Statistics for Linguists with R – a SIGIL course

Unit 2: Corpus Frequency Data & Statistical Inference

Marco Baroni[1] & Stefan Evert[2]
http://SIGIL.R-Forge.R-Project.org/

[1]Center for Mind/Brain Sciences, University of Trento
[2]Institute of Cognitive Science, University of Osnabrück

Frequency estimates & comparison

- How often is *kick the bucket* really used?
- What are the characteristics of "translationese"?
- Do Americans use more split infinitives than Britons? What about British teenagers?
- What are the typical collocates of *cat*?
- Can the next word in a sentence be predicted?
- Do native speakers prefer constructions that are grammatical according to some linguistic theory?

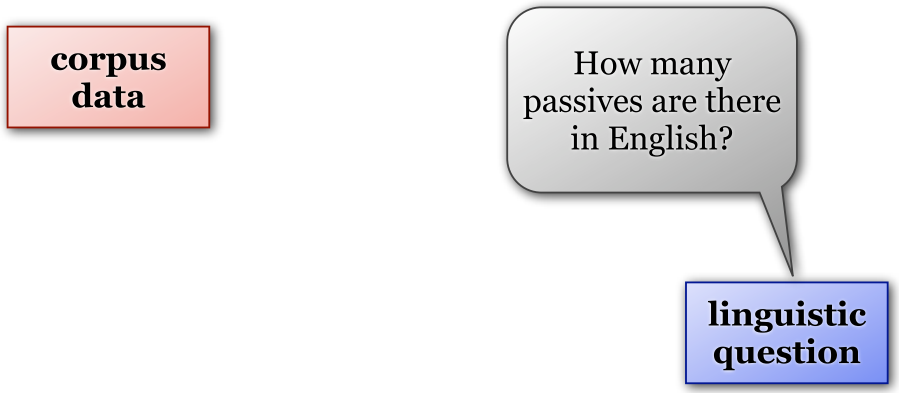
➡ evidence from frequency comparisons / estimates

A simple toy problem

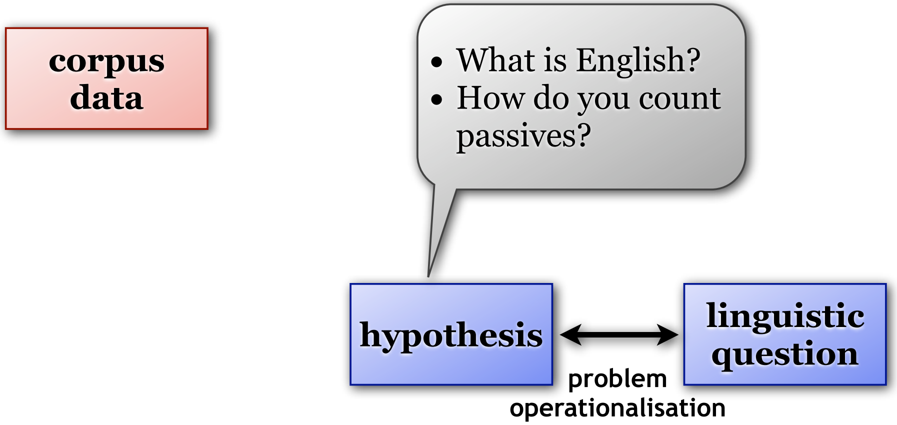
How many passives are there in English?

- American English style guide claims that
  - *"In an average English text, no more than 15% of the sentences are in passive voice. So use the passive sparingly, prefer sentences in active voice."*
  - http://www.ego4u.com/en/business-english/grammar/passive actually states that only 10% of English sentences are passives (as of June 2006)!
- We have doubts and want to verify this claim

From research question to statistical analysis



From research question to statistical analysis



What is English?

- Sensible definition: group of speakers
  - e.g. American English as language spoken by native speakers raised and living in the U.S.
  - may be restricted to certain communicative situation
- Also applies to definition of sublanguage
  - dialect (Bostonian, Cockney), social group (teenagers), genre (advertising), domain (statistics), ...
- Here: professional writing by AmE native speakers (➪ target group of style guide)

How do you count passives?

- Types vs. tokens
  - type count: How many *different* passives are there?
  - token count: How many *instances* are there?
- How many passive tokens are there in English?
  - ∞ — infinitely many, of course!
- Only **relative frequency** can be meaningful

How do you count passives?

- How many passives are there ...

  ... per million words?

  ... per thousand sentences?

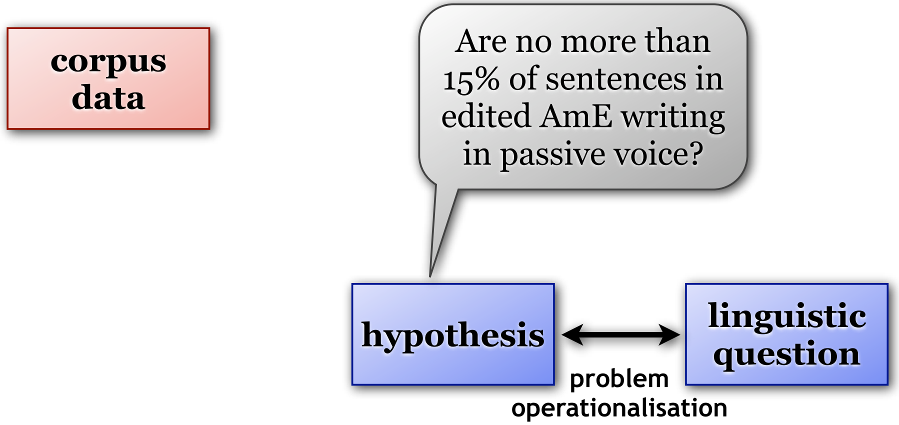
  ... per hour of recorded speech?

  ... per book?
- Are these measurements meaningful?

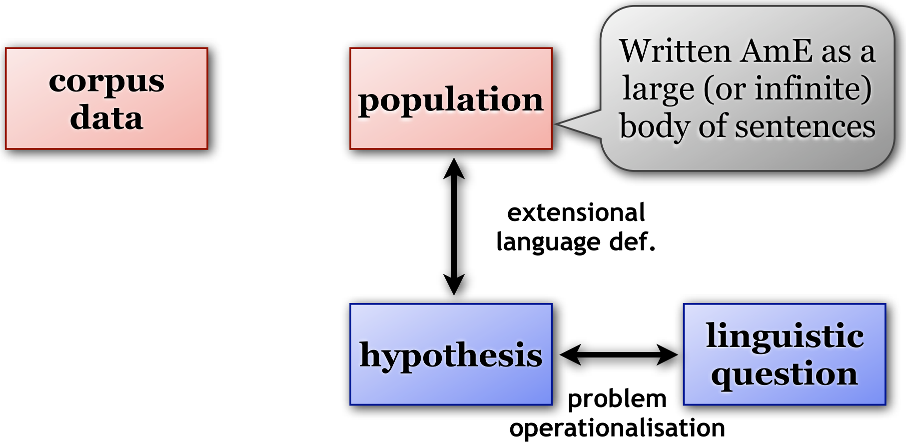
How do you count passives?

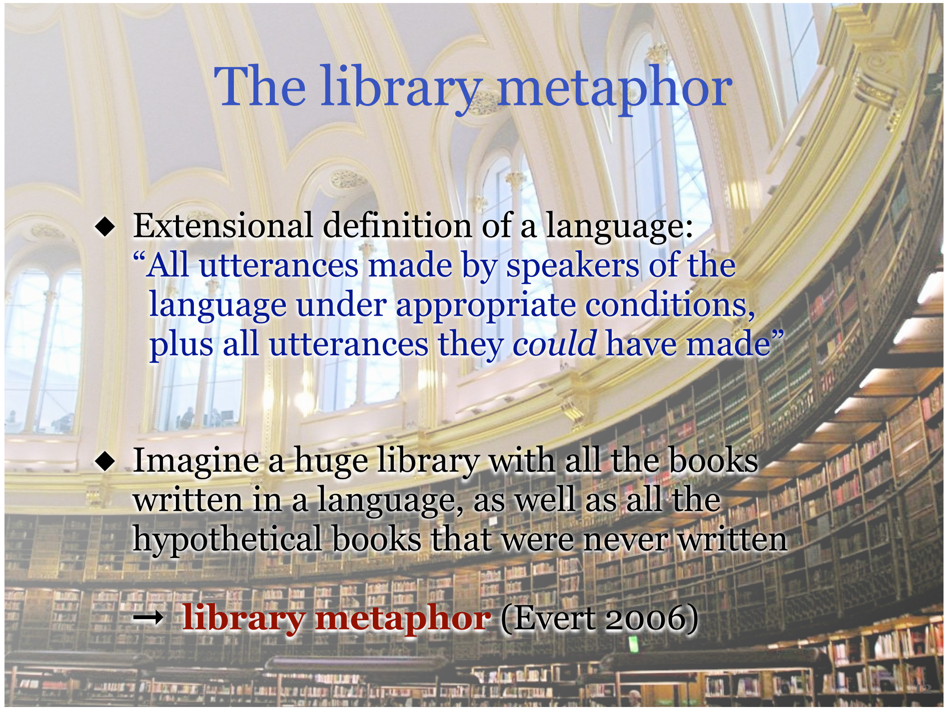
- How many passives could there be at most?
  - every VP can be in active or passive voice
  - frequency of passives is only interpretable by comparison with frequency of potential passives
- What proportion of VPs are in passive voice?
  - easier: proportion of sentences that contain a passive
  - in general with respect to some **unit of measurement**
- **Relative frequency** = **proportion** π

From research question to statistical analysis

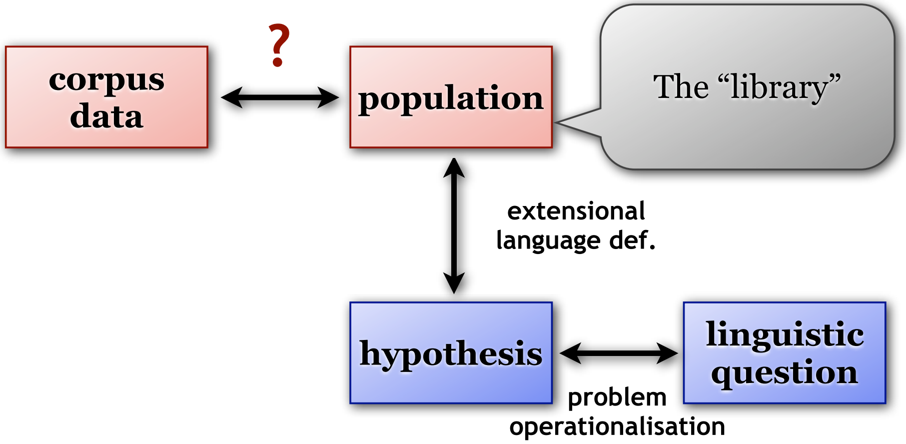


From research question to statistical analysis





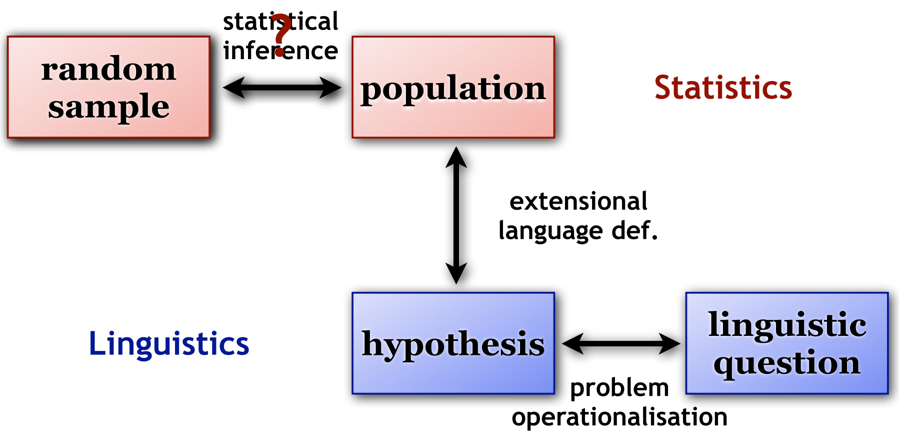
From research question to statistical analysis



How do you count tokens in an infinite library?

- Statistics deals with similar problems:
  - goal: determine properties of **large population** (human populace, objects produced in factory, …)
  - method: take (completely) **random sample** of objects, then extrapolate from sample to population
  - this works only because of **random** sampling!
- Many statistical methods are readily available

From research question to statistical analysis



Statistics & language

- Apply statistical procedure to linguistic problem
  ➪ need random sample from population
- What are the objects in our population?
  - words? sentences? texts? …
- Objects = whatever **unit of measurement** the proportions of interest are based on
  - we need to take a random sample of these units

The library metaphor

- Random sampling in the library metaphor
  - take sample of VPs (to be correct) or sentences (for convenience)
  - walk to a random shelf … … pick a random book … … open a random page … … and choose a random VP from the page
  - this gives us 1 item for our sample
  - repeat n times for **sample size n**

Types, tokens and proportions

- Proportions and relative sample frequencies are measured in terms of types & tokens
- Relative frequency of type v = proportion of tokens t_i that belong to this type

$$p = \frac{\overbrace{f(v)}^{\text{frequency of type}}}{\underbrace{n}_{\text{sample size}}}$$

- Compare relative sample frequency p against (hypothesised) population proportion π

Types, tokens and proportions

- Example: word frequencies
  - word type = dictionary entry (distinct word)
  - word token = instance of a word in library texts
- Example: passives
  - relevant VP types = active or passive (→ abstraction)
  - VP token = instance of VP in library texts
- Example: verb sucategorisation
  - relevant types = itr., tr., ditr., PP-comp., X-comp, …
  - verb token = occurrence of selected verb in text

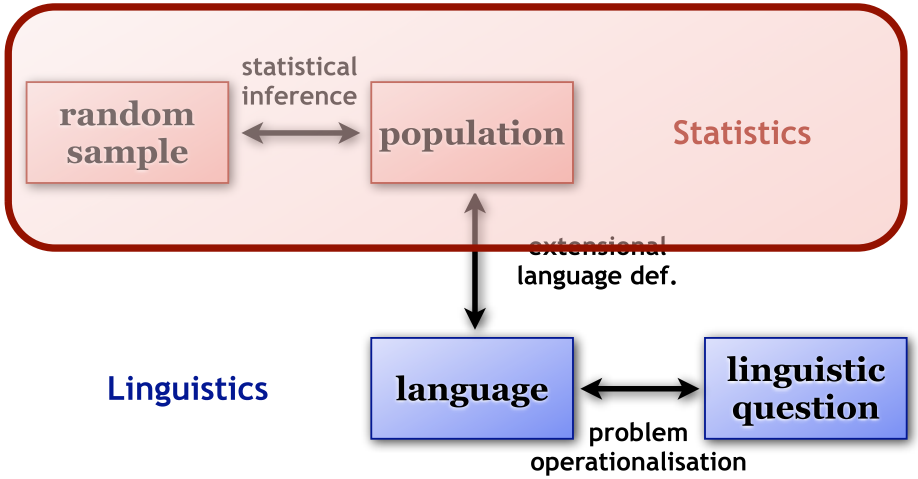
Inference from a sample

- Principle of inferential statistics
  - if a sample is picked at random, proportions should be roughly the same in sample and population
- Take a sample of, say, 100 VPs
  - observe 19 passives → p = 19% = .19
  - style guide → population proportion π = 15%
  - $p > \pi$ → reject claim of style guide?
- Take another sample, just to be sure
  - observe 13 passives → p = 13% = .13
  - $p < \pi$ → claim of style guide confirmed?

Sampling variation

- Random choice of sample ensures proportions are the same on average in sample & population
- But it also means that for every sample we will get a different value because of chance effects → **sampling variation**
- The main purpose of statistical methods is to estimate & correct for sampling variation
  - that's all there is to inferential statistics, really

Reminder: The role of statistics



Estimating sampling variation

- Assume that the style guide's claim is correct
  - the **null hypothesis** H_0, which we aim to refute

$$H_0 : \pi = .15$$

  - we also refer to $\pi_0 = .15$ as the **null proportion**
- Many corpus linguists set out to test H_0
  - each one draws a random sample of size $n = 100$
  - how many of the samples have the expected $k = 15$ passives, how many have $k = 19$, etc.?

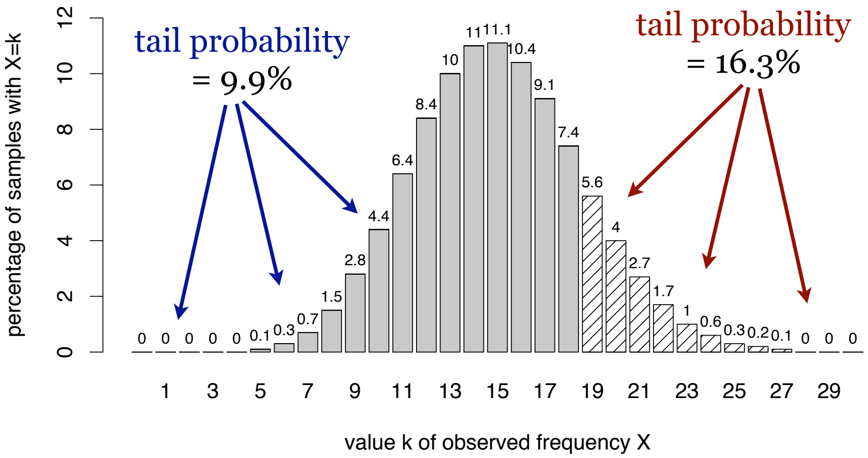
Estimating sampling variation

- We don't need an infinite number of monkeys (or corpus linguists) to answer these questions
  - randomly picking VPs from our metaphorical library is like drawing balls from an infinite urn
  - red ball = passive VP / white ball = active VP
  - H_0: assume proportion of red balls in urn is 15%
- This leads to a **binomial distribution**

$$\Pr(k) = \binom{n}{k} (\pi_0)^k (1 - \pi_0)^{n-k}$$

Binomial sampling distribution

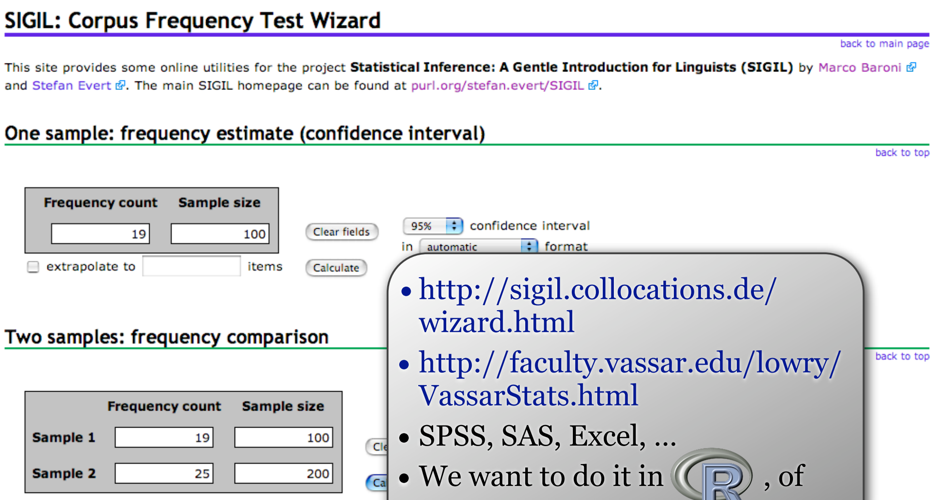
→ risk of false rejection = **p-value** = 26.2%



Statistical hypothesis testing

- Statistical **hypothesis tests**
  - define a **rejection criterion** for refuting H_0
  - control the risk of false rejection (**type I error**) to a "socially acceptable level" (**significance level**)
  - **p-value** = risk of false rejection for observation
  - p-value interpreted as amount of evidence against H_0
- Two-sided vs. one-sided tests
  - in general, two-sided tests should be preferred
  - one-sided test is plausible in our example

Hypothesis tests in practice



- http://sigil.collocations.de/wizard.html
- http://faculty.vassar.edu/lowry/VassarStats.html
- SPSS, SAS, Excel, ...
- We want to do it in R, of course

Binomial hypothesis test in R

- Relevant R function: `binom.test()`
- We need to specify
  - **observed data**: **19** passives out of **100** sentences
  - **null hypothesis**: H_0: π = **15%**
- Using the `binom.test()` function:

```
> binom.test(19, 100, p=.15)  # two-sided
> binom.test(19, 100, p=.15,  # one-sided
    alternative="greater")
```

Binomial hypothesis test in R

```
> binom.test(19, 100, p=.15)

   Exact binomial test

data:  19 and 100

number of successes = 19, number of
trials = 100, p-value = 0.2623

alternative hypothesis: true probability of
success is not equal to 0.15

95 percent confidence interval:
 0.1184432 0.2806980

sample estimates:
probability of success
                  0.19
```


Binomial hypothesis test in R

```
> binom.test(19, 100, p=.15)$p.value
[1] 0.2622728

> binom.test(23, 100, p=.15)$p.value
[1] 0.03430725

> binom.test(190, 1000, p=.15)$p.value
[1] 0.0006356804
```


Power

- Type II error = failure to reject incorrect H_0
  - the larger the discrepancy between H_0 and the true situation, the more likely it will be rejected
  - e.g. if the true proportion of passives is $\pi = .25$, then most samples provide enough evidence to reject; but true $\pi = .16$ makes rejection very difficult
  - a **powerful** test has a low type II error
- Basic insight: larger sample = more power
  - relative sampling variation becomes smaller
  - might become powerful enough to reject for $\pi = 15.1\%$

Parametric vs. non-parametric

- People often speak about parametric and non-parametric tests without precise definition
- Parametric tests make stronger assumptions
  - not just those assuming a normal distribution
  - binomial test: strong random sampling assumption
    ➞ might be considered a parametric test in this sense!
- Parametric tests are usually more powerful
  - strong assumptions allow less conservative estimate of sampling variation ➞ less evidence needed against H_0

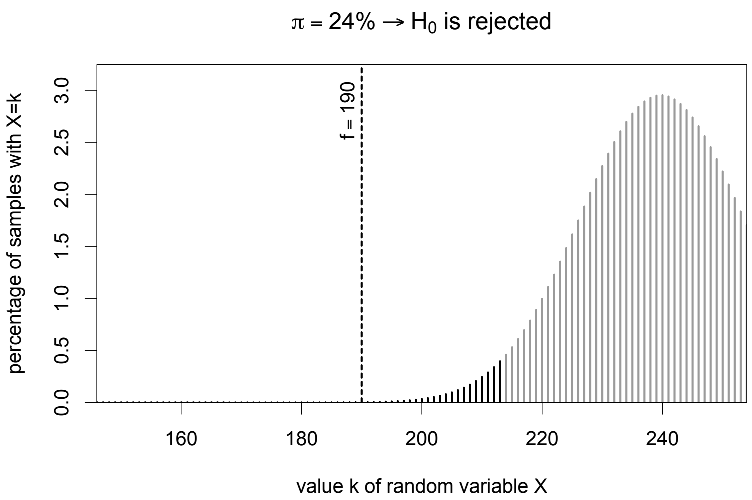
Trade-offs in statistics

- Inferential statistics is a trade-off between type I errors and type II errors
  - i.e. between **significance** and **power**
- Significance level
  - determines trade-off point
  - low significance level (p-value) → low power
- Conservative tests
  - put more weight on avoiding type I errors → weaker
  - most non-parametric methods are conservative

Confidence interval

- We now know how to test a null hypothesis H_0, rejecting it only if there is sufficient evidence
- But what if we do not have an obvious null hypothesis to start with?
  - this is typically the case in (computational) linguistics
- We can estimate the true population proportion from the sample data (relative frequency)
  - sampling variation → range of plausible values
  - such a **confidence interval** can be constructed by inverting hypothesis tests (e.g. binomial test)

Confidence interval



Confidence intervals

I'm cheating here a tiny little bit (not always an interval)

- Confidence interval = range of plausible values for true population proportion
- Size of confidence interval depends on sample size and the significance level of the test

	$n = 100$ $k = 19$	$n = 1{,}000$ $k = 190$	$n = 10{,}000$ $k = 1{,}900$
$\alpha = .05$	11.8%...28.1%	$\boxed{16.6\%\ldots21.6\%}$	18.2%...19.8%
$\alpha = .01$	10.1%...31.0%	15.9%...22.4%	18.0%...20.0%
$\alpha = .001$	8.3%...34.5%	15.1%...23.4%	17.7%...20.3%

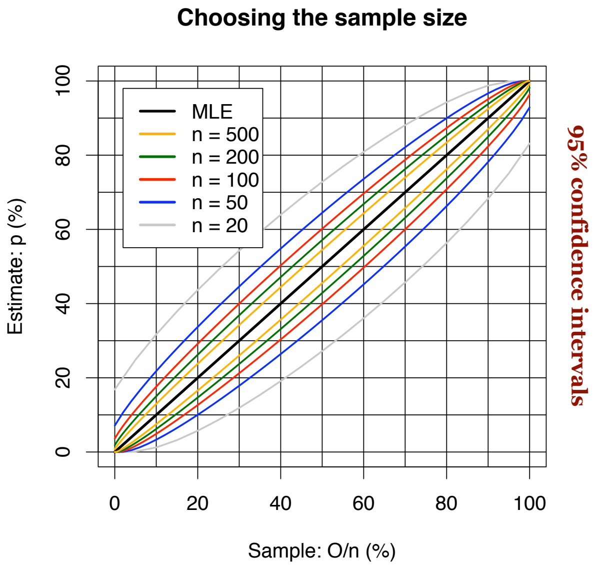
Confidence intervals in R

- Most hypothesis tests in R also compute a confidence interval (including `binom.test()`)
  - omit H_0 if only interested in confidence interval
- Significance level of underlying hypothesis test is controlled by `conf.level` parameter
  - expressed as confidence, e.g. `conf.level=.95` for significance level $\alpha = .05$, i.e. 95% confidence
- Can also compute one-sided confidence interval
  - controlled by `alternative` parameter
  - two-sided confidence intervals strongly recommended

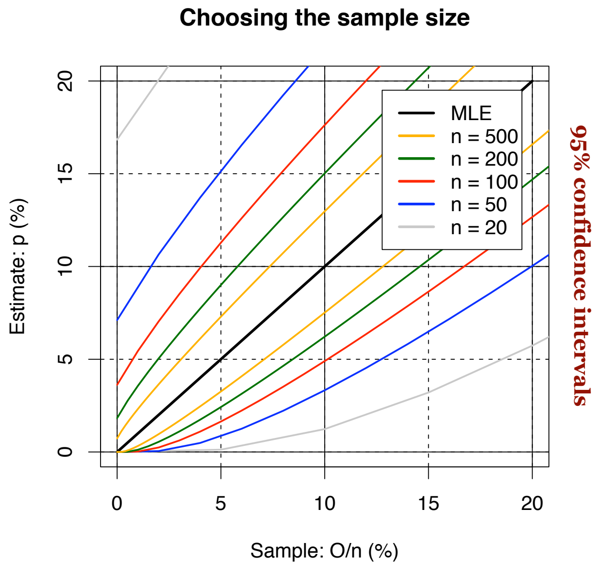
Confidence intervals in R

```
> binom.test(190, 1000, conf.level=.99)

  Exact binomial test

data:  190 and 1000

number of successes = 190, number of
trials = 1000, p-value < 2.2e-16

alternative hypothesis: true probability of
success is not equal to 0.5

99 percent confidence interval:
 0.1590920 0.2239133

sample estimates:
probability of success
                  0.19
```


Choosing sample size



Choosing sample size



Using R to choose sample size

- Call `binom.test()` with hypothetical values
- Plots on previous slides also created with R
  - requires calculation of large number of hypothetical confidence intervals
  - `binom.test()` is both inconvenient and inefficient
- The `corpora` package has a vectorised function

```
> library(corpora)
> prop.cint(190, 1000, conf.level=.99)
> ?prop.cint  # "conf. intervals for proportions"
```


Frequency comparison

- Many linguistic research questions can be operationalised as a frequency comparison
  - Are split infinitives more frequent in AmE than BrE?
  - Are there more definite articles in texts written by Chinese learners of English than native speakers?
  - Does *meow* occur more often in the vicinity of *cat* than elsewhere in the text?
  - Do speakers prefer *I couldn't agree more* over alternative compositional realisations?
- Compare observed frequencies in two samples

Frequency comparison

k_1	k_2
n_1-k_1	n_2-k_2

19	25
81	175

- Contingency table for frequency comparison
  - e.g. samples of sizes n_1 = 100 and n_2 = 200, containing 19 and 25 passives
  - H_0: same proportion in both underlying populations
- Chi-squared X^2, likelihood ratio G^2, Fisher's test
  - based on same principles as binomial test

Frequency comparison

- Chi-squared, log-likelihood and Fisher are appropriate for different (numerical) situations
  - Fisher: computationally expensive, small samples; X^2: small balanced samples; G^2: highly skewed data
- Estimates of effect size (confidence intervals)
  - e.g. difference or ratio of true proportions
  - exact confidence intervals are difficult to obtain
- Frequency comparison in practice
  - all relevant tests can be performed in R

Frequency comparison in R

- Frequency comparison with `prop.test()`
  - easy to use: specify counts k_i and sample sizes n_i
  - uses chi-squared test "behind the scenes"
  - also computes confidence interval for difference of population proportions
- E.g. for 19 passives out of 100 vs. 25 out of 200

```
> prop.test(c(19,25), c(100,200))
```

  - parameters `conf.level` and `alternative` can be used in the familiar way

Frequency comparison in R

```
> prop.test(c(19,25), c(100,200))

  2-sample test for equality of proportions with
continuity correction

data:  c(19, 25) out of c(100, 200)

X-squared = 1.7611, df = 1, p-value = 0.1845

alternative hypothesis: two.sided

95 percent confidence interval:
 -0.03201426  0.16201426

sample estimates:
prop 1 prop 2
 0.190  0.125
```


Frequency comparison in R

- Can also carry out chi-squared (`chisq.test`) and Fisher's exact test (`fisher.test`)
  - requires full contingency table as 2×2 matrix
  - NB: likelihood ratio test not in standard library
- Table for 19 out of 100 vs. 25 out of 200

```
> ct <- cbind(c(19,81),
              c(25,175))
> chisq.test(ct)
> fisher.test(ct)
```

19	**25**
81	**175**

Trade-offs in statistics: Significance vs. relevance

- Much focus on significant p-value, but ...
  - large differences may be non-significant if sample size is too small (e.g. 10/80 = 12.5% vs. 20/80 = 25%)
  - increase sample size for more powerful/sensitive test
  - very large samples lead to highly significant p-values for minimal and irrelevant differences (e.g. 1M tokens with 100,000 = 10% vs. 101,000 = 10.1% occurrences)
- It is important to assess both **significance** and **relevance** of frequency data!
  - confidence intervals combine both aspects

Some fine print

- Convenient `cont.table` function for building continency tables in `corpora` package

```
> library(corpora)
> ct <- cont.table(19, 100, 25, 200)
```

- Difference of proportions no always suitable as **measure of effect size**
  - especially if proportions can have different magnitudes (e.g. for lexical frequency data)
  - more intuitive: ratio of proportions (**relative risk**)
  - Conf. int. for similar **odds ratio** from Fisher's test

A case study: passives

- As a case study, we will compare the frequency of passives in Brown (AmE) and LOB (BrE)
  - pooled data
  - separately for each genre category
- Data files provided in CSV format
  - **`passives.brown.csv`** & **`passives.lob.csv`**
  - `cat` = genre category, `passive` = number of passives, `n_w` = number of word, `n_s` = number of sentences, `name` = description of genre category

Preparing the data

```
> Brown <- read.csv("passives.brown.csv")
> LOB <- read.csv("passives.lob.csv")

> Brown        # take a first look at the data tables
> LOB

# pooled data for entire corpus = column sums (col. 2 … 4)
> Brown.all <- colSums(Brown[, 2:4])
> LOB.all <- colSums(LOB[, 2:4])
```


Frequency tests for pooled data

```
> ct <- cbind(c(10123, 49576-10123),  # Brown
              c(10934, 49742-10934))  # LOB
> ct          # contingency table for chi-squared / Fisher
> fisher.test(ct)

# proportions test provides more interpretable effect size
> prop.test(c(10123, 10934), c(49576, 49742))

# we could in principle do the same for all 15 genres …
```

Automation: user functions

```
# user function do.test() executes proportions test for samples
# k1/n1 and k2/n2, and summarizes relevant results in compact form
> do.test <- function (k1, n1, k2, n2) {
    # res contains results of proportions test (list = data structure)
    res <- prop.test(c(k1, k2), c(n1, n2))
    # data frames are a nice way to display summary tables
    fmt <- data.frame(p=res$p.value,
      lower=res$conf.int[1], upper=res$conf.int[2])
    fmt  # return value of function = last expression
}
> do.test(10123, 49576, 10934, 49742)  # pooled data
> do.test(146, 975, 134, 947)          # humour genre
```


A nicer user function

```
# extract relevant information directly from data frames
> do.test(Brown$passive[15], Brown$n_s[15],
          LOB$passive[15], LOB$n_s[15])
# nicer version of user function with genre category labels
> do.test <- function (k1, n1, k2, n2, cat="") {
   res <- prop.test(c(k1, k2), c(n1, n2))
   fmt <- data.frame(p=res$p.value,
     lower=res$conf.int[1], upper=res$conf.int[2])
   rownames(fmt) <- cat  # add genre as row label
   fmt
}
> do.test(Brown$passive[15], Brown$n_s[15],
          LOB$passive[15], LOB$n_s[15],
          cat=Brown$name[15])
```


Automation: the `for` loop

```
# our code relies on same ordering of genre categories!
> all(Brown$cat == LOB$cat)
# carry out tests for all genres with a simple for loop
> for (i in 1:15) {
   res <- do.test(Brown$passive[i], Brown$n_s[i],
                  LOB$passive[i], LOB$n_s[i],
                  cat=Brown$name[i]))
   print(res)
}
# it would be nice to collect all these results in a single overview
# table; for this, we need a little bit of R wizardry ...
```


Collecting rows

```
# lapply collects results from iteration steps in a list
> result.list <- lapply(1:15, function (i) {
   do.test(Brown$passive[i], Brown$n_s[i],
           LOB$passive[i], LOB$n_s[i],
           cat=Brown$name[i])
})
> result <- do.call(rbind, result.list)
# think of this as an idiom that you just have to remember ...
> round(result, 5)     # easier to read after rounding
```

It's your turn now …

- Questions:
  - Which differences are significant?
  - Are the effect sizes linguistically relevant?

- A different approach:
  - You can construct a list of contingency tables with the `cont.table()` function from the `corpora` package
  - Apply `fisher.test()` or `chisq.test()` directly to each table in the list using the `lapply()` function
  - Try to extract relevant information with `sapply()`