Non-parametric tests & Discrete data

PSM 2

Bennett Kleinberg

5 Feb 2019

UCL

Probability, Statistics & Modeling II

Lecture 5

Non-paramtric tests & discrete data

What question do you have?

Today

- What to do if the parametric assumptions violated
- Non-parametric equivalents
- Discrete data (chisquare, loglinear model)

Non-parametric tests

~~Non~~ Parametric tests

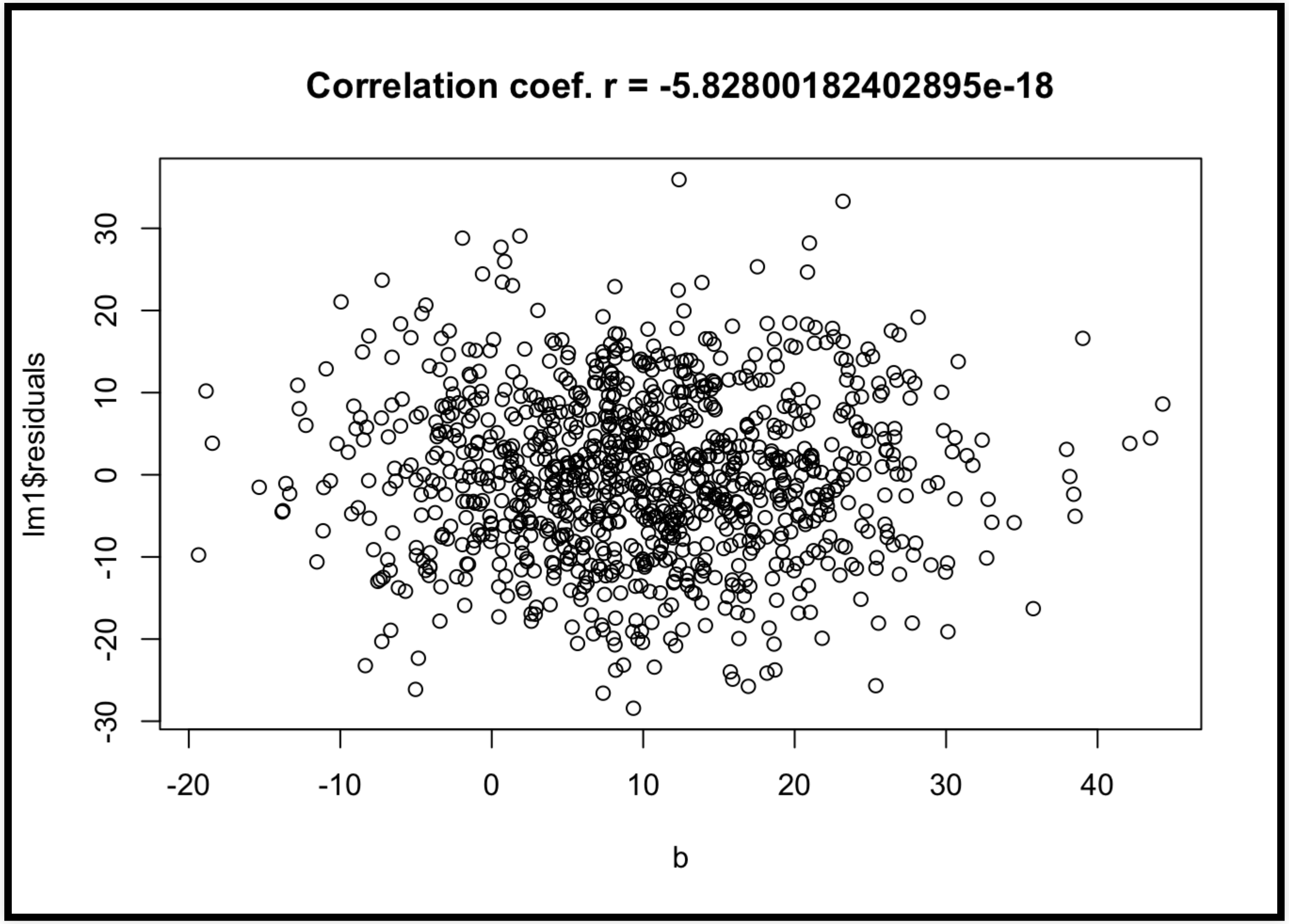
Parametric assumptions

- Independence of errors
- Homogeneity of variance
- Normality

Independence of errors

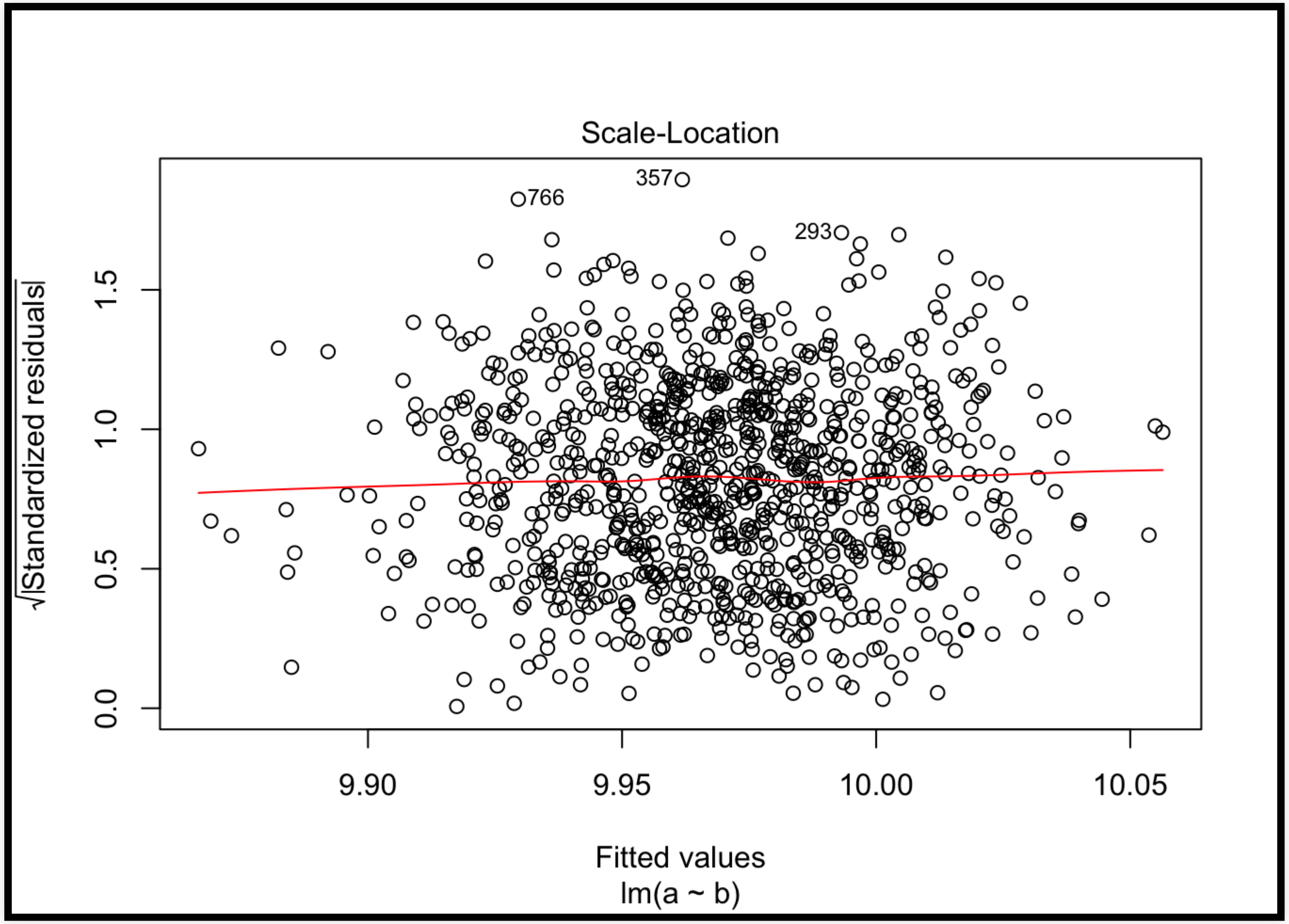
Independence of errors

- errors (estimated through residuals) should be ‘randomly’ distributed around 0
- … for all observations
- rule to investigate this: correlation
  - between residuals and predictor variable(s)


Correlation coef. r = -5.82800182402895e-18
lm1$residuals
30
20
10
0
-10
-20
-30
-20
-10
0
10
20
30
40
b

Homegeneity of variance

Also called: homoscedasticity

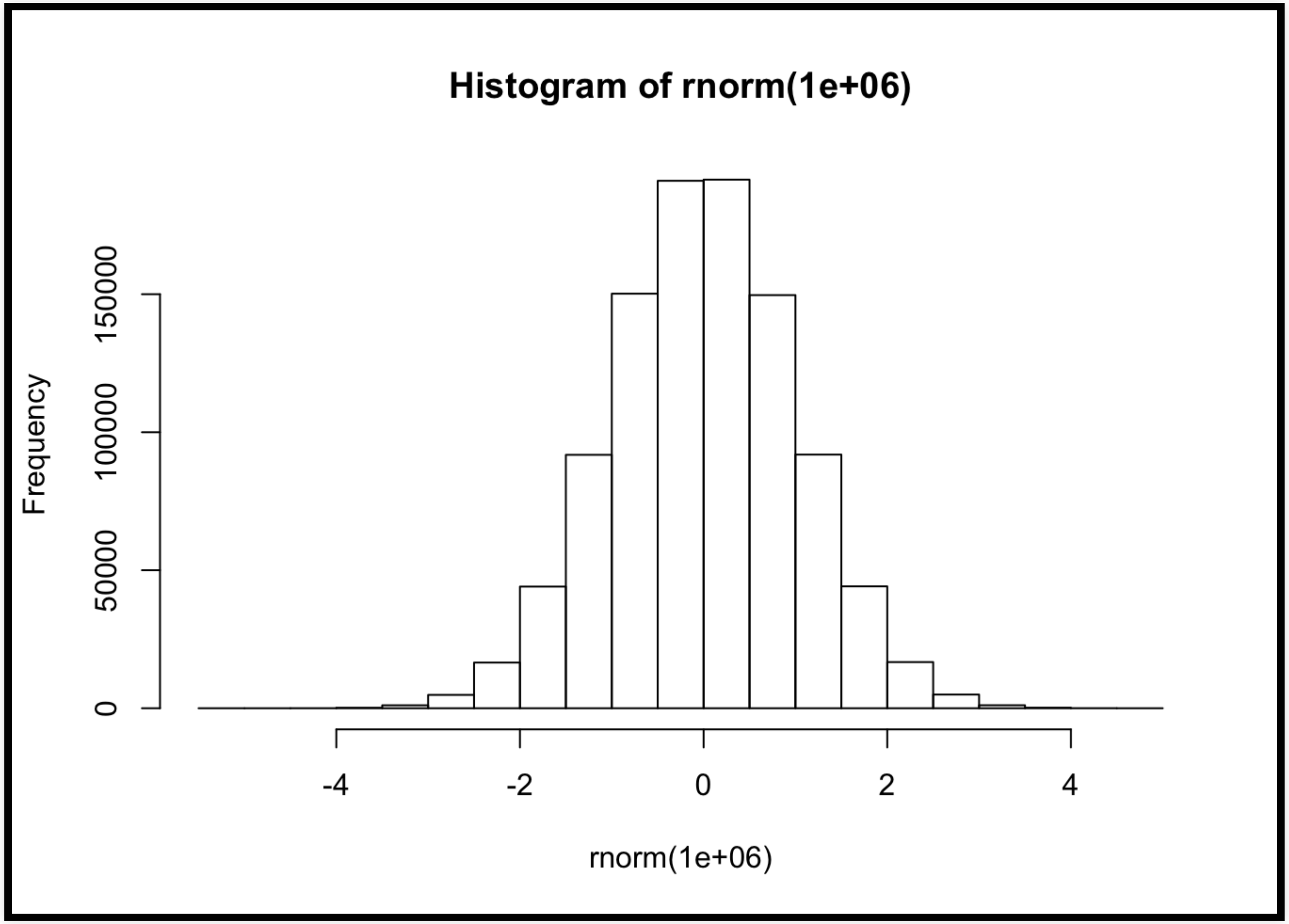


Homegeneity of variance

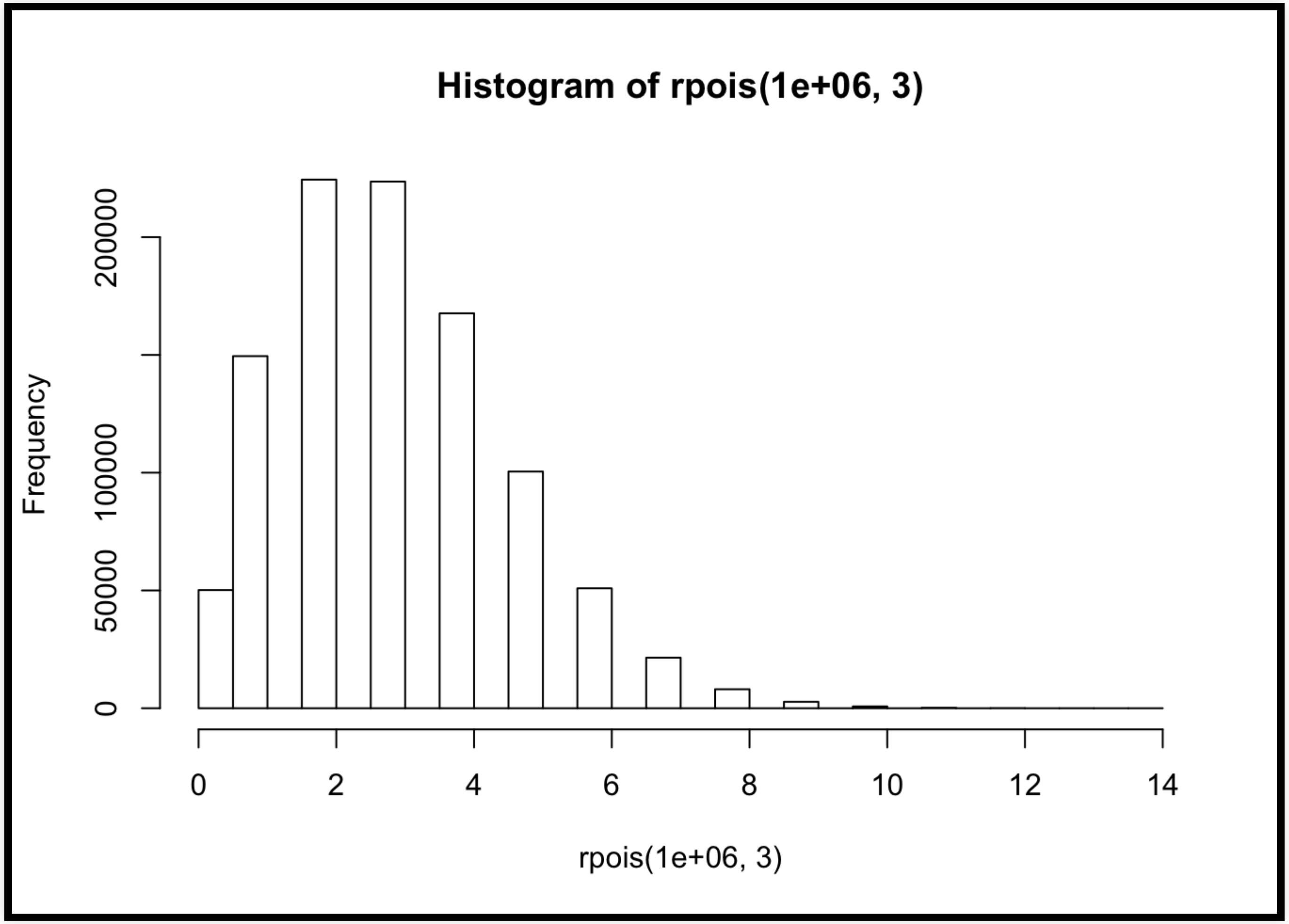
Can be tested with:

- Levene's Test `car:leveneTest(...)`
- Breush Pagan Test `lmtest::bptest(...)`
- both have H0 = data is homoscedastic

Normality


Histogram of rnorm(1e+06)
Frequency
0
50000
100000
150000
-4
-2
0
2
4
rnorm(1e+06)

Normality


Histogram of rpois(1e+06, 3)
Frequency
0
50000
100000
200000
0
2
4
6
8
10
12
14
rpois(1e+06, 3)

Normality

Normally met when:

- sample size is considerably large (e.g. $n > 50$)

Can be tested with:

- Kolmogorov-Smirnov Test
- Shapiro's Test `shapiro.test()`
- both have H_0 = data is normally distributed

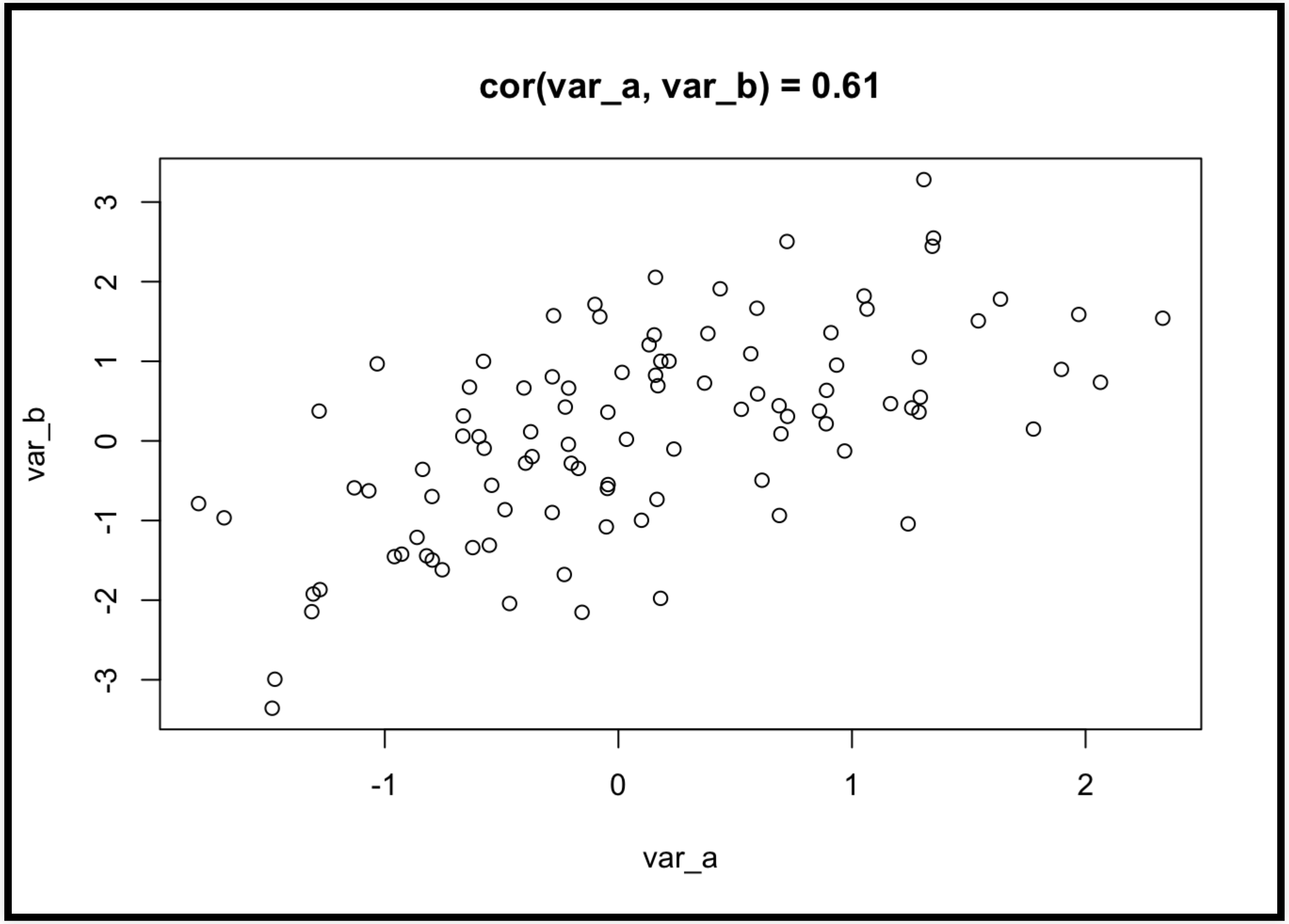
What to do if these are violated?

Violation of parametric assumptions

Assumption	Test	Potential fix
Independence of errors	Resiudal-predictor plot, correlation	Autocorrelation-sensitive methods
Homoscedasticity	Levene's Test, plot	Box-Cox transformation
Normality	K-S test, Shapiro's Test	Transforming data

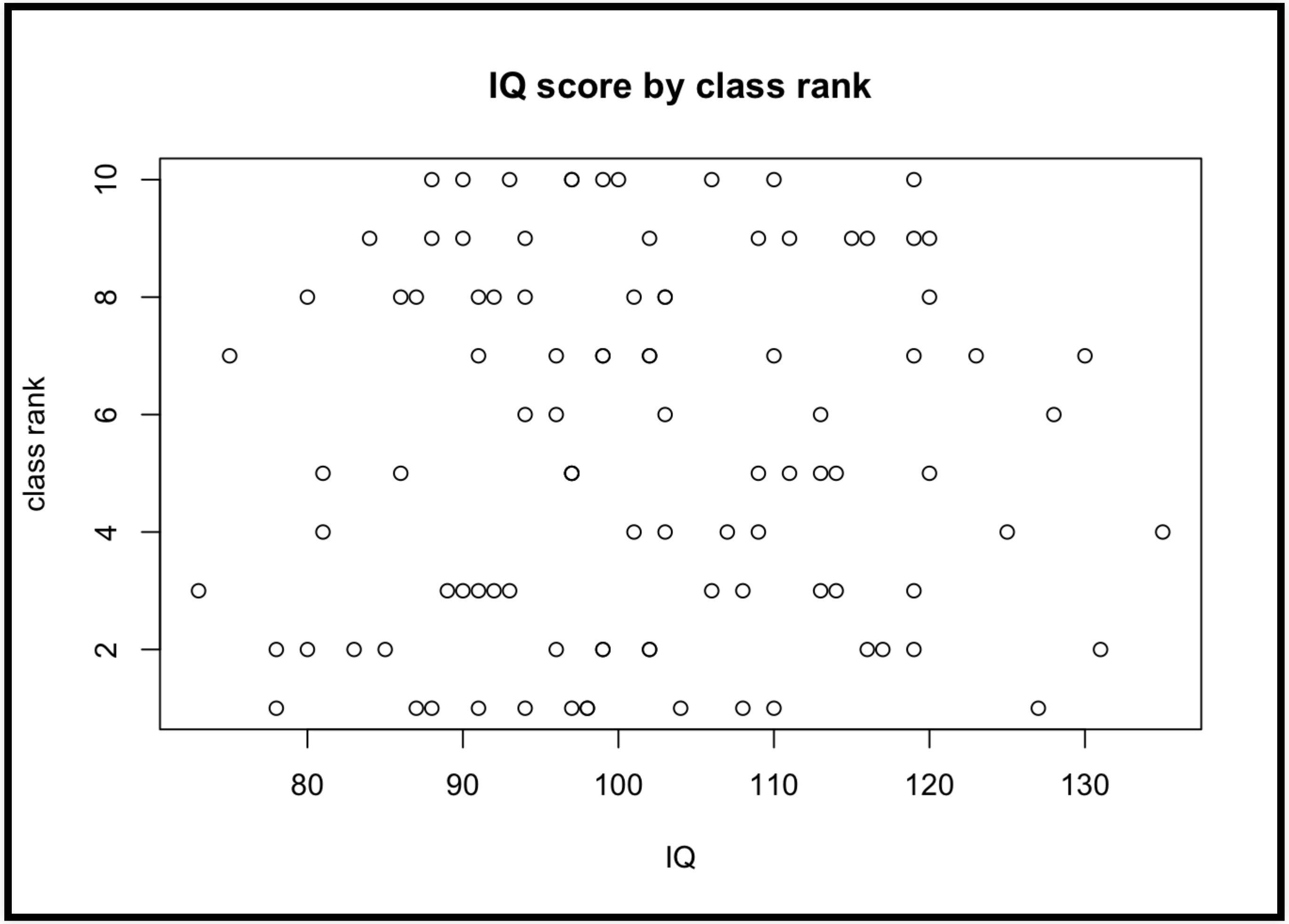
Non-paramtric tests: Correlation

Parametric case:



Non-paramtric tests: Correlation

Non-parametric case



Problem: class rank not parametric (e.g. not normally distributed)

Non-paramtric tests: Correlation

Spearman’s correlation test

Idea:

- rank the data
- run correlation on ranked data

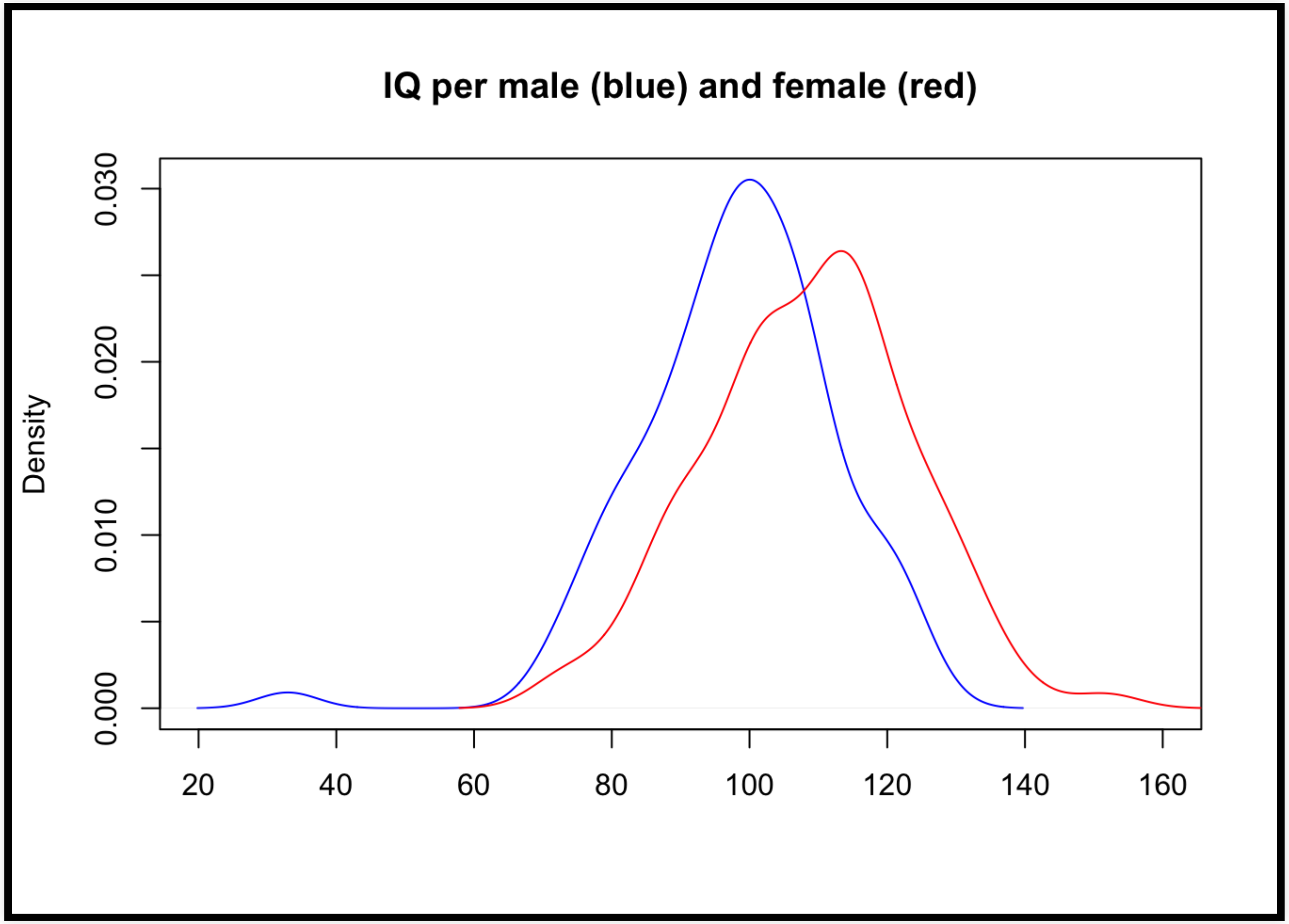
```
cor.test(iq, class_rank, method = 'spearman')
```

```
## Warning in cor.test.default(iq, class_rank, method = "spearman"): Cann
## compute exact p-value with ties
```

```
##
##  Spearman's rank correlation rho
##
## data:  iq and class_rank
## S = 158810, p-value = 0.6421
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##        rho
## 0.04704638
```

Non-paramtric tests: T-tests

Parametric case: Independent samples t-test



Non-paramtric tests: T-tests

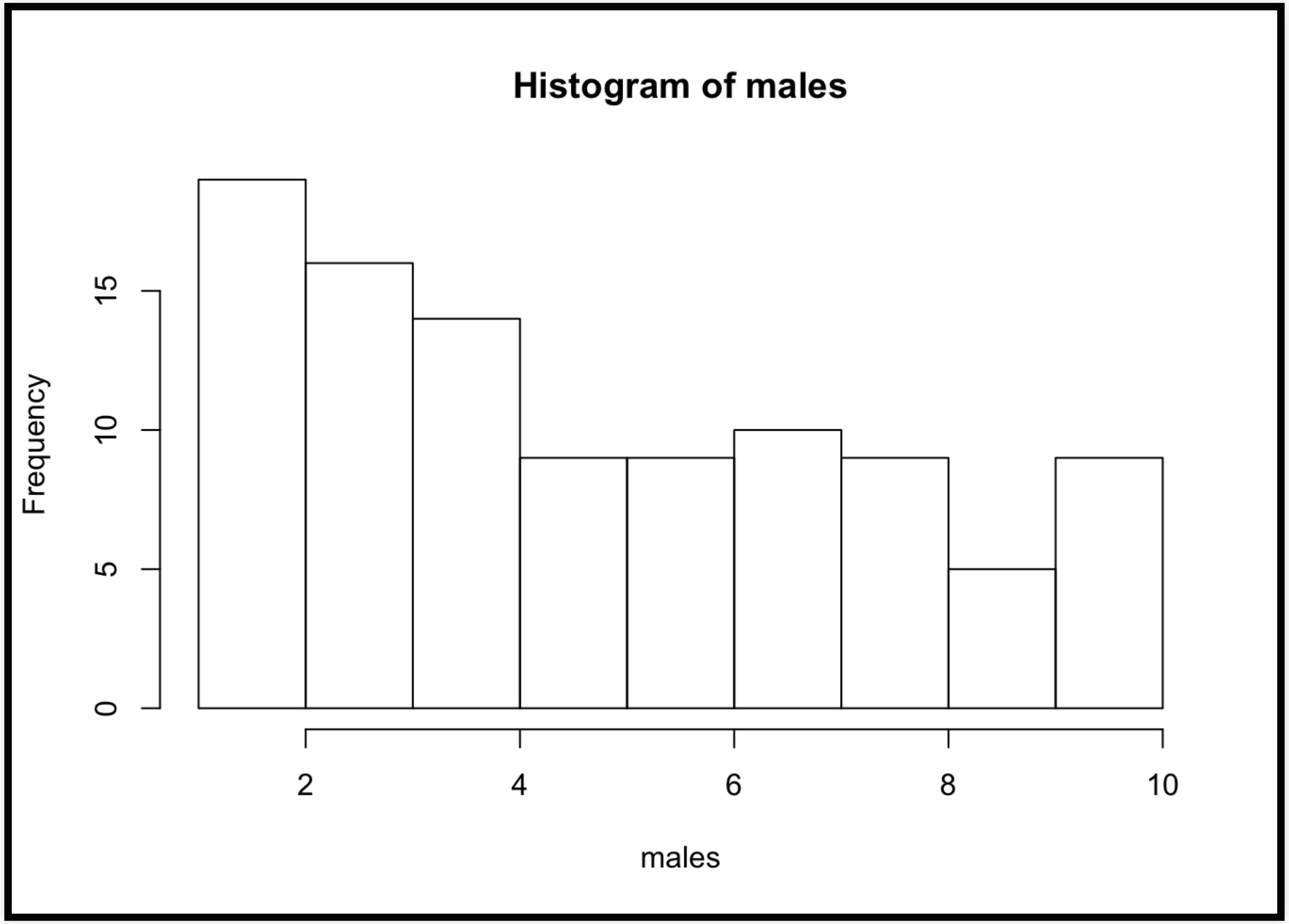
Parametric case: Independent samples t-test

```r
t.test(males, females)
```

```
##
##  Welch Two Sample t-test
##
## data:  males and females
## t = -5.0547, df = 197.52, p-value = 9.796e-07
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -14.496541  -6.359724
## sample estimates:
## mean of x mean of y
##  98.14252 108.57065
```

Non-paramtric tests: T-tests

Non-parametric case: Independent samples t-test



Problem: variable “rank” not parametric (e.g. not normally distributed)

Non-paramtric tests: T-tests

Wilcoxon Rank Sum Test

Idea:

- rank the data
- sum the ranks
- use the smallest rank sum as test statistic
- assess the significance of the test-statistic

males	rank.males.	females	rank.females.
3	27.5	8	75.5
3	27.5	7	65.0
8	82.0	2	15.5
6	63.0	3	25.5
7	72.5	8	75.5
10	96.0	1	5.5

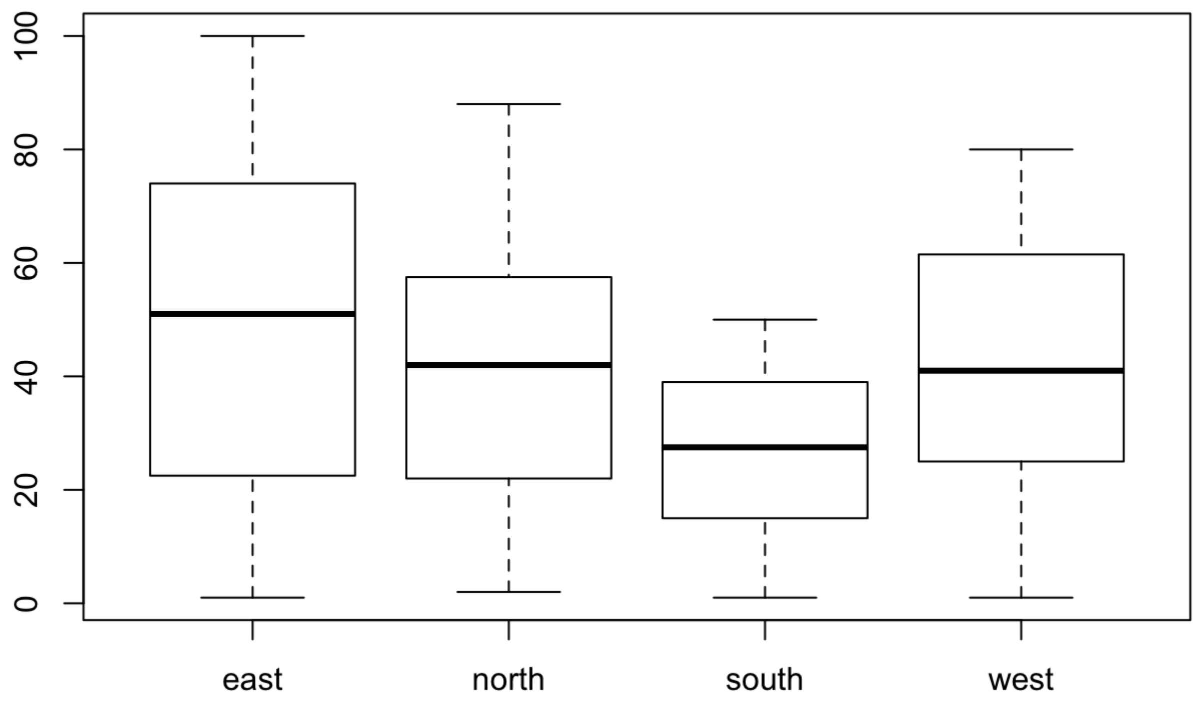
Wilcoxon rank sum test

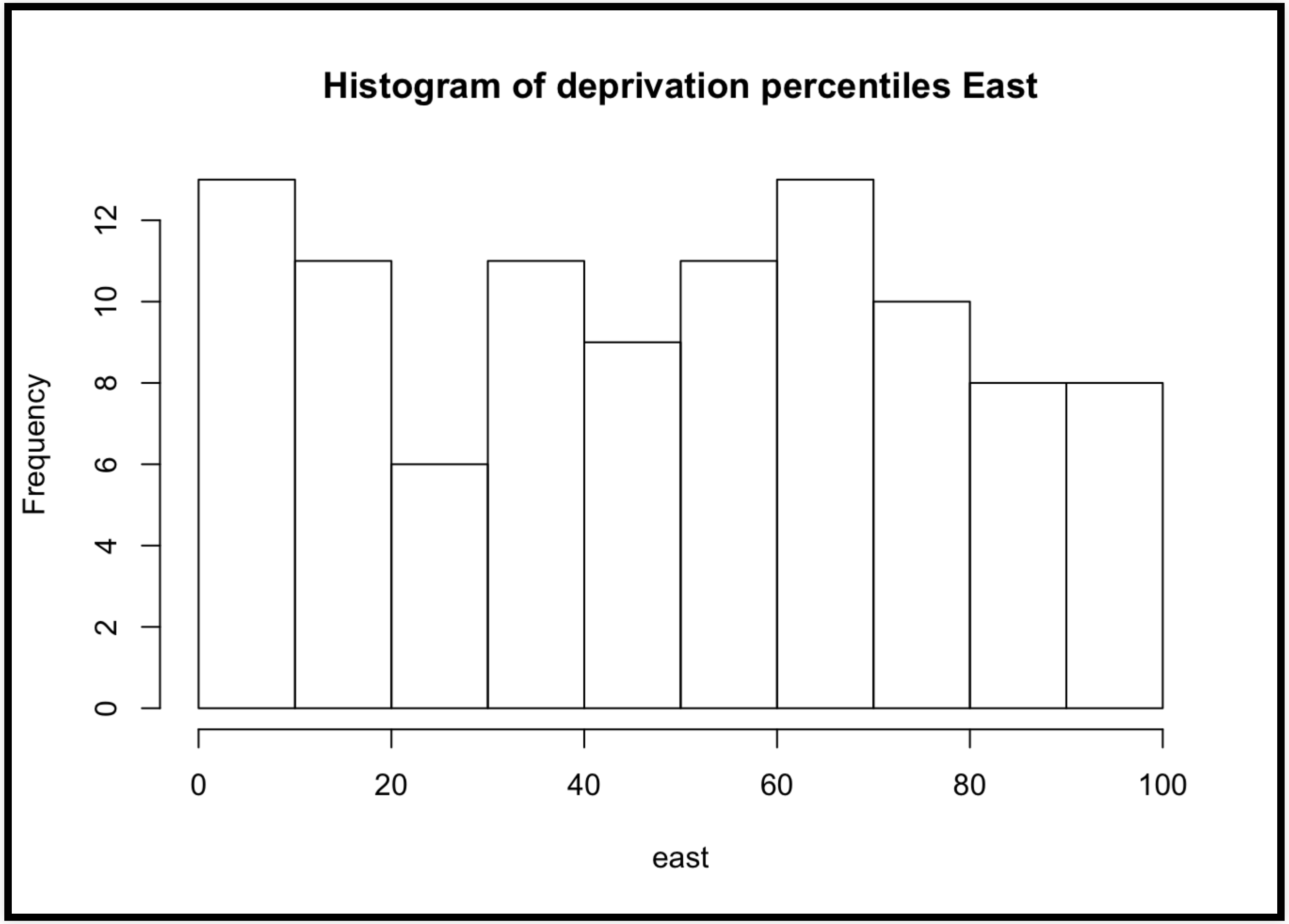
```r
wilcox.test(males, females)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  males and females
## W = 4564.5, p-value = 0.2853
## alternative hypothesis: true location shift is not equal to 0
```

For the non-parametric “dependent t-test”, you’d have to use the ‘paired’ argument (same is in the `t.test` function).

Non-paramtric tests: ANOVA




Histogram of deprivation percentiles East
Frequency
0
2
4
6
8
10
12
0
20
40
60
80
100
east

Non-paramtric tests: ANOVA

The Kruskal-Wallis Test

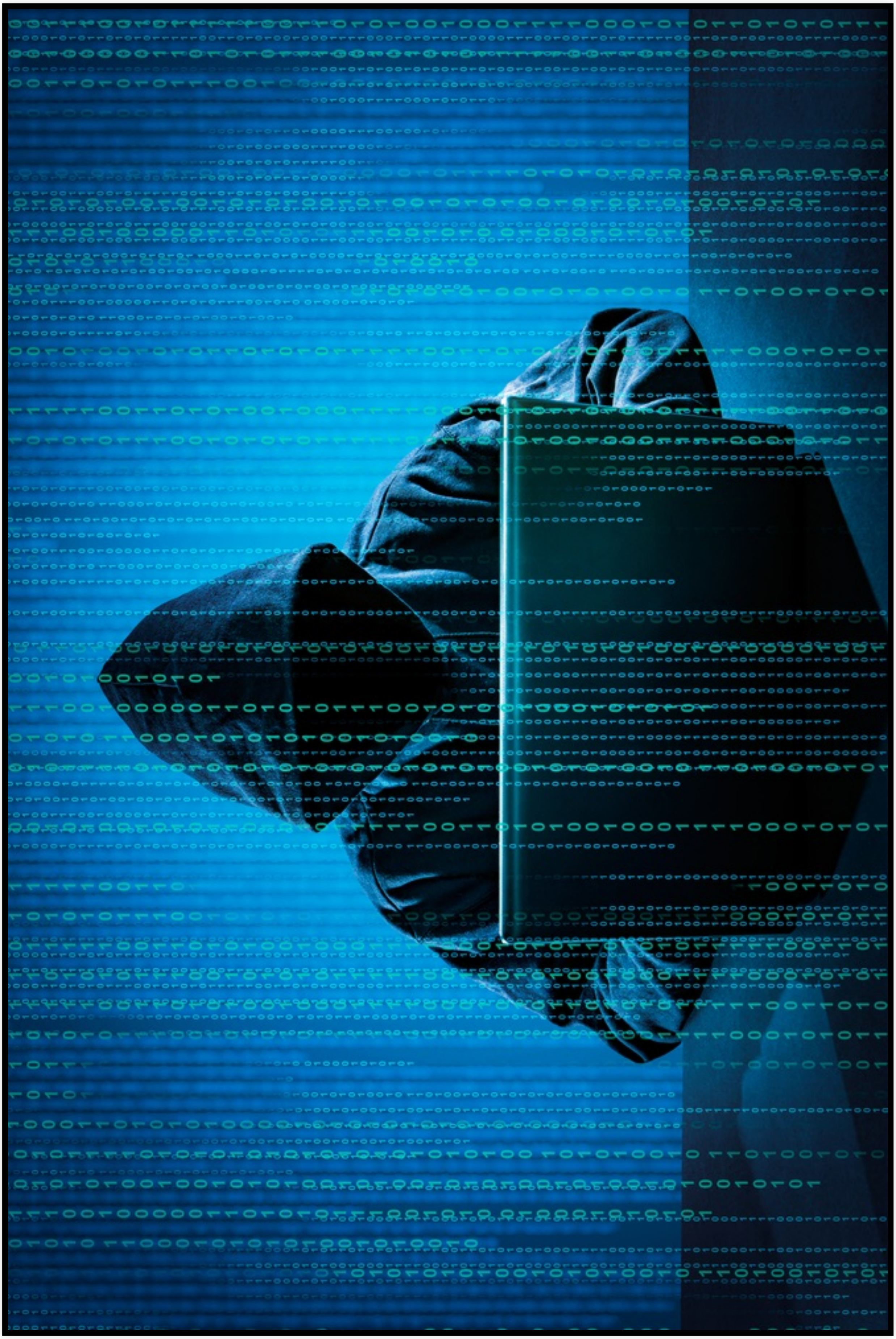
- rank the data
- sum the ranks per group
- apply Kruskal-Wallis formula to calculate the test-statistic H
- test significance of H

Non-paramtric tests: ANOVA

```r
kruskal.test(deprivation ~ area, data = deprivation)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  deprivation by area
## Kruskal-Wallis chi-squared = 40.92, df = 3, p-value = 6.8e-09
```

Discrete data



Problem

	No anti-virus software	Anti-virus software
Hacked	300	250
Not hacked	200	250

- uni-directionality?
- bi-directionality?
- third variable?

Association test

2 by 2 tables

Chi-square test

	No anti-virus software	Anti-virus software	Sum
Hacked	300	250	550
Not hacked	200	250	450
Sum	500	500	1000

Discrete data

Idea of the Chi-square test:

- Observed values O
- Expected values (if there were no association) E
- rows: i
- columns: j

$$E_{i,j} = (total_i * total_j)/total$$

Expected values

$$E_{i,j} = (total_i * total_j)/total$$

	No anti-virus software	Anti-virus software	Sum
Hacked	?	?	550
Not hacked	?	?	450
Sum	500	500	1000

Example: cell [hacked, no anti-virus software] –> cell [1,1]

$$E_{i,j} = (total_i * total_j)/total$$

$ = (550 * 500)/1000$

$ = 275$

Expected values

	No anti-virus software	**Anti-virus software**	**Sum**
Hacked	275	275	550
Not hacked	225	225	450
Sum	500	500	1000

Calculating the Chi-square value

$$\chi^2 = \sum \frac{(O-E^2)}{E}$$

For cell[2,1]:

$$cell[2,1] = \frac{(200-225)^2}{225} = \frac{-25^2}{225} = \frac{625}{225} = 2.78$$

Calculating the Chi-square value

- Repeat procedure for all cells
- Sum the values

$$\chi^2 = 9.701$$

	No anti-virus software	**Anti-virus software**
Hacked	300 (275)	250 (275)
Not hacked	200 (225)	250 (225)

- Null-hypothesis: there is no association between the two factors
- Alt. hypothesis: there is a significant association

The Chi-square test for 2*2 tables

```
##
##  Pearson's Chi-squared test
##
## data:  data1
## X-squared = 10.101, df = 1, p-value = 0.001482
```

Now what?

There is a significant association between being hacked (hacked vs not hacked) and the use of anti-virus software (no anti-virus software vs anit-virus software), $X^2(1) = 10.10$, $p = .001$.

But where does this association stem from? What drives it?

	No anti-virus software	**Anti-virus software**
Hacked	300	250
Not hacked	200	250

Standardized residuals

Interpret as:

- the number of standard deviations away from zero
- we know: +/- 2.58 SD = 0.01 and 0.99 percentile

```
knitr::kable(chisq.test(data1, correct = F)$stdres)
```

	No anti-virus software	Anti-virus software
Hacked	3.178209	-3.178209
Not hacked	-3.178209	3.178209

Interpretation

	No anti-virus software	**Anti-virus software**
Hacked	3.18 (O > E)	-3.18 (O < E)
Not hacked	-3.18 (O < E)	3.18 (O > E)

From 2-by-2 to r-by-c

	Non AV	**standard AV**	**premium AV**
No access	200	250	150
Files stolen	400	300	200
Ransomware	350	150	150

Extension of the 2 by 2 approach

```r
knitr::kable(addmargins(data2, c(1,2)))
```

	Non AV	standard AV	premium AV	Sum
No access	200	250	150	600
Files stolen	400	300	200	900
Ransomware	350	150	150	650
Sum	950	700	500	2150

Same steps:

- for each cell $\frac{(O-E^2)}{E}$
- sum to obtain χ^2
- assess *omnibus* significance
- follow-up interpretation

(O - E)^2/E

	Non AV	**standard AV**	**premium AV**
No access	15.99	15.29	0.78
Files stolen	0.01	0.17	0.41
Ransomware	13.73	17.95	0.01

Extension of the 2 by 2 approach

```
##
##  Pearson's Chi-squared test
##
## data:  data2
## X-squared = 64.344, df = 4, p-value = 3.537e-13
```

Follow-up tests for the interpretation.

–> What drives the sign. association?

Intepreting the r-by-c extension

```
knitr::kable(round(chisq.test(data2, correct = F)$stdres, 2))
```

	Non AV	standard AV	premium AV
No access	-6.30	5.61	1.19
Files stolen	0.20	0.65	-0.96
Ransomware	5.94	-6.18	-0.13

Remember: interpretation like z-scores

- +/- 1.96 –> sign. at $p < .05$
- +/- 2.58 –> sign. at $p < .05$

	Non AV	**standard AV**	**premium AV**
No access	(O < E)	(O > E)	(O == E)
Files stolen	(O == E)	(O == E)	(O == E)
Ransomware	(O > E)	(O < E)	(O == E)

Interpretation

The significant association between … was driven by four significant deviations between the observed and expected values.

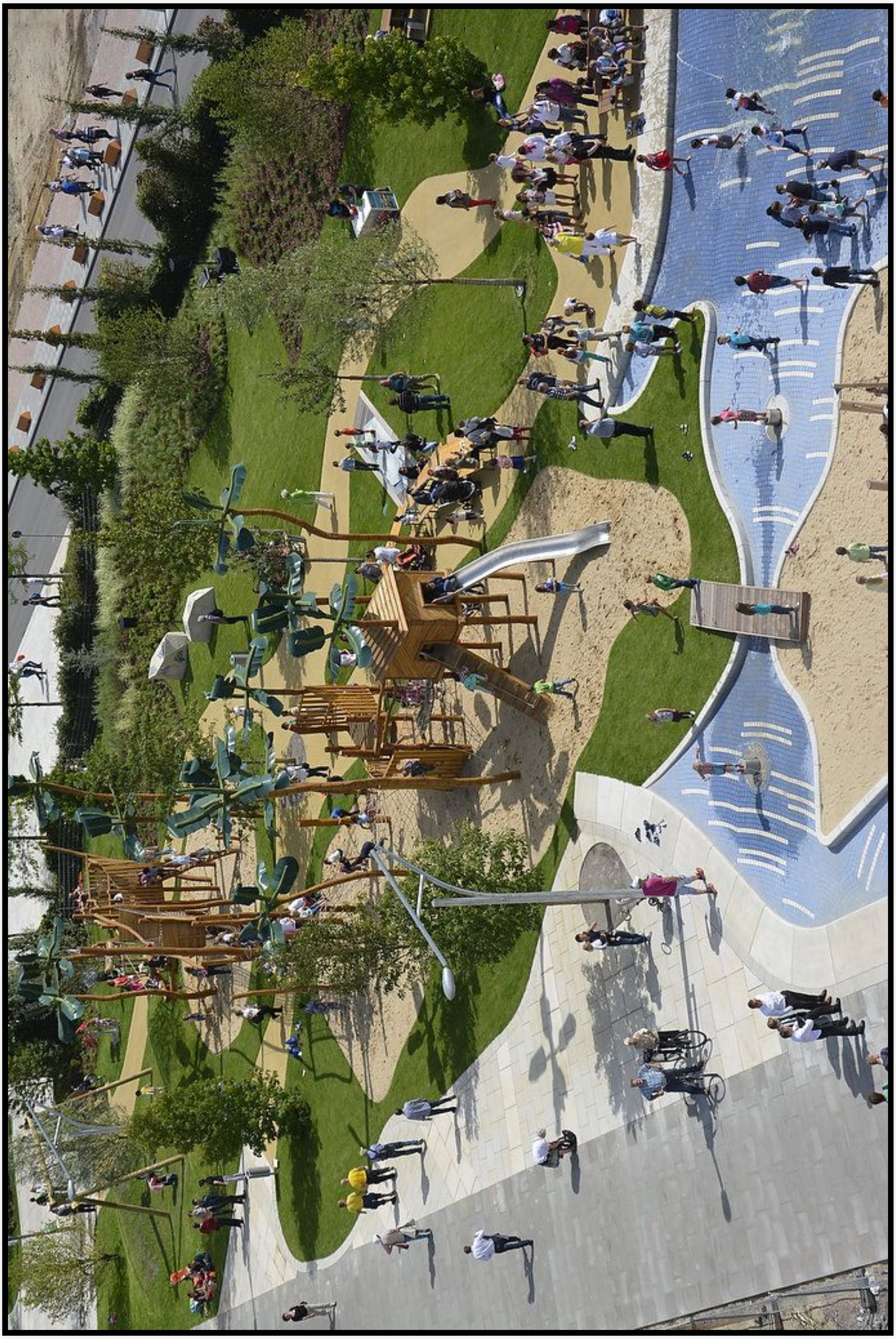
- Hackers failed to get access to significantly fewer computers when there was no anti-virus than expected ($z = -6.30$).
- Hackers inserted ransomware on computers without anti-virus more often than expected ($z = 5.94$).
- Hackers got access to computers with standard anti-virus software more often than expected ($z = 5.61$).
- Hackers inserted ransomware on computers with standard anti-virus less often than expected ($z = -6.18$).

Discrete data

Extension to multi-level models

X by Y by Z cases

- 2-dimensional arrays
  - 2-by-2 tables
  - r-by-c tables
- multidimensional arrays
  - **X**-by-**Y**-by-**Z**



X by Y by Z arrays

```
##                                vandalised yes  no
## area   natural surveillance
## urban  yes                               911  44
##        no                                538 456
## suburb yes                                 3   2
##        no                                 43 279
```

- 3 factors
  - vandalised: yes vs no
  - natural surveillance: yes vs no
  - area: urban vs rural

Simple extension of the r*c calculation?

Multilevel discrete data

```
## , , area = urban
##
##            natural surveillance
## vandalised       yes        no
##        yes 0.6287095 0.3712905
##        no  0.0880000 0.9120000
##
## , , area = suburb
##
##            natural surveillance
## vandalised         yes        no
##        yes 0.065217391 0.9347826
##        no  0.007117438 0.9928826
```

Idea of multilevel discrete modelling

If the data were independent…

then the expected count = joint prob. * n, where

joint prob. = product of the marginal probabilities

$$\mu_{i,j} = n * marginal_i * marginal_j$$

Probability data

	No anti-virus software	**Anti-virus software**	**Sum**
Hacked	0.3	0.25	0.55
Not hacked	0.2	0.25	0.45
Sum	0.5	0.50	1.00

Probability data

	No anti-virus software	**Anti-virus software**	**Sum**
Hacked	0.3	0.25	0.55
Not hacked	0.2	0.25	0.45
Sum	0.5	0.50	1.00

$$cell[1, 2] = 0.50 * 0.55 * 1000 = 275$$

$$cell[2, 2] = 0.50 * 0.45 * 1000 = 225$$

Towards a linear model

Log transformation

$$\mu_{i,j} = n * marginal_i * marginal_j$$

==

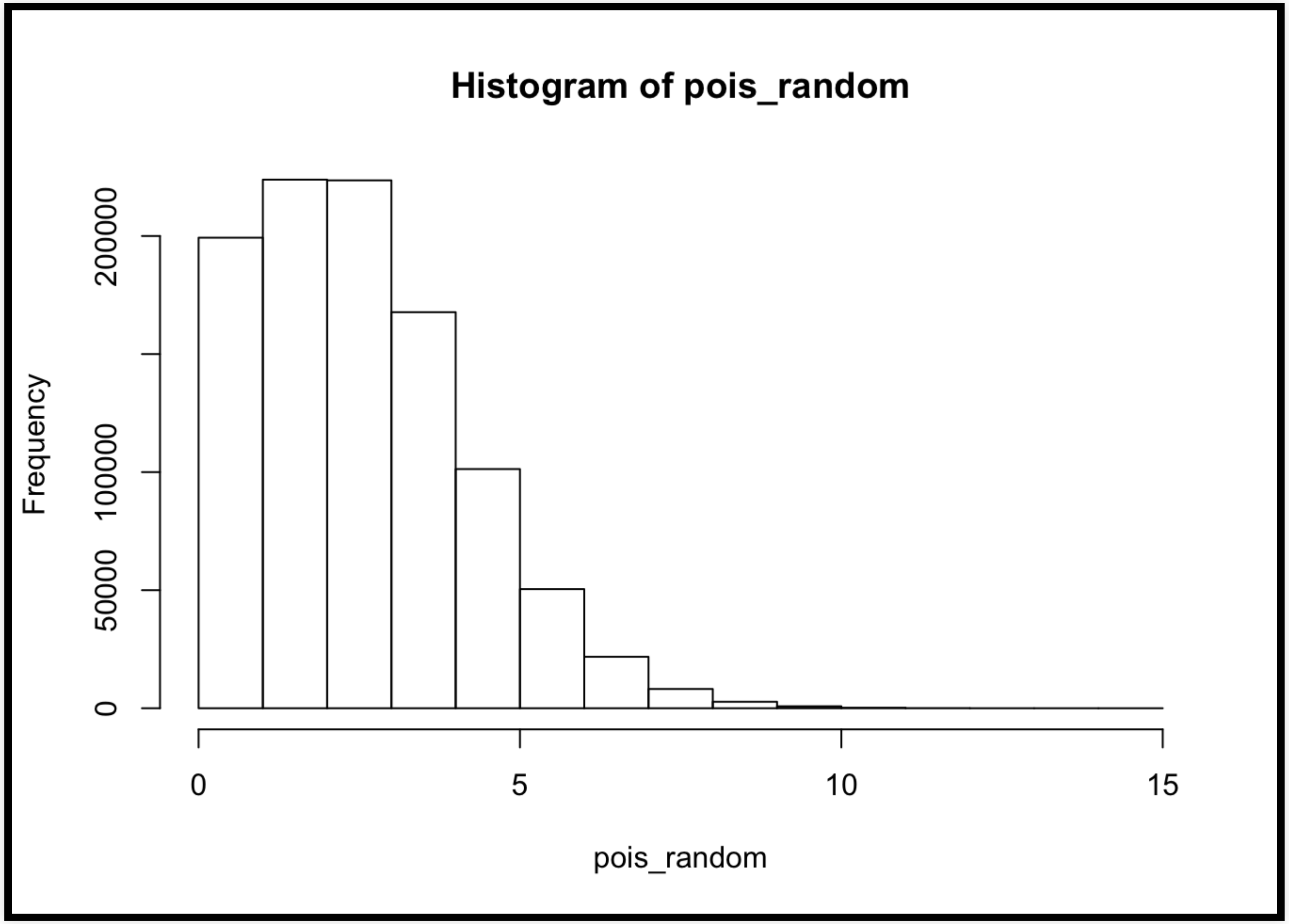
$$log(\mu_{i,j}) = log(n) + log(marginal_i) + log(marginal_j)$$

Hence: “loglinear” model

The Log-Linear Model

- GLM with link function for count data
- count data aptly modelled as a Poisson distrubuted variable

The poisson distribution



The loglinear model

```
##   vandalised natural.surveillance   area Freq
## 1        yes                  yes  urban  911
## 2         no                  yes  urban   44
## 3        yes                   no  urban  538
## 4         no                   no  urban  456
## 5        yes                  yes suburb    3
## 6         no                  yes suburb    2
## 7        yes                   no suburb   43
## 8         no                   no suburb  279
```

```
indep_model = glm(formula = Freq ~ vandalised + natural.surveillance + ar
               , data = data3_
               , family = poisson)
```

The loglinear model

```
##
## Call:
## glm(formula = Freq ~ vandalised + natural.surveillance + area,
##     family = poisson, data = data3_)
##
## Deviance Residuals:
##       1        2        3        4        5        6        7        8
##  14.522  -17.683   -7.817    3.426  -12.440   -8.832   -8.436   19.63
##
## Coefficients:
##                        Estimate Std. Error z value Pr(>|z|)
## (Intercept)             6.29154    0.03667 171.558  < 2e-16 ***
## vandalisedno           -0.64931    0.04415 -14.707  < 2e-16 ***
## natural.surveillanceno  0.31542    0.04244   7.431 1.08e-13 ***
## areasuburb             -1.78511    0.05976 -29.872  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

Interpretation

1. Look at the Residual deviance
    - Higher deviance = poorer model fit
    - We can test the H0 of model adequacy

```
pchisq(1286, 4, lower.tail = F)
```

```
## [1] 3.610223e-277
```

Reject H0 that the model is a good representation.

Interpretation

2. Look at the fitted values

```
knitr::kable(cbind(indep_model$data, round(fitted(indep_model), 2)))
```

vandalised	natural.surveillance	area	Freq	round(fitted(indep_model), 2)
yes	yes	urban	911	539.98
no	yes	urban	44	282.09
yes	no	urban	538	740.23
no	no	urban	456	386.70
yes	yes	suburb	3	90.60
no	yes	suburb	2	47.33
yes	no	suburb	43	124.19
no	no	suburb	279	64.88

Interpretation

3. Look at the anit-logged coefficients

Coefficient for “vandalised=no”

```
exp(-0.64931)
```

```
## [1] 0.5224061
```

Odds of a playground being vandalised are 0.52:1, regardless of whether there was natural surveillance and regardless of the area.

The full model

Also called: the saturated model

```
full_model = glm(formula = Freq ~ vandalised * natural.surveillance * are
                , data = data3_
                , family = poisson)
```

What do you expect?

The full model

```
knitr::kable(cbind(full_model$data, round(fitted(full_model), 2)))
```

vandalised	natural.surveillance	area	Freq	round(fitted(full_model), 2)
yes	yes	urban	911	911
no	yes	urban	44	44
yes	no	urban	538	538
no	no	urban	456	456
yes	yes	suburb	3	3
no	yes	suburb	2	2
yes	no	suburb	43	43
no	no	suburb	279	279

Loglinear model strategy

- Find a model less complex than the full model
- … where you cannot reject the H0 of model adequacy

Model selection

```
step(full_model)
```

```
## Start:  AIC=65.04
## Freq ~ vandalised * natural.surveillance * area
##
##                                          Df Deviance    AIC
## - vandalised:natural.surveillance:area  1  0.37399 63.417
## <none>                                      0.00000 65.043
##
## Step:  AIC=63.42
## Freq ~ vandalised + natural.surveillance + area + vandalised:natural.s
##     vandalised:area + natural.surveillance:area
##
##                                  Df Deviance    AIC
## <none>                                 0.37  63.42
## - natural.surveillance:area       1    92.02 153.06
## - vandalised:area                 1   187.75 248.80
## - vandalised:natural.surveillance  1   497.37 558.41
```

```
##
## Call:  glm(formula = Freq ~ vandalised + natural.surveillance + area +
##     vandalised:natural.surveillance + vandalised:area + natural.surve
##     family = poisson, data = data3_)
##
## Coefficients:
##                           (Intercept)                          vandalised
```

```
##                     6.8139                        -3.0
##      natural.surveillanceno                      areasubu
##                    -0.5249                        -5.52
## vandalisedno:natural.surveillanceno    vandalisedno:areasubu
##                     2.8479                        2.05
##   natural.surveillanceno:areasuburb
##                     2.9860
##
## Degrees of Freedom: 7 Total (i.e. Null);  1 Residual
## Null Deviance:	    2851
```

"Best" model

```
summary(best_model)
```

```
##
## Call:
## glm(formula = Freq ~ vandalised + natural.surveillance + area +
##     vandalised:natural.surveillance + vandalised:area + natural.surve
##     family = poisson, data = data3_)
##
## Deviance Residuals:
##        1         2         3         4         5         6         7
##  0.02044  -0.09256  -0.02658   0.02890  -0.33428   0.49134   0.09452
##        8
## -0.03690
##
## Coefficients:
##                                     Estimate Std. Error z value Pr(>|
## (Intercept)                          6.81387    0.03313 205.699  < 2e
## vandalisedno                        -3.01575    0.15162 -19.891  < 2e
## natural.surveillanceno              -0.52486    0.05428  -9.669  < 2e
## areasuburb                          -5.52827    0.45221 -12.225  < 2e
```

“Best” model

Can we reject the H0 of model adequacy?

```
pchisq(0.37, 1, lower.tail = F)
```

```
## [1] 0.5430043
```

No!

Fitted values of the “best” model

vandalised	natural.surveillance	area	Freq	round(fitted(best_model), 2)
yes	yes	urban	911	910.38
no	yes	urban	44	44.62
yes	no	urban	538	538.62
no	no	urban	456	455.38
yes	yes	suburb	3	3.62
no	yes	suburb	2	1.38
yes	no	suburb	43	42.38
no	no	suburb	279	279.62

Interpreting the coefficients

```
coefficients(best_model)
```

```
##                               (Intercept)                               vandalisedn
##                                 6.8138656                                -3.015754
##                      natural.surveillanceno                              areasubur
##                                -0.5248611                                -5.528267
## vandalisedno:natural.surveillanceno             vandalisedno:areasubur
##                                 2.8478892                                 2.054534
##   natural.surveillanceno:areasuburb
##                                 2.9860144
```

Interpreting the coefficients

```
vandalisedno:areasuburb
2.0545341
```

```
exp(2.055)
```

```
## [1] 7.806838
```

Exponentiated interaction ==> OR

Playgrounds that are in the suburb have estimated odds of not being vandalised that is 7.81 times the estimated odds for playgrounds that are in urban areas. This is independent of "natural surveillance".

- Log-linear models work in higher dimensions
- Allow you to model count data of 2+ dimensions

Follow the steps here (https://data.library.virginia.edu/an-introduction-to-loglinear-models/)

Recap

Outlook

Next week: Reading week

Week 6: Open Science (lecture + tutorial)

END