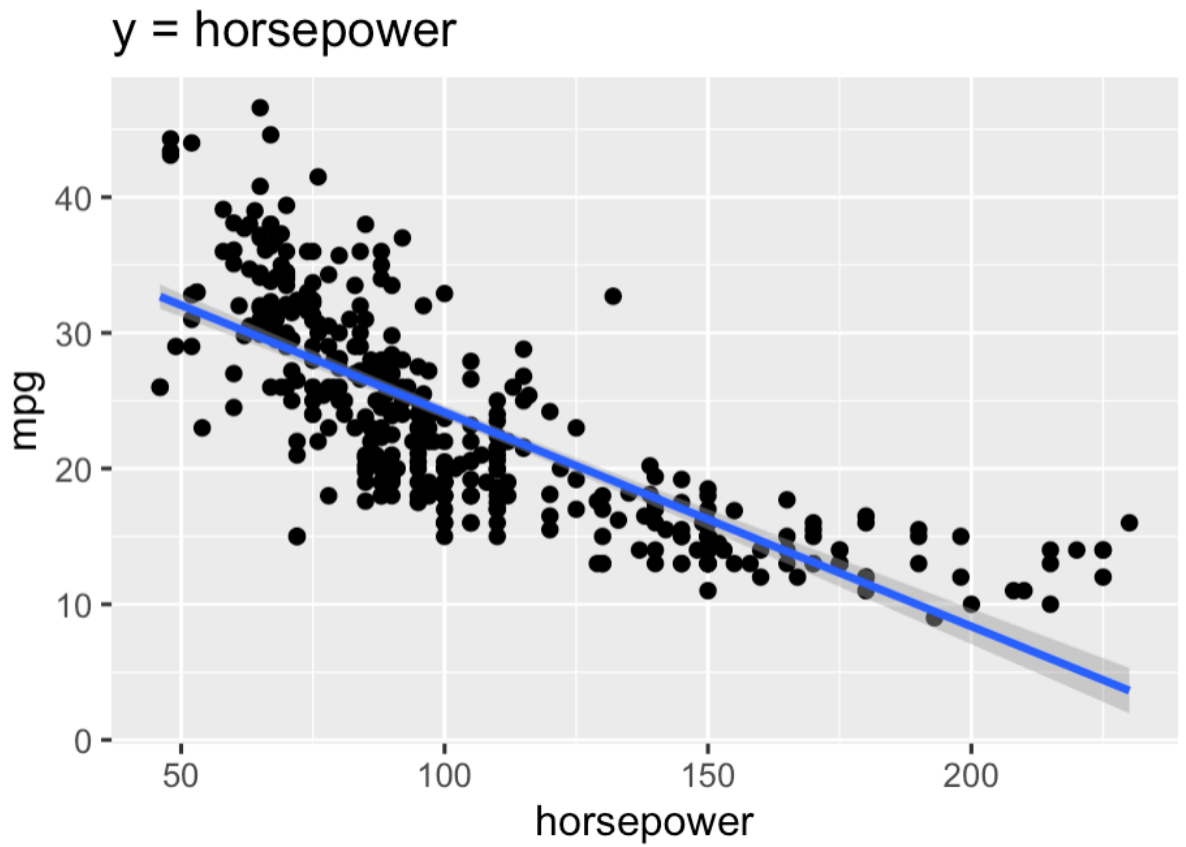
```
Call:
lm(formula = grossM ~ budgetM + imdb_score + director_easy, data = movies_train)

Residuals:
    Min      1Q  Median      3Q     Max
-394.72  -41.63  -17.02   18.62  673.14

Coefficients:
                              Estimate Std. Error t value Pr(>|t|)
(Intercept)                 -49.230235  18.042251  -2.729  0.00639 **
budgetM                       0.027101   0.005308   5.105 3.49e-07 ***
imdb_score                   16.748572   1.162255  14.410  < 2e-16 ***
director_easySteven Spielberg 85.683640  21.208287   4.040 5.47e-05 ***
director_easyOther           -2.137517  16.056864  -0.133  0.89411
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 69.67 on 3254 degrees of freedom
  (432 observations deleted due to missingness)
Multiple R-squared:  0.08398,   Adjusted R-squared:  0.08285
F-statistic: 74.58 on 4 and 3254 DF,  p-value: < 2.2e-16
```

“Re-leveling” Base of Factor with relevel()

```
Call:
lm(formula = grossM ~ budgetM + imdb_score + relevel(director_easy,
    ref = "Other"), data = movies_train)

Residuals:
    Min      1Q  Median      3Q     Max
-394.72  -41.63  -17.02   18.62  673.14

Coefficients:
                                                    Estimate Std. Error t value Pr(>|t|)
(Intercept)                                       -51.367752   7.554705  -6.799 1.24e-11 ***
budgetM                                             0.027101   0.005308   5.105 3.49e-07 ***
imdb_score                                         16.748572   1.162255  14.410  < 2e-16 ***
relevel(director_easy, ref = "Other")Clint Eastwood  2.137517  16.056864   0.133    0.894
relevel(director_easy, ref = "Other")Steven Spielberg 87.821157  14.049186   6.251 4.61e-10 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 69.67 on 3254 degrees of freedom
  (432 observations deleted due to missingness)
Multiple R-squared:  0.08398,   Adjusted R-squared:  0.08285
F-statistic: 74.58 on 4 and 3254 DF,  p-value: < 2.2e-16
```

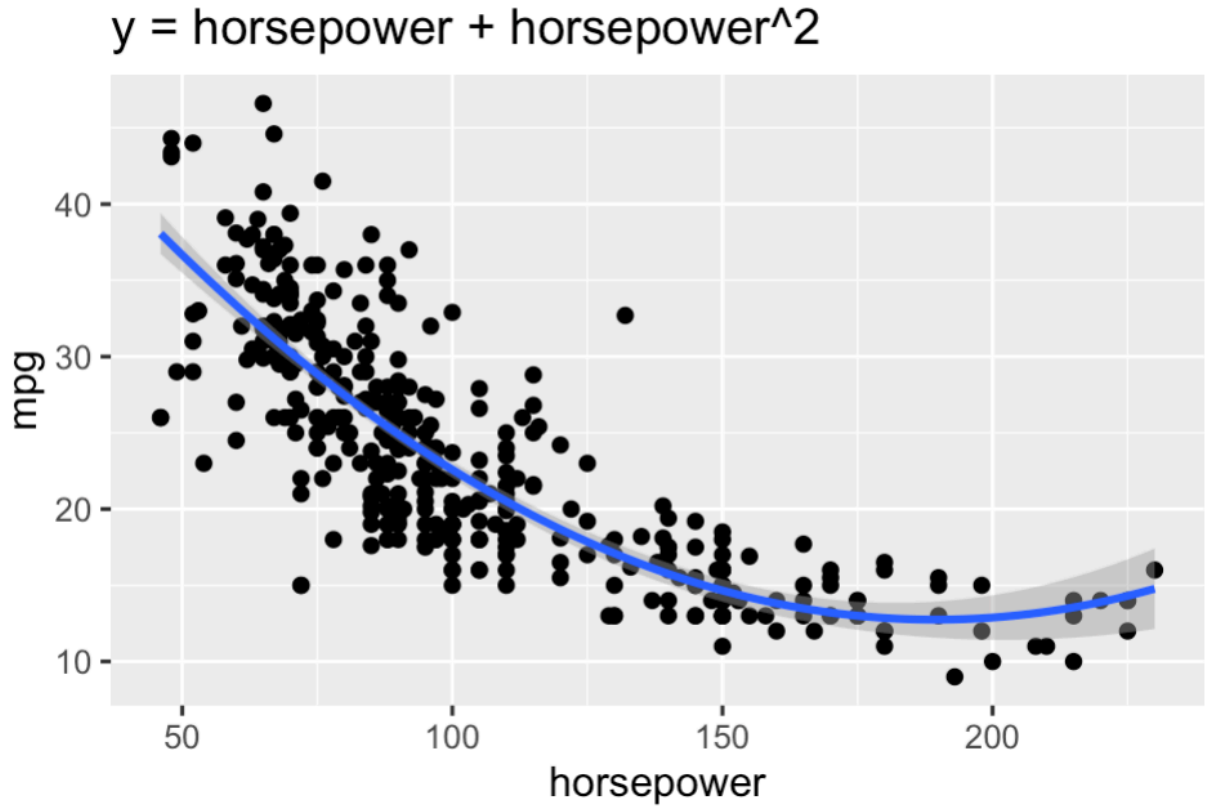
Nonlinear (Squared) Predictors

$$mpg_i = \beta_0 + \beta_1 \cdot horsepower + \epsilon_i$$



Nonlinear (Squared) Predictors

$$mpg_i = \beta_0 + \beta_1 \cdot horsepower + \epsilon_i$$



$$mpg_i = \beta_0 + \beta_1 \cdot horsepower + \beta_2 \cdot horsepower^2 + \epsilon_i$$

Nonlinear (Squared) Predictors

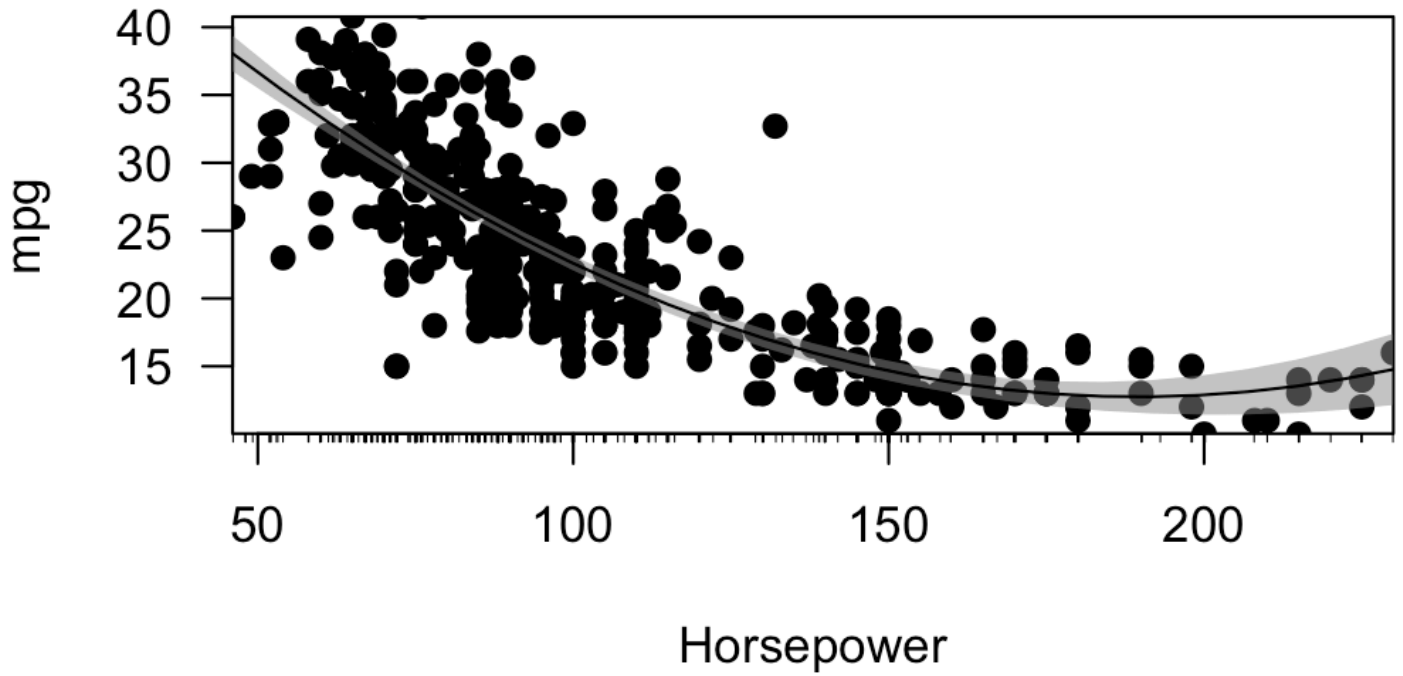
$$mpg_i = \beta_0 + \beta_1 \cdot horsepower + \beta_2 \cdot horsepower^2 + \epsilon_i$$

```
> summary(mod1)$coefficients
                    Estimate   Std. Error   t value      Pr(>|t|)
(Intercept)     56.900099702 1.8004268063  31.60367 1.740911e-109
horsepower      -0.466189630 0.0311246171 -14.97816  2.289429e-40
I(horsepower^2)  0.001230536 0.0001220759  10.08009  2.196340e-21
```

$Horsepower$	$Horsepower^2$	$\widehat{mpg} = \hat{\beta}_0 + \hat{\beta}_1 \cdot horsepower + \hat{\beta}_2 \cdot horsepower^2$
50	$50^2 = 2{,}500$	56.9 + -0.466*50 + 0.00125*2,500 = 36.66696
60	$60^2 = 3{,}600$	56.9 + -0.466*60 + 0.00125*60^2 =33.35865

Nonlinear (Squared) Predictors

$Horsepower$	$Horsepower^2$	$\widehat{mpg} = \hat{\beta}_0 + \hat{\beta}_1 \cdot horsepower + \hat{\beta}_2 \cdot horsepower^2$
50	$50^2 = 2{,}500$	56.9 + -0.466*50 + 0.00125*50^2 = 36.66696
60	$60^2 = 3{,}600$	56.9 + -0.466*60 + 0.00125*60^2 =33.35865



Squared X Terms (Feature Selection)

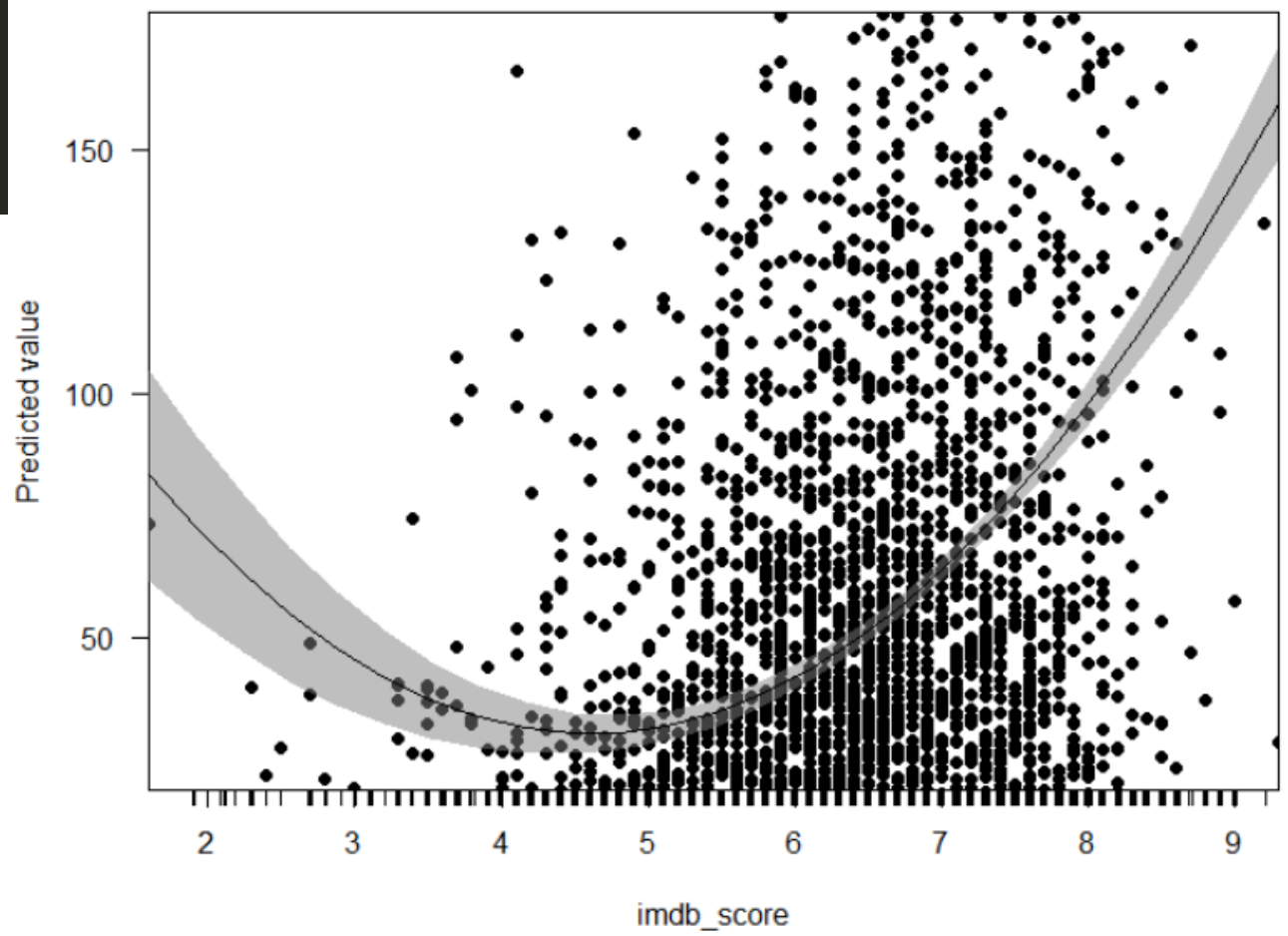
```
> mod8 <- lm(grossM ~ logbudgetM + imdb_score + I(imdb_score^2), data = movies_train)
> summary(mod8)

Call:
lm(formula = grossM ~ logbudgetM + imdb_score + I(imdb_score^2),
    data = movies_train)

Residuals:
    Min      1Q  Median      3Q     Max
-153.46  -34.11   -9.74   20.56  585.76

Coefficients:
                Estimate Std. Error t value Pr(>|t|)
(Intercept)      34.4480    20.1669   1.708   0.0877 .
logbudgetM       36.7661     0.9689  37.946  < 2e-16 ***
imdb_score      -54.0149     6.6363  -8.139 5.51e-16 ***
I(imdb_score^2)   5.8641     0.5426  10.807  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 58.05 on 3420 degrees of freedom
  (332 observations deleted due to missingness)
Multiple R-squared:  0.3477,    Adjusted R-squared:  0.3471
F-statistic: 607.7 on 3 and 3420 DF,  p-value: < 2.2e-16
```

Interpreting Squared X Terms Using Margins Package

```
# marginal effects of imdb score
install.packages('margins')
library('margins')
m <- margins::margins(mod8, at = list(imdb_score = c(1:10)))
m
margins::cplot(mod8, x = "imdb_score", what = "prediction",
               scatter = TRUE)
```

at(imdb_score)	logbudgetM	imdb_score
1	36.77	-42.287
2	36.77	-30.559
3	36.77	-18.830
4	36.77	-7.102
5	36.77	4.626
6	36.77	16.354
7	36.77	28.082
8	36.77	39.810
9	36.77	51.539
10	36.77	63.267

```
>
```

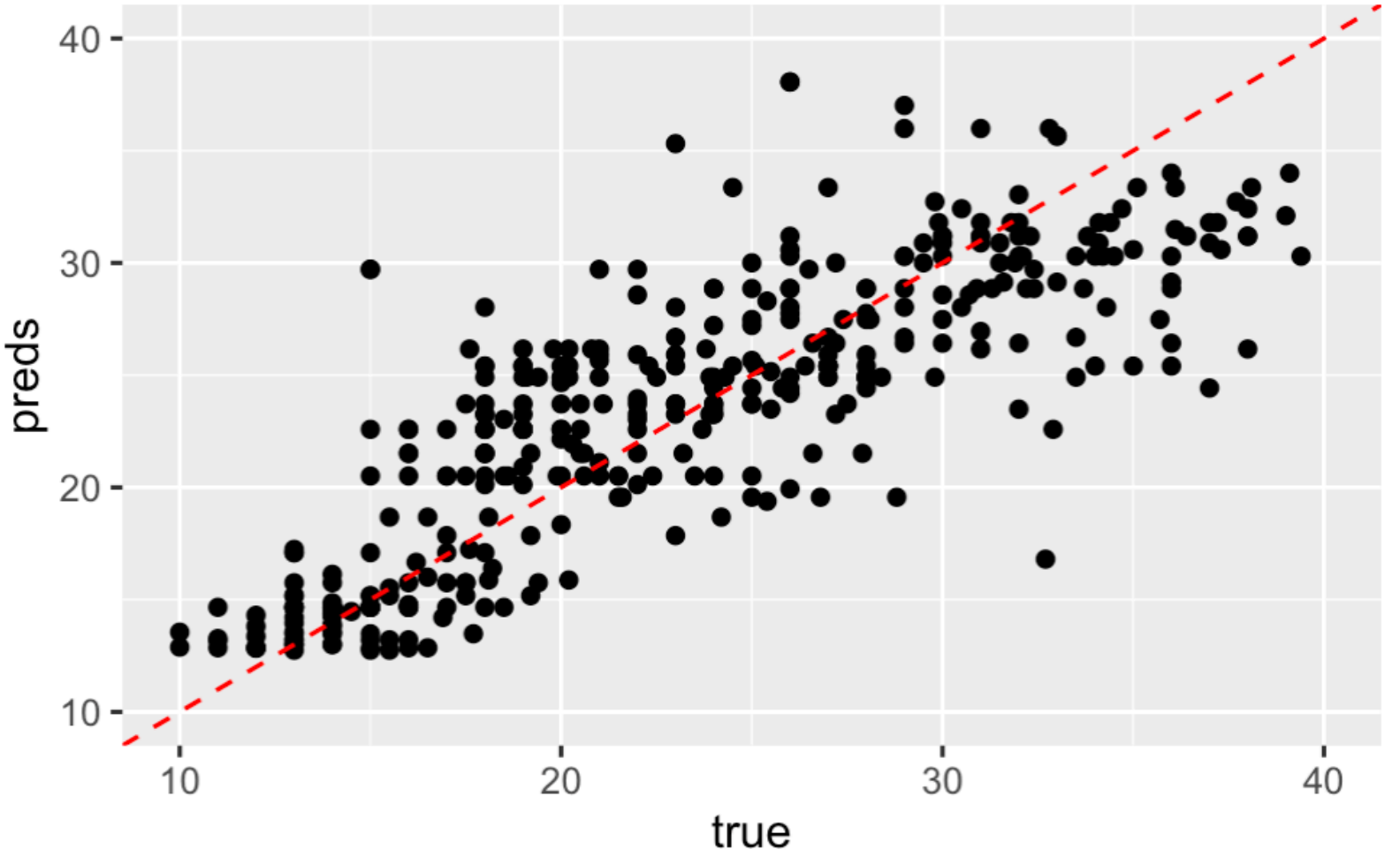


Generating Model Predictions

```r
# generating predictions
preds <- predict(mod1)
```

```r
# put model preds and true in a data frame
mod1_df <- data.frame(
  preds = preds,
  true = Auto$mpg
)
```

Predicted True Plots




```
ggplot(mod1_df, aes(x = true, y = preds)) + geom_point() +
  geom_abline(color = "red", linetype = "dashed") +
  xlim(10,40) + ylim(10,40)
```

Residuals: mod1$residuals

$$\hat{\epsilon}_i = y_i - \hat{y}_i$$

$$\hat{\epsilon}_i = y_i - \beta_0 - \beta_1 \cdot horsepower + \beta_2 \cdot horsepower^2$$

```
# model residuals
mod1_df <- data.frame(
  preds = preds,
  true = Auto$mpg,
  resids = mod1$residuals
)
```

In Class Exercise

1. Install the ‘ISLR’ package and load the ‘Auto’ dataset (data(Auto))
2. Build your favorite model predicting mpg
3. Feel free to transform variables as you so desire
4. Produce various predicted/true plots (add your name as a title) and we can compare

Auto {ISLR} R Documentatio

Auto Data Set

Description

Gas mileage, horsepower, and other information for 392 vehicles.

Usage

```
Auto
```

Format

A data frame with 392 observations on the following 9 variables.

`mpg`
: miles per gallon

`cylinders`
: Number of cylinders between 4 and 8

`displacement`
: Engine displacement (cu. inches)

`horsepower`
: Engine horsepower

`weight`
: Vehicle weight (lbs.)

`acceleration`
: Time to accelerate from 0 to 60 mph (sec.)

`year`
: Model year (modulo 100)

`origin`
: Origin of car (1. American, 2. European, 3. Japanese)

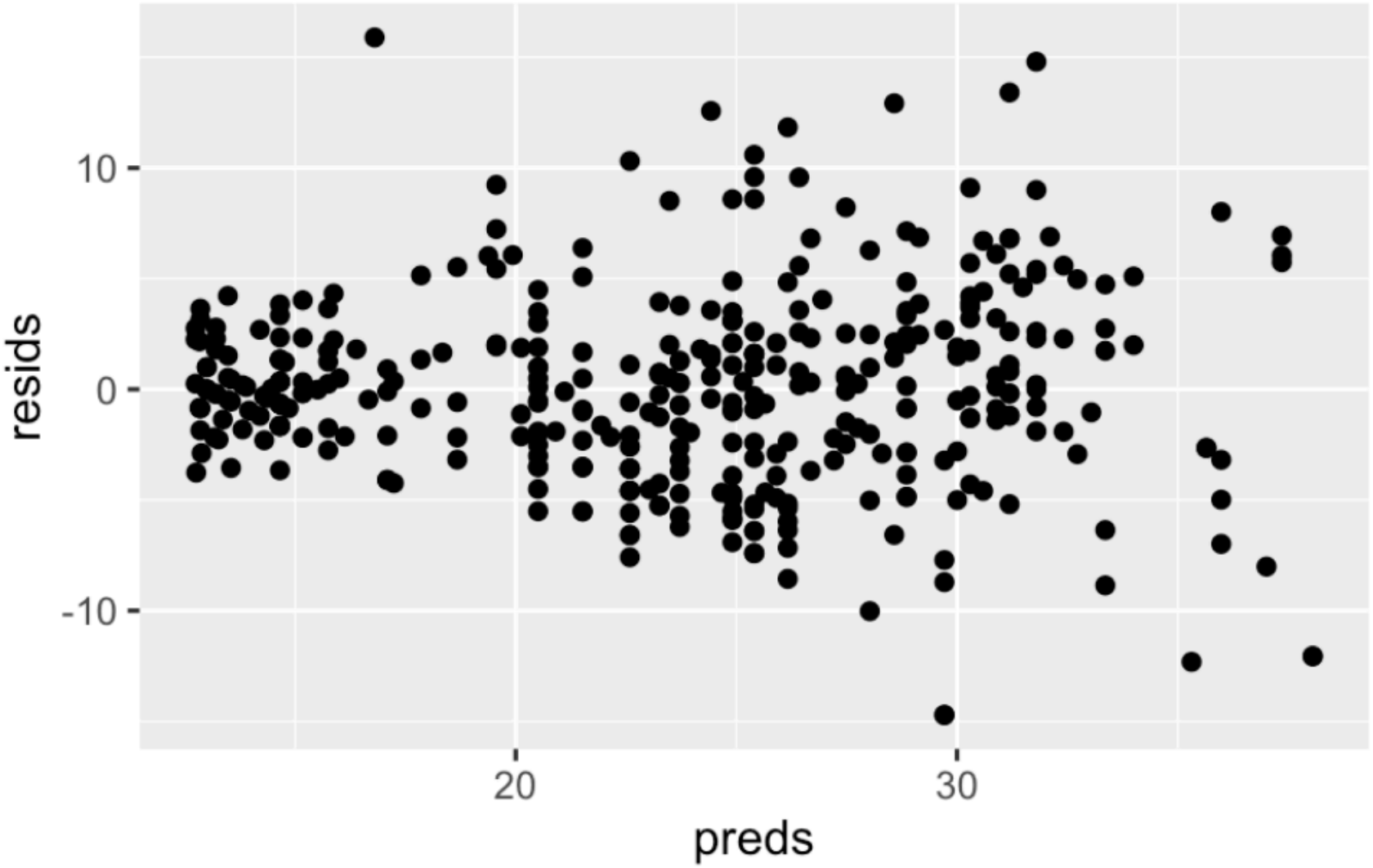
`name`
: Vehicle name

The orginal data contained 408 observations but 16 observations with missing values were removed.

Source

This dataset was taken from the StatLib library which is maintained at Carnegie Mellon University. The dataset was used in the 1983 American Statistical Association Exposition.

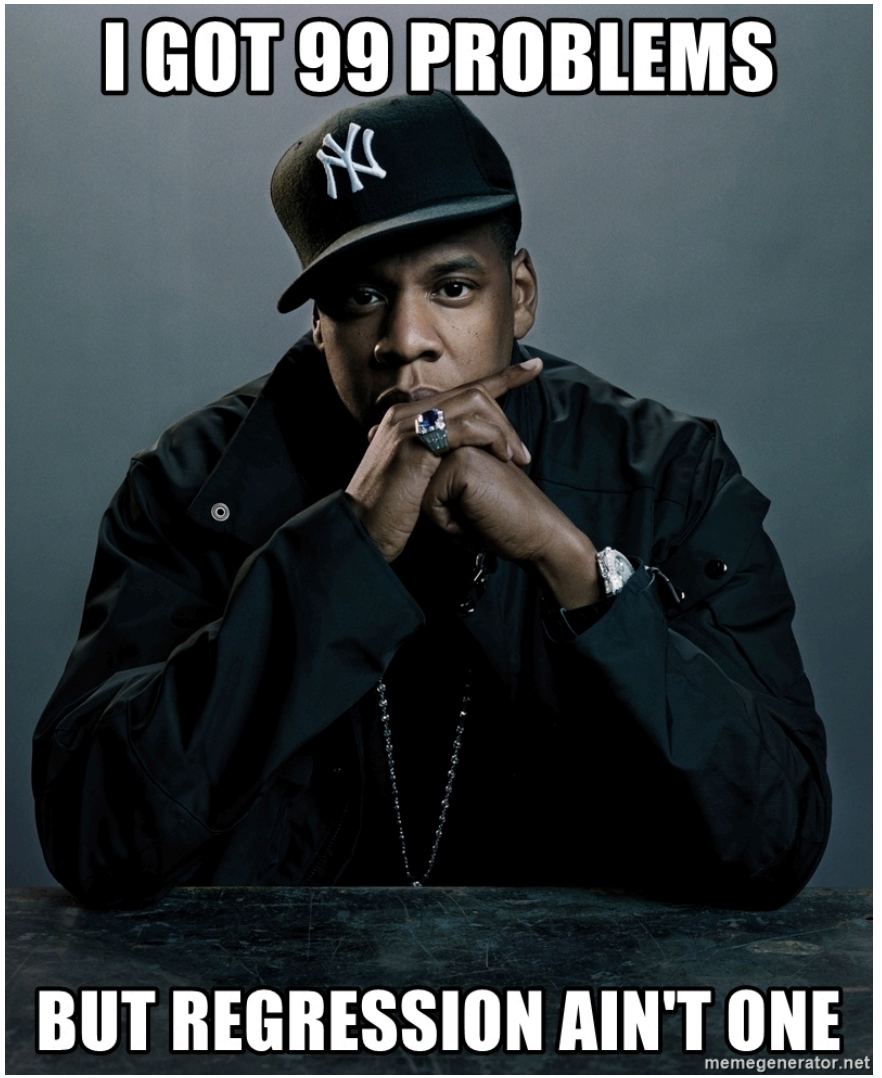
Plot of Residuals Against Fitted Values



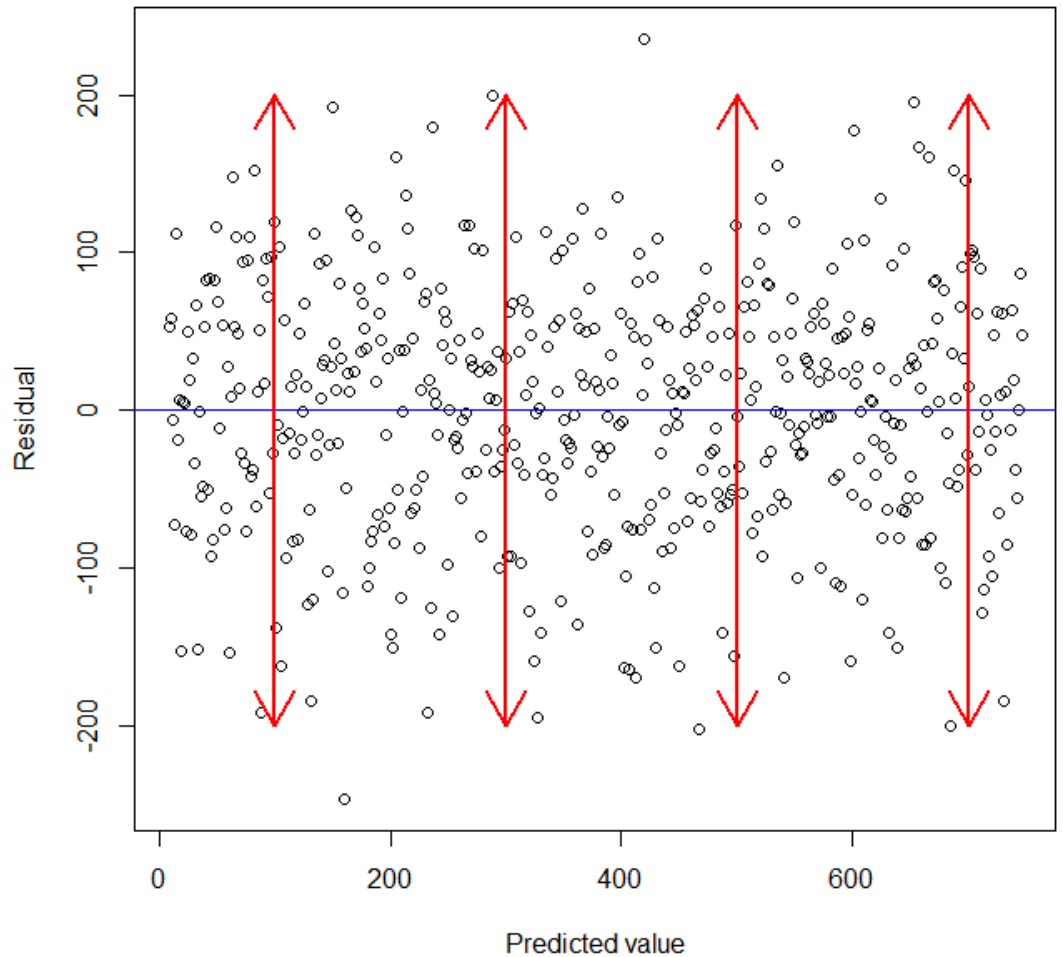

```
ggplot(mod1_df, aes(x = preds, y = resids)) + geom_point()
```

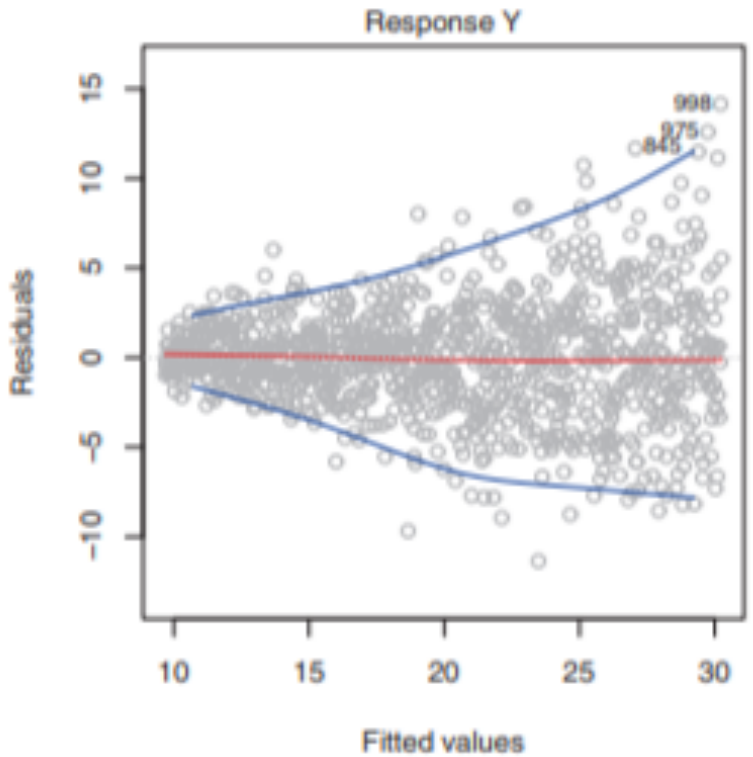
Potential Problems with Regressions

1. Non-linearity of Xs
2. Non-constant variance of error term
3. Correlation of error terms
4. Outliers
5. High-leverage points
6. Collinearity
7. Correlation vs Causation



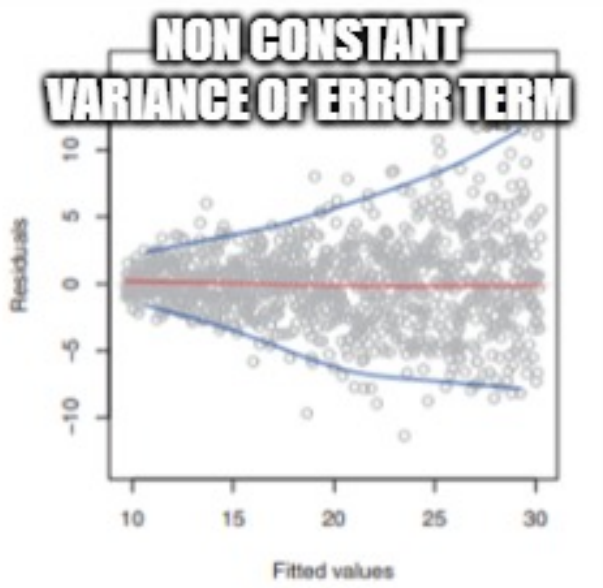
Heteroskedasticity: non-constant variance of errors

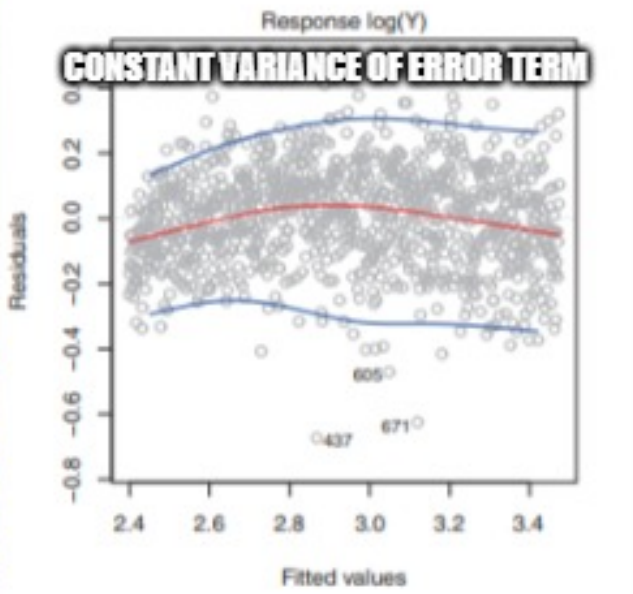




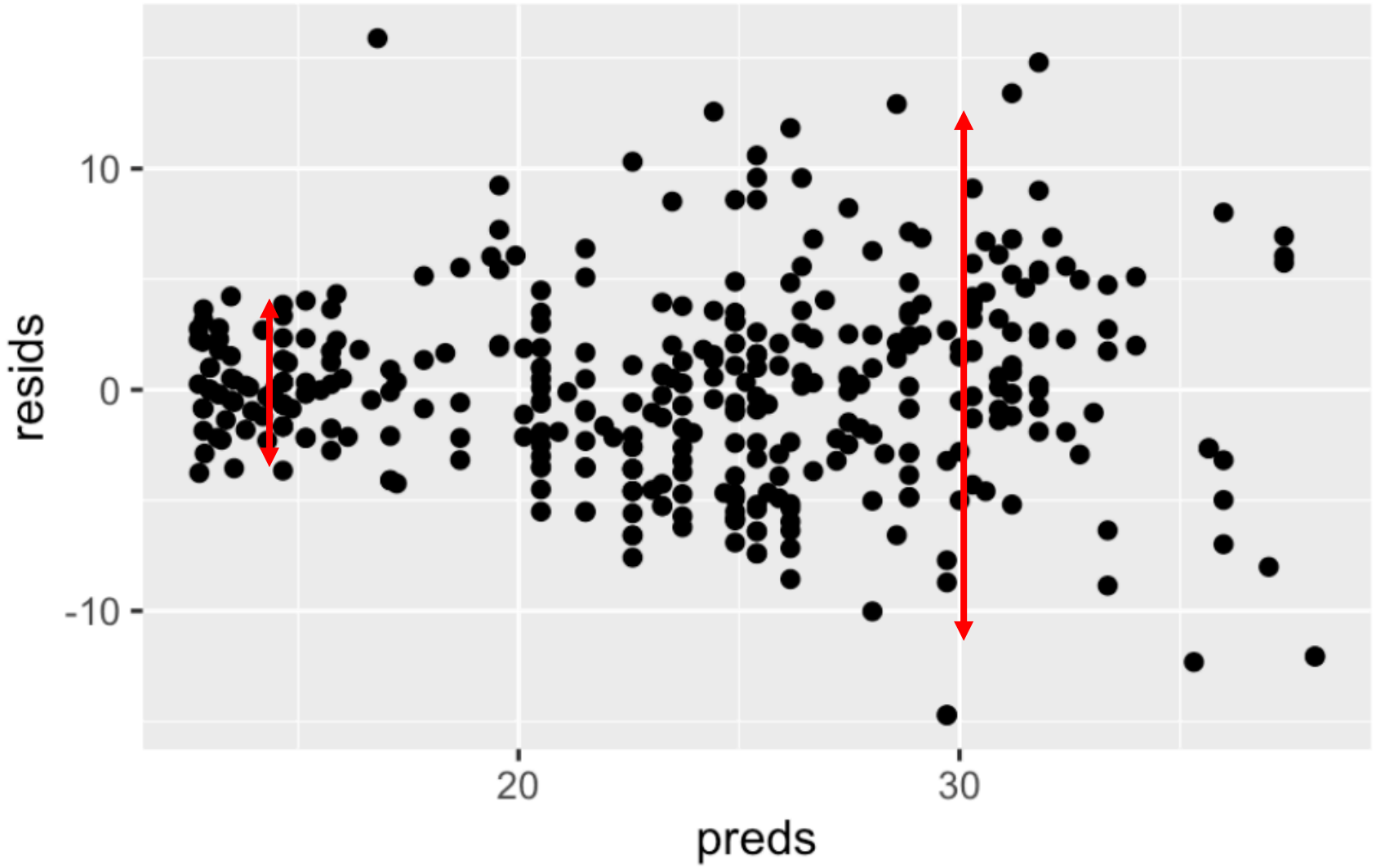

FUTUBANDERA.CL


imgflip.com

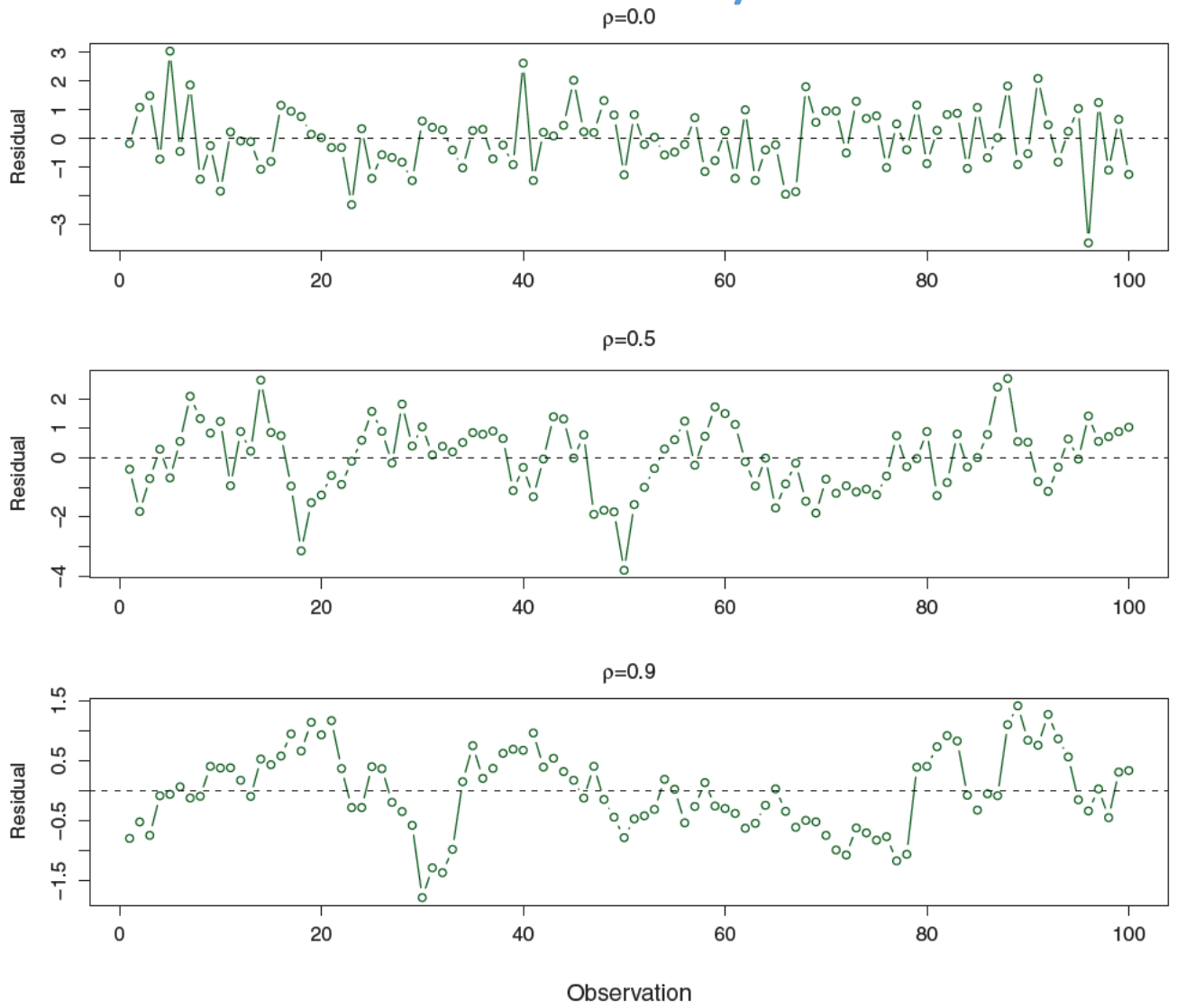

NON CONSTANT
VARIANCE OF ERROR TERM
Residuals
10
5
0
−5
−10
10
15
20
25
30
Fitted values


Response log(Y)
CONSTANT VARIANCE OF ERROR TERM
Residuals
0.2
0.0
−0.2
−0.4
−0.6
−0.8
605
671
437
2.4
2.6
2.8
3.0
3.2
3.4
Fitted values

Do We Have Constant Error Variance?



Correlation of Error Terms: Property of Time Series. Don't worry about this



Outliers

```
Auto_sub <- data.frame(Auto, resids = mod1$residuals) %>%
  arrange(desc(resids))
Auto_sub %>% slice(1)
Auto_sub %>% slice(392)
```

```
  mpg cylinders displacement horsepower weight acceleration year origin          name    resids
1  15         6          250         72   3158         19.5   75      1 ford maverick -14.71355
> Auto_sub %>% slice(1)
   mpg cylinders displacement horsepower weight acceleration year origin          name   resids
1 32.7         6          168        132   2910         11.4   80      3 datsun 280-zx 15.89607
```

Ford Maverick

Datsun (Nissan) 280ZX

Why are these outliers?

High Leverage Points

Leverage (aka hat value) of observation *i*: how much would all our predictions change if we exclude observation *I*

$$h_i = \frac{1}{n} + \frac{(x_i - \bar{x})^2}{\sum_{i'=1}^{n}(x_{i'} - \bar{x})^2}$$

$$e.g., x_i = jeff\ bezos\ net\ wealth, \quad n = 100$$

$$h_i \in \left[\frac{1}{n}, 1\right]$$

$$h_{billgates} = \frac{1}{100} + \frac{(114B - 100k)^2}{\sum_{i'=1}^{n}(x_{i'} - \bar{x})^2}$$

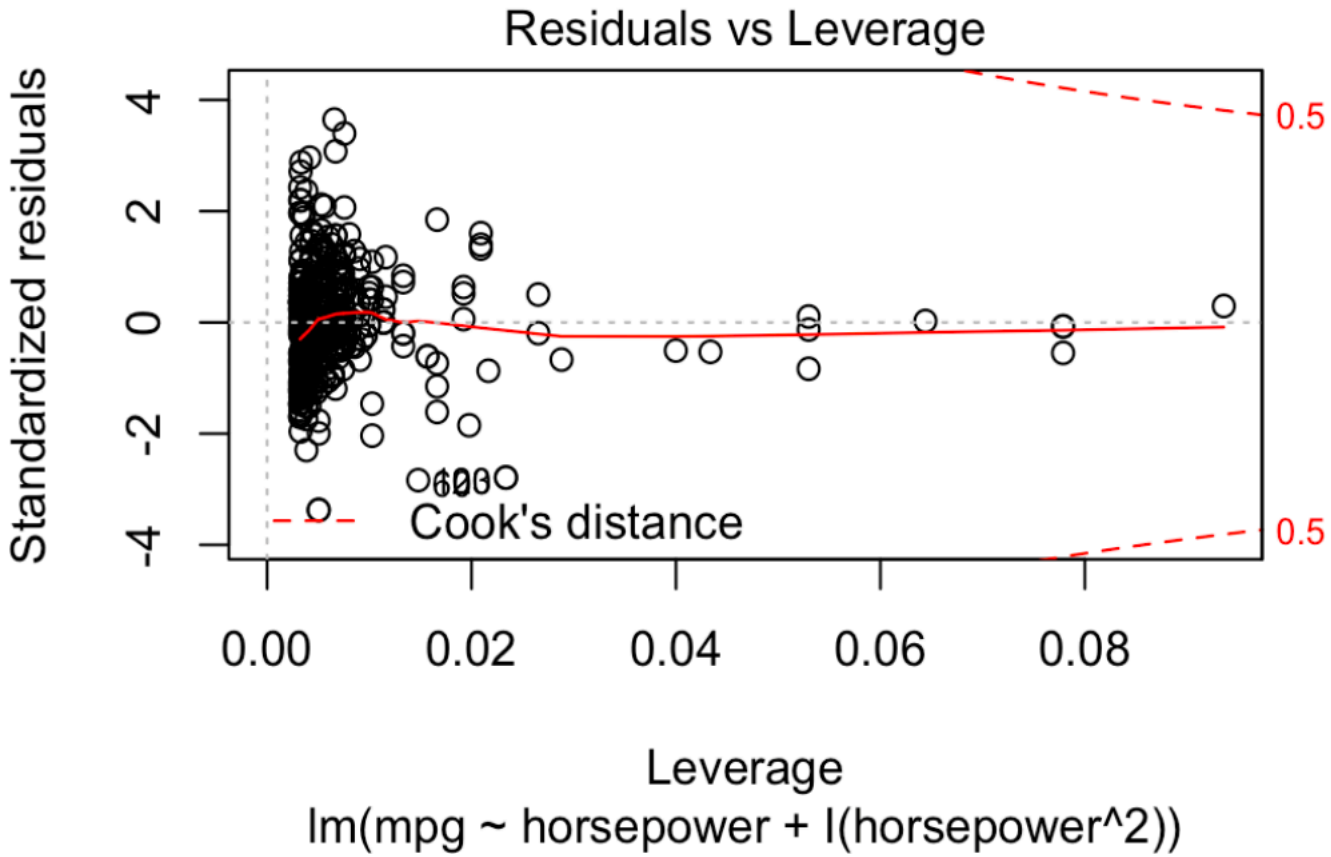
$$Average = (p+1)/n$$

High Leverage Points in Auto

```
Auto_sub <- data.frame(Auto, hats = hatvalues(mod1))
Auto_sub %>% arrange(desc(hats)) %>% slice(1:2)
```

```
  mpg cylinders displacement horsepower weight acceleration year origin                name       hats
1  16         8          400        230   4278          9.5   73      1 pontiac grand prix 0.09359709
2  14         8          455        225   4425         10.0   70      1   pontiac catalina 0.07787099
```

Plotting High Leverage Points Using plot(mod)



Collinearity

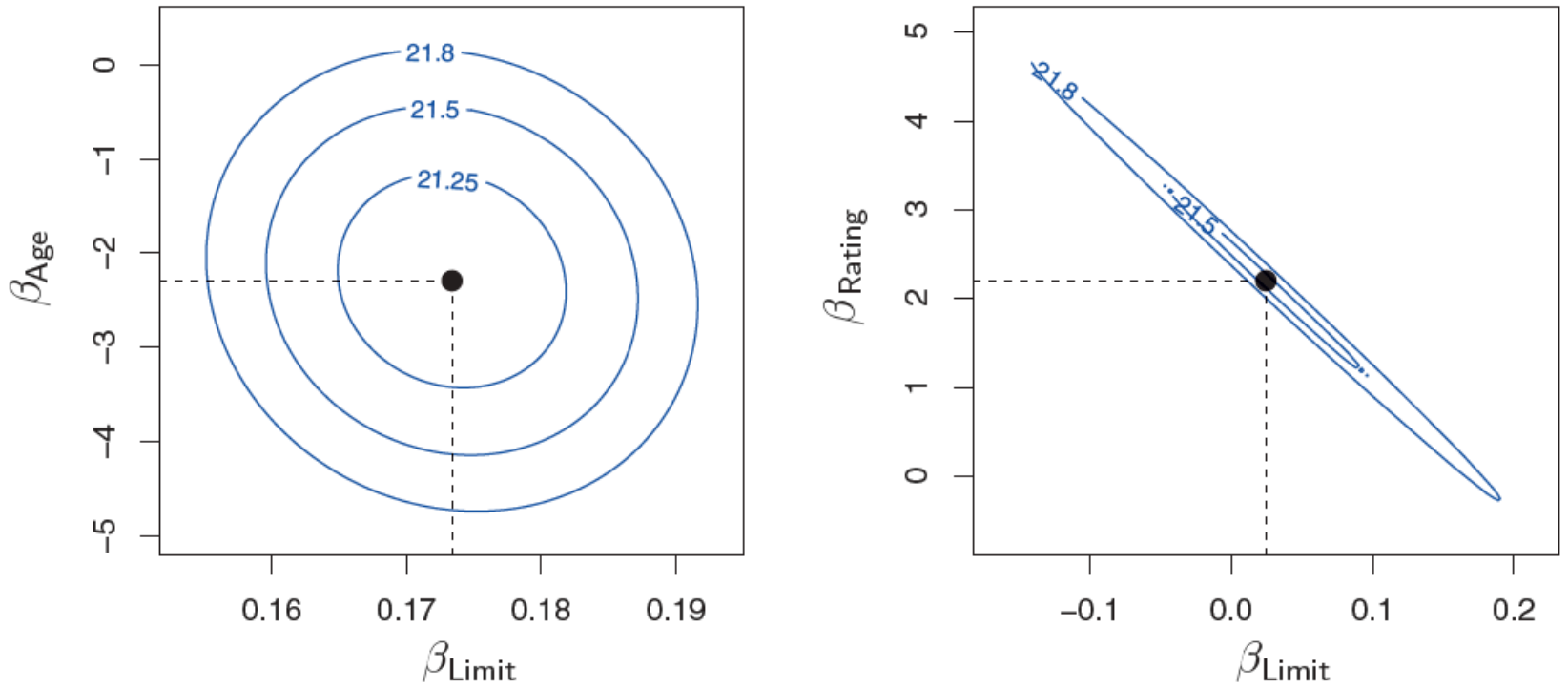


FIGURE 3.15. *Contour plots for the RSS values as a function of the parameters β for various regressions involving the* `Credit` *data set. In each plot, the black dots represent the coefficient values corresponding to the minimum RSS.* Left: *A contour plot of RSS for the regression of* `balance` *onto* `age` *and* `limit`*. The minimum value is well defined.* Right: *A contour plot of RSS for the regression of* `balance` *onto* `rating` *and* `limit`*. Because of the collinearity, there are many pairs* $(\beta_{\text{Limit}}, \beta_{\text{Rating}})$ *with a similar value for RSS.*

Impact of collinearity

		Coefficient	Std. error	t-statistic	p-value
Model 1	Intercept	−173.411	43.828	−3.957	< 0.0001
	age	−2.292	0.672	−3.407	0.0007
	limit	0.173	0.005	34.496	< 0.0001
Model 2	Intercept	−377.537	45.254	−8.343	< 0.0001
	rating	2.202	0.952	2.312	0.0213
	limit	0.025	0.064	0.384	0.7012

TABLE 3.11. *The results for two multiple regression models involving the* `Credit` *data set are shown. Model 1 is a regression of* `balance` *on* `age` *and* `limit`*, and Model 2 a regression of* `balance` *on* `rating` *and* `limit`*. The standard error of* $\hat{\beta}_{\text{limit}}$ *increases 12-fold in the second regression, due to collinearity.*

Testing for collinearity: VIF (olsrr package)

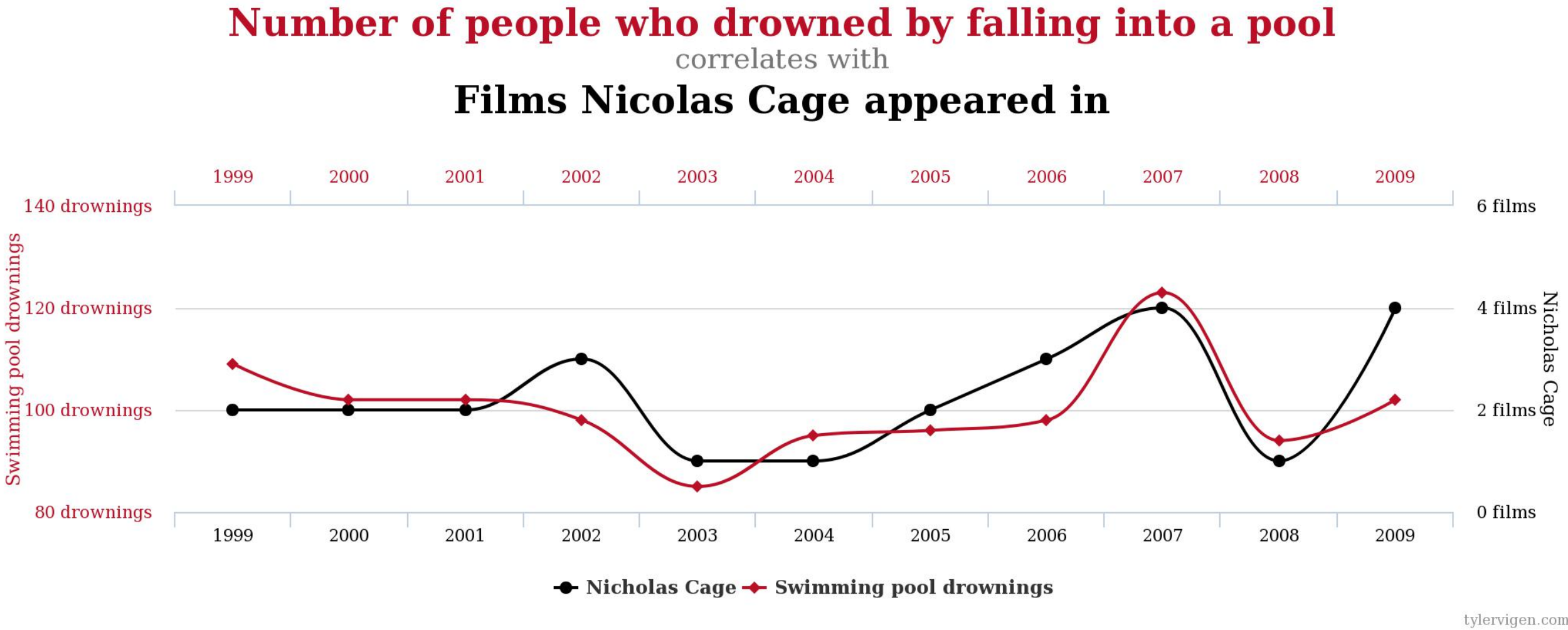
$$\mathrm{VIF}(\hat{\beta}_j) = \frac{1}{1-R^2_{X_j|X_{-j}}}$$

Where $R^2_{X_j|X_{-j}}$ is the R^2 on all other predictors

```
> library('olsrr')
> ols_vif_tol(mod1)
# A tibble: 2 x 3
  Variables       Tolerance   VIF
  <chr>               <dbl> <dbl>
1 horsepower         0.0341  29.3
2 I(horsepower^2)    0.0341  29.3
```

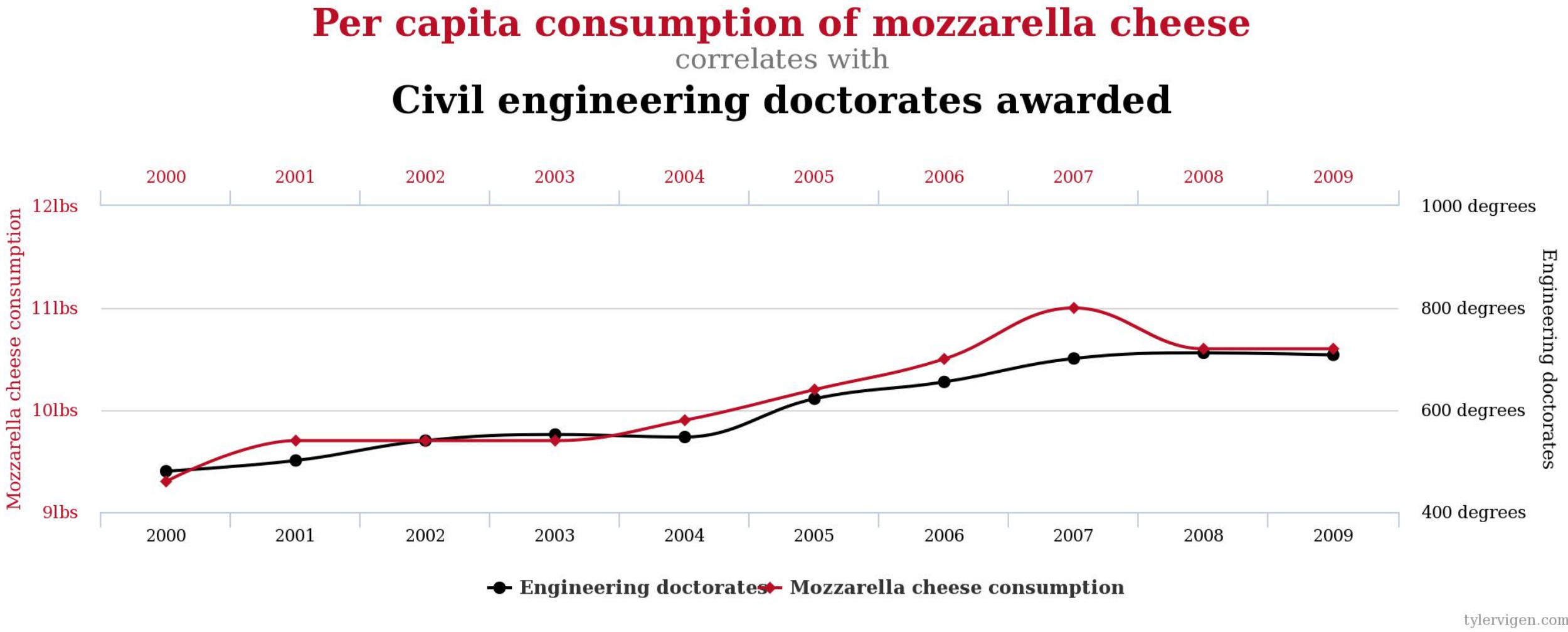
VIF > 5 or 10 indicates problematic level of multicollinearity

Spurious Relationship?



Source: http://www.tylervigen.com/spurious-correlations

Spurious Relationship?



Source: http://www.tylervigen.com/spurious-correlations