Stats Modeling Overview

AG Schissler

1/25/2018

Statistical modeling workflow

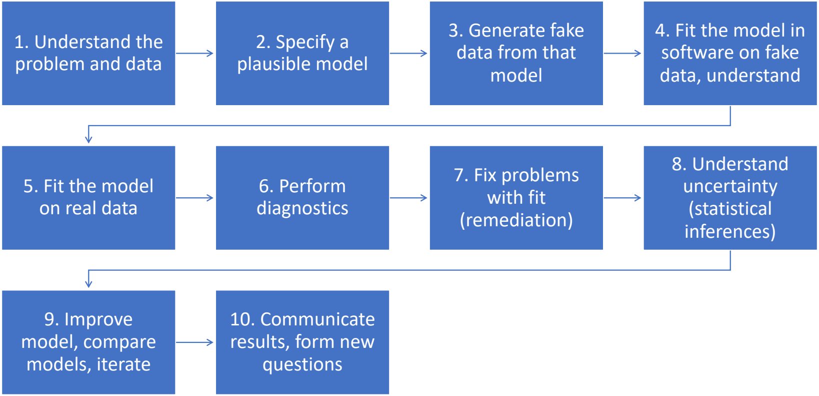


Figure 1: Modeling workflow

Quick example: Stopping distance for speeding cars

```r
## 'pressure data set' is automatically loaded in
## workspace through package{datasets}
head(cars)
```

```
##   speed dist
## 1     4    2
## 2     4   10
## 3     7    4
## 4     7   22
## 5     8   16
## 6     9   10
```

Data set structure

```r
str(cars)
```

```
## 'data.frame':	50 obs. of  2 variables:
##  $ speed: num  4 4 7 7 8 9 10 10 10 11 ...
##  $ dist : num  2 10 4 22 16 10 18 26 34 17 ...
```

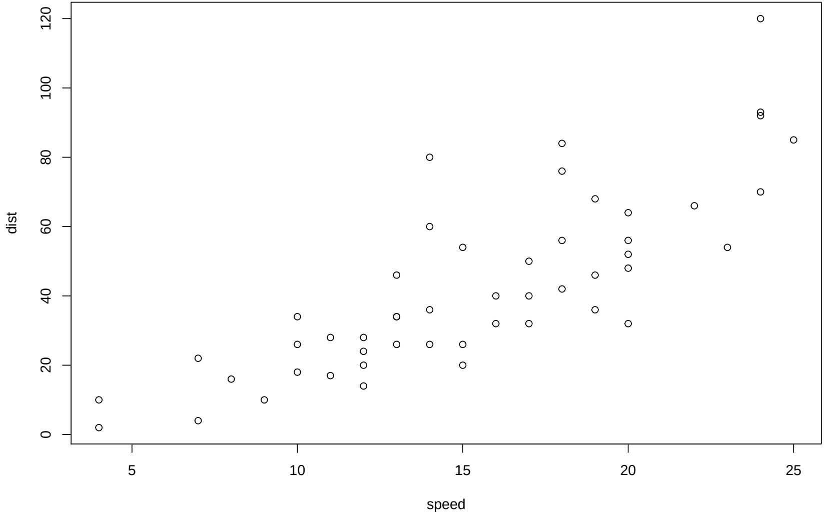
Numeric summary of pressure and temperature

```r
summary(cars)
```

```
##      speed           dist       
##  Min.   : 4.0   Min.   :  2  
##  1st Qu.:12.0   1st Qu.: 26  
##  Median :15.0   Median : 36  
##  Mean   :15.4   Mean   : 43  
##  3rd Qu.:19.0   3rd Qu.: 56  
##  Max.   :25.0   Max.   :120  
```

Visualization of speed and dist relationship

```
plot(cars)
```

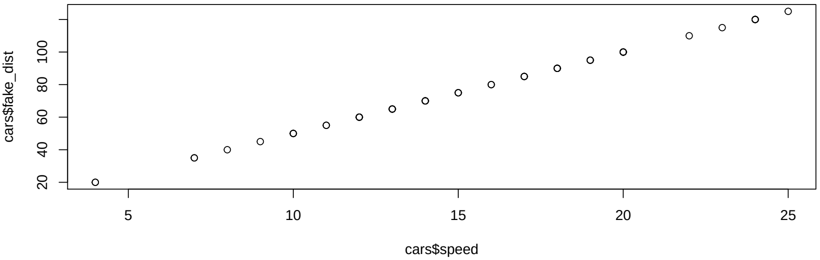


Maybe stopping distance increases linearly with speed?

- If that's true then a model could be:
- dist = b0 + b1*speed
- Let's generate fake data and fit a linear model (simple linear regression!)

Generate fake data from pressure = b0 + b1*temperature

```r
## pick values b0 and b1
b0 <- 0
b1 <- 5
## store in dataset as a new variable
cars$fake_dist <- b0 + b1*cars$speed
plot(x = cars$speed, y = cars$fake_dist)
```



Fit a simple linear regression model

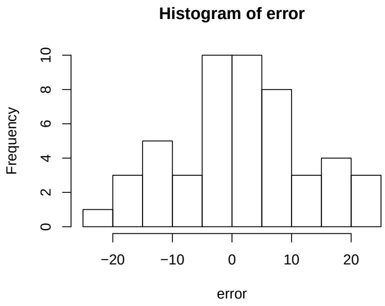
```
fake_lm <- lm(formula = fake_dist ~ speed, data = cars)
summary(fake_lm)
```

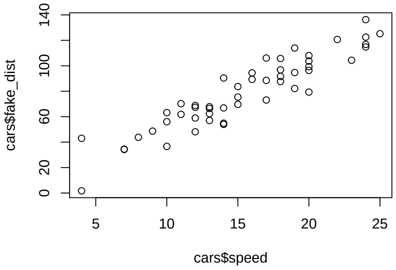
```
## Warning in summary.lm(fake_lm): essentially perfect fit
## unreliable

##
## Call:
## lm(formula = fake_dist ~ speed, data = cars)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -1.82e-14 -7.95e-15 -2.51e-15  1.53e-15  7.11e-14
##
## Coefficients:
##             Estimate Std. Error  t value Pr(>|t|)
## (Intercept) 6.43e-14   7.02e-15 9.16e+00  4.1e-12
## speed       5.00e+00   4.32e-16 1.16e+16  < 2e-16
```

Hmm, data seem too perfect, add noise??

```
set.seed(440)
error <- rnorm(n = nrow(cars), mean = 0, sd = 10)
cars$fake_dist <- b0 + b1*cars$speed + error
par(mfrow=c(1,2))
hist(error, breaks = 12)
plot(x = cars$speed, y = cars$fake_dist)
```





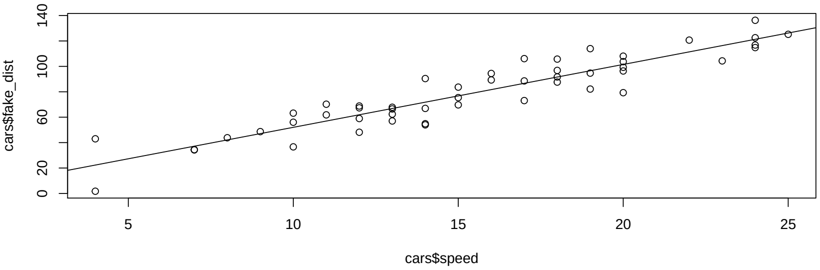
Fit a simple linear regression model with noise in data

```r
fake_lm <- lm(formula = fake_dist ~ speed, data = cars)
## attributes(fake_lm)
summary(fake_lm)
```

```
##
## Call:
## lm(formula = fake_dist ~ speed, data = cars)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -22.209  -5.077  -0.159   6.801  20.663
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)     2.54       4.72    0.54     0.59
## speed           4.95       0.29   17.03   <2e-16
##
## Residual standard error: 10.8 on 48 degrees of freedom
```

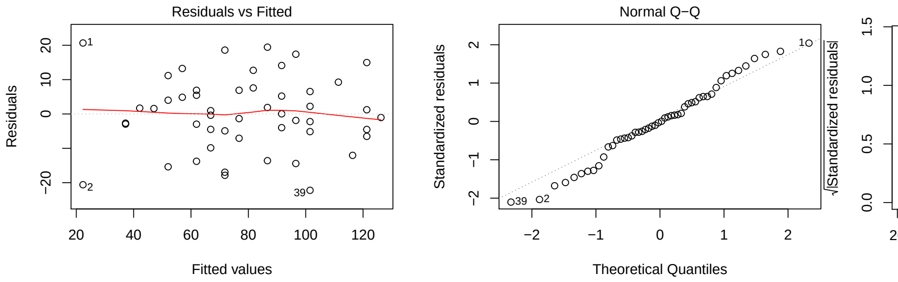
Plot the regression line

```
plot(x = cars$speed, y = cars$fake_dist)
abline(fake_lm)
```



Plot some diagnostics

```
par(mfrow=c(1,2))
plot(fake_lm)
```



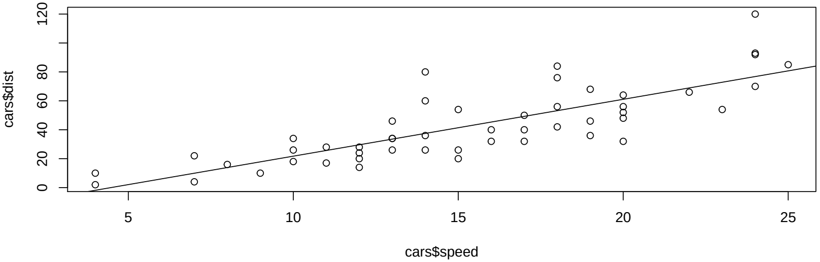
Fit model on real data

```r
real_lm <- lm(formula = dist ~ speed, data = cars)
summary(real_lm)
```

```
##
## Call:
## lm(formula = dist ~ speed, data = cars)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -29.07  -9.53  -2.27   9.21  43.20
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -17.579      6.758   -2.60    0.012
## speed          3.932      0.416    9.46  1.5e-12
##
## Residual standard error: 15.4 on 48 degrees of freedom
## Multiple R-squared:  0.651,  Adjusted R-squared:  0.644
```

Plot the regression line

```
par(mfrow=c(1,1))
plot(x = cars$speed, y = cars$dist)
abline(real_lm)
```



Plot some diagnostics

```r
par(mfrow=c(1,2))
plot(real_lm)
```