Uncertainty: (how to) deal with it

Peter Ralph

1 October 2020 – Advanced Biological Statistics

Course overview

image: Frank Klausz, woodandshop.com

2 . 2

Steps in data analysis

1. Care, or at least think, about the data.
2. Look at the data.
3. Query the data.
4. Check the results.
5. Communicate.

2 . 3

Steps in data analysis

1. Care, or at least think, about the data.
2. Look at the data.
3. Query the data.
4. Check the results.
5. Communicate.

Often “statistics” focuses on *querying*. Doing that effectively requires all the other steps, too.

2 . 3

Prerequisites

We'll be assuming that you have some familiarity with

- programming, and
- statistics

Prerequisites

We'll be assuming that you have some familiarity with

- programming, and
- statistics

For instance, you should be able to figure out what this means:

```
x = c(2, 4, 3, 6)
y = c(5, 12, 4, 10, 2)
t.test(x, y)
```

```
##
##  Welch Two Sample t-test
##
## data:  x and y
## t = -1.3761, df = 5.4988, p-value = 0.2222
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -8.031728  2.331728
## sample estimates:
## mean of x mean of y
##      3.75      6.60
```


Overview and mechanics

See the course website.

Break

Please take 10 minutes to

1. answer the “Welcome Survey” on Canvas,
2. get the course repository from github,
3. install Rstudio and/or
4. move around.

Questions?

Some core statistical concepts

Statistics or parameters?

A statistic is

a numerical description of a dataset.

Statistics or parameters?

A statistic is

a numerical description of a dataset.

A parameter is

a numerical attribute of a model of reality.

Statistics or parameters?

A statistic is

a numerical description of a dataset.

A parameter is

a numerical attribute of a model of reality.

Often, *statistics* are used to estimate *parameters*.

The two heads of classical statistics

estimating parameters, with uncertainty *(confidence intervals)*

evaluating (in-)consistency with a particular situation *(p-values)*

The two heads of classical statistics

estimating parameters, with uncertainty *(confidence intervals)*

evaluating (in-)consistency with a particular situation *(p-values)*

1. What do these data tell us about the world?
2. How strongly do we believe it?

The two heads of classical statistics

estimating parameters, with uncertainty *(confidence intervals)*

evaluating (in-)consistency with a particular situation *(p-values)*

1. What do these data tell us about the world?
2. How strongly do we believe it?

This week: digging in, with simple examples.

Lurking, behind everything:

is *uncertainty*

Lurking, behind everything:

is *uncertainty*

thanks to *randomness*.

Lurking, behind everything:

is *uncertainty*

thanks to *randomness*.

How do we understand randomness, concretely and quantitatively?

Lurking, behind everything:

is *uncertainty*

thanks to *randomness*.

How do we understand randomness, concretely and quantitatively?

With *models*.

A quick look at some data

Some data

AirBnB hosts in Portland, OR: website and download link.

```
airbnb <- read.csv("../Datasets/portland-airbnb-listings.csv")
nrow(airbnb)
```

```
## [1] 5634
```

```
names(airbnb)
```

```
##   [1] "id"                                 "listing_url"
##   [5] "name"                               "summary"
##   [9] "experiences_offered"                "neighborhood_overview"
##  [13] "access"                             "interaction"
##  [17] "medium_url"                         "picture_url"
##  [21] "host_url"                           "host_name"
##  [25] "host_about"                         "host_response_time"
##  [29] "host_is_superhost"                  "host_thumbnail_url"
##  [33] "host_listings_count"                "host_total_listings_count"
##  [37] "host_identity_verified"             "street"
##  [41] "neighbourhood_group_cleansed"       "city"
##  [45] "market"                             "smart_location"
##  [49] "latitude"                           "longitude"
##  [53] "room_type"                          "accommodates"
##  [57] "beds"                               "bed_type"
##  [61] "price"                              "weekly_price"
##  [65] "cleaning_fee"                       "guests_included"
##  [69] "maximum_nights"                     "minimum_minimum_nights"
##  [73] "maximum_maximum_nights"             "minimum_nights_avg_ntm"
##  [77] "has_availability"                   "availability_30"
##  [81] "availability_365"                   "calendar_last_scraped"
```


Questions: how much does an AirBnB typically cost in Portland? Do “instant bookable” ones cost more?

Second, look at the data

```r
summary(airbnb$price)
```

```
##    Length     Class      Mode
##      5634 character character
```



```r
summary(airbnb$instant_bookable)
```

```
##    Length     Class      Mode
##      5634 character character
```


Whoops

```r
airbnb$price <- as.numeric(gsub("$", "", airbnb$price, fixed=TRUE))
```

```
## Warning: NAs introduced by coercion
```

```r
airbnb$instant_bookable <- (airbnb$instant_bookable == "t")
```

```r
summary(airbnb$price)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
##     0.0    69.0    95.0   119.5   136.0   999.0      32
```

```r
summary(airbnb$instant_bookable)
```

```
##    Mode   FALSE    TRUE 
## logical    2960    2674
```



```r
table(airbnb$bed_type) # hm
```

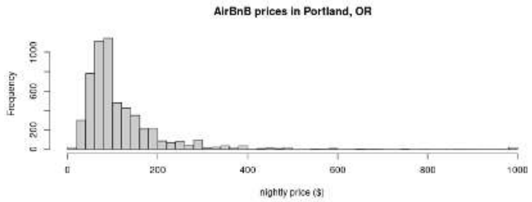
```
## 
##        Airbed         Couch         Futon Pull-out Sofa      Real Bed 
##             9             7            42            21          5555
```


How much is a typical night?

```r
mean(airbnb$price, na.rm=TRUE)
```

```
## [1] 119.5396
```

```r
hist(airbnb$price, breaks=40, xlab='nightly price ($)', col=grey(.8),
        xlim=range(airbnb$price, finite=TRUE), main='AirBnB prices in Portland, OR')
```

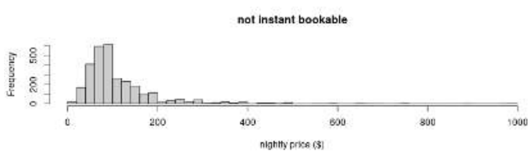


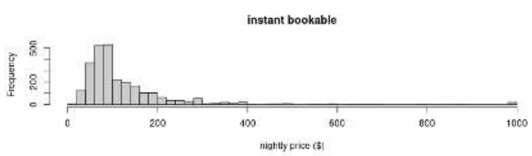

```r
hist(airbnb$price, breaks=40, xlab='nightly price ($)', col=grey(.8),
        xlim=range(airbnb$price, finite=TRUE), main='AirBnB prices in Portland, OR')
```

Conclusion?

Do "instant bookable" charge more?

```r
layout(1:2)
instant <- airbnb$price[airbnb$instant_bookable]
not_instant <- airbnb$price[!airbnb$instant_bookable]
hist(not_instant, breaks=40, xlab='nightly price ($)', col=grey(.8),
        xlim=range(airbnb$price, finite=TRUE), main='not instant bookable')
hist(instant, breaks=40, xlab='nightly price ($)', col=grey(.8), main='instant
        bookable')
```





```r
(tt <- t.test(instant, not_instant))
```

```
##
##  Welch Two Sample t-test
##
## data:  instant and not_instant
## t = 3.6482, df = 5039.8, p-value = 0.0002667
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##   4.475555 14.872518
## sample estimates:
## mean of x mean of y
##  124.6409  114.9668
```


Conclusion

> *Instant bookable hosts cost more than others (P=0.00027, t-test with df=5039.7695486).*

Conclusion

> *Instant bookable hosts cost more than others (P=0.00027, t-test with df=5039.7695486).*

Critique this conclusion, and write your own.

Scribe: person with the smallest `sample.int(1000, 1)`.

Don't forget Steps 1 and 5!

1. Care, or at least think, about the data.
2. Communicate.

Don't forget Steps 1 and 5!

1. Care, or at least think, about the data.
2. Communicate.

How *big* is the difference? How sure are we?

Don't forget Steps 1 and 5!

1. Care, or at least think, about the data.
2. Communicate.

How *big* is the difference? How sure are we?

Statistical significance does not imply real-world significance.

Revised conclusion (in class)

Instant bookable hosts cost on average $10 more than not instant bookable, with a 95% confidence interval of $4.50 to $15. The distribution of prices in the two groups were very similar: for instance, the first and third quantiles of instant bookable hosts are $70 and $145, and those of not instant bookable hosts are $68 and $130, respectively. The average instant bookable cost was about $125, with a 95% confidence interval of +/- about $4; non-instant bookable hosts cost on average $115 per night, with a 95% CI of about +/- $3. Note that the difference of $10 is smallish compared to the price of a room, but the difference is highly significant (p=.0003, t-test with 5039 degrees of freedom) because of the large sample sizes.

So: what did we just do?

Hypothesis testing and p-values

A p-value is

A p-value is

> *the probability of seeing a result at least as surprising as what was observed in the data, if the null hypothesis is true.*

A p-value is

the probability of seeing a result at least as surprising as what was observed in the data, if the null hypothesis is true.

Usually, this means

- *a result* - numerical value of a statistic
- *surprising* - big
- *null hypothesis* - the model we use to calculate the p-value

which can all be defined to suit the situation.

What does a small p-value mean?

If the null hypothesis *were* true, then you'd be really unlikely to see something like what you actually *did*.

What does a small p-value mean?

If the null hypothesis *were* true, then you'd be really unlikely to see something like what you actually *did*.

So, either the "null hypothesis" is not a good description of reality or something surprising happened.

What does a small p-value mean?

If the null hypothesis *were* true, then you'd be really unlikely to see something like what you actually *did*.

So, either the "null hypothesis" is not a good description of reality or something surprising happened.

How useful this is depends on the null hypothesis.

For instance

```
##
##  Welch Two Sample t-test
##
## data:  instant and not_instant
## t = 3.6482, df = 5039.8, p-value = 0.0002667
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##   4.475555 14.872518
## sample estimates:
## mean of x mean of y
##  124.6409  114.9668
```


Also for instance

```
t.test(airbnb$price)
```

```
##
##  One Sample t-test
##
## data:  airbnb$price
## t = 91.32, df = 5601, p-value < 2.2e-16
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##  116.9734 122.1058
## sample estimates:
## mean of x
##  119.5396
```


Also for instance

```
t.test(airbnb$price)
```

```
##
##  One Sample t-test
##
## data:  airbnb$price
## t = 91.32, df = 5601, p-value < 2.2e-16
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##  116.9734 122.1058
## sample estimates:
## mean of x
##  119.5396
```

Is *that* p-value useful?

Exercise:

My hypothesis: People tend to have longer index fingers on the hand they write with because writing stretches the ligaments.

(class survey) How many people have a longer index finger on the hand they write with?

Exercise:

My hypothesis: People tend to have longer index fingers on the hand they write with because writing stretches the ligaments.

(class survey) How many people have a longer index finger on the hand they write with?

(class survey) Everyone flip a coin:

```
ifelse(runif(1) < 0.5, "H", "T")
```

and put the result in this google doc.

Exercise:

My hypothesis: People tend to have longer index fingers on the hand they write with because writing stretches the ligaments.

(class survey) How many people have a longer index finger on the hand they write with?

(class survey) Everyone flip a coin:

```
ifelse(runif(1) < 0.5, "H", "T")
```

and put the result in this google doc.

We want to estimate the parameter

$$\theta = \mathbb{P}(\text{random person has writing finger longer}),$$

and now we have a *fake dataset with* $\theta = 1/2$.

Let's get some more data:

```
n <- 37 # class size
sum(ifelse(runif(1) < 1/2, "H", "T") == "H")
```

and put the result in the same google doc.

Let's get some more data:

```
n <- 37 # class size
sum(ifelse(runif(1) < 1/2, "H", "T") == "H")
```

and put the result in the same google doc.

Now we can estimate the p-value for the hypothesis that $\theta = 1/2$.

A faster method:

```
replicate(1000, sum(rbinom(n, 1, 1/2) > 0))
```

A faster method:

```
replicate(1000, sum(rbinom(n, 1, 1/2) > 0))
```

or, equivalently,

```
rbinom(1000, n, 1/2)
```


(in class)

```
mean(replicate(10000, rbinom(1, 36, 1/2) >= 20))
```

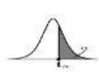
```
## [1] 0.3057
```

Here, we've estimated that the difference in numbers of people with a longer finger on each hand is not statistically significant (p\appox$0.3$, by simulation).

So, where do p-values come from?

Either math:

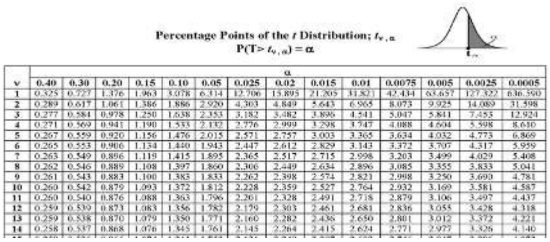
Percentage Points of the t Distribution; $t_{v,\alpha}$
$P(T > t_{v,\alpha}) = \alpha$



	α													
v	0.40	0.30	0.20	0.15	0.10	0.05	0.025	0.02	0.015	0.01	0.0075	0.005	0.0025	0.0005
1	0.325	0.727	1.376	1.963	3.078	6.314	12.706	15.895	21.205	31.821	42.434	63.657	127.322	636.590
2	0.289	0.617	1.061	1.386	1.886	2.920	4.303	4.849	5.643	6.965	8.073	9.925	14.089	31.598
3	0.277	0.584	0.978	1.250	1.638	2.353	3.182	3.482	3.896	4.541	5.047	5.841	7.453	12.924
4	0.271	0.569	0.941	1.190	1.533	2.132	2.776	2.999	3.298	3.747	4.088	4.604	5.598	8.610
5	0.267	0.559	0.920	1.156	1.476	2.015	2.571	2.757	3.003	3.365	3.634	4.032	4.773	6.869
6	0.265	0.553	0.906	1.134	1.440	1.943	2.447	2.612	2.829	3.143	3.372	3.707	4.317	5.959
7	0.263	0.549	0.896	1.119	1.415	1.895	2.365	2.517	2.715	2.998	3.203	3.499	4.029	5.408
8	0.262	0.546	0.889	1.108	1.397	1.860	2.306	2.449	2.634	2.896	3.085	3.355	3.833	5.041
9	0.261	0.543	0.883	1.100	1.383	1.833	2.262	2.398	2.574	2.821	2.998	3.250	3.690	4.781
10	0.260	0.542	0.879	1.093	1.372	1.812	2.228	2.359	2.527	2.764	2.932	3.169	3.581	4.587
11	0.260	0.540	0.876	1.088	1.363	1.796	2.201	2.328	2.491	2.718	2.879	3.106	3.497	4.437
12	0.259	0.539	0.873	1.083	1.356	1.782	2.179	2.303	2.461	2.681	2.836	3.055	3.428	4.318
13	0.259	0.538	0.870	1.079	1.350	1.771	2.160	2.282	2.436	2.650	2.801	3.012	3.372	4.221
14	0.258	0.537	0.868	1.076	1.345	1.761	2.145	2.264	2.415	2.624	2.771	2.977	3.326	4.140

So, where do p-values come from?

Either math:



Percentage Points of the t Distribution; $t_{\nu,\alpha}$

$P(T > t_{\nu,\alpha}) = \alpha$

	α													
ν	0.40	0.30	0.20	0.15	0.10	0.05	0.025	0.02	0.015	0.01	0.0075	0.005	0.0025	0.0005
1	0.325	0.727	1.376	1.963	3.078	6.314	12.706	15.895	21.205	31.821	42.434	63.657	127.322	636.590
2	0.289	0.617	1.061	1.386	1.886	2.920	4.303	4.849	5.643	6.965	8.073	9.925	14.089	31.598
3	0.277	0.584	0.978	1.250	1.638	2.353	3.182	3.482	3.896	4.541	5.047	5.841	7.453	12.924
4	0.271	0.569	0.941	1.190	1.533	2.132	2.776	2.999	3.298	3.747	4.088	4.604	5.598	8.610
5	0.267	0.559	0.920	1.156	1.476	2.015	2.571	2.757	3.003	3.365	3.634	4.032	4.773	6.869
6	0.265	0.553	0.906	1.134	1.440	1.943	2.447	2.612	2.829	3.143	3.372	3.707	4.317	5.959
7	0.263	0.549	0.896	1.119	1.415	1.895	2.365	2.517	2.715	2.998	3.203	3.499	4.029	5.408
8	0.262	0.546	0.889	1.108	1.397	1.860	2.306	2.449	2.634	2.896	3.085	3.355	3.833	5.041
9	0.261	0.543	0.883	1.100	1.383	1.833	2.262	2.398	2.574	2.821	2.998	3.250	3.690	4.781
10	0.260	0.542	0.879	1.093	1.372	1.812	2.228	2.359	2.527	2.764	2.932	3.169	3.581	4.587
11	0.260	0.540	0.876	1.088	1.363	1.796	2.201	2.328	2.491	2.718	2.879	3.106	3.497	4.437
12	0.259	0.539	0.873	1.083	1.356	1.782	2.179	2.303	2.461	2.681	2.836	3.055	3.428	4.318
13	0.259	0.538	0.870	1.079	1.350	1.771	2.160	2.282	2.436	2.650	2.801	3.012	3.372	4.221
14	0.258	0.537	0.868	1.076	1.345	1.761	2.145	2.264	2.415	2.624	2.771	2.977	3.326	4.140

Or, computers. (maybe math, maybe simulation, maybe both)

So, where did *this* p-value come from?

```
(tt <- t.test(instant, not_instant))
```

```
##
##  Welch Two Sample t-test
##
## data:  instant and not_instant
## t = 3.6482, df = 5039.8, p-value = 0.0002667
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##   4.475555 14.872518
## sample estimates:
## mean of x mean of y
##  124.6409  114.9668
```


So, where did *this* p-value come from?

```
(tt <- t.test(instant, not_instant))
```

```
##
##  Welch Two Sample t-test
##
## data:  instant and not_instant
## t = 3.6482, df = 5039.8, p-value = 0.0002667
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##   4.475555 14.872518
## sample estimates:
## mean of x mean of y
##  124.6409  114.9668
```


The t distribution! (see separate slides)

// reveal.js plugins

/