
# Linear Regression

Measurement & Evaluation of HCC Systems


# Linear Regression

Today's goal:

Evaluate the effect of multiple variables on an outcome variable (regression)

Outline:

- Basic theory
- Simple regression in R
- Extending this to multiple regression
- Multiple regression in R


# Theory

of linear regression

# Theory

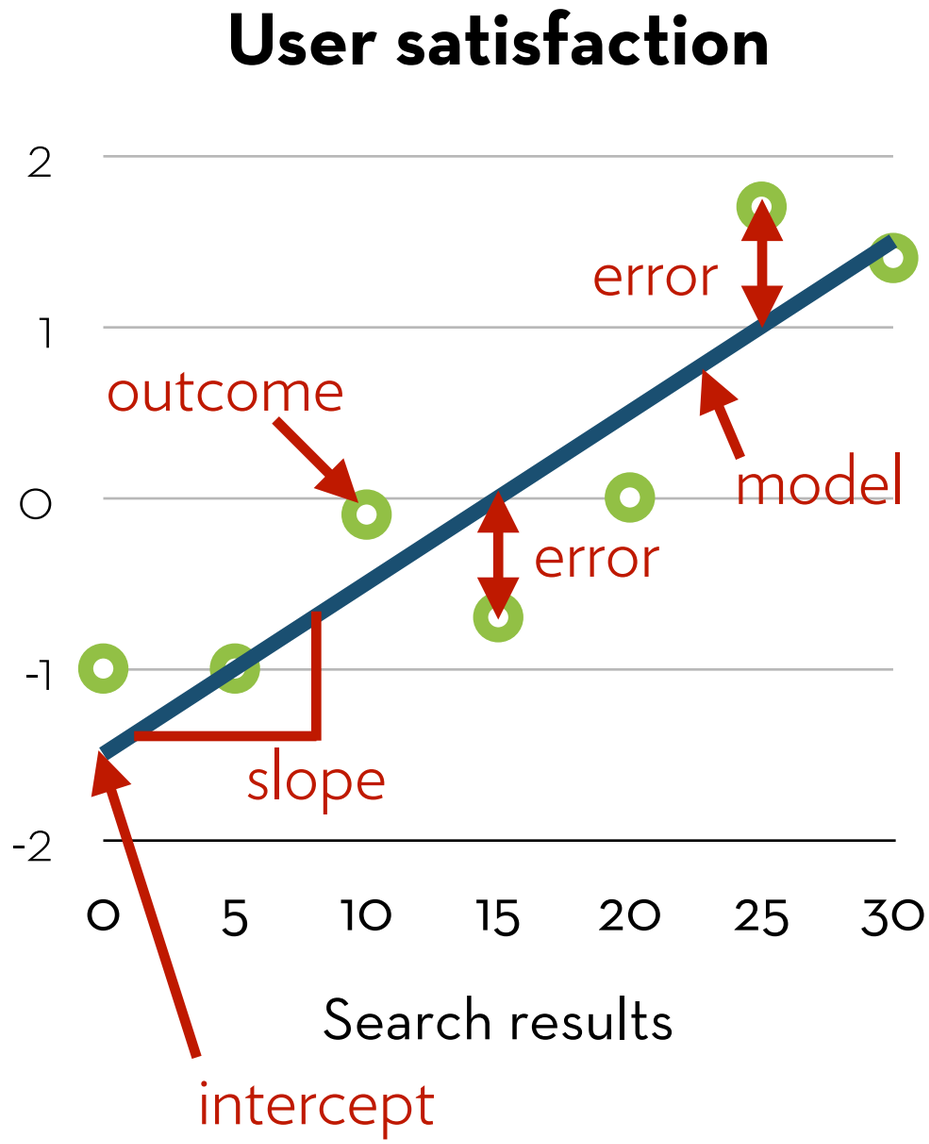
Any type of model:

$$outcome_i = model + error_i$$

Linear regression:

The model is a line with an intercept (a) and a slope (b)

$$Y_i = a + bX_i + e_i$$



# Finding the best line

$Y_i = a + bX_i + e_i$

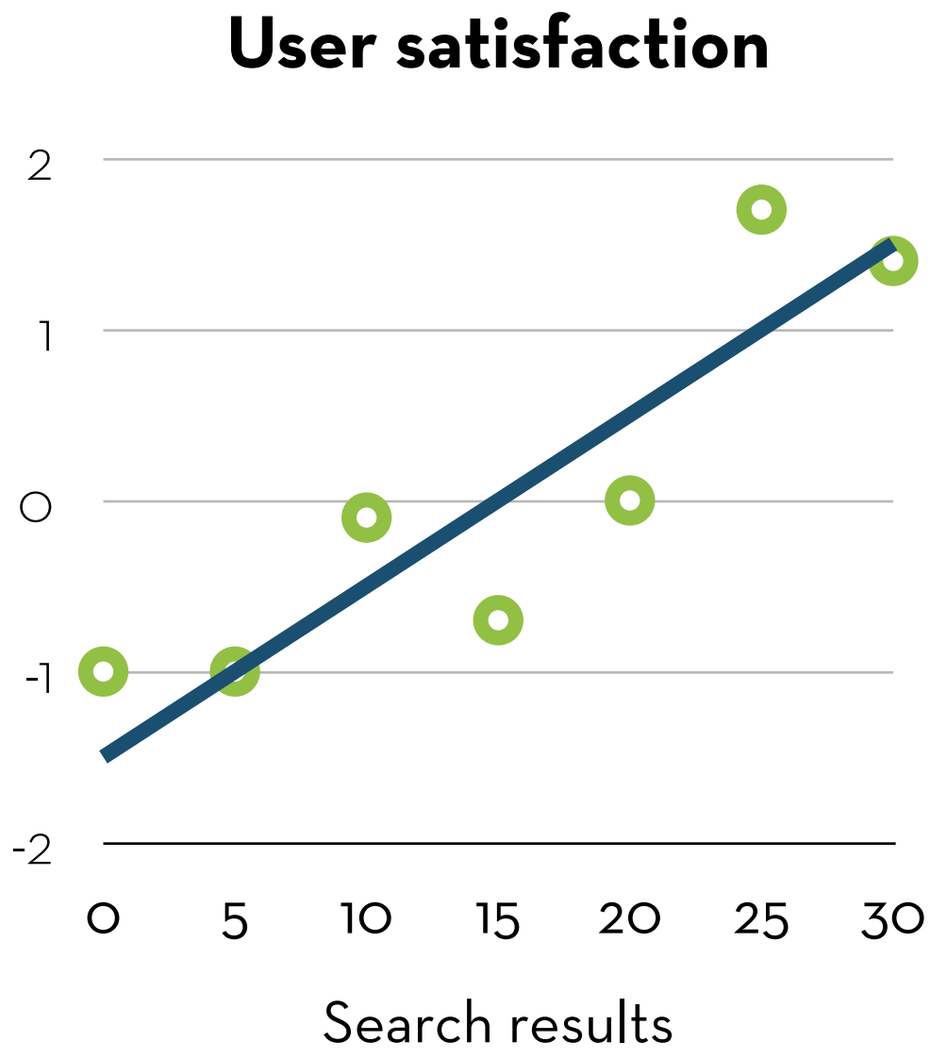
a and b are chosen so that the deviations (residuals) are minimized

We know this! General:

deviation =

$\sum(\text{observation}_i - \text{model})^2$

Goal: minimize sum of squared errors (SSr)



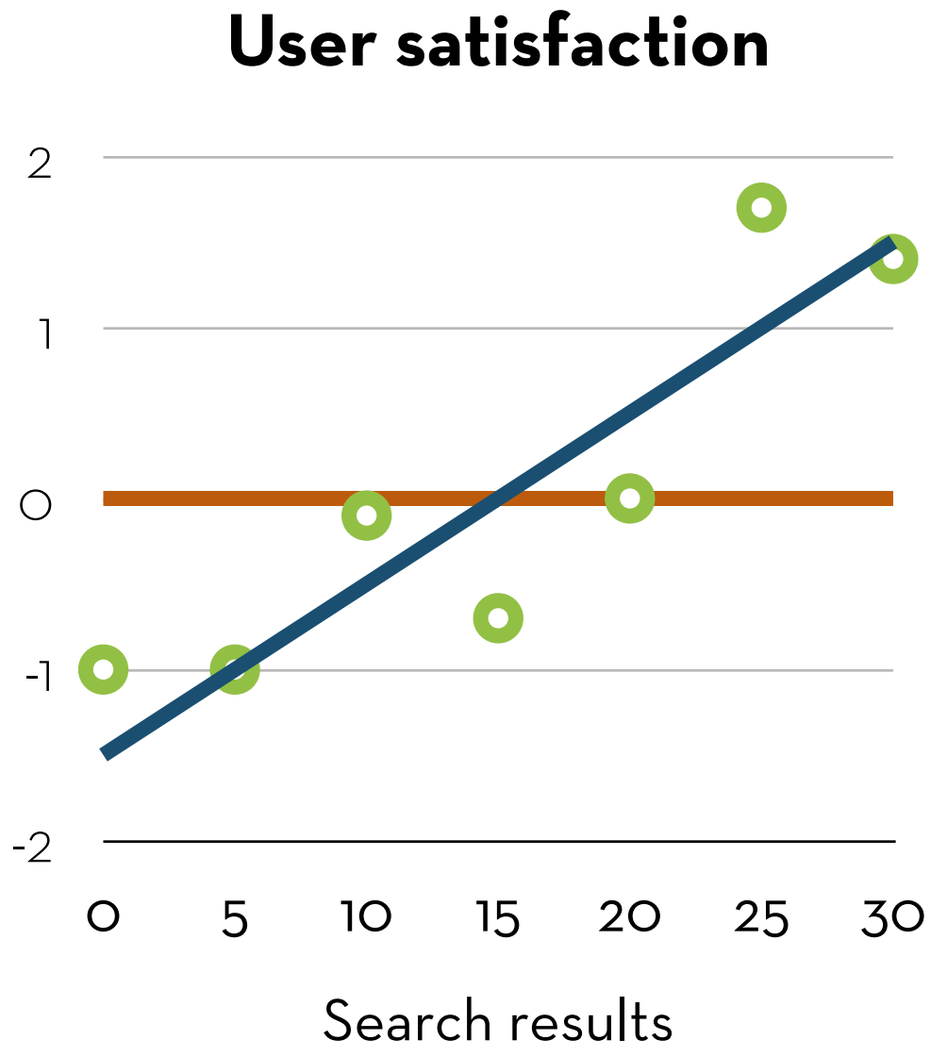
# Goodness of fit

How good is the model?

We can use deviation for this as well!

Compare against the deviation of the simplest model

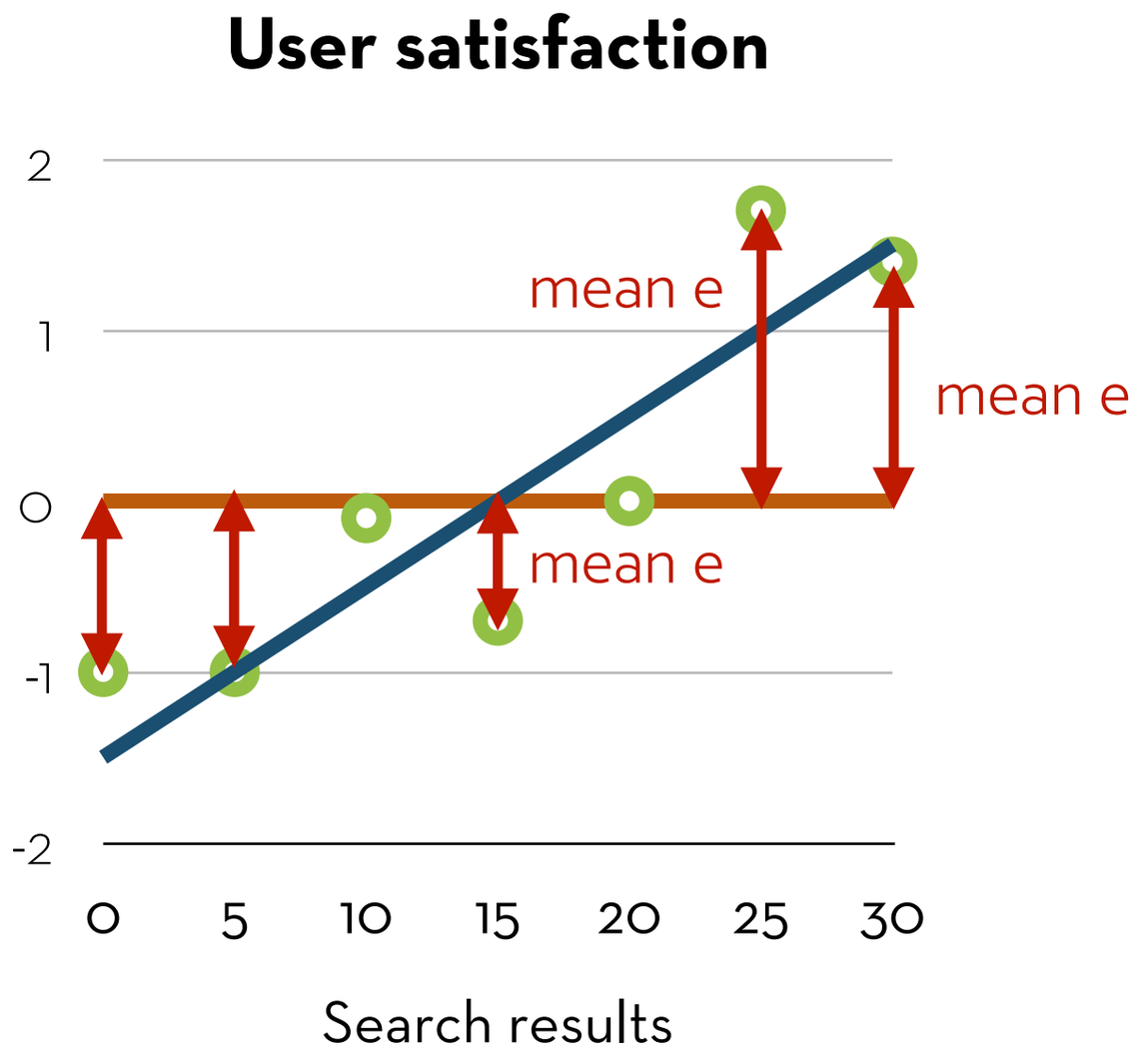
In this case: the mean



# Goodness of fit

Total sum of squares (SSt)

Squared e from the mean



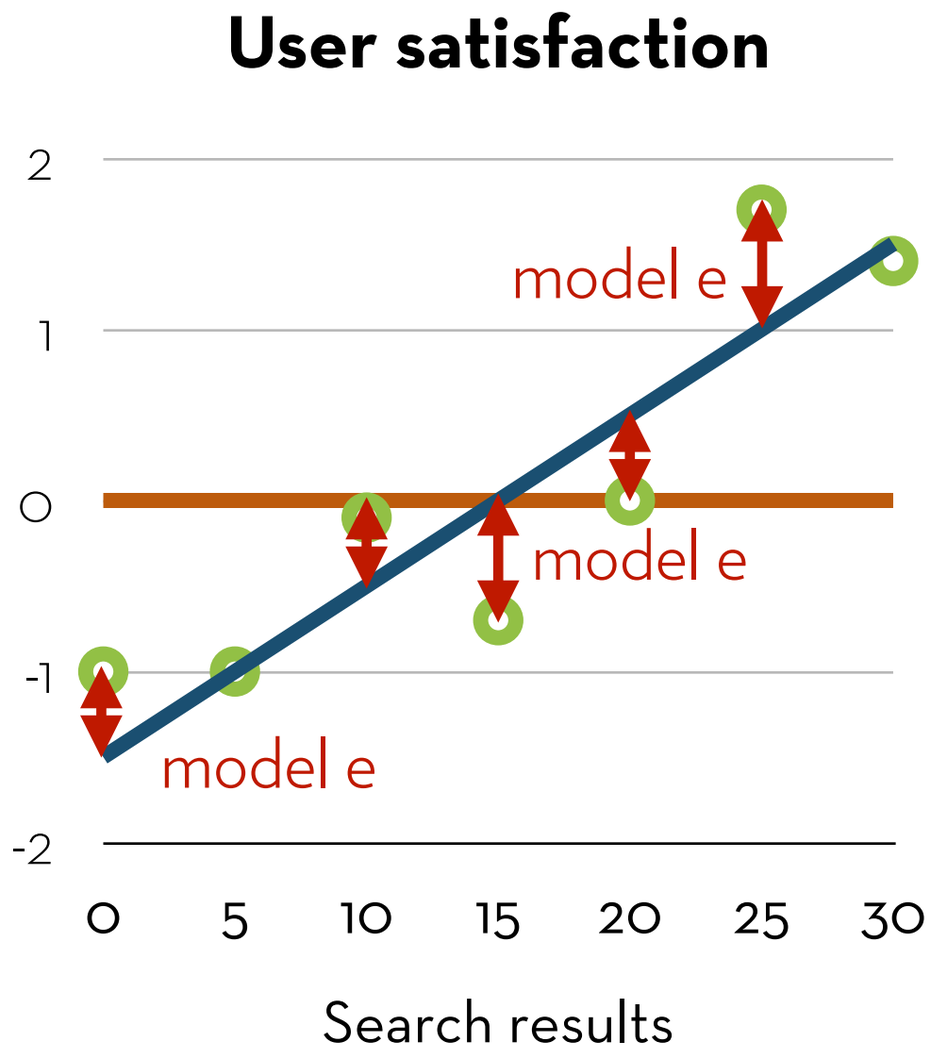
# Goodness of fit

Total sum of squares (SSt)

Squared e from the mean

Residual sum of sq. (SSr)

Squared e from the model



# Goodness of fit

Total sum of squares (SSt)

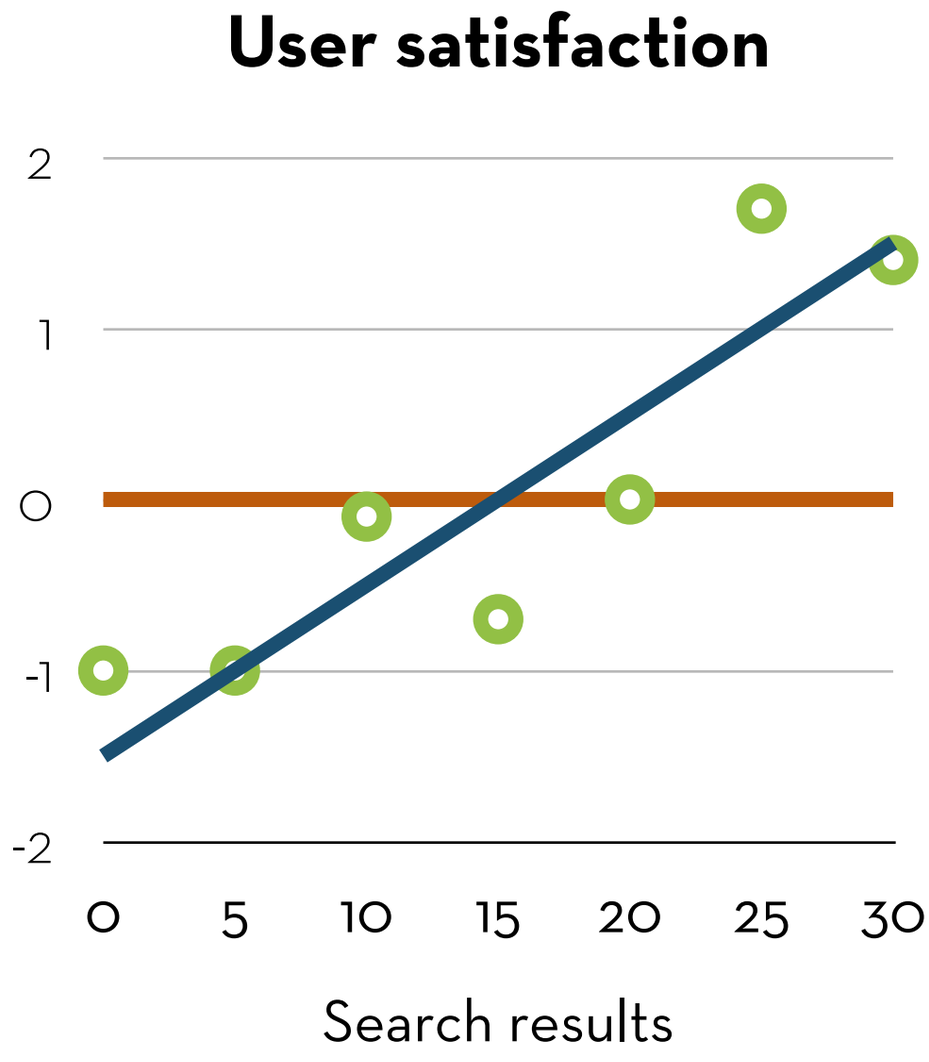
Squared e from the mean

Residual sum of sq. (SSr)

Squared e from the model

Model sum of squares (SSm)

SSt – SSr



# Goodness of fit

R-square model fit

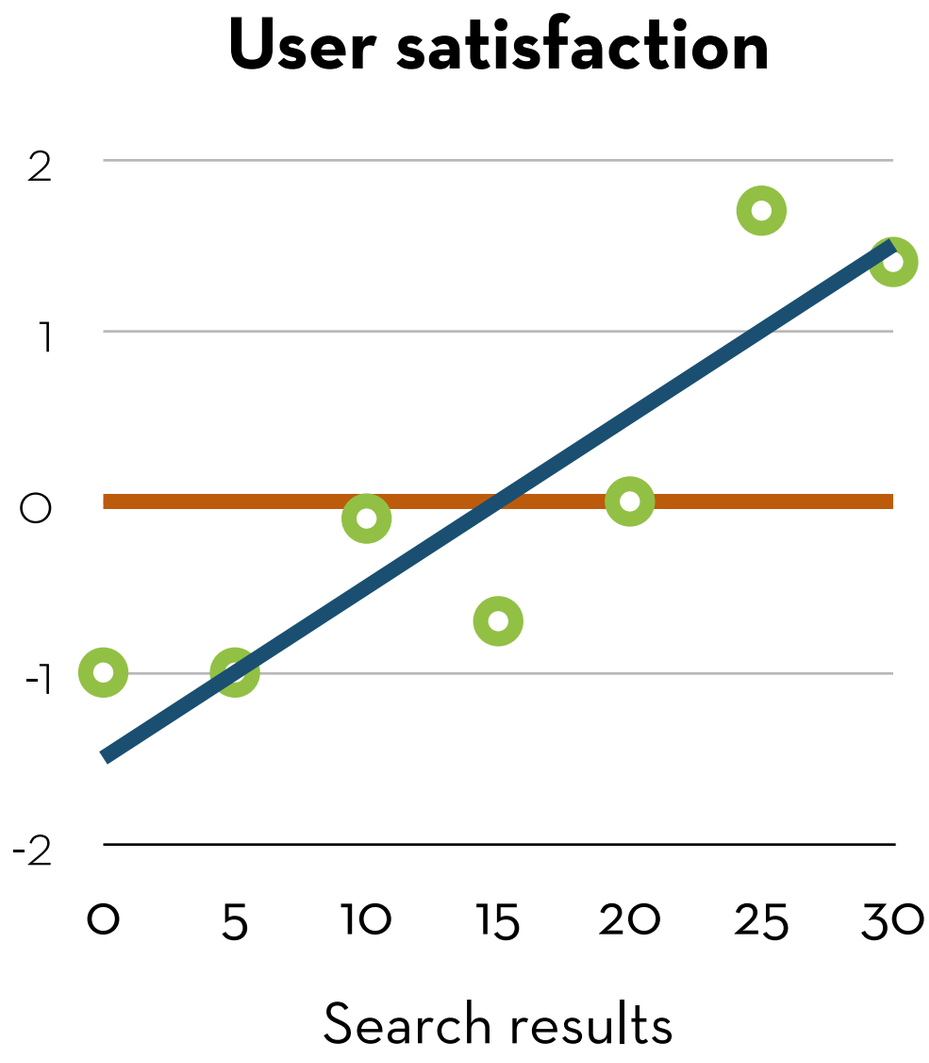
$R^2 = SSm / SSt$

Amount of variation in Y explained by the model

F-ratio (and p-value)

$F = MSm/MSr$

How much did the model improve, compared to the error?



# Testing a predictor

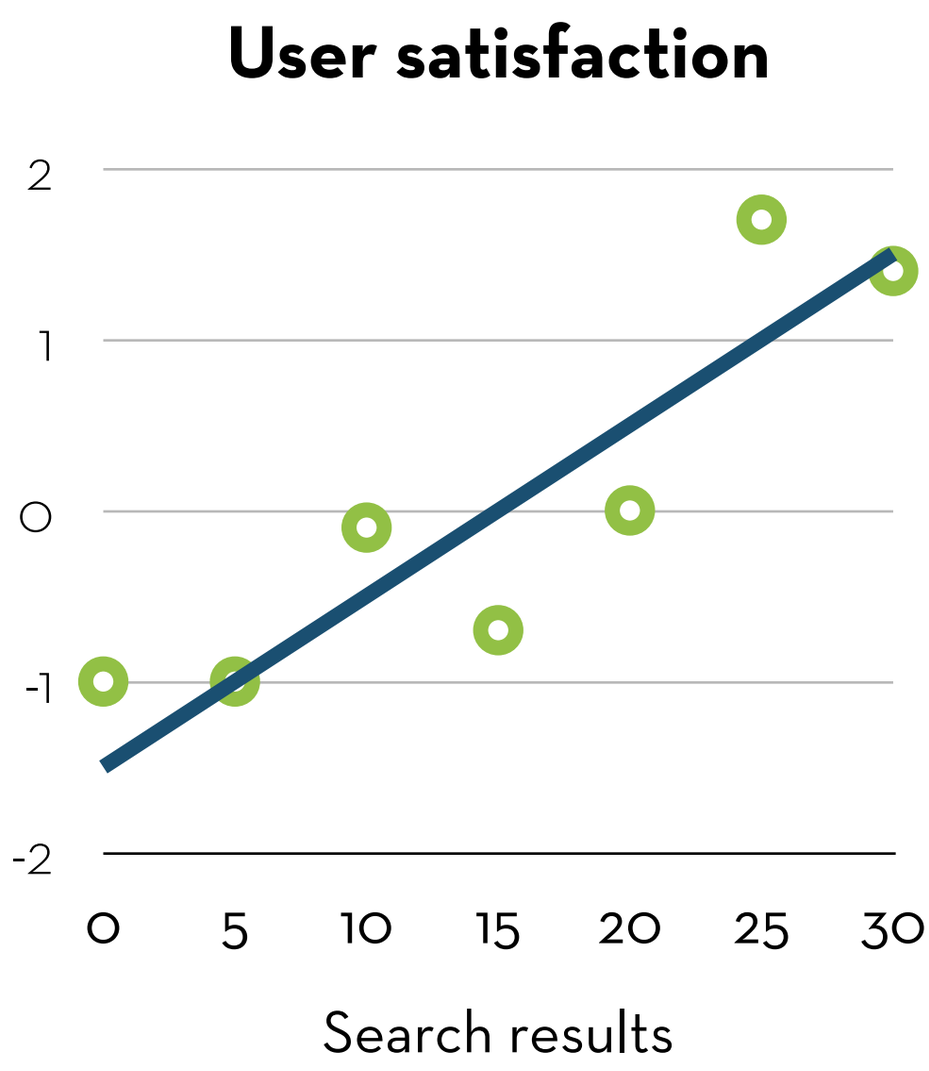
If a predictor is bad, its slope (b) will be almost zero (like the mean)

A good predictor has a slope that is significantly different from zero

Compare slope (b) against variability of slope (SEb):

t = b/SEb

with df = N - p - 1




# Regression in R

look mom! no formulas!

# Regression in R

Let's start simple:

File: Album Sales 2.dat, set Name to album2

Dataset: album sales by promotion method

Variables:

adverts: $1000s spent on advertisement

sales: 1000s of copies sold

airplay: plays on the radio the week before release

attract: attractiveness of the band

# Scatterplot

Scatterplot of sales and adverts, with regression line and mean:

```
ggplot(album2,aes(y=sales,x=adverts))+geom_point()
+geom_smooth(method="lm",color="red",se=F)
+geom_line(aes(y=mean(album2$sales)),color="blue")
```

Result:

- A positive relationship
- Regression line is noticeably different from the mean

# A linear model

Write the regression model:

```
salesModel <- lm(sales ~ adverts, data = album2)
```

Get the results:

```
summary(salesModel)
```

# Output

```
Call:
lm(formula = sales ~ adverts, data = album2)

Residuals:
     Min       1Q   Median       3Q      Max 
-152.949  -43.796   -0.393   37.040  211.866 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept) 1.341e+02  7.537e+00  17.799   <2e-16 ***
adverts     9.612e-02  9.632e-03   9.979   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 65.99 on 198 degrees of freedom
Multiple R-squared:  0.3346,   Adjusted R-squared:  0.3313 
F-statistic: 99.59 on 1 and 198 DF,  p-value: < 2.2e-16
```

# Overall fit

The “Multiple R-squared” tells us the percentage of variance in sales explained by advert

- Seems to be 33.46%
- Can turn this into a correlation coefficient: $\sqrt{.3346} = .5784$

“F-statistic” gives us MSm/MSr, and a p-value

- Seems to be significant
- The model makes significantly a better prediction than the mean

# Model parameters

$Y_i = a + bX_i + e_i$

a: the estimate for "(Intercept)"

Is equal to the average sales with zero dollars spent on advertising (X=0) —> $Y_i = a + e_i$

b: the estimate for "adverts"

The change in the outcome associated with a 1 unit change in predictor

For an +$1000 difference in advertising, the model predicts 96 extra album sales

# Significance

b is significant if it is large relative to its standard error

t = b/SEb, with df = N - p - 1

Here, t = 9.979, and p < .001

The advertising budget makes a significant contribution to predicting album sales

Wait… where did I see that t-value before?

# Using the model

album sales = 134.14 + 0.096*advertising budget

How many albums am I likely to sell with a $100,000 advertisement budget?

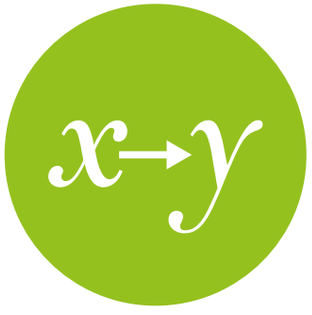
134.14 + 0.096*100 = 143.74 -> around 143,740 albums

What is the predicted effect of spending $50,000 extra on advertising?

0.096*50 = 4.8 -> you will likely sell 4,800 more albums


# Multiple Regression

regression with more than one predictor



# Multiple Regression

$outcome_i = model + error_i$

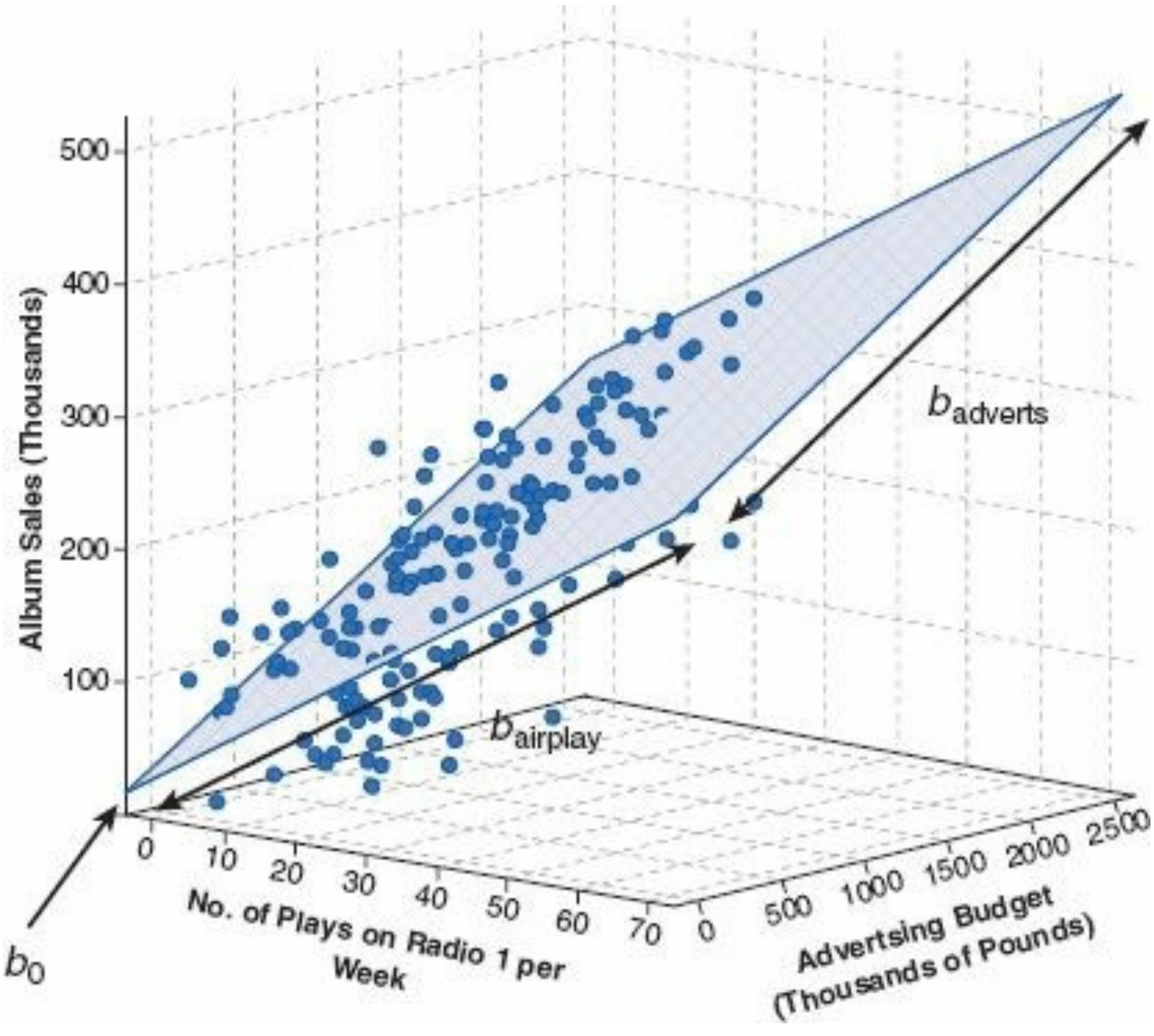
Multiple regression:

The model is a line with an intercept (a) and **several** slopes ($b_1 ... b_n$)

$$Y_i = a + b_1X_{1i} + b_2X_{2i} + \ldots + b_nX_{ni} + e_i$$

This means you can predict album sales using advertising **and** airplay

# Multiple Regression



# Goodness of fit

$R^2$ = SSm / SSt

Same as before, but $R^2$ is now called the "multiple $R^2$"

Combined effect of all predictors

Total variance in Y explained by all Xes in the model

$R^2$ always gets larger when more predictors are used

More predictors -> better fit

# Comparing models

Compare models using the F-ratio:

$F(k, df_{new}) = (N-k-1)R^2_{change} / k(1-R^2_{new})$

k = # of additional parameters, $R^2_{change}$ is the increase in $R^2$, $R^2_{new}$ is the $R^2$ of the new model

This only works for nested models

The new model has the same parameters and data as the old model, plus more

Otherwise, compare AIC or BIC (but makes sure the models use the same data!)

# Selecting variables

Decide on your outcome variable (Y)

Decide which Xes you definitely want to test

Which other variables should you include?

- Correlated with X but not Y? No, reduce model power
- Correlated with Y but not X? Yes, increases precision
- Correlated with both X and Y? Yes, may change the effect (think back to partial correlation!)

Use this rule also when you're designing your experiment!

# Running regression

Use a combination of the following methods:

- Hierarchical regression (start with most important vars, add more step-by-step)
- Forced entry (run with all vars at once)
- Stepwise (start with all, remove the worst-fitting variables one by one until only significant variables are left)*

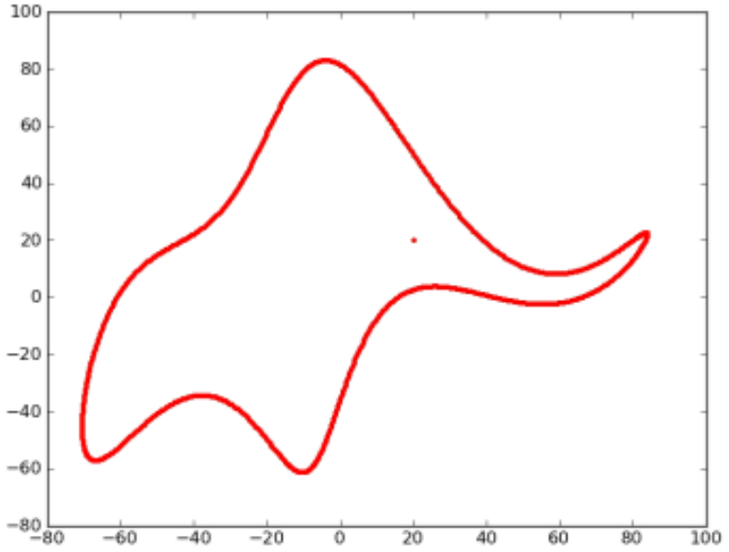
* be careful with this method! Better to do it by hand...

Theory is a far better guideline than the data itself


# Over-fitting

John von Neumann famously said:

> With four parameters I can fit an elephant, and with five I can make him wiggle his trunk.




*"Drawing an elephant with four complex parameters" by Jurgen Mayer, Khaled Khairy, and Jonathon Howard, Am. J. Phys. 78, 648 (2010)*

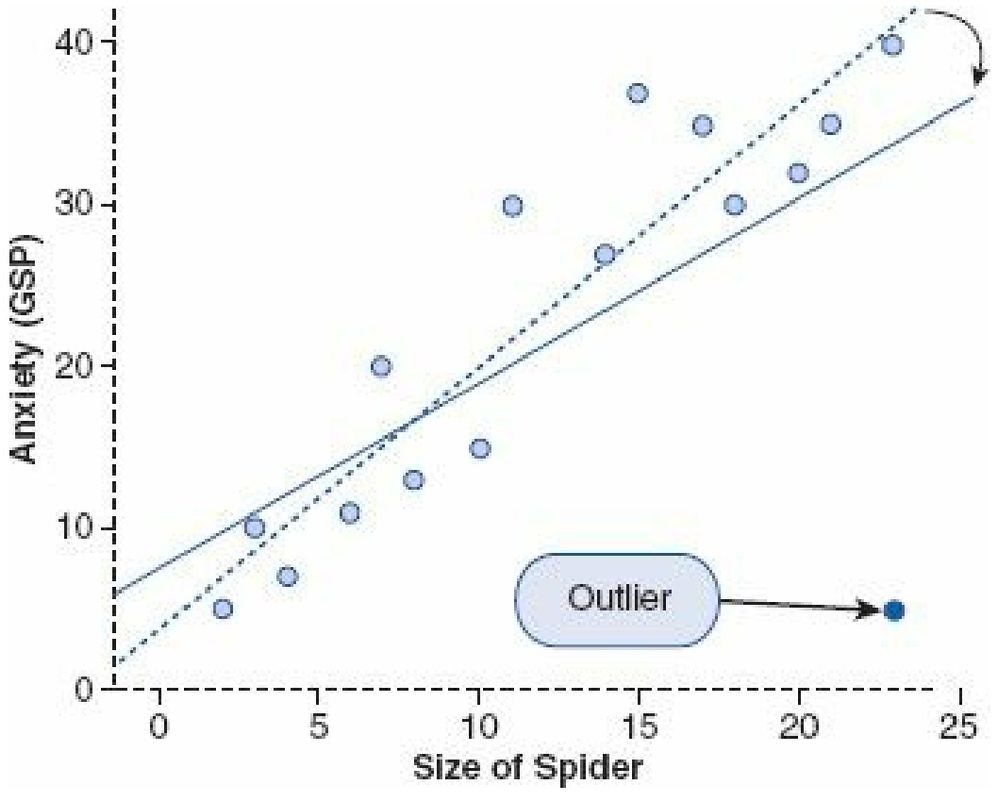
# Outliers

An outlier is a data point that differs substantially from the model

They have a very large residual (error)

Outliers can bias your regression coefficients

How can we detect them?



# Residuals

Residual method:

Take the residual (error) of each data point

Divide it by the standard deviation (this creates a z-score)

Assess the situation:

- Any cases z > 3.29: clear outliers
- If >1% of the cases z > 2.58: model is a very poor fit for some cases
- If > 5% of the cases z > 1.96: model is a rather poor fit for many cases

# Influential cases

Influencer method 1: How does each data point influence the predicted outcomes?

Cook's distance: overall influence of the data point on the model

>1 is cause for concern

Hat values (leverage): the influence of the data point on the predicted values

Average influence is (k+1)/n, anything twice (or trice) that is cause for concern

# Influential cases

Influencer method 2: How does each data point influence the model parameters?

DFBeta: run the model with and without the data point, observe the difference in each parameter

You get one for each parameter, unstandardized

Covariance ratio: convenient summary of the influence of a data point on the variances of the model parameters

Should be $< 1+(3(k+1)/n)$ and $> 1-(3(k+1)/n)$

Note: influential cases are worse than merely outlying cases!

# Assumptions

Outcome should be quantitative, continuous, and unbounded

Predictors should not be too highly correlated (see next slide)

No variables correlated with both X and Y should be left out

Homoscedasticity and independence

Linearity (although we can test for some non-linear effects)

# Multicollinearity

Both $X_1$ and $X_2$ are predictors of Y, but highly correlated with each other

Correlation of $X_1$ with Y is .4 but controlling for $X_2$ it is .2

Correlation of $X_2$ with Y is .4, but controlling for $X_1$ it is .2

Two possibilities:

$X_1$ has a high b (e.g. $b_1$ = .6) and $X_2$ has a low b (e.g. $b_2$ = .3)

$X_1$ has a low b (e.g. $b_1$ = .3) and $X_2$ has a high b (e.g. $b_2$ = .6)

Which one is correct?

# Multicollinearity

The wizard is having a hard time deciding on $b_1$ and $b_2$!

Consequences:

- $b_1$ and $b_2$ are untrustworthy, so it is hard to tell which X is most important
- the benefit of having them both is small: with either $X_1$ or $X_2$, $R^2$ = .40, with both, $R^2$ = .45

# Multicollinearity

Tests for multicollinearity:

- High correlation between Xes
- Variance inflation factor (VIF), should be lower than 10 (or 5)

# Cross-validation

How can we make sure the model would work the same on the population? (or on a different sample?)

Use the adjusted $R^2$

Use data splitting

Build your model on half the data, test it on the other half


# MLR in R

MLR = Multiple Linear Regression

# MLR in R

Run a linear model predicting album sales by adverts, airplay, and attractiveness:

```
salesModel2 <- lm(sales ~ adverts + airplay + attract, data =
album2)
summary (salesModel2)
```

$R^2$ has increased from .335 to .665

Airplay and attractiveness account for an additional 33% of variance in sales

# Parameters

Model: $sales_i = -26.61 + 0.085^*advert_i + 3.37^*airplay_i + 11.09^*attract_i + e_i$

A $1000 difference spent on ads is associated with a predicted 85 extra albums sold, keeping attractiveness and airplay constant

Two bands with the same level of advertising and attractiveness but a 1-airplay difference are expected to differ 3367 in album sales

# Parameters

Model: $sales_i$ = -26.61 + 0.085*$advert_i$ + 3.37*$airplay_i$ + 11.09*$attract_i$ + $e_i$

A band rated one unit higher in attractiveness is expected to sell 11,086 more albums than a band with a one unit lower attractiveness but the same airplay and advertising

# Parameters

What does the parameter for “a” (-26.61) mean?

With zero advertisement, zero airplay, and zero attractiveness, the album is expected to sell -26610 copies

That’s nonsense!

Also, zero attractiveness is impossible (ranges for 1 to 10)

But even with attract = 1, the result is negative

This is why your outcome variables should ideally be unbounded!

# Parameters

How can we compare the effect of adverts ($b_1$ = 0.085) with the effect of airplay ($b_2$ = 3.37)?

Difficult; they are not measured on the same scale!

Solution: standardize them! —> Beta

Load package: QuantPsyc

lm.beta(salesModel2)

Interpretation: a 1 SD difference in advertisement is related to a 0.51 SD difference in sales

# CIs

Confidence intervals are easy to obtain:

```
confint(salesModel2)
```

This gives us an idea of how certain we can be of our model parameters

If you want to be more certain, collect more data!

# Compare models

Since the models are nested, we can conduct the F-ratio test:

anova(salesModel, salesModel2)

Fit is significantly improved

$F(2, 196) = 96.447, p < .001$

# What to report

First model:

R-square, F-test

b-parameters and their significance

Subsequent models:

R-square increase, F-test comparison with previous model

b-parameters and their significance

# Outliers etc.

Standardized residuals:

```
album2$rstand <- rstandard(salesModel2)
```

Large standardized residuals are > 1.96

```
album2$rstand.large < - (album2$rstand > 1.96 |
album2$rstand < -1.96)
```

Show them!

```
album2[album2$rstand.large,c("adverts", "sales", "airplay",
"attract", "rstand")]
```

# Outliers etc.

Cook's distances, hat values (leverage), and covariance ratios

```
album2$cook <- cooks.distance(salesModel2)
album2$leverage <- hatvalues(salesModel2)
album2$covratio <- covratio(salesModel2)
```

Check them out (for data points with large residual):

```
album2[album2$rstand.large,c("cook", "leverage",
"covratio")]
```

Only covariance ratio of case 169 is a problem, but still okay given the Cook's distance << 1

# Multicollinearity

Check the Variance Inflation Factor:

load the *car* package, if you haven't already

```
vif(salesModel2)
```

well below 10 (and even 5)

If we find problems, we may need to remove some Xes

# Plotting residuals

Run the plots of the model:

plot(salesModel2)

First plot: residuals by estimated value of Y (fitted value)

This should look random

If it fans out: heteroscedasticity! If it curves: non-linearity!

Second plot: Q-Q plot to test deviations from normality

Straight line: residuals are normal

Can check the latter also with a histogram of rstud

# Solving problems

What if we have problems? (e.g. heteroscedasticity, non-normality, outliers, non-linearity)

Try transforming your outcome variable and/or predictors

What if that doesn't work?

Use bootstrapping!

# Bootstrapping

Write a bootstrap function:

```
bootReg <- function(samples, i){
    fit <-lm(sales~adverts+airplay+attract, data=samples[i,])
    return(coef(fit))
}
```

Run a bootstrap sample:

```
bootResults <- boot(album2, bootReg, 2000)
bootResults
```

# Bootstrapping

Get confidence intervals:

boot.ci(bootResults, type="bca", index = 1)

boot.ci(bootResults, type="bca", index = 2)

boot.ci(bootResults, type="bca", index = 3)

boot.ci(bootResults, type="bca", index = 4)

Compare to confint(salesModel2)


# Categorical predictors

A look forward to the t-test and ANOVA

# Categorical X

What if X is binary, e.g. Male/female?

Simply include in the regression, coded 0-1

b = the difference between the two groups

What if X is k groups, e.g. religion, city, …?

Designate one group as the **baseline**

Create k-1 dummy variables for the other groups, and put them in the regression

bs = the difference between each group and the baseline

# Dummies in R

Read the data

File: GlastonburyFestivalRegression.dat, set Name to gfr

Dataset: festival-goer hygiene (repeated measures)

Variables:

ticknumb: participant id

music: music affiliation—crusty, indy, metaller, or NMA

day1, day2, day3: hygiene level at days 1-3 (0-4 scale)

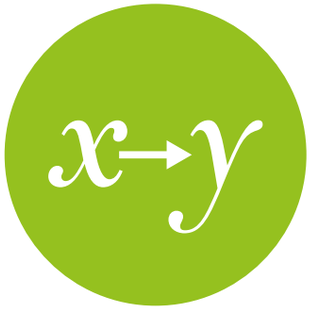
change: change in hygiene levels from day 1 to day 3

# Dummies in R

Let's take "No Musical Affiliation" as a baseline

Create contrast for the rest:

```
crusty_v_NMA<-c(1,0,0,0)
indie_v_NMA<-c(0,1,0,0)
metal_v_NMA<-c(0,0,1,0)
contrasts(gfr$music)<-cbind(crusty_v_NMA,
indie_v_NMA, metal_v_NMA)
```



# Dummies in R

Model: $change_i = a + b_1*crusty\_v\_NMA_i + b_2*indie\_v\_NMA_i + b_3*metal\_v\_NMA_i + e_i$

Interpret the results

a shows the mean change of people with NMA

bs show differences between each music type and NMA

What is the level of Y in a certain group? Simple: a + b

What about other differences? And overall effects? That's ANOVA!

# Dummies in R

Run the regression:

```
dummyModel <- lm(change ~ music, data=gfr)
summary(dummyModel)
```

Interpret the results

- What is the mean change for NMA?
- What is the difference between indie and NMA? Is it significant?
- What is the level of change for indie?

# 1- vs. 2-sided tests

Lets say you test the satisfaction between a red background color and a blue background color...What is your hypothesis?

$M_{red} > M_{blue}$?

$M_{red} < M_{blue}$?

$M_{red} \neq M_{blue}$?

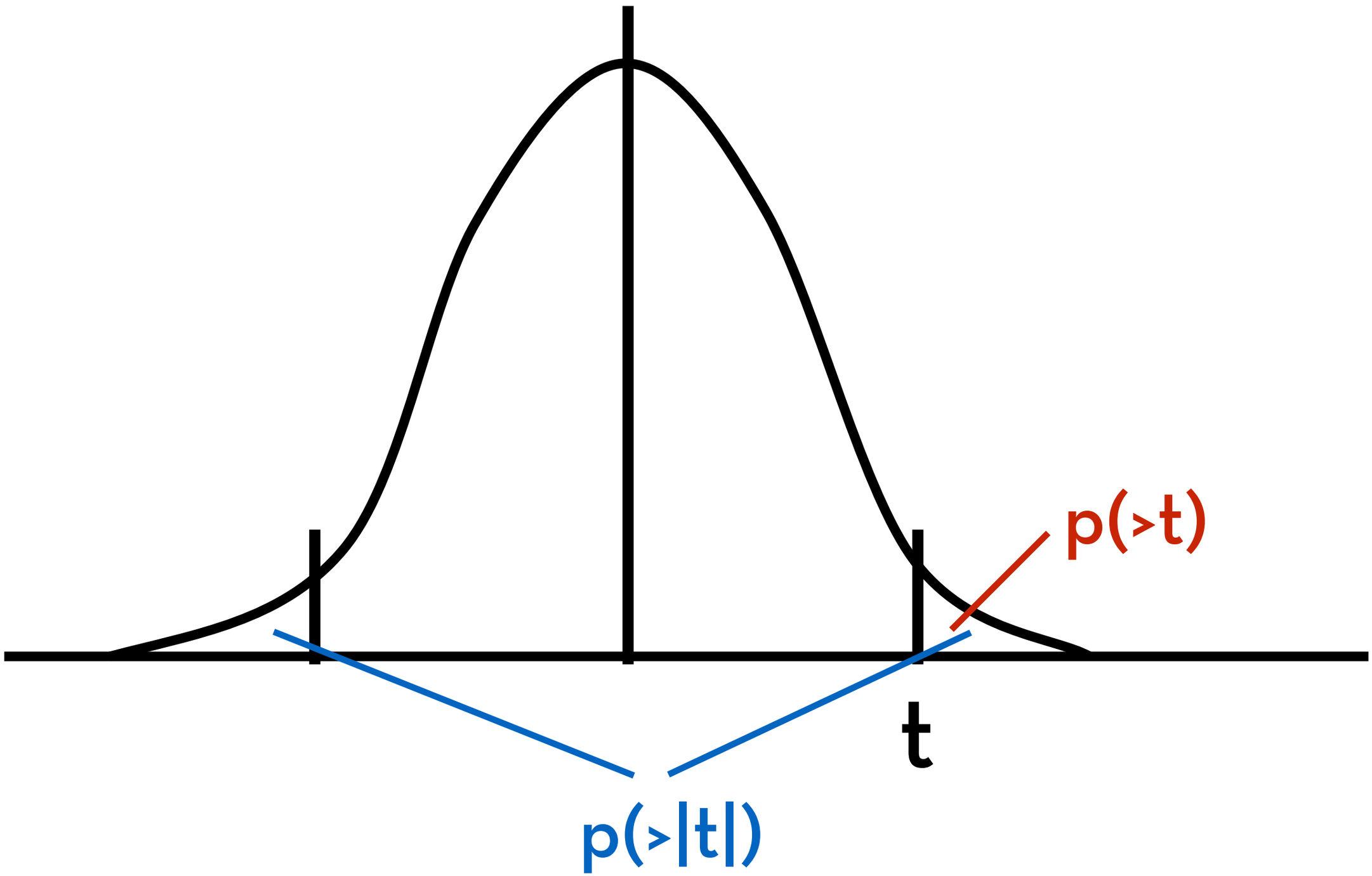
If you don't know, your test should be 2-sided (default)

You test p(>|t|)

If you have an idea, your test should be 1-sided

You test p(>t) or p(<–t), which is 1/2*p(>|t|)

# 1- vs. 2-sided tests



“It is the mark of a truly intelligent person
to be moved by statistics.”

T H A N K S !

George Bernard Shaw