Machine learning: General Linear Models

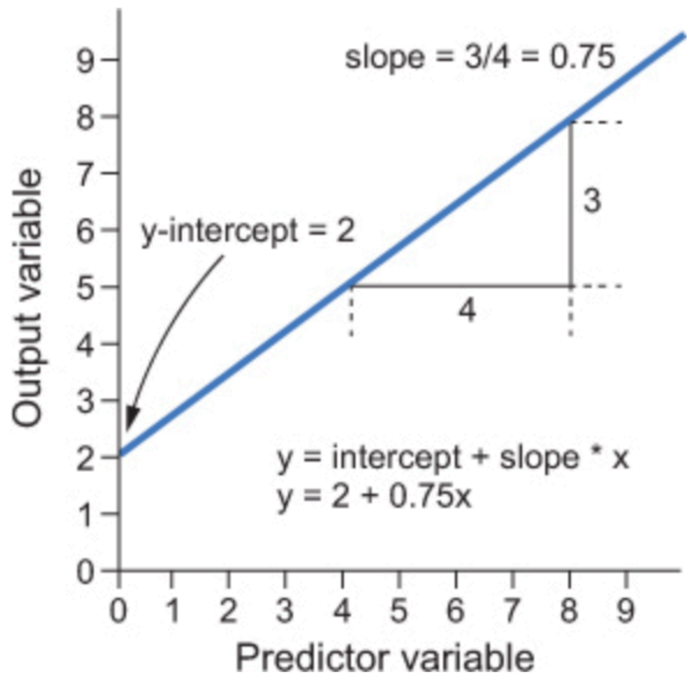
Alex Di Genova

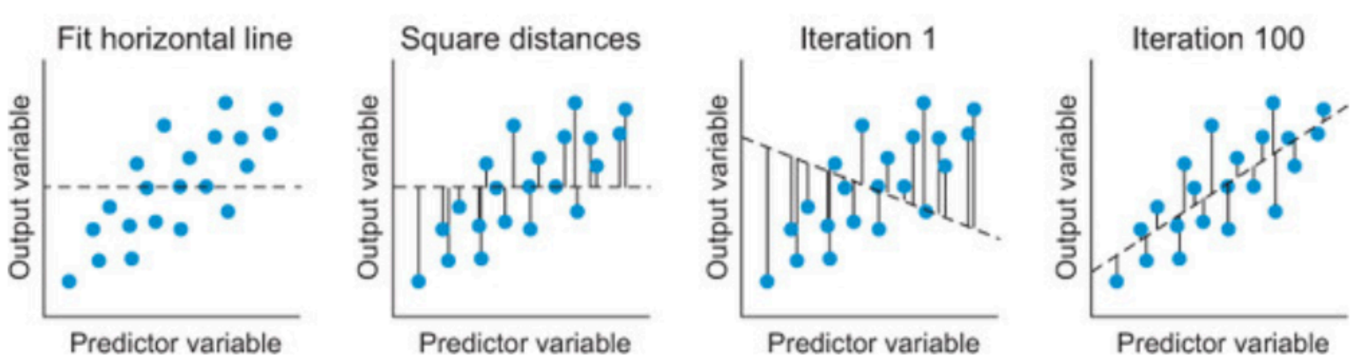
25/06/2024

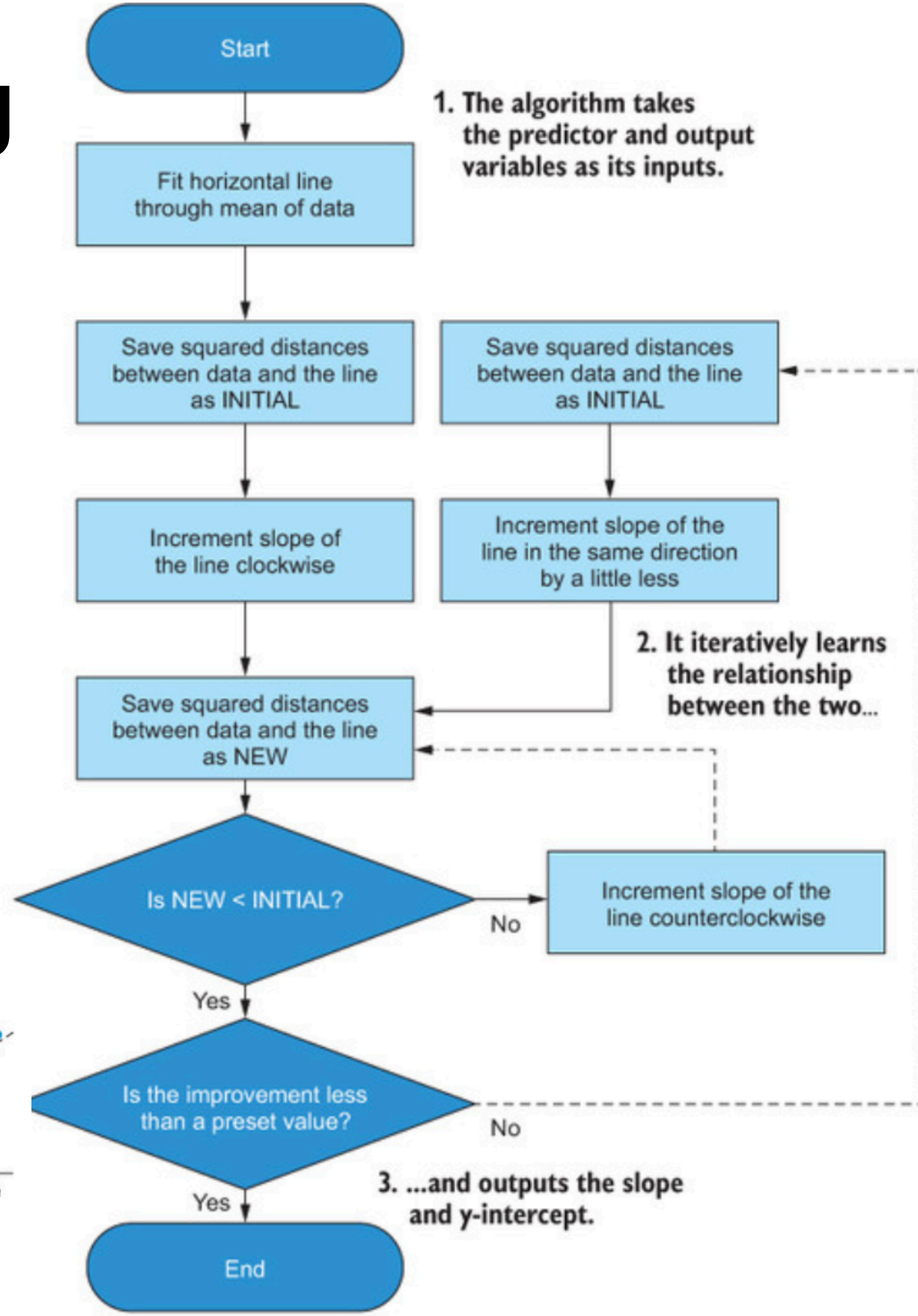
Machine learning

Model and algorithm

Y = intercept + slope X
(parameters)



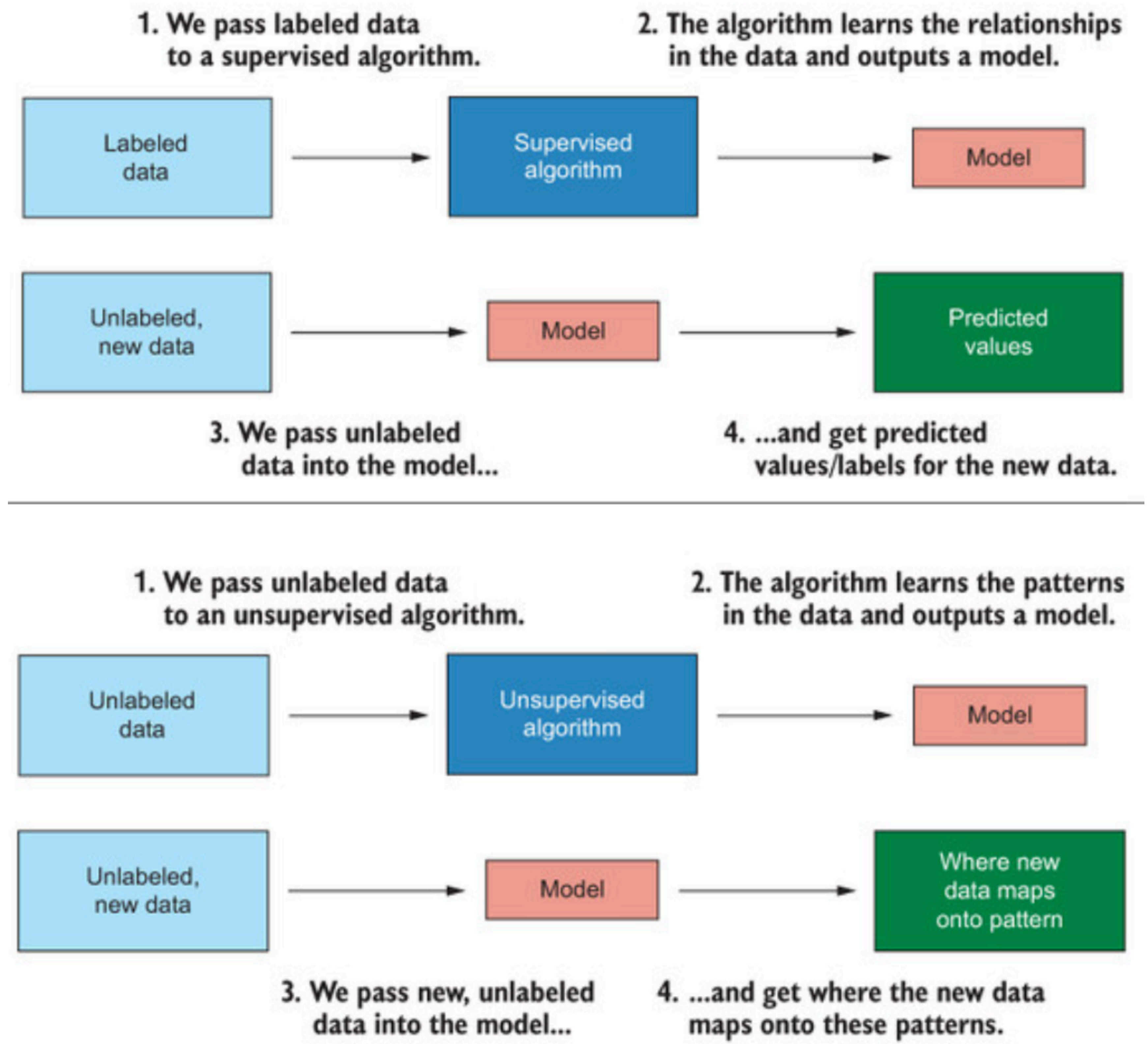




Machine learning algorithms

Classes

- Supervised
  - Classification
  - Regression
- Unsupervised
  - Dimension Reduction
  - Clustering
- Semi-supervised



Machine Learning

Logistic regression

- **Log-Odds**
  - Model the probability that a given input belongs to a particular class
  - Odds = p/1-p —- the ratio of the probability of the event occurring to the probability of it not occurring.
  - Log-Odds = log (p/1-p)
  - log-odds are modeled as a linear combination of the input variables:
    - $\log(1/1-p) = \beta 0 + \beta 1\ x1 + \beta 2\ x2 + \ldots + \beta n\ xn$
- **Sigmoid Function** (Logistic function)
  - Maps the log-odds to a probability value between 0 and 1
  - $\sigma(z) = 1 / (1 + e^{-z})$ where $z = \beta 0 + \beta 1\ x1 + \beta 2\ x2 + \ldots + \beta n\ xn$
- **Binary Classification**
  - If the probability is greater than a certain threshold (commonly 0.5), the input is classified as class 1; otherwise, it is classified as class 0.

Linear Regression in R

An Introduction with Examples

Introduction to Linear Regression

- Linear Regression is a statistical method for modeling the relationship between a dependent variable and one or more independent variables.
- Applications:
- - Predictive Analysis
- - Risk Management
- - Trend Analysis

Theory of Linear Regression

- Simple Linear Regression: Models relationship between two variables by fitting a linear equation.
- Multiple Linear Regression: Extends simple linear regression to include multiple predictors.
- Assumptions:
- - Linearity
- - Independence
- - Homoscedasticity
- - Normality

Fitting a Linear Model in R

1. Data Preparation
2. Building the Model
3. Evaluating the Model

```
> glimpse(mtcars)
Rows: 32
Columns: 11
$ mpg  <dbl> 21.0, 21.0, 22.8, 21.4, 18.7,…
$ cyl  <dbl> 6, 6, 4, 6, 8, 6, 8, 4, 4, 6, 6, 8,…
$ disp <dbl> 160.0, 160.0, 108.0, 258.0,…
$ hp   <dbl> 110, 110, 93, 110, 175, 105,…
$ drat <dbl> 3.90, 3.90, 3.85, 3.08, 3.15,…
$ wt   <dbl> 2.620, 2.875, 2.320, 3.215,…
$ qsec <dbl> 16.46, 17.02, 18.61, 19.44.…
$ vs   <dbl> 0, 0, 1, 1, 0, 1, 0, 1, 1, 1, 1, 0…
$ am   <dbl> 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0,…
$ gear <dbl> 4, 4, 4, 3, 3, 3, 3, 4, 4, 4, 4,…
$ carb <dbl> 4, 4, 1, 1, 2, 1, 4, 2, 2, 4, 4,…
```

```
# Load necessary libraries
library(ggplot2)

# Load the data
data(mtcars)
head(mtcars)

# Fit a simple linear regression model
model <- lm(mpg ~ wt, data=mtcars)

# Summary of the model
summary(model)

# Plotting the data and the model
ggplot(mtcars, aes(x=wt, y=mpg)) +
  geom_point() +
  geom_smooth(method="lm", col="blue")
```

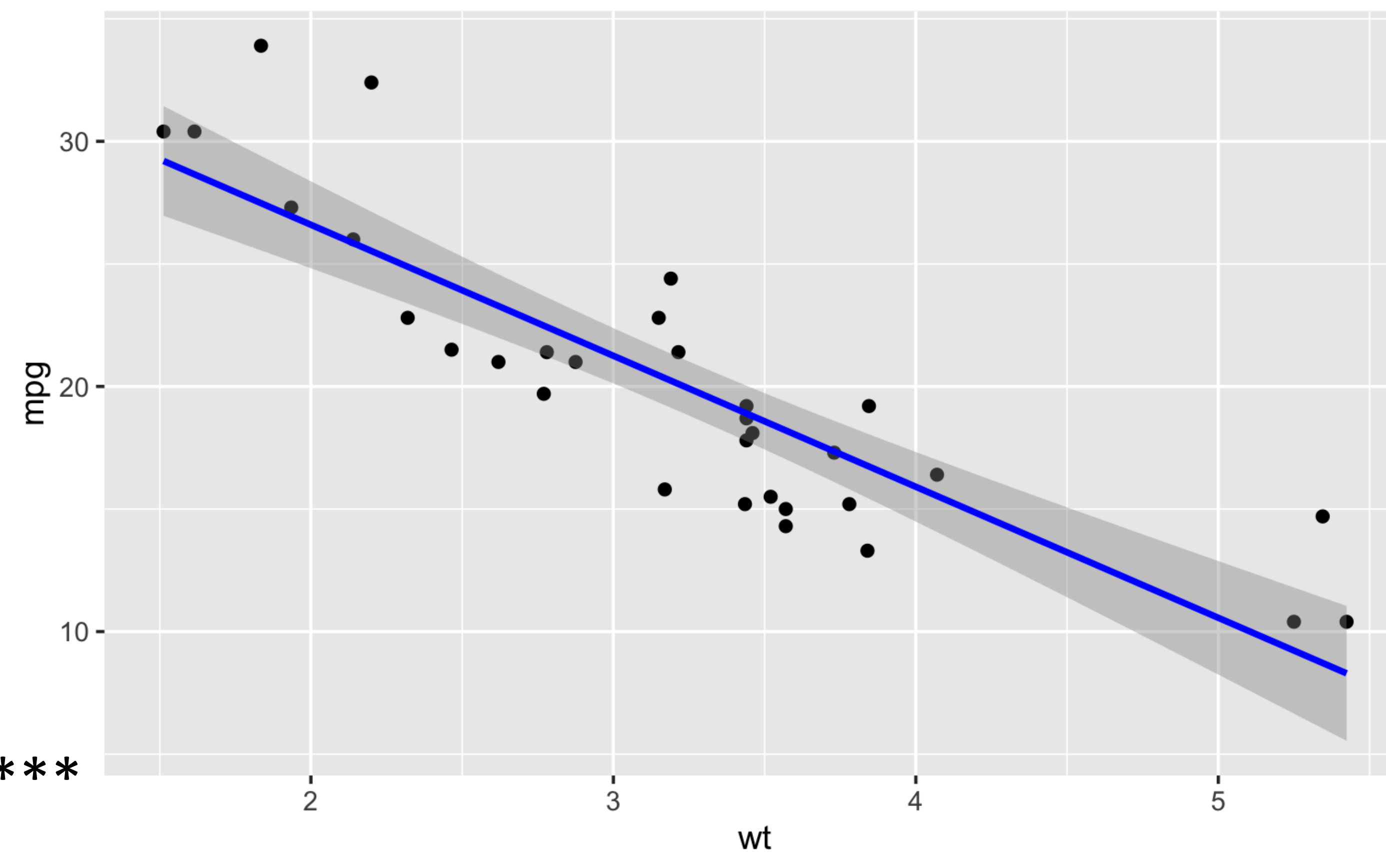
Fitting a Linear Model in R

```
Call:
lm(formula = mpg ~ wt, data = mtcars)

Residuals:
    Min      1Q  Median      3Q     Max
-4.5432 -2.3647 -0.1252  1.4096  6.8727

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  37.2851     1.8776  19.858  < 2e-16 ***
wt           -5.3445     0.5591  -9.559 1.29e-10 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3.046 on 30 degrees of freedom
Multiple R-squared:  0.7528,	Adjusted R-squared:  0.7446
F-statistic: 91.38 on 1 and 30 DF,  p-value: 1.294e-10
```



Multiple Linear Regression in R

```
# Fit a multiple linear regression model
model_mult <- lm(mpg ~ wt + hp + qsec, data=mtcars)

# Summary of the model
summary(model_mult)

# Diagnostic plots
par(mfrow=c(2,2))
plot(model_mult)
```

```
Call:
lm(formula = mpg ~ wt + hp + qsec, data = mtcars)

Residuals:
    Min      1Q  Median      3Q     Max
-3.8591 -1.6418 -0.4636  1.1940  5.6092

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 27.61053    8.41993   3.279  0.00278 **
wt          -4.35880    0.75270  -5.791 3.22e-06 ***
hp          -0.01782    0.01498  -1.190  0.24418
qsec         0.51083    0.43922   1.163  0.25463
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.578 on 28 degrees of freedom
Multiple R-squared:  0.8348,     Adjusted R-squared:  0.8171
F-statistic: 47.15 on 3 and 28 DF,  p-value: 4.506e-11
```

Interpreting the Results

- Coefficients:
  - Each coefficient represents the expected change in the dependent variable for a one-unit change in the independent variable, holding all other variables constant.
  - If wt has a coefficient of -4.35, it means that for each unit increase in wt, mpg decreases by 4.35 units, assuming other variables are constant.

````
```{r}
# Coefficients
coef(model_mult)
# R-squared
summary(model_mult)$r.squared
# P-values
summary(model_mult)$coefficients[,4]
```
````

```
(Intercept)          wt          hp        qsec
27.61052686 -4.35879720 -0.01782227  0.51083369
[1] 0.8347678
 (Intercept)           wt           hp         qsec
2.784556e-03 3.217222e-06 2.441762e-01 2.546284e-01
```

Interpreting the Results

- R-squared:
  - indicates the proportion of the variance in the dependent variable that is predictable from the independent variables
  - An R-squared value of 0.83 means that 83% of the variance in mpg can be explained by wt, hp and qsec.

````
```{r}
# Coefficients
coef(model_mult)
# R-squared
summary(model_mult)$r.squared
# P-values
summary(model_mult)$coefficients[,4]
```
````

```
(Intercept)          wt          hp        qsec
27.61052686 -4.35879720 -0.01782227  0.51083369
[1] 0.8347678
 (Intercept)           wt           hp         qsec
2.784556e-03 3.217222e-06 2.441762e-01 2.546284e-01
```

Interpreting the Results

- P-values:
  - P-values assess the statistical significance of each coefficient. A small p-value (typically < 0.05) indicates strong evidence against the null hypothesis, suggesting that the coefficient is significantly different from zero.
  - If the p-value for wt is 0.007, it means there is a 0.007% chance that wt has no effect on mpg, indicating statistical significance.

```{r}
# Coefficients
coef(model_mult)
# R-squared
summary(model_mult)$r.squared
# P-values
summary(model_mult)$coefficients[,4]
```

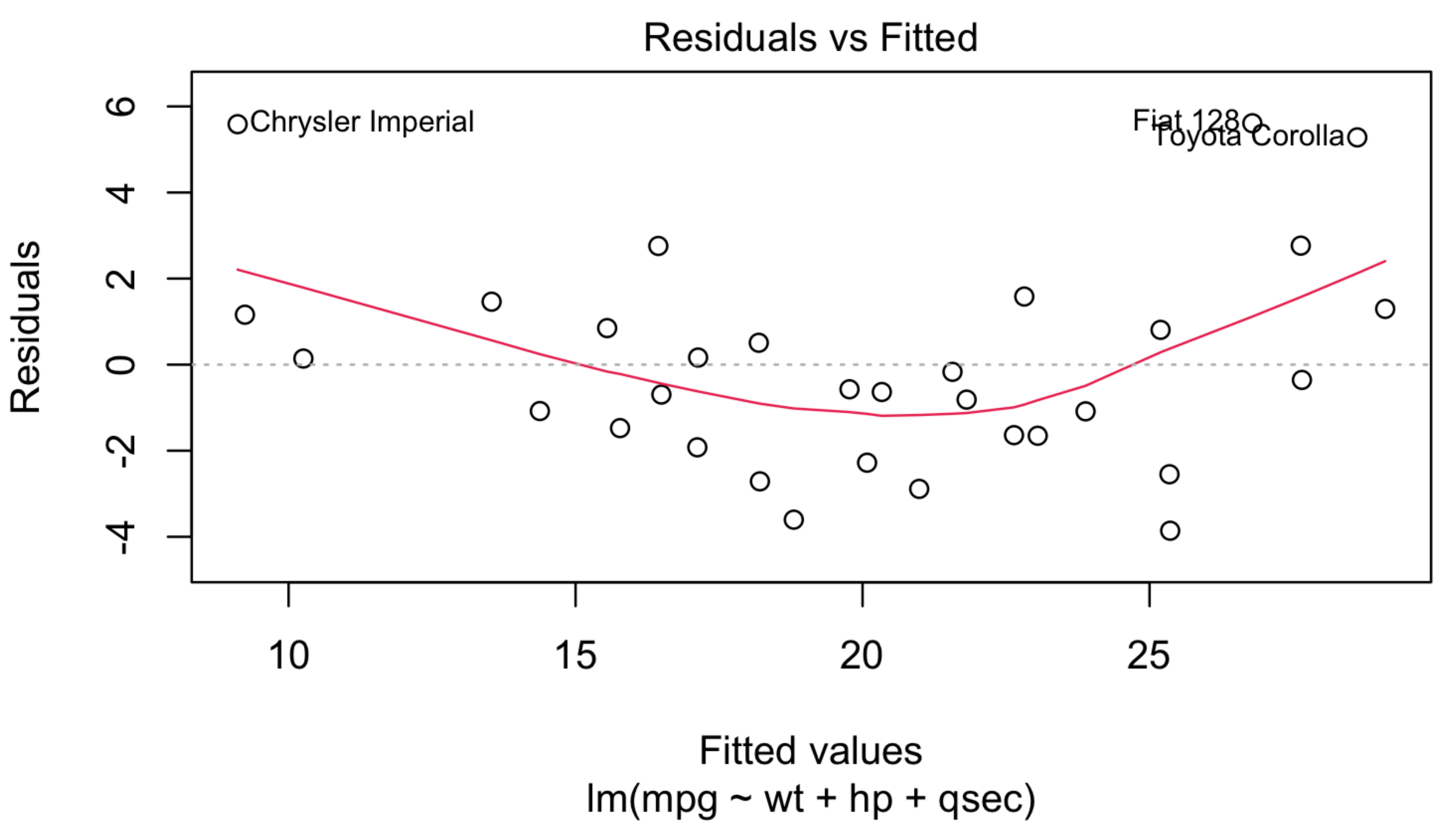
```
(Intercept)          wt          hp        qsec
27.61052686 -4.35879720 -0.01782227  0.51083369
[1] 0.8347678
 (Intercept)           wt           hp         qsec
2.784556e-03 3.217222e-06 2.441762e-01 2.546284e-01
```

Model Diagnostics

- Residual Analysis:
  - Check residuals to ensure they are randomly distributed.
- Checking Assumptions:
  - Independence: Check for autocorrelation.
  - Homoscedasticity: Look for constant variance in residual plots.
  - Normality: Q-Q plot for residuals.

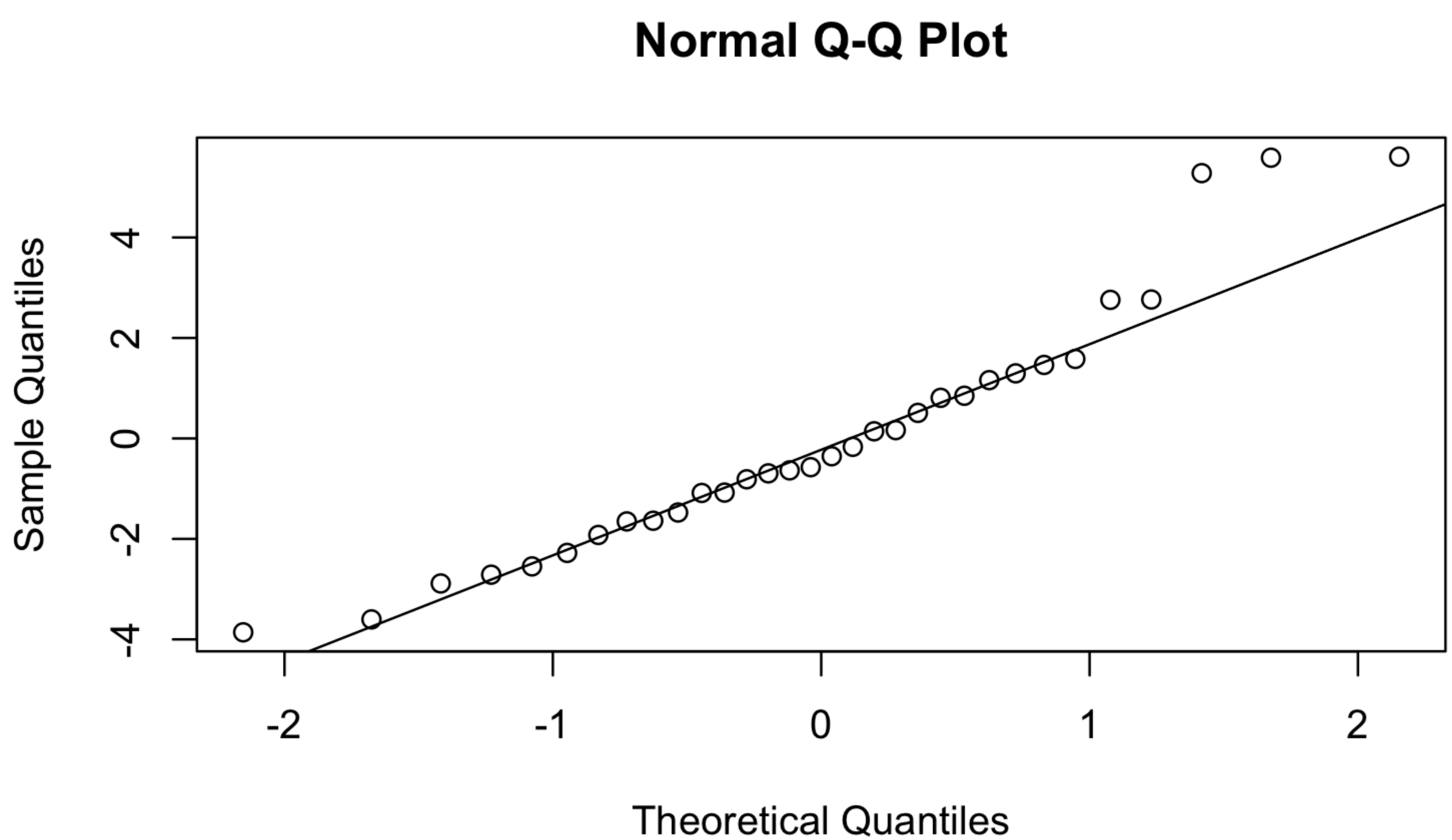
Residual Analysis

- Goal: Ensure residuals are randomly distributed.
- Code Example: plot(model_mult, which=1) # Residuals vs Fitted plot
- Look for a random scatter of residuals around the horizontal line at zero.



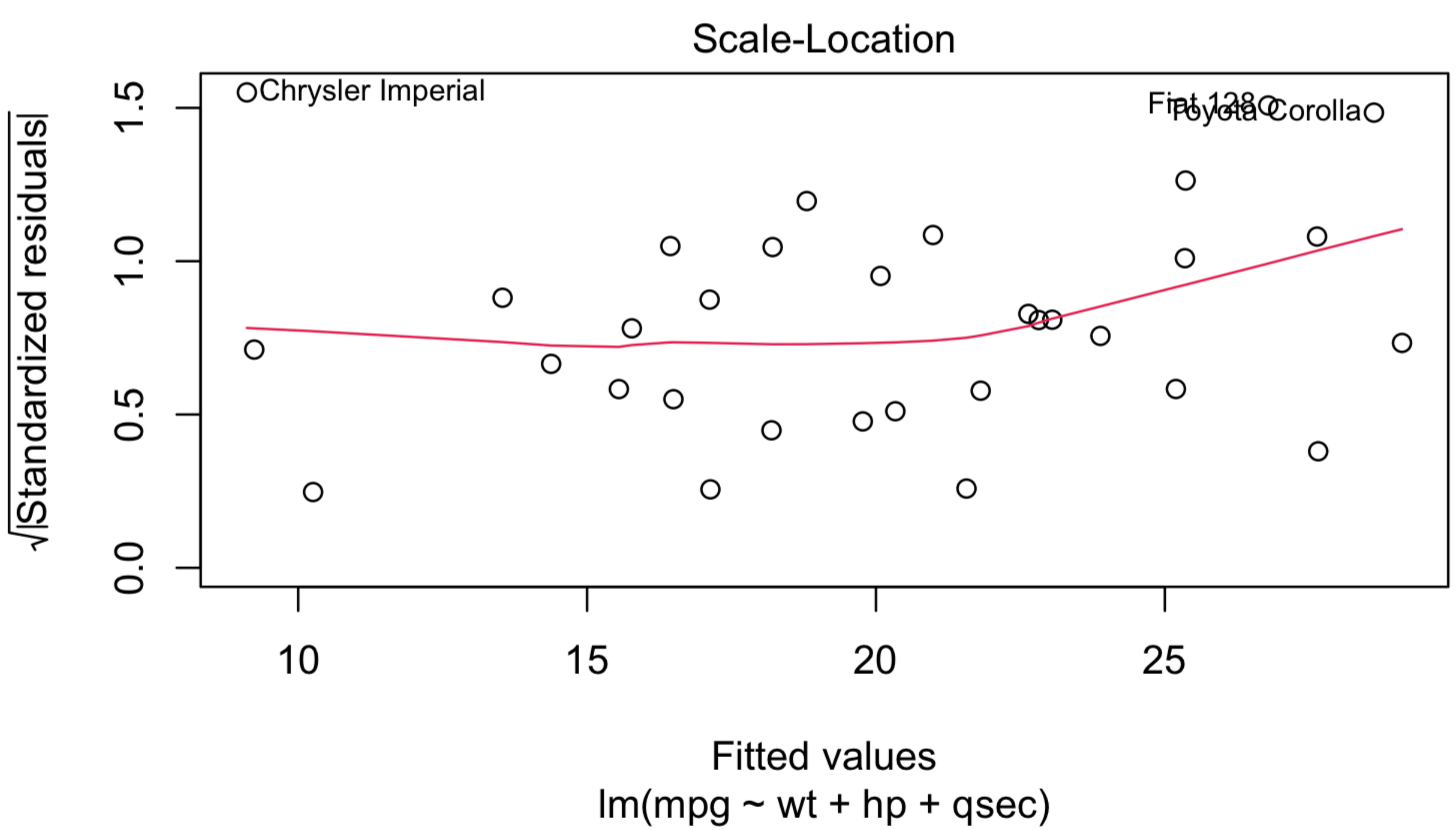
Normality of Residuals

- Goal: Residuals should be approximately normally distributed.
- Code : qqnorm(resid(model_mult)) qqline(resid(model_mult))
- Points should fall approximately along the reference line in a Q-Q plot.



Homoscedasticity Check

- Goal: Residuals should have constant variance.
- Code: plot(model_mult, which=3) # Scale-Location plot
- Look for a horizontal line with equally spread points.



Independence of Residuals

- Goal: Residuals should not be correlated.
- Code: durbinWatsonTest(model_mult)
- A Durbin-Watson statistic around 2 (1.5-2.5) suggests no autocorrelation

lag	Autocorrelation	D-W Statistic	p-value
1	0.2742427	1.422421	0.048

https://www.investopedia.com/terms/d/durbin-watson-statistic.asp

How to select features?

- Examples

Book:
Machine learning with R

Questions?
Practice!!!