

# Assumptions & Diagnostics

Utrecht University Winter School: Regression in R



**Utrecht  
University**

Kyle M. Lang

Department of Methodology & Statistics  
Utrecht University

2022-02-02

# Outline

---

Assumptions of Linear Regression

Regression Diagnostics

Influential Observations

Treating Influential Observations



# Assumptions of MLR

---

The assumptions of the linear model can be stated as follows:

1. The model is linear in the parameters.
  - This is OK:  $Y = \beta_0 + \beta_1 X + \beta_2 Z + \beta_3 XZ + \beta_4 X^2 + \beta_5 X^3 + \varepsilon$
  - This is not:  $Y = \beta_0 X^{\beta_1} + \varepsilon$
2. The predictor matrix is *full rank*.
  - $N > P$
  - No  $X_p$  can be a linear combination of other predictors.



# Assumptions of MLR

---

3. The predictors are strictly exogenous.
  - The predictors do not correlated with the errors.
  - $\text{Cov}(\hat{Y}, \varepsilon) = 0$
  - $E[\varepsilon_n] = 0$
4. The errors have constant, finite variance.
  - $\text{Var}(\varepsilon_n) = \sigma^2 < \infty$
5. The errors are uncorrelated.
  - $\text{Cov}(\varepsilon_i, \varepsilon_j) = 0, i \neq j$
6. The errors are normally distributed.
  - $\varepsilon \sim N(0, \sigma^2)$



# Assumptions of MLR

---

The assumption of *spherical errors* combines Assumptions 4 and 5.

$$\text{Var}(\varepsilon) = \begin{bmatrix} \sigma^2 & 0 & \cdots & 0 \\ 0 & \sigma^2 & \cdots & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & \cdots & \sigma^2 \end{bmatrix} = \sigma^2 \mathbf{I}_N$$

We can combine Assumptions 3, 4, 5, and 6 by assuming independent and identically distributed normal errors:

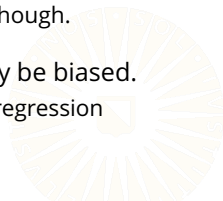
- $\varepsilon \stackrel{iid}{\sim} \mathbf{N}(\mathbf{0}, \sigma^2)$



# Consequences of Violating Assumptions

---

1. If the model is not linear in the parameters, then we're not even working with linear regression.
  - We need to move to entirely different modeling paradigm.
2. If the predictor matrix is not full rank, the model is not estimable.
  - The parameter estimates cannot be uniquely determined from the data.
3. If the predictors are not exogenous, the estimated regression coefficients will be biased.
4. If the errors are not spherical, the standard errors will be biased.
  - The estimated regression coefficients will be unbiased, though.
5. If errors are non-normal, small-sample inferences may be biased.
  - The justification for some tests and procedures used in regression analysis may not hold.



# Regression Diagnostics

---

If some of the assumptions are (grossly) violated, the inferences we make using the model may be wrong.

- We need to check the tenability of our assumptions before leaning too heavily on the model estimates.

These checks are called *regression diagnostics*.

- Graphical visualizations
- Quantitative indices/measures
- Formal statistical tests

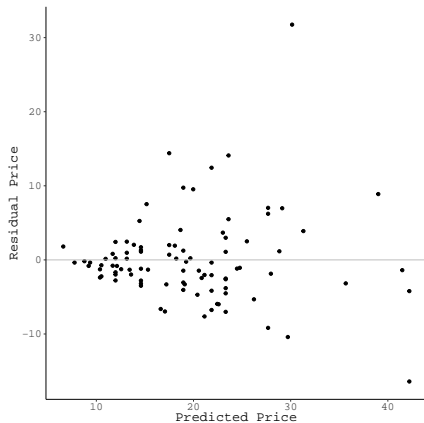
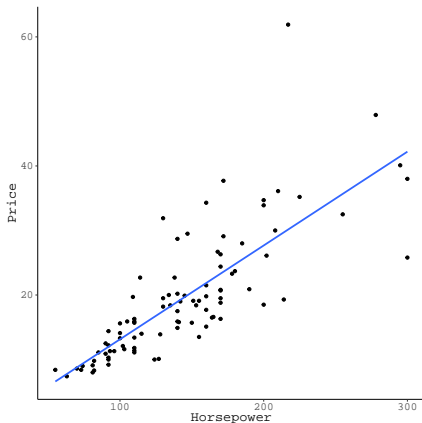


# Residual Plots

Plots of residuals vs. predicted values are very useful.

- Here we see clear evidence of heteroscedasticity.

```
out1 <- lm(Price ~ Horsepower, data = Cars93)
```



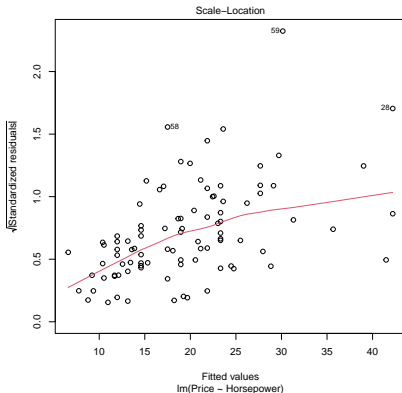


# Scale-Location Plots

```
plot(out1, 3)
```

Scale-location plots also offer an excellent means of detecting non-constant error variance.

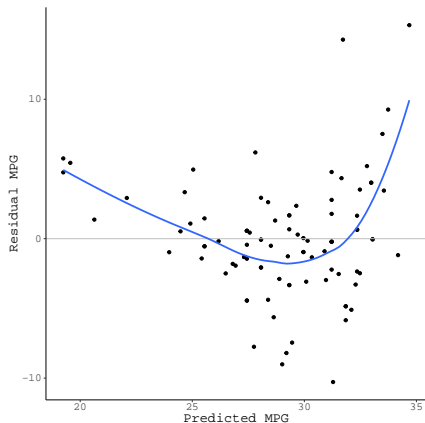
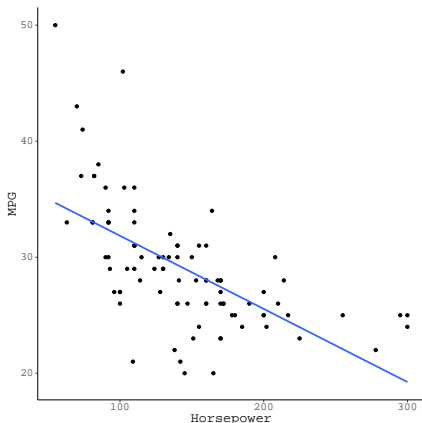
- Plot an approximation of the pointwise residual variance against the fitted values.
- Any trend indicates systematic changes in the residual variance.



# Residual Plots

Residual plots can also show violations of the linearity assumption.

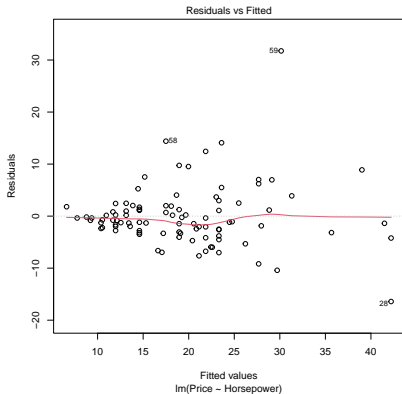
```
out2 <- lm(MPG.highway ~ Horsepower, data = Cars93)
```



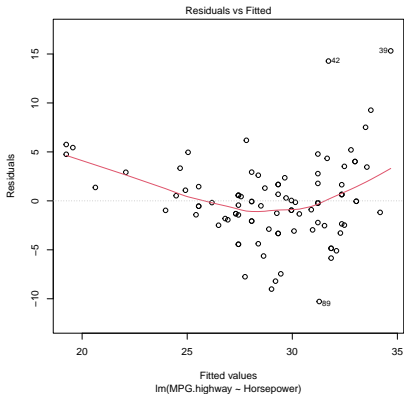
# Residual Plots

We can easily create residual plots from a fitted model object.

```
plot(out1, 1)
```



```
plot(out2, 1)
```



# Partial Residual Plots

---

In multiple linear regression, ordinary residual plots may not reveal nonlinearity.

- If we do find nonlinearity, ordinary residual plots can't tell us which term is causing the issue.

Partial residual plots show the trend for individual predictors, after controlling for all other variables in the model.

1. First, define the partial residual for the  $p$ th predictor.

$$\hat{\varepsilon}_n^{(p)} = \hat{\varepsilon}_n + \hat{\beta}_p X_{np}$$

2. Then, plot the partial residuals,  $\hat{\varepsilon}_n^{(p)}$ , against  $X_p$ , for all predictors.



# Partial Residual Plots

Let's look at an example. Consider the following model.

```
out3 <- lm(MPG.highway ~ Horsepower + Turn.circle, data = Cars93)
partSummary(out3, -1)
```

Residuals:

Min	1Q	Median	3Q	Max
-10.1346	-2.4247	-0.2107	2.1684	14.1929

Coefficients:

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	58.361713	5.339245	10.931	< 2e-16
Horsepower	-0.042480	0.009423	-4.508	1.96e-05
Turn.circle	-0.594652	0.153114	-3.884	0.000196

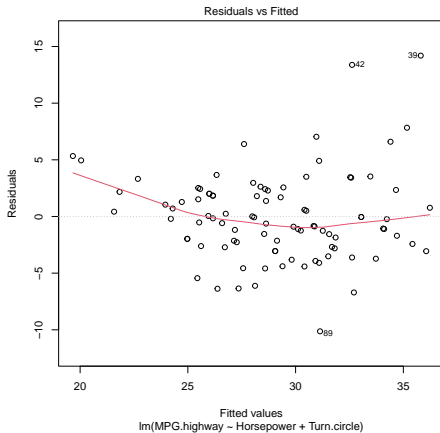
Residual standard error: 3.918 on 90 degrees of freedom

Multiple R-squared: 0.4717, Adjusted R-squared: 0.46

F-statistic: 40.19 on 2 and 90 DF, p-value: 3.372e-13

# Partial Residual Plots

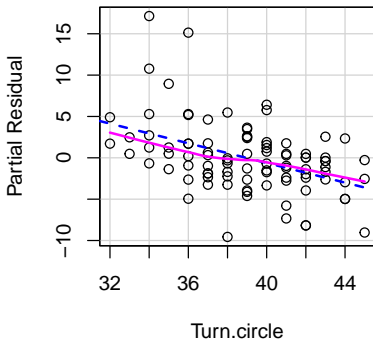
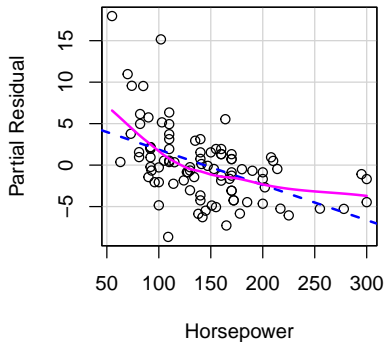
```
plot(out3, 1)
```



# Partial Residual Plots

```
crPlots(out3, ylab = "Partial Residual")
```

Component + Residual Plots



# Partial Residual Plots

Let's add the quadratic expansion of Horsepower.

```
out4 <- update(out3, ". ~ . + poly(Horsepower, 2) - Horsepower")  
partSummary(out4, -1)
```

Residuals:

Min	1Q	Median	3Q	Max
-10.2650	-2.2447	-0.2369	2.3775	13.2971

Coefficients:

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	43.6770	6.0648	7.202	1.82e-10
Turn.circle	-0.3745	0.1554	-2.411	0.017982
poly(Horsepower, 2)1	-25.1594	4.5537	-5.525	3.24e-07
poly(Horsepower, 2)2	14.6465	3.9757	3.684	0.000394

Residual standard error: 3.67 on 89 degrees of freedom

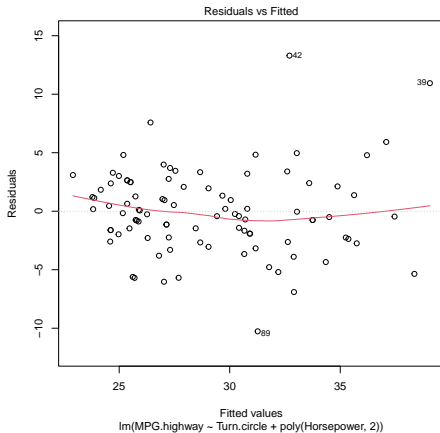
Multiple R-squared: 0.5416, Adjusted R-squared: 0.5262

F-statistic: 35.06 on 3 and 89 DF, p-value: 4.729e-15



# Partial Residual Plots

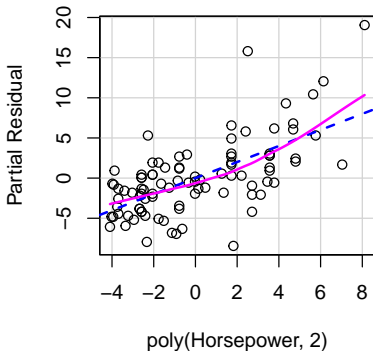
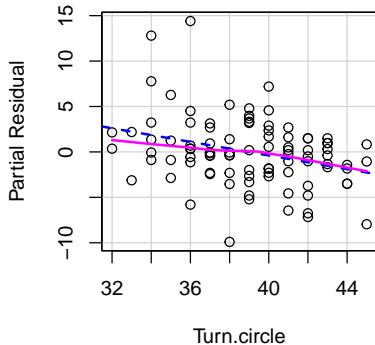
```
plot(out4, 1)
```



# Partial Residual Plots

```
crPlots(out4, ylab = "Partial Residual")
```

Component + Residual Plots

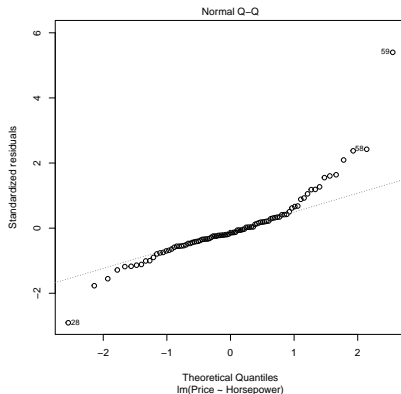


# QQ-Plots

```
plot(out1, 2)
```

A normal Q-Q Plot is one of the best ways to evaluate the normality assumption.

- Plot the quantiles of the residual distribution against the theoretically ideal quantiles.
- We can actually use a Q-Q Plot to compare any two distributions.



# Influential Observations



# Influential Observations

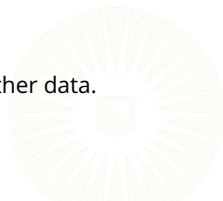
---

Influential observations contaminate analyses in two ways:

1. Exert too much influence on the fitted regression model
2. Invalidate estimates/inferences by violating assumptions

There are two distinct types of influential observations:

1. Outliers
  - Observations with extreme outcome values, relative to the other data.
  - Observations with outcome values that fit the model very badly.
2. High-leverage observations
  - Observation with extreme predictor values, relative to other data.



# Outliers

---

Outliers can be identified by scrutinizing the residuals.

- Observations with residuals of large magnitude may be outliers.
- The difficulty arises in quantifying what constitutes a “large” residual.

If the residuals do not have constant variance, then we cannot directly compare them.

- We need to standardize the residuals in some way.



# Studentized Residuals

---

Begin by defining the concept of a *deleted residual*:

$$\hat{\varepsilon}_{(n)} = Y_n - \hat{Y}_{(n)}$$

- $\hat{\varepsilon}_{(n)}$  quantifies the distance of  $Y_n$  from the regression line estimated after excluding the  $n$ th observation.

If we standardize the deleted residual,  $\hat{\varepsilon}_{(n)}$ , we get the externally studentized residual:

$$t_{(n)} = \frac{\hat{\varepsilon}_{(n)}}{SE_{\hat{\varepsilon}_{(n)}}}$$

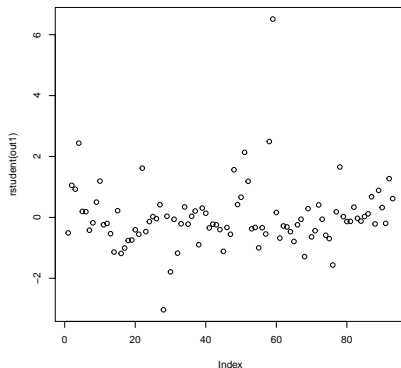


# Studentized Residual Plots

```
plot(rstudent(out1))
```

Index plots of the externally studentized residuals can help spotlight potential outliers.

- Look for observations that clearly “stand out from the crowd.”





# High-Leverage Points

---

We identify high-leverage observations through their *leverage* values.

- An observation's leverage,  $h_n$ , quantifies the extent to which its predictors affect the fitted regression model.
- Observations with  $X$  values very far from the mean,  $\bar{X}$ , affect the fitted model disproportionately.



# High-Leverage Points

---

We identify high-leverage observations through their *leverage* values.

- An observation's leverage,  $h_n$ , quantifies the extent to which its predictors affect the fitted regression model.
- Observations with  $X$  values very far from the mean,  $\bar{X}$ , affect the fitted model disproportionately.

In simple linear regression, the  $n$ th leverage is given by:

$$h_n = \frac{1}{N} + \frac{(X_n - \bar{X})^2}{\sum_{m=1}^N (X_m - \bar{X})^2}$$

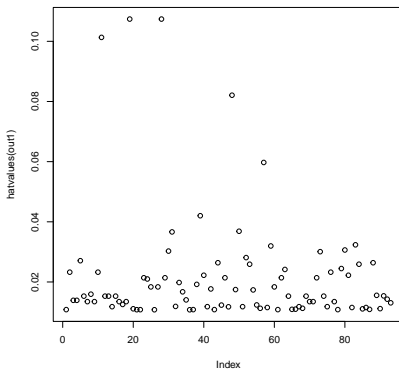


# Leverage Plots

```
plot(hatvalues(out1))
```

Index plots of the leverage values can help spotlight high-leverage points.

- Again, look for observations that clearly “stand out from the crowd.”



# Outliers & Leverages → Influential Points

---

Observations with high leverage or large (externally) studentized residuals are not necessarily influential.

- High-leverage observations tend to be more influential than outliers.
- The worst problems arise from observations that are both outliers and have high leverage.

*Measures of influence* simultaneously consider extremity in both  $X$  and  $Y$  dimensions.

- Observations with high measures of influence are very likely to cause problems.



# Measures of Influence

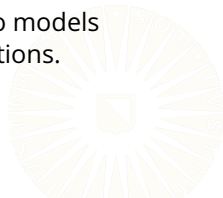
---

Measures of influence come in two flavors.

1. Global measures of influence
  - Cook's Distance
2. Coefficient-specific measures of influence
  - DFBETAS

All measures of influence use the same logic as the deleted residual.

- Compare models estimated from the whole sample to models estimated from samples excluding individual observations.



# Global Measures of Influence

---

Each observation gets a Cook's Distance value.

$$\begin{aligned}\text{Cook's } D_n &= \frac{\sum_{n=1}^N \left( \hat{Y}_n - \hat{Y}_{(n)} \right)^2}{(P+1) \hat{\sigma}^2} \\ &= (P+1)^{-1} t_n^2 \frac{h_n}{1-h_n}\end{aligned}$$

Each regression coefficient (including the intercept) gets a DFBETAS value for each observation.

$$\text{DFBETAS}_{np} = \frac{\hat{\beta}_p - \hat{\beta}_{p(n)}}{\text{SE}_{\hat{\beta}_{p(n)}}}$$

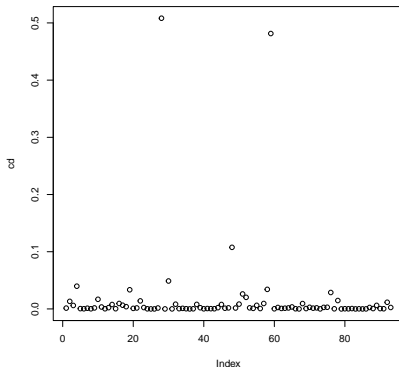


# Plots of Cook's Distance

```
cd <- cooks.distance(out1)  
plot(cd)
```

Index plots of Cook's distances can help spotlight the influential points.

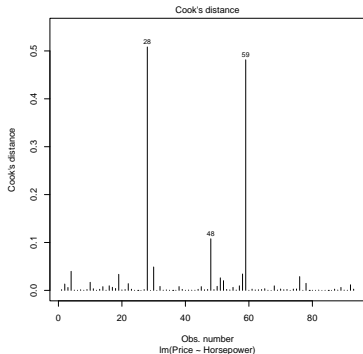
- Look for observations that clearly “stand out from the crowd.”



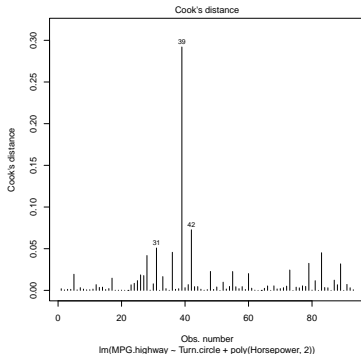
# Plots of Cook's Distance

We can create Cook's Distance plots by plotting a fitted model object.

```
plot(out1, 4)
```



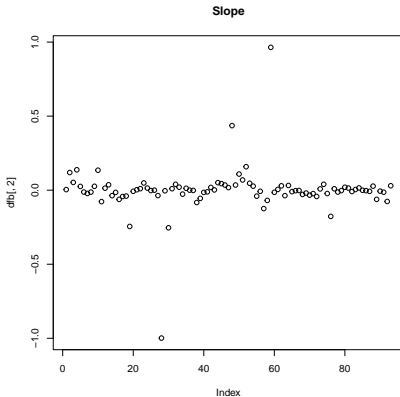
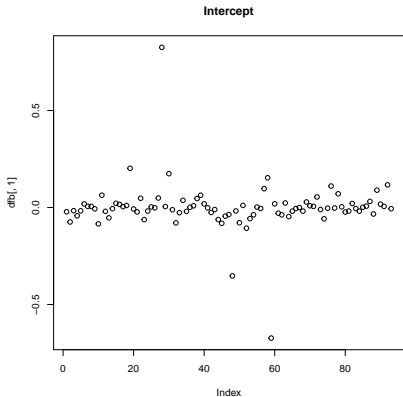
```
plot(out4, 4)
```





# Plots of DFBETAS

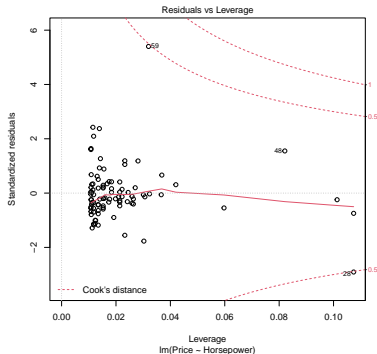
```
dfb <- dfbetas(out1)
plot(dfb[, 1], main = "Intercept")
plot(dfb[, 2], main = "Slope")
```



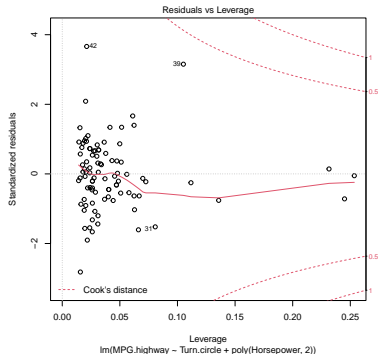
# Influence Plots

Plotting studentized residuals against leverages can help identify influential cases.

```
plot(out1, 5)
```



```
plot(out4, 5)
```



# Removing Influential Observations

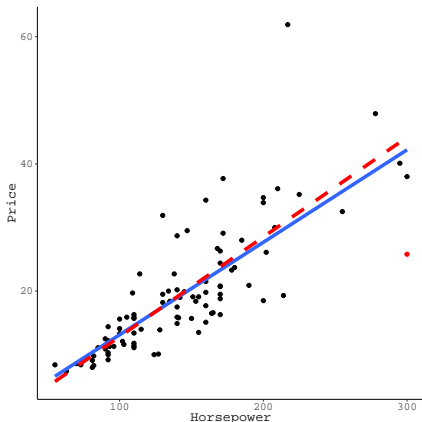
```
(maxD <- which.max(cd))
```

```
28
```

```
28
```

Observation number 28 was the most influential according to Cook's Distance.

- Removing that observation has a small impact on the fitted regression line.
- Influential observations don't only affect the regression line, though.



# Removing Influential Observations

```
## Exclude the influential case:
```

```
Cars93.2 <- Cars93[-maxD, ]
```

```
## Fit model with reduced sample:
```

```
out2 <- lm(Price ~ Horsepower, data = Cars93.2)
```

```
summary(out1)$coefficients %>% round(6)
```

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	-1.398769	1.820016	-0.768548	0.444152
Horsepower	0.145371	0.011898	12.218325	0.000000

```
summary(out2)$coefficients %>% round(6)
```

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	-2.837646	1.806418	-1.570868	0.119722
Horsepower	0.156750	0.011996	13.066942	0.000000

# Removing Influential Observations

---

```
partSummary(out1, 2)
```

Residuals:

Min	1Q	Median	3Q	Max
-16.413	-2.792	-0.821	1.803	31.753

```
partSummary(out2, 2)
```

Residuals:

Min	1Q	Median	3Q	Max
-11.4069	-3.0349	-0.5912	1.8530	30.7229

# Removing Influential Observations

---

```
summary(out1)[c("sigma", "r.squared", "fstatistic")] %>%  
  unlist() %>%  
  head(3)
```

sigma	r.squared	fstatistic.value
5.976953	0.621287	149.287468

```
summary(out2)[c("sigma", "r.squared", "fstatistic")] %>%  
  unlist() %>%  
  head(3)
```

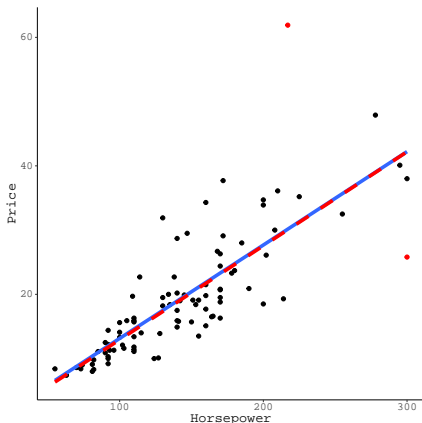
sigma	r.squared	fstatistic.value
5.7243112	0.6548351	170.7449721

# Removing Influential Observations

```
(maxDs <- sort(cd) %>% names() %>% tail(2) %>% as.numeric())  
[1] 59 28
```

If we remove the two most influential observations, 59 and 28, the fitted regression line barely changes at all.

- The influences of these two observations were counteracting one another.
- We're probably still better off, though.



# Removing Influential Observations

```
## Exclude influential cases:
```

```
Cars93.2 <- Cars93[-maxDs, ]
```

```
## Fit model with reduced sample:
```

```
out2.2 <- lm(Price ~ Horsepower, data = Cars93.2)
```

```
summary(out1)$coefficients %>% round(6)
```

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	-1.398769	1.820016	-0.768548	0.444152
Horsepower	0.145371	0.011898	12.218325	0.000000

```
summary(out2.2)$coefficients %>% round(6)
```

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	-1.695315	1.494767	-1.134166	0.25977
Horsepower	0.146277	0.009986	14.648807	0.00000



# Removing Influential Observations

---

```
partSummary(out1, 2)
```

Residuals:

Min	1Q	Median	3Q	Max
-16.413	-2.792	-0.821	1.803	31.753

```
partSummary(out2.2, 2)
```

Residuals:

Min	1Q	Median	3Q	Max
-10.3079	-2.5786	-0.6084	1.9775	14.5793

# Removing Influential Observations

---

```
summary(out1)[c("sigma", "r.squared", "fstatistic")] %>%  
  unlist() %>%  
  head(3)
```

sigma	r.squared	fstatistic.value
5.976953	0.621287	149.287468

```
summary(out2.2)[c("sigma", "r.squared", "fstatistic")] %>%  
  unlist() %>%  
  head(3)
```

sigma	r.squared	fstatistic.value
4.7053314	0.7068391	214.5875491