Chi-squared tests:
Test of independence

Research Methods for Human Inquiry
Andrew Perfors

Categorical data

Chi-squared tests are used for **categorical data**: the outcome variable is nominal scale

Goodness of fit tests compare observed frequencies of one variable vs a hypothesis about the true probabilities of that variable

---

Today: *two* nominal variables

Test of association / test of independence tests if two nominal-scale variables are related to each other

Today's story…

Nobody is sure what to do, so they decide to do another vote

Today's story…

To get as many votes as possible, they set up two ballot boxes at opposite ends of the same room. Votes are anonymous ballot cards with three choices

Select one:

- ☐ Attack 
- ☐ Rescue 
- ☐ Analyse 

There might be a problem…

Doggie is spotted lurking suspiciously near box 2.
Did he tamper with the ballots in it?

Select one:

- ☐ Attack 
- ☐ Rescue 
- ☐ Analyse 

There might be a problem…

Doggie is spotted lurking suspiciously near box 2.
Did he tamper with the ballots in it?

We want to be able to compare the distribution of responses in Box 1 and Box 2.

Are they “the same” as each other, or does one have suspiciously inflated votes for Doggie?

What test should we use?

- This problem is similar to the last one
- Turns out to also be a chi-square test, almost identical to the last one.
- But it's a lot trickier to figure out how to calculate the "expected frequencies" under the null hypothesis....

What test should we use?

1) A diagnostic test statistic, *T*
2) Sampling distribution of *T* if the null is true
3) The observed *T* in your data
4) A rule that maps every value of *T* onto a decision (accept or reject H0)

Very similar to the X^2 statistic we calculated last time; but this time we aren't given the expected frequencies, so we have to estimate them first…

Our data set is the cross-tabulation

The box the ballot was in

Candidate chosen on the ballot

	BOX 1	BOX 2	
Doggie	89	97	
Gladly	41	39	
Shadow	13	11	

But we also want the overall totals

The box the ballot was in

Candidate chosen on the ballot

	BOX 1	BOX 2	Total
Doggie	89	97	**186**
Gladly	41	39	**80**
Shadow	13	11	**24**
Total	**143**	**147**	**290**

Some notation for the observations

	BOX 1	BOX 2	Total
Doggie	O_{D1}	O_{D2}	O_D
Gladly	O_{G1}	O_{G2}	O_G
Shadow	O_{S1}	O_{S2}	O_S
Total	O_1	O_2	N

O_{ij} is the observed count of the number of ballots for candidate i in box j

We'll also need some notation to describe our population parameters

	BOX 1	BOX 2
Doggie	θ_{D1}	θ_{D2}
Gladly	θ_{G1}	θ_{G2}
Shadow	θ_{S1}	θ_{S2}

H_0:

$\theta_{D1} = \theta_{D2}$

$\theta_{G1} = \theta_{G2}$

$\theta_{S1} = \theta_{S2}$

θ_{ij} is the probability that a ballot in box j turns out to be a vote for candidate i

null hypothesis: no difference between the boxes

So if the null is true, there are only three parameters

	BOX 1	BOX 2
Doggie	θ_D	θ_D
Gladly	θ_G	θ_G
Shadow	θ_S	θ_S

	Both
Doggie	θ_D
Gladly	θ_G
Shadow	θ_S

so we can collapse it

But what are these parameters?!? In the goodness of fit test, we were given them, but here we don't know what they are…

Estimate them based on the data!

	Both
Doggie	θ_D
Gladly	θ_G
Shadow	θ_S

	BOX 1	BOX 2	Total
Doggie	89	97	186
Gladly	41	39	80
Shadow	13	11	24
Total	143	147	290

Doggie got 186 of 290 votes, so the best estimate of θ_D is 186 / 290 (i.e., about 0.641)

More formally

	Both
Doggie	θ_D
Gladly	θ_G
Shadow	θ_S

	BOX 1	BOX 2	Total
Doggie	O_{D1}	O_{D2}	O_D
Gladly	O_{G1}	O_{G2}	O_G
Shadow	O_{S1}	O_{S2}	O_S
Total	O_1	O_2	N

$$\hat{\theta}_i = \frac{O_i}{N}$$

The null hypothesis population parameter for candidate i is given by the total observations for i divided by the sample size

Now we can get the expected frequencies under H_0 the same way as last time…

$$E_{ij} = O_j \times \hat{\theta}_i$$

Expected number of ballots for candidate i in box j

Total number of ballots in box j

Probability (according to H_0) of ballots in favour of candidate i

$$\hat{\theta}_i = \frac{O_i}{N}$$

Now we can get the expected frequencies under H_0 the same way as last time…

$$E_{ij} = O_j \times \frac{O_i}{N}$$

Expected number of ballots for candidate i in box j

Total number of ballots in box j

Probability (according to H_0) of ballots in favour of candidate i

$$\hat{\theta}_i = \frac{O_i}{N}$$

Observed versus **expected**.

	BOX 1	BOX 2
Doggie	O_{D1}	O_{D2}
Gladly	O_{G1}	O_{G2}
Shadow	O_{S1}	O_{S2}

	BOX 1	BOX 2
Doggie	E_{D1}	E_{D2}
Gladly	E_{G1}	E_{G2}
Shadow	E_{S1}	E_{S2}

Observed versus **expected**.

	BOX 1	BOX 2	Total
Doggie	89	97	186
Gladly	41	39	80
Shadow	13	11	24
Total	143	147	290

	BOX 1	BOX 2
Doggie	$\frac{O_D O_1}{N}$	$\frac{O_D O_2}{N}$
Gladly	$\frac{O_G O_1}{N}$	$\frac{O_G O_2}{N}$
Shadow	$\frac{O_S O_1}{N}$	$\frac{O_S O_2}{N}$

Observed versus **expected**.

	BOX 1	BOX 2	Total
Doggie	89	97	186
Gladly	41	39	80
Shadow	13	11	24
Total	143	147	290

	BOX 1	BOX 2
Doggie	$\frac{186*143}{290}$	$\frac{186*147}{290}$
Gladly	$\frac{80*143}{290}$	$\frac{80*147}{290}$
Shadow	$\frac{24*143}{290}$	$\frac{24*147}{290}$

Observed versus **expected**.

	BOX 1	BOX 2	Total
Doggie	89	97	186
Gladly	41	39	80
Shadow	13	11	24
Total	143	147	290

	BOX 1	BOX 2
Doggie	91.717	94.283
Gladly	39.448	40.552
Shadow	11.834	12.166

Now that we have E_{ij} and O_{ij}, we can calculate an analogous goodness of fit statistic!!!

$$X^2 = \sum_{i=1}^{R} \sum_{j=1}^{C} \frac{(O_{ij} - E_{ij})^2}{E_{ij}}$$

Note:

- R is the number of rows
- C is the number of columns

$$X^2 = \sum_{i=1}^{R} \sum_{j=1}^{C} \frac{(O_{ij} - E_{ij})^2}{E_{ij}}$$

Observed (O_{ij})

	BOX 1	BOX 2
Doggie	89	97
Gladly	41	39
Shadow	13	11

Expected (E_{ij})

	BOX 1	BOX 2
Doggie	91.717	94.283
Gladly	39.448	40.552
Shadow	11.834	12.166

O_{ij}-E_{ij}

	BOX 1	BOX 2
Doggie	89-91.717	97-94.283
Gladly	41-39.448	39-40.552
Shadow	13-11.834	11-12.166

$$X^2 = \sum_{i=1}^{R} \sum_{j=1}^{C} \frac{(O_{ij} - E_{ij})^2}{E_{ij}}$$

Observed (O_{ij})

	BOX 1	BOX 2
Doggie	89	97
Gladly	41	39
Shadow	13	11

Expected (E_{ij})

	BOX 1	BOX 2
Doggie	91.717	94.283
Gladly	39.448	40.552
Shadow	11.834	12.166

O_{ij}-E_{ij}

(These are also called the raw residuals and give a sense of which cells deviate the most*)

	BOX 1	BOX 2
Doggie	-2.717	2.717
Gladly	1.552	-1.552
Shadow	1.166	-1.166

* sorta. More on that later in the video

$$X^2 = \sum_{i=1}^{R} \sum_{j=1}^{C} \frac{(O_{ij} - E_{ij})^2}{E_{ij}}$$

Observed (O_{ij})

	BOX 1	BOX 2
Doggie	89	97
Gladly	41	39
Shadow	13	11

Expected (E_{ij})

	BOX 1	BOX 2
Doggie	91.717	94.283
Gladly	39.448	40.552
Shadow	11.834	12.166

$(O_{ij}-E_{ij})^2$

	BOX 1	BOX 2
Doggie	7.382	7.382
Gladly	2.409	2.409
Shadow	1.3596	1.3596

$$X^2 = \sum_{i=1}^{R} \sum_{j=1}^{C} \frac{(O_{ij} - E_{ij})^2}{E_{ij}}$$

Observed (O_{ij})

	BOX 1	BOX 2
Doggie	89	97
Gladly	41	39
Shadow	13	11

Expected (E_{ij})

	BOX 1	BOX 2
Doggie	91.717	94.283
Gladly	39.448	40.552
Shadow	11.834	12.166

$$\frac{(O_{ij}-E_{ij})^2}{E_{ij}}$$

	BOX 1	BOX 2
Doggie	0.0805	0.0783
Gladly	0.0611	0.0594
Shadow	0.1149	0.1118

$$X^2 = \sum_{i=1}^{R} \sum_{j=1}^{C} \frac{(O_{ij} - E_{ij})^2}{E_{ij}}$$

Observed (O_{ij})

	BOX 1	BOX 2
Doggie	89	97
Gladly	41	39
Shadow	13	11

Expected (E_{ij})

	BOX 1	BOX 2
Doggie	91.717	94.283
Gladly	39.448	40.552
Shadow	11.834	12.166

$$\sum_{j=1}^{C} \frac{(O_{ij}-E_{ij})^2}{E_{ij}}$$

	BOX 1	BOX 2	
Doggie	0.0805	0.0783	0.1588
Gladly	0.0611	0.0594	0.1205
Shadow	0.1149	0.1118	0.2267

$$X^2 = \sum_{i=1}^{R} \sum_{j=1}^{C} \frac{(O_{ij} - E_{ij})^2}{E_{ij}}$$

Observed (O_{ij})

	BOX 1	BOX 2
Doggie	89	97
Gladly	41	39
Shadow	13	11

Expected (E_{ij})

	BOX 1	BOX 2
Doggie	91.717	94.283
Gladly	39.448	40.552
Shadow	11.834	12.166

$$\sum_{i=1}^{R} \sum_{j=1}^{C} \frac{(O_{ij}-E_{ij})^2}{E_{ij}}$$

	BOX 1	BOX 2	
Doggie	0.0805	0.0783	0.1588
Gladly	0.0611	0.0594	0.1205
Shadow	0.1149	0.1118	0.2267

0.506

So what do we have?

X^2 ✔ 1) A diagnostic test statistic, *T*

2) Sampling distribution of *T* if the null is true

0.506 ✔ 3) The observed *T* in your data

4) A rule that maps every value of *T* onto a decision (accept or reject H0)

So what do we have?

X^2 ✔ 1) A diagnostic test statistic, *T*

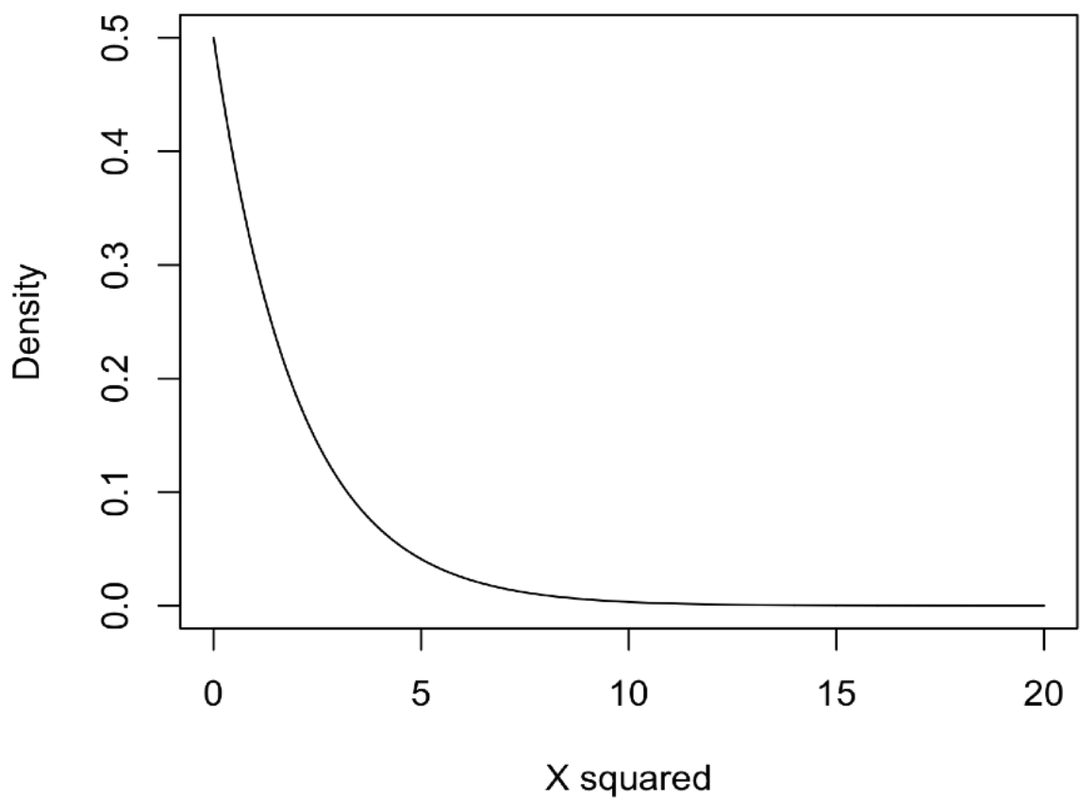
2) Sampling distribution of *T* if the null is true

0.506 ✔ 3) The observed *T* in your data

4) A rule that maps every value of *T* onto a decision (accept or reject H0)

Sampling distribution is still chi-square

- Just like last time, for the same reason: it's created by squaring and summing normally distributed variables
  - Degrees of freedom: df = (R – 1)(C – 1)
    - Why is kind of technical and not really worth going into; it's in your textbook if you care
  - For our ballot-stuffing data, this means df = 2



So what do we have?

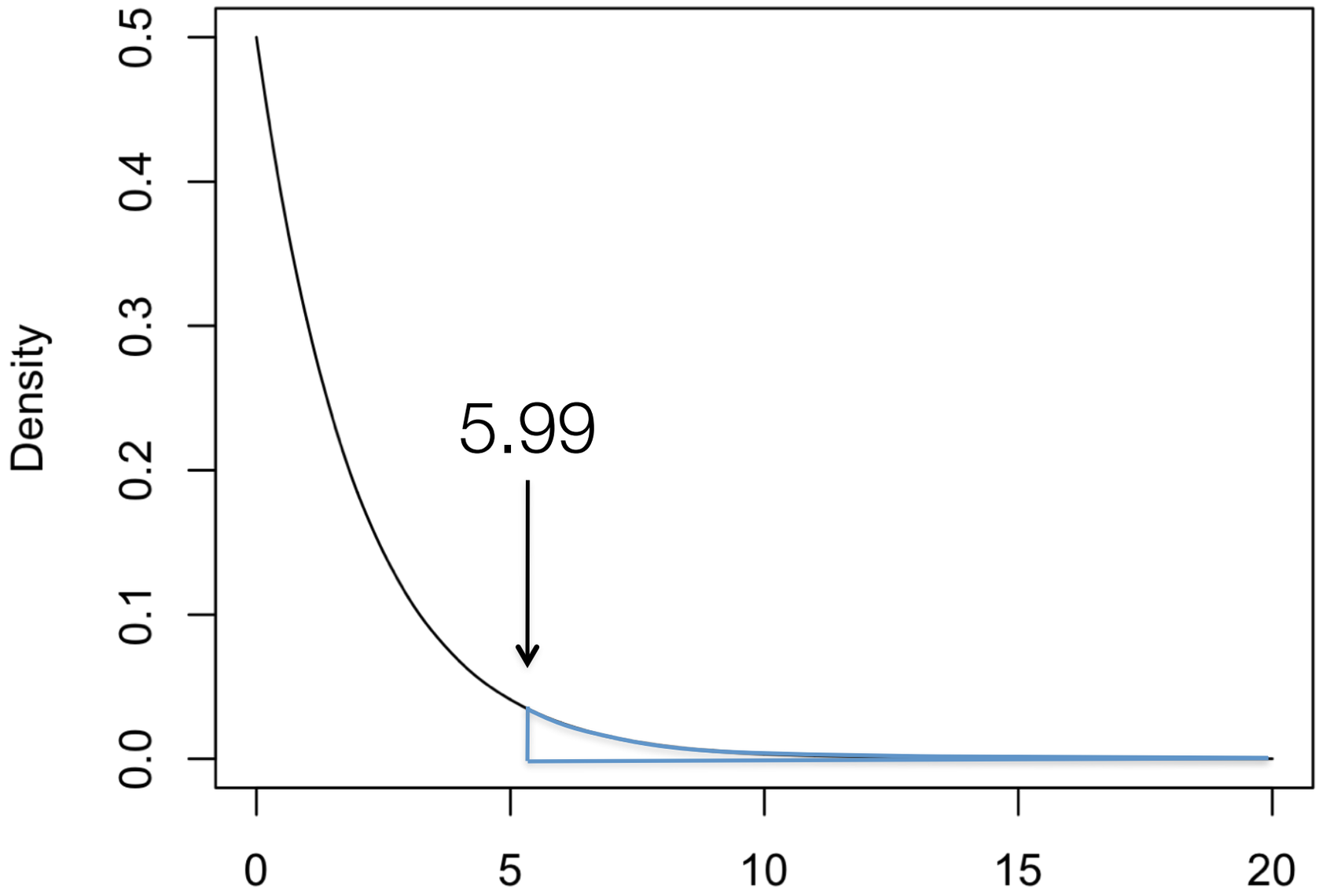
X^2 ✔ 1) A diagnostic test statistic, *T*

χ^2, df=2 ✔ 2) Sampling distribution of *T* if the null is true

0.506 ✔ 3) The observed *T* in your data

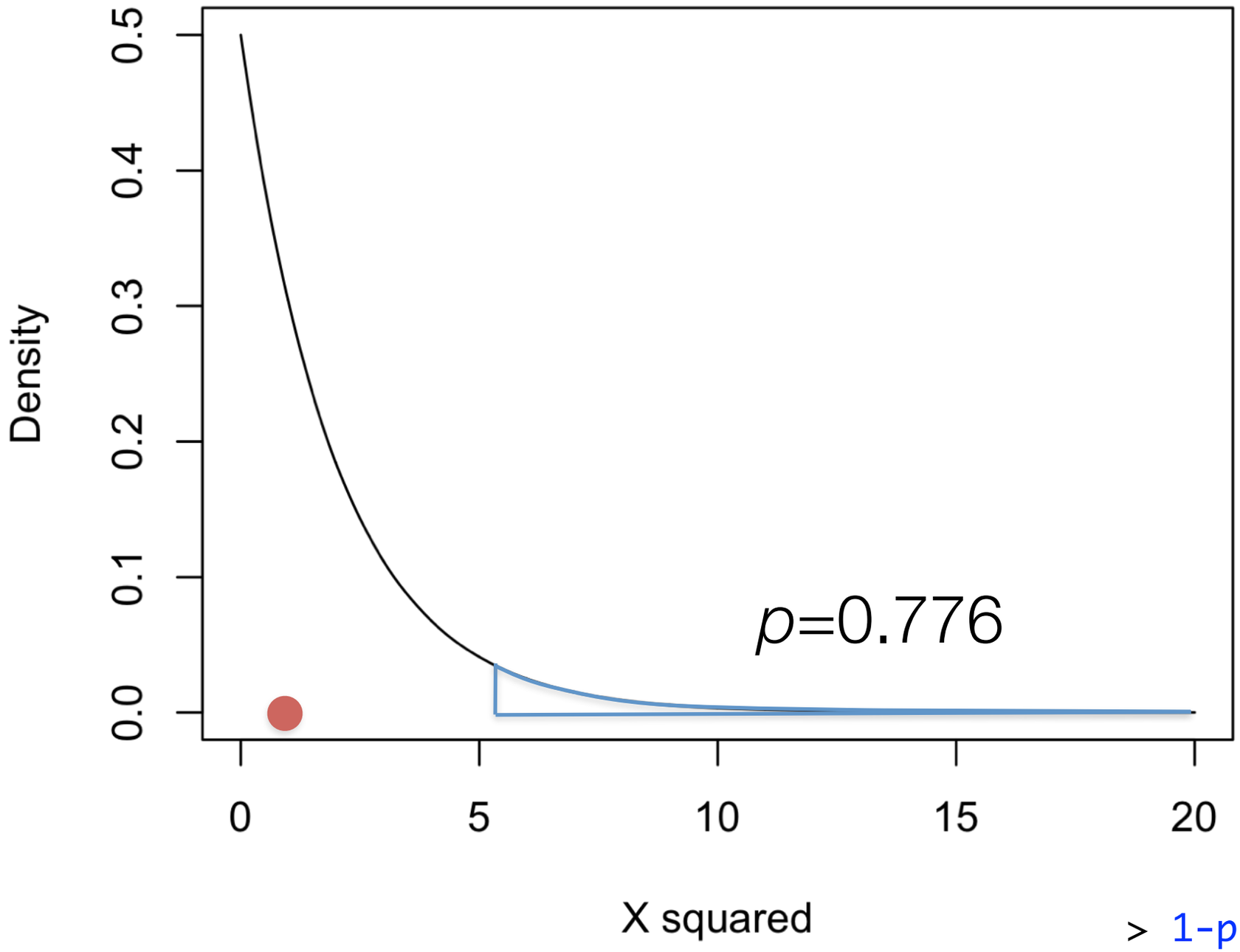
4) A rule that maps every value of *T* onto a decision (accept or reject H0)

Reject the null if X^2 is large...



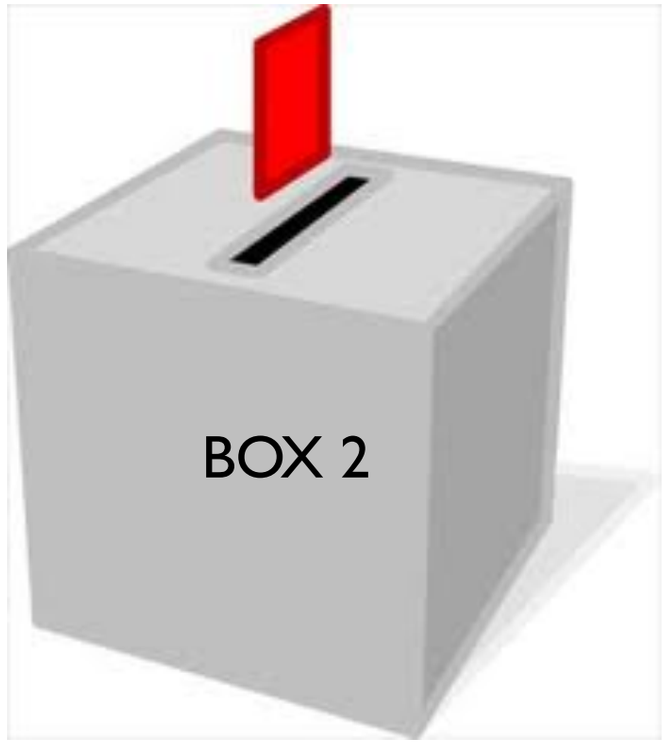

```
> qchisq(.95,df= 2)
[1] 5.991465
```

Our data are *not* in the rejection region!




```
> 1-pchisq(0.506,df=2)
[1] 0.7764679
```

So we do not reject the null hypothesis, and cannot conclude that Doggie cheated



Summary: A quick comparison

Test	Goodness of fit	Independence
variables?	one nominal	two nominal
related to?	a hypothesis about true probabilities of the variable	one another
test statistic	$X^2 = \sum_{i=1}^{k} \frac{(O_i - E_i)^2}{E_i}$	$X^2 = \sum_{i=1}^{R} \sum_{j=1}^{C} \frac{(O_{ij} - E_{ij})^2}{E_{ij}}$
sampling distribution of test statistic	χ^2	χ^2
degrees of freedom	*k-1*, where *k*=# of categories	*(r-1)(c-1)* where *r*=# of categories of one variable, and *c*=# of categories of other

Running this in R

```
> loc2 <- here("twoboxesvotes.csv")
> d2 <- read_csv(file=loc2)
> d2

# A tibble: 290 x 2
   box   vote
   <chr> <chr>
 1 box1  gladly
 2 box1  doggie
 3 box1  doggie
 4 box1  gladly
 5 box1  doggie
 6 box1  doggie
 7 box1  doggie
 8 box1  doggie
 9 box1  doggie
10 box1  doggie
# … with 280 more rows
```

There are two ways to do the test

Running it in R

```
> boxesTable <- table(d2$vote,d2$box)
> boxesTable

        box1 box2
  doggie   89   97
  gladly   41   39
  shadow   13   11
```

First we need to use `table()` to construct the cross-tabulation from the raw data

One way to do a test of independence

- The `chisq.test()` function allows us to run this test in two different ways
- Method #1: set the `x` argument to be equal to the cross-tabulated frequencies...

```
> chisq.test( x = boxesTable )

    Pearson's Chi-squared test

data:  boxesTable
X-squared = 0.50568, df = 2, p-value = 0.7766
```

One way to do a test of independence

- The `chisq.test()` function allows us to run this test in two different ways
- Method #2: set the `x` argument to be equal to one variable, and the `y` argument to be equal to the other one

```
> chisq.test( x = d2$vote, y=d2$box )

    Pearson's Chi-squared test

data:  d2$vote and d2$box
X-squared = 0.50568, df = 2, p-value = 0.7766
```

Note that this is different than the goodness of fit test!
(in which you specify `x` and `p`, not `x` and `y`)

Some other cool stuff

- The `chisq.test()` function secretly calculates a bunch of things, which we can see if we assign it to a variable

```
> ctResult <- chisq.test( x = boxesTable )
> ctResult

    Pearson's Chi-squared test

data:  boxesTable
X-squared = 0.50568, df = 2, p-value = 0.7766
```

But ctResult is actually secretly a list, which means it contains lots of individual variables that you can access with the $ operator!

Some other cool stuff

- The `chisq.test()` function secretly calculates a bunch of things, which we can see if we assign it to a variable

```
> ctResult$statistic
X-squared
0.5056765
```

Returns the χ^2 statistic that was calculated

```
> ctResult$p.value
[1] 0.7765935
```

Returns the p-value that was calculated

```
> ctResult$parameter
df
 2
```

Returns the degrees of freedom

Some other cool stuff

- The `chisq.test()` function secretly calculates a bunch of things, which we can see if we assign it to a variable

```
> ctResult$observed
        box1 box2
doggie    89   97
gladly    41   39
shadow    13   11
```

Returns the set of observations O

	BOX 1	BOX 2
Doggie	89	97
Gladly	41	39
Shadow	13	11

```
> ctResult$expected
             box1     box2
doggie 91.71724 94.28276
gladly 39.44828 40.55172
shadow 11.83448 12.16552
```

Returns the calculated expected values E

	BOX 1	BOX 2
Doggie	91.717	94.283
Gladly	39.448	40.552
Shadow	11.834	12.166

Some other cool stuff

- The `chisq.test()` function secretly calculates a bunch of things, which we can see if we assign it to a variable

```
> ctResult$residuals
              box1       box2
doggie -0.2837283  0.2798415
gladly  0.2470589 -0.2436744
shadow  0.3388005 -0.3341592
```

Returns the Pearson residuals

Pearson residuals are the raw residuals divided by $\sqrt{E}$, which helps normalise so larger cells don't "count" too much

O_{ij}-E_{ij}

(These are also called the raw residuals and give a sense of which cells deviate the most*)

	BOX 1	BOX 2
Doggie	-2.717	2.717
Gladly	1.552	-1.552
Shadow	1.166	-1.166

Some people call these "standardised residuals" but others call something else the standardised residuals so I prefer to avoid that terminology altogether

Some other cool stuff

- The `chisq.test()` function secretly calculates a bunch of things, which we can see if we assign it to a variable

```
> ctResult$stdres
              box1       box2
doggie  -0.6654653  0.6654653
gladly   0.4077840 -0.4077840
shadow   0.4968698 -0.4968698
```

Returns the adjusted residuals

These are the raw residuals divided by the standard error $\sqrt{E(1 - O_i/N)(1 - O_j/N)}$

Essentially what this does is allow us to interpret these as *kind of* indicating how many "standard deviations" away individual cells are.

A (weak) rule of thumb some use is that if you have an overall significant test and any adjusted residuals are more extreme than +/- 1.96 or so, those individual items are significant as well

You don't need to know either this or the Pearson residual equation!

Adjusted residual example

- Let's look at them for the data from Day 1

```
> votingTable <- table(d$vote)
> votingTable

 bunny doggie gladly shadow
     2     55     36      7
```

```
> ctVote <- chisq.test(x=votingTable,p=ed)
> ctVote

    Chi-squared test for given probabilities

data:  votingTable
X-squared = 11.304, df = 3, p-value = 0.01019
```

Suggests that people really didn't want to leave (votes were lower for that option than you'd expect based on Bunny's probability alone)

```
> ctVote$stdres

     bunny     doggie     gladly     shadow
-3.1749016  1.9077421  0.5512683 -0.5706866
```

(This is a weak rule of thumb; in Week 8 I'll tell you about a more principled way to do this)

Exercises are in w6day2exercises.Rmd