ALGSTAT: AN R PACKAGE FOR ALGEBRAIC STATISTICS

Luis David García–Puente

Department of Mathematics and Statistics
Sam Houston State University

Algebraic Statistics 2015
Department of Mathematics
University of Genova
June 8, 2015

Joint work with
David Kahle (Baylor University)
and
Ruriko Yoshida (University of Kentucky)

SOFTWARE IN ALGEBRAIC STATISTICS

In this tutorial we will introduce **algstat**, an R package for algebraic statistics.

- Software for **algebraic geometry**:
    - Bertini
    - CoCoA-5
    - LattE
    - Macaulay2
    - Risa/Asir
    - Sage
    - Singular
- Software for **algebraic statistics**:
    - 4ti2
    - GraphicalModels.m2 (Macaulay2 package)
    - Bigatti & Caboara's algebraic statistics (CoCoA-5 package)
    - **Algstat** (R package)
    - . . .

SOFTWARE IN ALGEBRAIC STATISTICS

In this tutorial we will introduce **algstat**, an R package for algebraic statistics.

- Software for **algebraic geometry**:
  - Bertini
  - CoCoA-5
  - LattE
  - Macaulay2
  - Risa/Asir
  - Sage
  - Singular
- Software for **algebraic statistics**:
  - 4ti2
  - GraphicalModels.m2 (Macaulay2 package)
  - Bigatti & Caboara's algebraic statistics (CoCoA-5 package)
  - **Algstat** (R package)
  - ...

SOFTWARE IN ALGEBRAIC STATISTICS

In this tutorial we will introduce **algstat**, an R package for algebraic statistics.

- Software for **algebraic geometry**:
  - Bertini
  - CoCoA-5
  - LattE
  - Macaulay2
  - Risa/Asir
  - Sage
  - Singular
- Software for **algebraic statistics**:
  - 4ti2
  - GraphicalModels.m2 (Macaulay2 package)
  - Bigatti & Caboara's algebraic statistics (CoCoA-5 package)
  - **Algstat** (R package)
  - ...

SOFTWARE IN ALGEBRAIC STATISTICS

In this tutorial we will introduce **algstat**, an R package for algebraic statistics.

- Software for **algebraic geometry**:
    - Bertini
    - CoCoA-5
    - LattE
    - Macaulay2
    - Risa/Asir
    - Sage
    - Singular
- Software for **algebraic statistics**:
    - 4ti2
    - GraphicalModels.m2 (Macaulay2 package)
    - Bigatti & Caboara's algebraic statistics (CoCoA-5 package)
    - **Algstat** (R package)
    - . . .

SOFTWARE IN ALGEBRAIC STATISTICS

In this tutorial we will introduce **algstat**, an R package for algebraic statistics.

- Software for **algebraic geometry**:
    - Bertini
    - CoCoA-5
    - LattE
    - Macaulay2
    - Risa/Asir
    - Sage
    - Singular
- Software for **algebraic statistics**:
    - 4ti2
    - GraphicalModels.m2 (Macaulay2 package)
    - Bigatti & Caboara's algebraic statistics (CoCoA-5 package)
    - **Algstat** (R package)
    - ...

THE PURPOSE OF THIS TUTORIAL

Introduce **algstat**, an R package for algebraic statistics. We will focus for the most part on **log-linear models for contingency tables, Markov bases**, and the **Metropolis algorithm**.

Robbiano ($\sim$ 2002): "We designed `CoCoA` to be **user-friendly**."

Our goal is to design a **friendly** software package for algebraic statistics in R .

Friendly, for us, means a software that statisticians and general data analysts can **willingly** use.

The input must be **data**.

THE PURPOSE OF THIS TUTORIAL

Introduce **algstat**, an R package for algebraic statistics. We will focus for the most part on **log-linear models for contingency tables, Markov bases**, and the **Metropolis algorithm**.

Robbiano ($\sim$ 2002): "We designed `CoCoA` to be **user-friendly**."

Our goal is to design a **friendly** software package for algebraic statistics in R .

Friendly, for us, means a software that statisticians and general data analysts can **willingly** use.

The input must be **data**.

THE PURPOSE OF THIS TUTORIAL

Introduce **algstat**, an R package for algebraic statistics. We will focus for the most part on **log-linear models for contingency tables, Markov bases**, and the **Metropolis algorithm**.

Robbiano (∼ 2002): "We designed `CoCoA` to be **user-friendly**."

Our goal is to design a **friendly** software package for algebraic statistics in R .

Friendly, for us, means a software that statisticians and general data analysts can **willingly** use.

The input must be **data**.

THE PURPOSE OF THIS TUTORIAL

Introduce **algstat**, an R package for algebraic statistics. We will focus for the most part on **log-linear models for contingency tables, Markov bases**, and the **Metropolis algorithm**.

Robbiano ($\sim$ 2002): "We designed `CoCoA` to be **user-friendly**."

Our goal is to design a **friendly** software package for algebraic statistics in R .

Friendly, for us, means a software that statisticians and general data analysts can **willingly** use.

The input must be **data**.

THE PURPOSE OF THIS TUTORIAL

Introduce **algstat**, an R package for algebraic statistics. We will focus for the most part on **log-linear models for contingency tables, Markov bases**, and the **Metropolis algorithm**.

Robbiano ($\sim$ 2002): "We designed `CoCoA` to be **user-friendly**."

Our goal is to design a **friendly** software package for algebraic statistics in R .

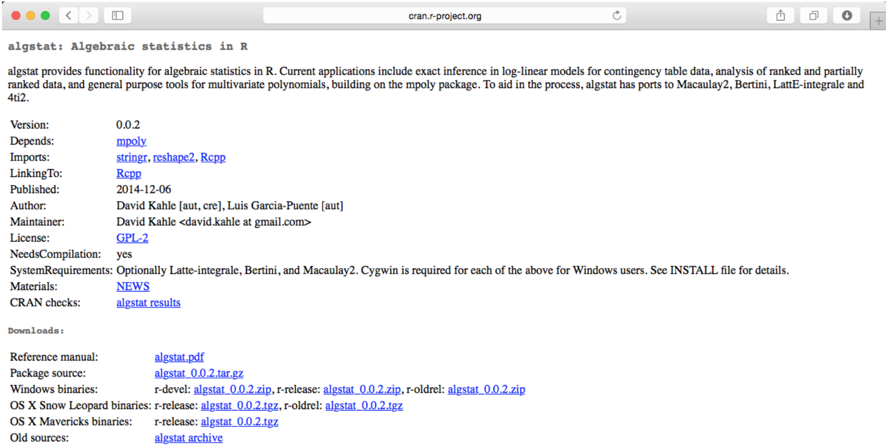
Friendly, for us, means a software that statisticians and general data analysts can **willingly** use.

The input must be **data**.

ALGSTAT V0.0.2 (CRAN VERSION)

algstat is an R package.

It is currently available through R's main package repository, **CRAN**.



algstat: Algebraic statistics in R

algstat provides functionality for algebraic statistics in R. Current applications include exact inference in log-linear models for contingency table data, analysis of ranked and partially ranked data, and general purpose tools for multivariate polynomials, building on the mpoly package. To aid in the process, algstat has ports to Macaulay2, Bertini, LattE-integrale and 4ti2.

Version:	0.0.2
Depends:	mpoly
Imports:	stringr, reshape2, Rcpp
LinkingTo:	Rcpp
Published:	2014-12-06
Author:	David Kahle [aut, cre], Luis Garcia-Puente [aut]
Maintainer:	David Kahle <david.kahle at gmail.com>
License:	GPL-2
NeedsCompilation:	yes
SystemRequirements:	Optionally Latte-integrale, Bertini, and Macaulay2. Cygwin is required for each of the above for Windows users. See INSTALL file for details.
Materials:	NEWS
CRAN checks:	algstat results

Downloads:

Reference manual:	algstat.pdf
Package source:	algstat_0.0.2.tar.gz
Windows binaries:	r-devel: algstat_0.0.2.zip, r-release: algstat_0.0.2.zip, r-oldrel: algstat_0.0.2.zip
OS X Snow Leopard binaries:	r-release: algstat_0.0.2.tgz, r-oldrel: algstat_0.0.2.tgz
OS X Mavericks binaries:	r-release: algstat_0.0.2.tgz
Old sources:	algstat archive

ALGSTAT V0.0.2 (CRAN VERSION)

algstat is an R package.

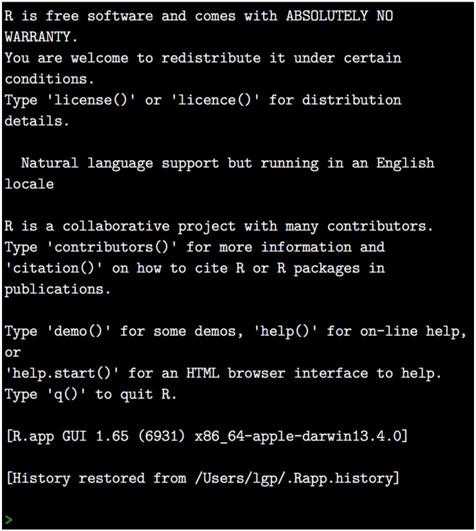
It is currently available through R's main package repository, **CRAN**.

`http://cran.r-project.org/web/packages/algstat/index.html`

The **latest version** is available on **github**.

`https://github.com/dkahle/algstat`

WHAT IS R ?



```
R is free software and comes with ABSOLUTELY NO
WARRANTY.
You are welcome to redistribute it under certain
conditions.
Type 'license()' or 'licence()' for distribution
details.

  Natural language support but running in an English
locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in
publications.

Type 'demo()' for some demos, 'help()' for on-line help,
or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

[R.app GUI 1.65 (6931) x86_64-apple-darwin13.4.0]

[History restored from /Users/lgp/.Rapp.history]

>
```

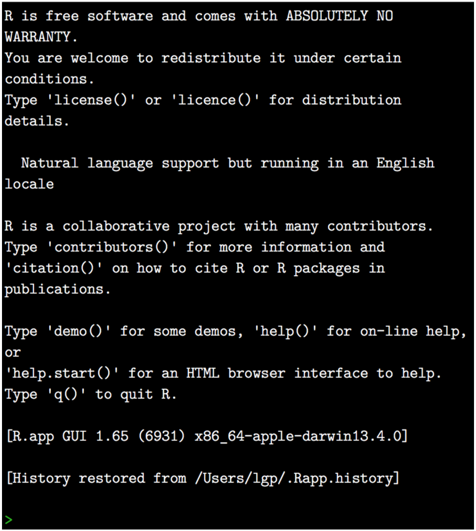
R is a free software environment for statistical computing and graphics.

To download R , got to `www.r-project.org`

R is the most reasonable platform for the **dissemination** of algebraic statistics methods.

… not necessarily for **implementing** them.

WHAT IS R ?



```
R is free software and comes with ABSOLUTELY NO
WARRANTY.
You are welcome to redistribute it under certain
conditions.
Type 'license()' or 'licence()' for distribution
details.

  Natural language support but running in an English
locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in
publications.

Type 'demo()' for some demos, 'help()' for on-line help,
or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

[R.app GUI 1.65 (6931) x86_64-apple-darwin13.4.0]

[History restored from /Users/lgp/.Rapp.history]


>
```

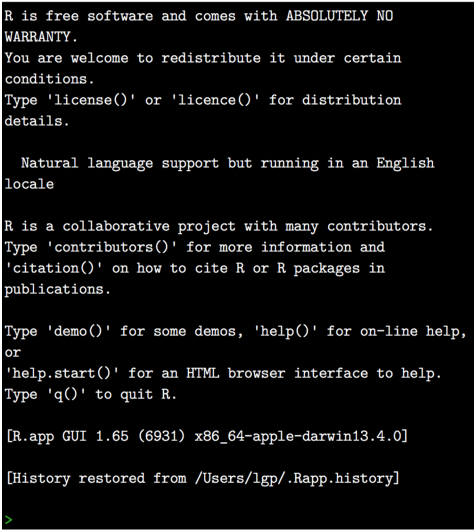
R is a free software environment for statistical computing and graphics.

To download R , got to `www.r-project.org`

R is the most reasonable platform for the **dissemination** of algebraic statistics methods.

… not necessarily for **implementing** them.

WHAT IS R ?



```
R is free software and comes with ABSOLUTELY NO
WARRANTY.
You are welcome to redistribute it under certain
conditions.
Type 'license()' or 'licence()' for distribution
details.

  Natural language support but running in an English
locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in
publications.

Type 'demo()' for some demos, 'help()' for on-line help,
or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

[R.app GUI 1.65 (6931) x86_64-apple-darwin13.4.0]

[History restored from /Users/lgp/.Rapp.history]

>
```

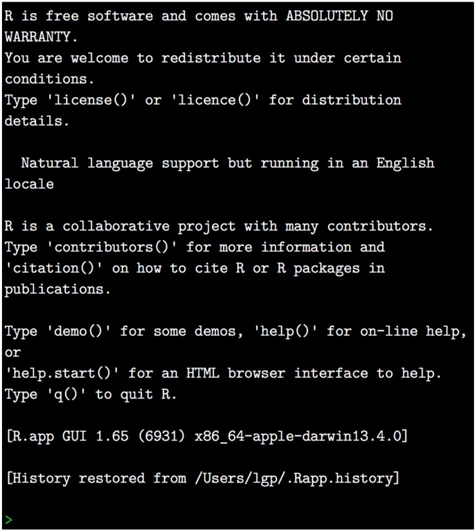
R is a free software environment for statistical computing and graphics.

To download R , got to `www.r-project.org`

R is the most reasonable platform for the **dissemination** of algebraic statistics methods.

… not necessarily for **implementing** them.

WHAT IS R ?



```
R is free software and comes with ABSOLUTELY NO
WARRANTY.
You are welcome to redistribute it under certain
conditions.
Type 'license()' or 'licence()' for distribution
details.

  Natural language support but running in an English
locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in
publications.

Type 'demo()' for some demos, 'help()' for on-line help,
or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

[R.app GUI 1.65 (6931) x86_64-apple-darwin13.4.0]

[History restored from /Users/lgp/.Rapp.history]

>
```

R is a free software environment for statistical computing and graphics.

To download R , got to `www.r-project.org`

R is the most reasonable platform for the **dissemination** of algebraic statistics methods.

… not necessarily for **implementing** them.

CHALLENGES

R has **no** base support for symbolic computing.

CHALLENGES

R has **no** base support for symbolic computing.

No base support for multivariate polynomial structures or operations.

CHALLENGES

R has **no** base support for symbolic computing.

No base support for multivariate polynomial structures or operations.

However it does have …

CHALLENGES

R has **no** base support for symbolic computing.

No base support for multivariate polynomial structures or operations.

However it does have …

Basic multivariate polynomial support through David Kahle' **mpoly** package.

CHALLENGES

R has **no** base support for symbolic computing.

No base support for multivariate polynomial structures or operations.

However it does have …

Basic multivariate polynomial support through David Kahle' **mpoly** package.

Access to the system.

CHALLENGES

R has **no** base support for symbolic computing.

No base support for multivariate polynomial structures or operations.

However it does have …

Basic multivariate polynomial support through David Kahle' **mpoly** package.

Access to the system.

So we can farm out computations to computer algebra systems (á la **Sage**).

CHALLENGES

R has **no** base support for symbolic computing.

No base support for multivariate polynomial structures or operations.

However it does have …

Basic multivariate polynomial support through David Kahle' **mpoly** package.

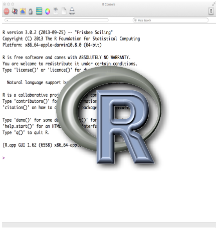
Access to the system.

So we can farm out computations to computer algebra systems (á la **Sage**).

Rcpp package to incorporate `C++` code for very fast implementations (Metropolis-Hastings algorithm).

ALGSTAT DESIGN

mpoly representations
as needed

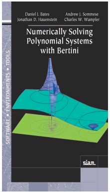


```
R version 3.0.2 (2013-09-25) -- "Frisbee Sailing"
Copyright (C) 2013 The R Foundation for Statistical Computing
Platform: x86_64-apple-darwin10.8.0 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for di

  Natural language support bu

R is a collaborative proj          cor
Type 'contributors()' fo          rmation
'citation()' on how to c          package

Type 'demo()' for some de          o()' for
'help.start()' for an HTML          interfa
Type 'q()' to quit R.

[R.app GUI 1.62 (6558) x86_64-apple

>
```

C++ For MCMC sampling

`4ti2` via `LattE-Integrale`

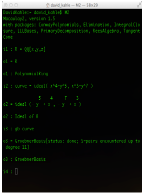
For computation of Markov bases for M-H algorithm proposal distribution



`Bertini`

Coming soon!

Solution to systems of multivariate polynomials



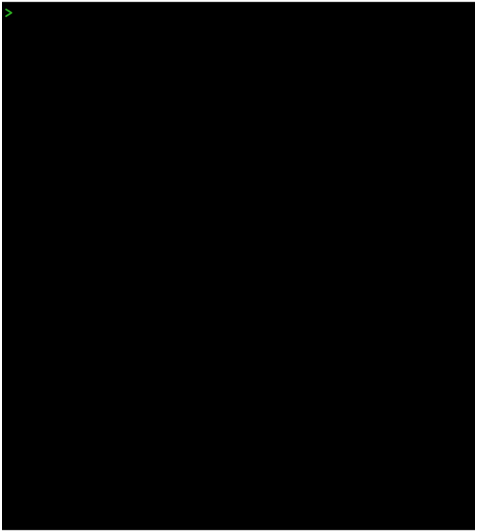
```
DavidKahle:~ david_kahle$ M2
Macaulay2, version 1.5
with packages: ConwayPolynomials, Elimination, IntegralClo
sure, LLLBases, PrimaryDecomposition, ReesAlgebra, Tangent
Cone

i1 : R = QQ[x,y,z]

o1 = R

o1 : PolynomialRing

i2 : curve = ideal( x^4-y^5, x^3-y^7 )

                5    4     7    3
o2 = ideal (- y  + x , - y  + x )

o2 : Ideal of R

i3 : gb curve

o3 = GroebnerBasis[status: done; S-pairs encountered up to
 degree 11]

o3 : GroebnerBasis

i4 : 
```

`Macaulay2`

Coming soon!

Algebraic geometry and commutative algebra

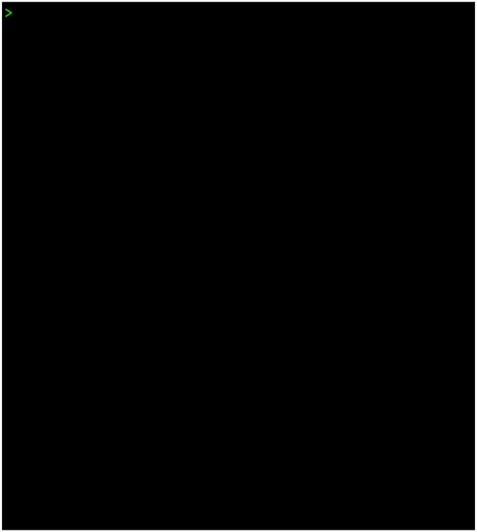
GETTING ALGSTAT



```
>
```

1. Download and install R
2. Download and install LattE integrale and Bertini.
3. Follow the code to the left

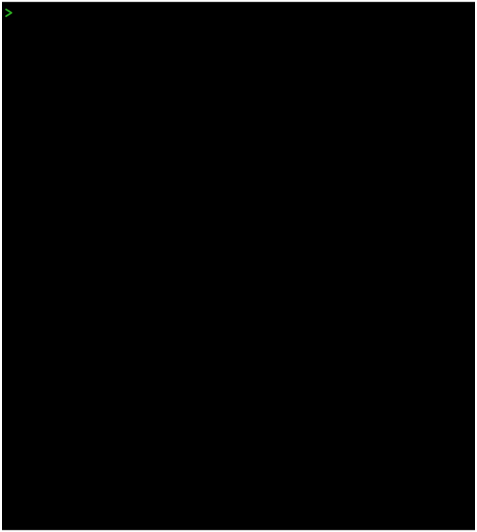
GETTING ALGSTAT



```
>
```

1. Download and install R
2. Download and install LattE integrale and Bertini.
3. Follow the code to the left

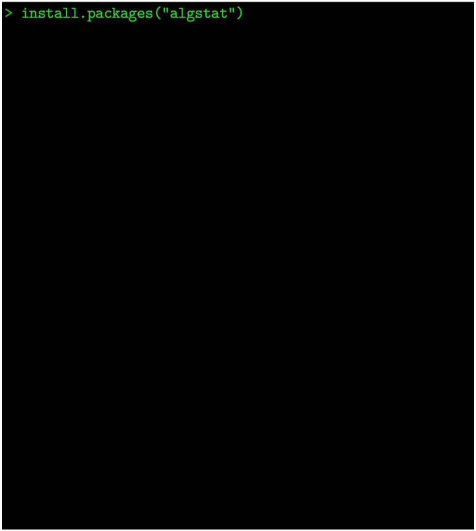
Getting algstat



```
>
```

1. Download and install R
2. Download and install LattE integrale and Bertini.
3. Follow the code to the left

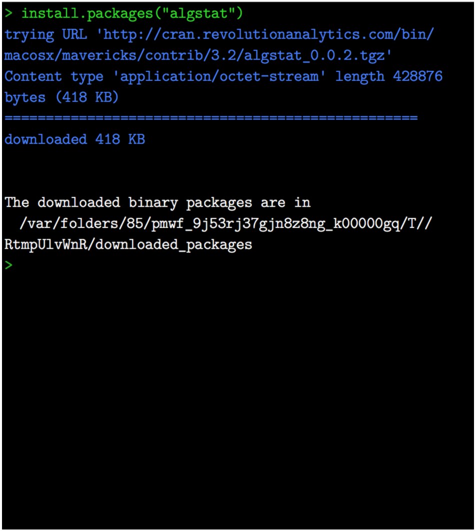
GETTING ALGSTAT



```
> install.packages("algstat")
```

1. Download and install R
2. Download and install LattE integrale and Bertini.
3. Follow the code to the left

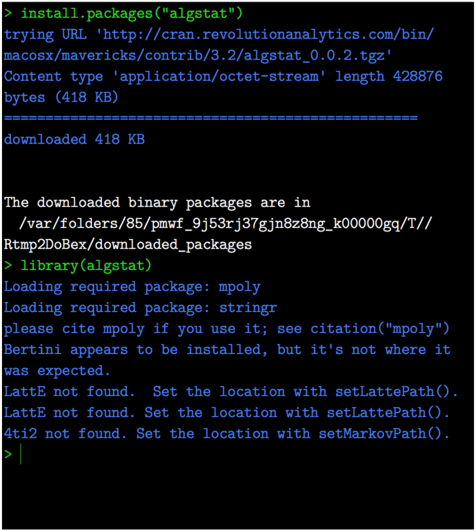
GETTING ALGSTAT



```
> install.packages("algstat")
trying URL 'http://cran.revolutionanalytics.com/bin/
macosx/mavericks/contrib/3.2/algstat_0.0.2.tgz'
Content type 'application/octet-stream' length 428876
bytes (418 KB)
==================================================
downloaded 418 KB


The downloaded binary packages are in
  /var/folders/85/pmwf_9j53rj37gjn8z8ng_k00000gq/T//
RtmpUlvWnR/downloaded_packages
>
```

1. Download and install R
2. Download and install LattE integrale and Bertini.
3. Follow the code to the left

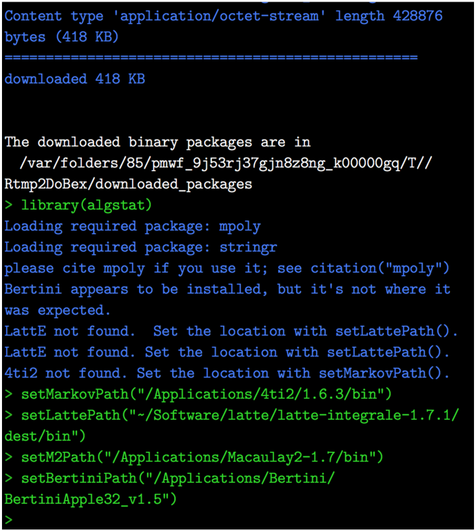
GETTING ALGSTAT



```
> install.packages("algstat")
trying URL 'http://cran.revolutionanalytics.com/bin/
macosx/mavericks/contrib/3.2/algstat_0.0.2.tgz'
Content type 'application/octet-stream' length 428876
bytes (418 KB)
==================================================
downloaded 418 KB


The downloaded binary packages are in
  /var/folders/85/pmwf_9j53rj37gjn8z8ng_k00000gq/T//
Rtmp2DoBex/downloaded_packages
> library(algstat)
Loading required package: mpoly
Loading required package: stringr
please cite mpoly if you use it; see citation("mpoly")
Bertini appears to be installed, but it's not where it
was expected.
LattE not found.  Set the location with setLattePath().
LattE not found. Set the location with setLattePath().
4ti2 not found. Set the location with setMarkovPath().
> 
```

1. Download and install R
2. Download and install LattE integrale and Bertini.
3. Follow the code to the left

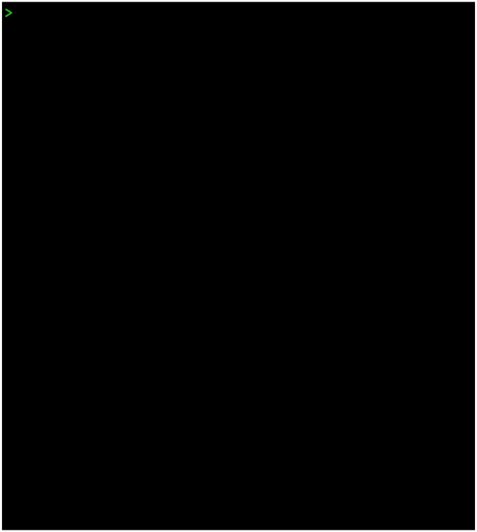
GETTING ALGSTAT



```
Content type 'application/octet-stream' length 428876
bytes (418 KB)
==================================================
downloaded 418 KB


The downloaded binary packages are in
  /var/folders/85/pmwf_9j53rj37gjn8z8ng_k00000gq/T//
Rtmp2DoBex/downloaded_packages
> library(algstat)
Loading required package: mpoly
Loading required package: stringr
please cite mpoly if you use it; see citation("mpoly")
Bertini appears to be installed, but it's not where it
was expected.
LattE not found.  Set the location with setLattePath().
LattE not found. Set the location with setLattePath().
4ti2 not found. Set the location with setMarkovPath().
> setMarkovPath("/Applications/4ti2/1.6.3/bin")
> setLattePath("~/Software/latte/latte-integrale-1.7.1/
dest/bin")
> setM2Path("/Applications/Macaulay2-1.7/bin")
> setBertiniPath("/Applications/Bertini/
BertiniApple32_v1.5")
>
```

1. Download and install R
2. Download and install LattE integrale and Bertini.
3. Follow the code to the left

ALGSTAT METHODS



```
>
```

A package = functions + data

algstat has many functions

Many are experimental

However, the functions for **exact inference in multi-way tables** are quite stable

ALGSTAT METHODS

```
> ls(pos = "package:algstat")
 [1] "Amaker"         "bertini"
 [3] "bump"           "condorcet"
 [5] "count"          "countTables"
 [7] "Emaker"         "hierarchical"
 [9] "hmat"           "is.bertini"
[11] "is.m2"          "kprod"
[13] "latteMax"       "latteMin"
[15] "lower"          "lpnorm"
[17] "m2"             "markov"
[19] "mchoose"        "metropolis"
[21] "Mmaker"         "ones"
[23] "Pmaker"         "polyOptim"
[25] "polySolve"      "print.spectral"
[27] "projectOnto"    "rvotes"
[29] "setBertiniPath" "setLattePath"
[31] "setM2Path"      "setMarkovPath"
[33] "Smaker"         "spectral"
[35] "subsets"        "tab2vec"
[37] "tableau"        "teshape"
[39] "Tmaker"         "upper"
[41] "variety"        "vec2tab"
>
```

A package = functions + data

algstat has many functions

Many are experimental

However, the functions for **exact inference in multi-way tables** are quite stable

ALGSTAT METHODS

```
> ls(pos = "package:algstat")
 [1] "Amaker"         "bertini"
 [3] "bump"           "condorcet"
 [5] "count"          "countTables"
 [7] "Emaker"         "hierarchical"
 [9] "hmat"           "is.bertini"
[11] "is.m2"          "kprod"
[13] "latteMax"       "latteMin"
[15] "lower"          "lpnorm"
[17] "m2"             "markov"
[19] "mchoose"        "metropolis"
[21] "Mmaker"         "ones"
[23] "Pmaker"         "polyOptim"
[25] "polySolve"      "print.spectral"
[27] "projectOnto"    "rvotes"
[29] "setBertiniPath" "setLattePath"
[31] "setM2Path"      "setMarkovPath"
[33] "Smaker"         "spectral"
[35] "subsets"        "tab2vec"
[37] "tableau"        "teshape"
[39] "Tmaker"         "upper"
[41] "variety"        "vec2tab"
>
```

A package = functions + data

algstat has many functions

Many are experimental

However, the functions for **exact inference in multi-way tables** are quite stable

ALGSTAT METHODS

```
> ls(pos = "package:algstat")
 [1] "Amaker"         "bertini"
 [3] "bump"           "condorcet"
 [5] "count"          "countTables"
 [7] "Emaker"         "hierarchical"
 [9] "hmat"           "is.bertini"
[11] "is.m2"          "kprod"
[13] "latteMax"       "latteMin"
[15] "lower"          "lpnorm"
[17] "m2"             "markov"
[19] "mchoose"        "metropolis"
[21] "Mmaker"         "ones"
[23] "Pmaker"         "polyOptim"
[25] "polySolve"      "print.spectral"
[27] "projectOnto"    "rvotes"
[29] "setBertiniPath" "setLattePath"
[31] "setM2Path"      "setMarkovPath"
[33] "Smaker"         "spectral"
[35] "subsets"        "tab2vec"
[37] "tableau"        "teshape"
[39] "Tmaker"         "upper"
[41] "variety"        "vec2tab"
>
```

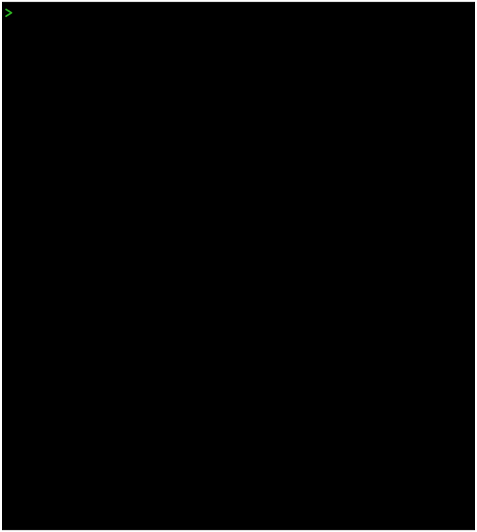
A package = functions + data

algstat has many functions

Many are experimental

However, the functions for **exact inference in multi-way tables** are quite stable

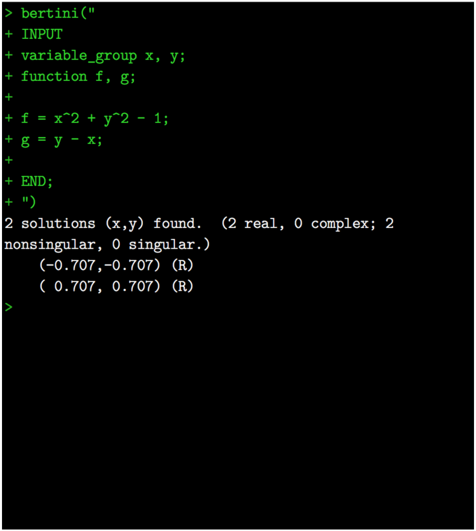
COMPUTATIONAL ALGEBRA (R + mpoly + Bertini)



```
>
```

Bertini functions

COMPUTATIONAL ALGEBRA (R + mpoly + Bertini)

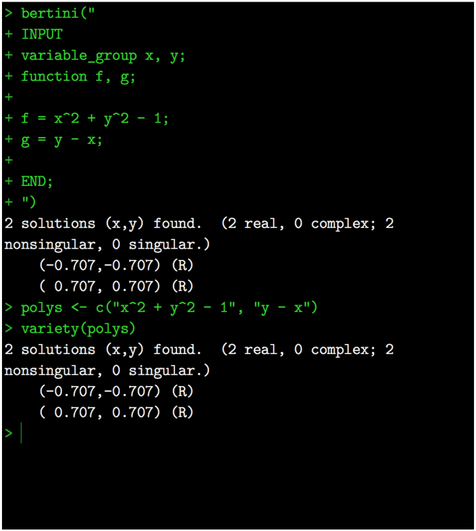


```
> bertini("
+ INPUT
+ variable_group x, y;
+ function f, g;
+
+ f = x^2 + y^2 - 1;
+ g = y - x;
+
+ END;
+ ")
2 solutions (x,y) found.  (2 real, 0 complex; 2
nonsingular, 0 singular.)
    (-0.707,-0.707) (R)
    ( 0.707, 0.707) (R)
>
```

Bertini functions

bertini: Evaluates raw Bertini code.

COMPUTATIONAL ALGEBRA (R + mpoly + Bertini)



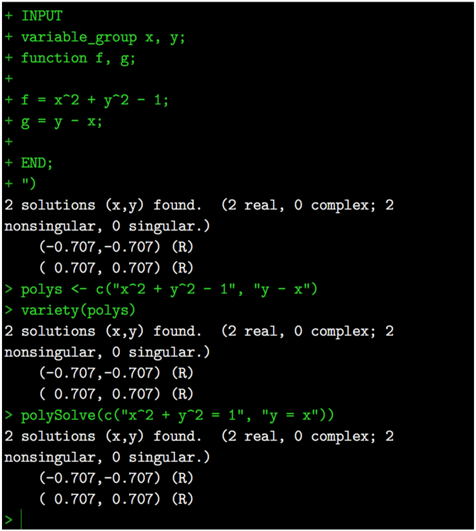
```
> bertini("
+ INPUT
+ variable_group x, y;
+ function f, g;
+
+ f = x^2 + y^2 - 1;
+ g = y - x;
+
+ END;
+ ")
2 solutions (x,y) found.  (2 real, 0 complex; 2
nonsingular, 0 singular.)
    (-0.707,-0.707) (R)
    ( 0.707, 0.707) (R)
> polys <- c("x^2 + y^2 - 1", "y - x")
> variety(polys)
2 solutions (x,y) found.  (2 real, 0 complex; 2
nonsingular, 0 singular.)
    (-0.707,-0.707) (R)
    ( 0.707, 0.707) (R)
>
```

Bertini functions

bertini: Evaluates raw Bertini code.

variety: Find the zero locus of a collection of multivariate polynomials. Currently limited to zero-dimensional varieties.

Computational Algebra (R + mpoly + Bertini)



```
+ INPUT
+ variable_group x, y;
+ function f, g;
+
+ f = x^2 + y^2 - 1;
+ g = y - x;
+
+ END;
+ ")
2 solutions (x,y) found.  (2 real, 0 complex; 2
nonsingular, 0 singular.)
   (-0.707,-0.707) (R)
   ( 0.707, 0.707) (R)
> polys <- c("x^2 + y^2 - 1", "y - x")
> variety(polys)
2 solutions (x,y) found.  (2 real, 0 complex; 2
nonsingular, 0 singular.)
   (-0.707,-0.707) (R)
   ( 0.707, 0.707) (R)
> polySolve(c("x^2 + y^2 = 1", "y = x"))
2 solutions (x,y) found.  (2 real, 0 complex; 2
nonsingular, 0 singular.)
   (-0.707,-0.707) (R)
   ( 0.707, 0.707) (R)
>
```

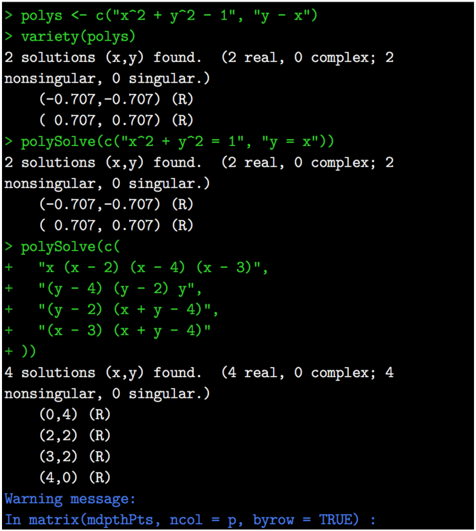
Bertini functions

bertini: Evaluates raw Bertini code.

variety: Find the zero locus of a collection of multivariate polynomials. Currently limited to zero-dimensional varieties.

polySolve: Solves a system of polynomial equations (with finitely many solutions).

COMPUTATIONAL ALGEBRA (R + mpoly + Bertini)



```
> polys <- c("x^2 + y^2 - 1", "y - x")
> variety(polys)
2 solutions (x,y) found.  (2 real, 0 complex; 2
nonsingular, 0 singular.)
    (-0.707,-0.707) (R)
    ( 0.707, 0.707) (R)
> polySolve(c("x^2 + y^2 = 1", "y = x"))
2 solutions (x,y) found.  (2 real, 0 complex; 2
nonsingular, 0 singular.)
    (-0.707,-0.707) (R)
    ( 0.707, 0.707) (R)
> polySolve(c(
+   "x (x - 2) (x - 4) (x - 3)",
+   "(y - 4) (y - 2) y",
+   "(y - 2) (x + y - 4)",
+   "(x - 3) (x + y - 4)"
+ ))
4 solutions (x,y) found.  (4 real, 0 complex; 4
nonsingular, 0 singular.)
    (0,4) (R)
    (2,2) (R)
    (3,2) (R)
    (4,0) (R)
Warning message:
In matrix(mdpthPts, ncol = p, byrow = TRUE) :
```

Bertini functions

bertini: Evaluates raw Bertini code.

variety: Find the zero locus of a collection of multivariate polynomials. Currently limited to zero-dimensional varieties.

polySolve: Solves a system of polynomial equations (with finitely many solutions).

ANALYSIS OF CONTINGENCY TABLES

ANALYSIS OF CONTINGENCY TABLES

Consider the contingency table

ANALYSIS OF CONTINGENCY TABLES

Consider the contingency table

		Political affiliation		
		Dem	Rep	
Personality	Introvert	3	7	10
	Extrovert	6	4	10
		9	11	20

ANALYSIS OF CONTINGENCY TABLES

Consider the contingency table

		Political affiliation		
		Dem	Rep	
Personality	Introvert	3	7	10
	Extrovert	6	4	10
		9	11	20

Q : Are personality and political affiliation independent?

ANALYSIS OF CONTINGENCY TABLES

Consider the contingency table

		Political affiliation		
		Dem	Rep	
Personality	Introvert	3	7	10
	Extrovert	6	4	10
		9	11	20

Q : Are personality and political affiliation independent?

The standard method tells us to

ANALYSIS OF CONTINGENCY TABLES

Consider the contingency table

		Political affiliation		
		Dem	Rep	
Personality	Introvert	3	7	10
	Extrovert	6	4	10
		9	11	20

Q : Are personality and political affiliation independent?

The standard method tells us to

1. Compute Pearson's chi-squared statistic, and

Analysis of contingency tables

Consider the contingency table

		Political affiliation		
		Dem	Rep	
Personality	Introvert	3	7	10
	Extrovert	6	4	10
		9	11	20

Q : Are personality and political affiliation independent?

The standard method tells us to

1. Compute Pearson's chi-squared statistic, and

$$\chi_2 = \sum_{\text{cells}} \frac{(O-E)^2}{E}$$

Analysis of contingency tables

Consider the contingency table

		Political affiliation		
		Dem	Rep	
Personality	Introvert	3	7	10
	Extrovert	6	4	10
		9	11	20

Q : Are personality and political affiliation independent?

The standard method tells us to

1. Compute Pearson's chi-squared statistic, and

$$\chi_2 = \sum_{\text{cells}} \frac{(O-E)^2}{E} = \frac{(3-\frac{10\times 9}{20})^2}{\frac{10\times 9}{20}} + \frac{(7-\frac{10\times 11}{20})^2}{\frac{10\times 11}{20}} + \frac{(6-\frac{10\times 9}{20})^2}{\frac{10\times 9}{20}} + \frac{(4-\frac{10\times 11}{20})^2}{\frac{10\times 11}{20}} = 1.8182$$

ANALYSIS OF CONTINGENCY TABLES

Consider the contingency table

		Political affiliation		
		Dem	Rep	
Personality	Introvert	3	7	10
	Extrovert	6	4	10
		9	11	20

Q : Are personality and political affiliation independent?

The standard method tells us to

1. Compute Pearson's chi-squared statistic, and

$$\chi_2 = \sum_{\text{cells}} \frac{(O-E)^2}{E} = \frac{(3-\frac{10\times 9}{20})^2}{\frac{10\times 9}{20}} + \frac{(7-\frac{10\times 11}{20})^2}{\frac{10\times 11}{20}} + \frac{(6-\frac{10\times 9}{20})^2}{\frac{10\times 9}{20}} + \frac{(4-\frac{10\times 11}{20})^2}{\frac{10\times 11}{20}} = 1.8182$$

2. Compute the p-value based on the chi-squared 1 distribution

ANALYSIS OF CONTINGENCY TABLES

Consider the contingency table

		Political affiliation		
		Dem	Rep	
Personality	Introvert	3	7	10
	Extrovert	6	4	10
		9	11	20

Q : Are personality and political affiliation independent?

The standard method tells us to

1. Compute Pearson's chi-squared statistic, and

$$\chi_2 = \sum_{\text{cells}} \frac{(O-E)^2}{E} = \frac{(3-\frac{10\times 9}{20})^2}{\frac{10\times 9}{20}} + \frac{(7-\frac{10\times 11}{20})^2}{\frac{10\times 11}{20}} + \frac{(6-\frac{10\times 9}{20})^2}{\frac{10\times 9}{20}} + \frac{(4-\frac{10\times 11}{20})^2}{\frac{10\times 11}{20}} = 1.8182$$

2. Compute the p-value based on the chi-squared 1 distribution

$$p - \text{value} = P\left[\chi_1^2 \geq 1.8182\right] = .1775$$

ANALYSIS OF CONTINGENCY TABLES

Consider the contingency table

		Political affiliation		
		Dem	Rep	
Personality	Introvert	3	7	10
	Extrovert	6	4	10
		9	11	20

Q : Are personality and political affiliation independent?

The standard method tells us to

1. Compute Pearson's chi-squared statistic, and

$$\chi_2 = \sum_{\text{cells}} \frac{(O-E)^2}{E} = \frac{(3-\frac{10\times 9}{20})^2}{\frac{10\times 9}{20}} + \frac{(7-\frac{10\times 11}{20})^2}{\frac{10\times 11}{20}} + \frac{(6-\frac{10\times 9}{20})^2}{\frac{10\times 9}{20}} + \frac{(4-\frac{10\times 11}{20})^2}{\frac{10\times 11}{20}} = 1.8182$$

2. Compute the p-value based on the chi-squared 1 distribution

$$p - \text{value} = P\left[\chi_1^2 \geq 1.8182\right] = .1775$$

3. Decide whether it is reasonable (i.e., reject independence if p is small)

ANALYSIS OF CONTINGENCY TABLES

Consider the contingency table

		Political affiliation		
		Dem	Rep	
Personality	Introvert	3	7	10
	Extrovert	6	4	10
		9	11	20

Q : Are personality and political affiliation independent?

Fisher’s method tells us to (this is the so-called “exact test”)

ANALYSIS OF CONTINGENCY TABLES

Consider the contingency table

		Political affiliation		
		Dem	Rep	
Personality	Introvert	3	7	10
	Extrovert	6	4	10
		9	11	20

Q : Are personality and political affiliation independent?

Fisher's method tells us to (this is the so-called "exact test")

1. Compute the probability of every table, conditional on the marginals

ANALYSIS OF CONTINGENCY TABLES

Consider the contingency table

		Political affiliation		
		Dem	Rep	
Personality	Introvert	3	7	10
	Extrovert	6	4	10
		9	11	20

Q : Are personality and political affiliation independent?

Fisher's method tells us to (this is the so-called "exact test")

1. Compute the probability of every table, conditional on the marginals

$$\begin{array}{|cc|}\hline 0 & 10 \\ 9 & 1 \\ \hline\end{array}\ \begin{array}{|cc|}\hline 1 & 9 \\ 8 & 2 \\ \hline\end{array}\ \begin{array}{|cc|}\hline 2 & 8 \\ 7 & 3 \\ \hline\end{array}\ \begin{array}{|cc|}\hline 3 & 7 \\ 6 & 4 \\ \hline\end{array}\ \begin{array}{|cc|}\hline 4 & 6 \\ 5 & 5 \\ \hline\end{array}\ \begin{array}{|cc|}\hline 5 & 5 \\ 4 & 6 \\ \hline\end{array}\ \begin{array}{|cc|}\hline 6 & 4 \\ 3 & 7 \\ \hline\end{array}\ \begin{array}{|cc|}\hline 7 & 3 \\ 2 & 8 \\ \hline\end{array}\ \begin{array}{|cc|}\hline 8 & 2 \\ 1 & 9 \\ \hline\end{array}\ \begin{array}{|cc|}\hline 9 & 1 \\ 0 & 10 \\ \hline\end{array}$$

ANALYSIS OF CONTINGENCY TABLES

Consider the contingency table

		Political affiliation		
		Dem	Rep	
Personality	Introvert	3	7	10
	Extrovert	6	4	10
		9	11	20

Q : Are personality and political affiliation independent?

Fisher's method tells us to (this is the so-called "exact test")

1. Compute the probability of every table, conditional on the marginals

$$\begin{array}{|cc|}\hline 0 & 10 \\ 9 & 1 \\ \hline\end{array}\;\begin{array}{|cc|}\hline 1 & 9 \\ 8 & 2 \\ \hline\end{array}\;\begin{array}{|cc|}\hline 2 & 8 \\ 7 & 3 \\ \hline\end{array}\;\overset{\text{OBSERVED}}{\begin{array}{|cc|}\hline 3 & 7 \\ 6 & 4 \\ \hline\end{array}}\;\begin{array}{|cc|}\hline 4 & 6 \\ 5 & 5 \\ \hline\end{array}\;\begin{array}{|cc|}\hline 5 & 5 \\ 4 & 6 \\ \hline\end{array}\;\begin{array}{|cc|}\hline 6 & 4 \\ 3 & 7 \\ \hline\end{array}\;\begin{array}{|cc|}\hline 7 & 3 \\ 2 & 8 \\ \hline\end{array}\;\begin{array}{|cc|}\hline 8 & 2 \\ 1 & 9 \\ \hline\end{array}\;\begin{array}{|cc|}\hline 9 & 1 \\ 0 & 10 \\ \hline\end{array}$$

ANALYSIS OF CONTINGENCY TABLES

Consider the contingency table

		Political affiliation		
		Dem	Rep	
Personality	Introvert	3	7	10
	Extrovert	6	4	10
		9	11	20

Q : Are personality and political affiliation independent?

Fisher's method tells us to (this is the so-called "exact test")

1. Compute the probability of every table, conditional on the marginals

$$\begin{bmatrix} 0 & 10 \\ 9 & 1 \end{bmatrix} \begin{bmatrix} 1 & 9 \\ 8 & 2 \end{bmatrix} \begin{bmatrix} 2 & 8 \\ 7 & 3 \end{bmatrix} \overset{\text{OBSERVED}}{\begin{bmatrix} 3 & 7 \\ 6 & 4 \end{bmatrix}} \begin{bmatrix} 4 & 6 \\ 5 & 5 \end{bmatrix} \begin{bmatrix} 5 & 5 \\ 4 & 6 \end{bmatrix} \begin{bmatrix} 6 & 4 \\ 3 & 7 \end{bmatrix} \begin{bmatrix} 7 & 3 \\ 2 & 8 \end{bmatrix} \begin{bmatrix} 8 & 2 \\ 1 & 9 \end{bmatrix} \begin{bmatrix} 9 & 1 \\ 0 & 10 \end{bmatrix}$$

There are $n = 10$ such tables

ANALYSIS OF CONTINGENCY TABLES

Consider the contingency table

		Political affiliation		
		Dem	Rep	
Personality	Introvert	3	7	10
	Extrovert	6	4	10
		9	11	20

Q : Are personality and political affiliation independent?

Fisher's method tells us to (this is the so-called "exact test")

1. Compute the probability of every table, conditional on the marginals

						OBSERVED													
0	10	1	9	2	8	3	7	4	6	5	5	6	4	7	3	8	2	9	1
9	1	8	2	7	3	6	4	5	5	4	6	3	7	2	8	1	9	0	10

There are $n = 10$ such tables

There probabilities are hypergeometric

ANALYSIS OF CONTINGENCY TABLES

Consider the contingency table

		Political affiliation		
		Dem	Rep	
Personality	Introvert	3	7	10
	Extrovert	6	4	10
		9	11	20

Q : Are personality and political affiliation independent?

Fisher's method tells us to (this is the so-called "exact test")

1. Compute the probability of every table, conditional on the marginals

			OBSERVED						
0 10 9 1	1 9 8 2	2 8 7 3	3 7 6 4	4 6 5 5	5 5 4 6	6 4 3 7	7 3 2 8	8 2 1 9	9 1 0 10
.0001	.0027	.0322	.1500	.3151	.3151	.1500	.0322	.0027	.0001

There are $n = 10$ such tables

There probabilities are hypergeometric

Analysis of contingency tables

Consider the contingency table

		Political affiliation		
		Dem	Rep	
Personality	Introvert	3	7	10
	Extrovert	6	4	10
		9	11	20

Q : Are personality and political affiliation independent?

Fisher's method tells us to (this is the so-called "exact test")

1. Compute the probability of every table, conditional on the marginals

			OBSERVED						
0 10	1 9	2 8	3 7	4 6	5 5	6 4	7 3	8 2	9 1
9 1	8 2	7 3	6 4	5 5	4 6	3 7	2 8	1 9	0 10
.0001	.0027	.0322	.1500	.3151	.3151	.1500	.0322	.0027	.0001

There are $n = 10$ such tables

There probabilities are hypergeometric

2. Compute the p-value by summing the probabilities of the tables with smaller probabilities

ANALYSIS OF CONTINGENCY TABLES

Consider the contingency table

		Political affiliation		
		Dem	Rep	
Personality	Introvert	3	7	10
	Extrovert	6	4	10
		9	11	20

Q : Are personality and political affiliation independent?

Fisher's method tells us to (this is the so-called "exact test")

1. Compute the probability of every table, conditional on the marginals

			OBSERVED						
0 10 9 1	1 9 8 2	2 8 7 3	3 7 6 4	4 6 5 5	5 5 4 6	6 4 3 7	7 3 2 8	8 2 1 9	9 1 0 10
.0001	.0027	.0322	.1500	.3151	.3151	.1500	.0322	.0027	.0001

There are $n = 10$ such tables

There probabilities are hypergeometric

2. Compute the p-value by summing the probabilities of the tables with smaller probabilities

$$p - \text{value} = .3698$$

ANALYSIS OF CONTINGENCY TABLES

Consider the contingency table

		Political affiliation		
		Dem	Rep	
Personality	Introvert	3	7	10
	Extrovert	6	4	10
		9	11	20

Q : Are personality and political affiliation independent?

Fisher's method tells us to (this is the so-called "exact test")

1. Compute the probability of every table, conditional on the marginals

			OBSERVED						
0 10	1 9	2 8	3 7	4 6	5 5	6 4	7 3	8 2	9 1
9 1	8 2	7 3	6 4	5 5	4 6	3 7	2 8	1 9	0 10
.0001	.0027	.0322	.1500	.3151	.3151	.1500	.0322	.0027	.0001

There are $n = 10$ such tables

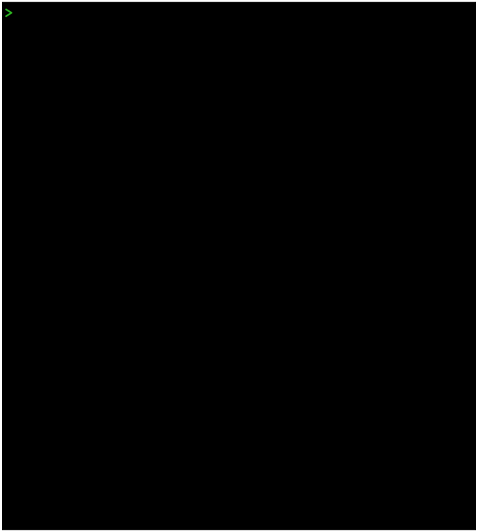
There probabilities are hypergeometric

2. Compute the p-value by summing the probabilities of the tables with smaller probabilities

$$p - \text{value} = .3698$$

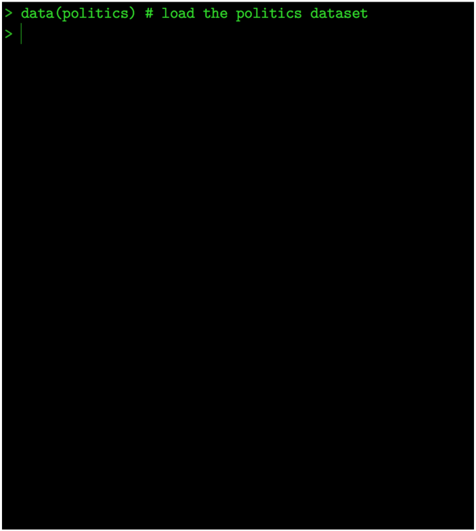
3. Decide whether it is reasonable

Analysis of contingency tables in R



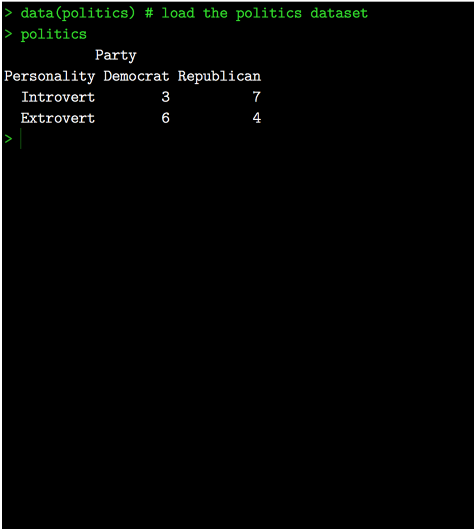
```
>
```

Analysis of contingency tables in R



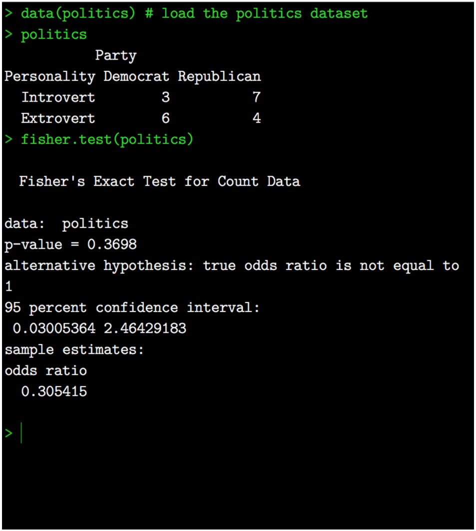
```
> data(politics) # load the politics dataset
> 
```

ANALYSIS OF CONTINGENCY TABLES IN R



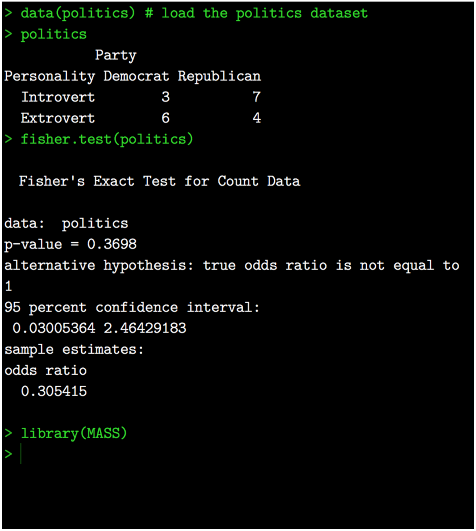
```
> data(politics) # load the politics dataset
> politics
           Party
Personality Democrat Republican
  Introvert        3          7
  Extrovert        6          4
> 
```

ANALYSIS OF CONTINGENCY TABLES IN R



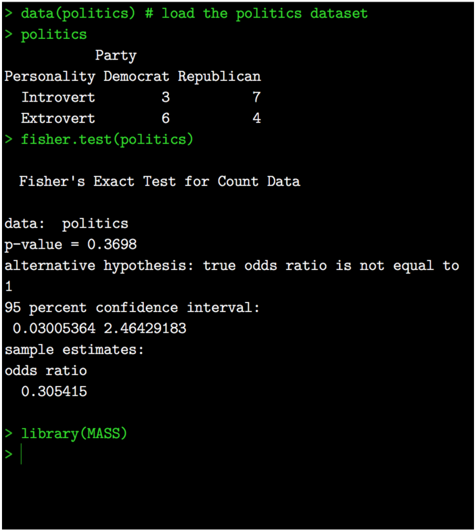
```
> data(politics) # load the politics dataset
> politics
           Party
Personality Democrat Republican
  Introvert        3          7
  Extrovert        6          4
> fisher.test(politics)

	Fisher's Exact Test for Count Data

data:  politics
p-value = 0.3698
alternative hypothesis: true odds ratio is not equal to
1
95 percent confidence interval:
 0.03005364 2.46429183
sample estimates:
odds ratio
  0.305415 

> 
```

ANALYSIS OF CONTINGENCY TABLES IN R

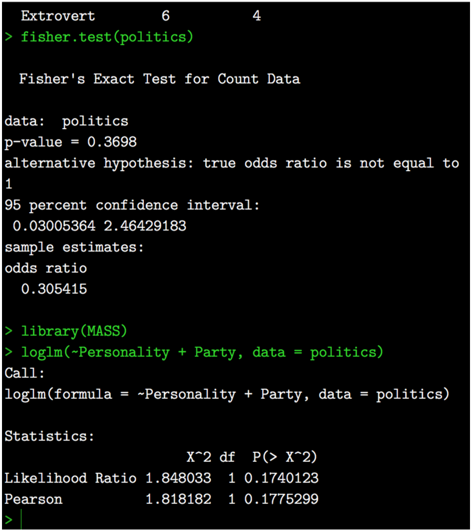


```
> data(politics) # load the politics dataset
> politics
           Party
Personality Democrat Republican
  Introvert        3          7
  Extrovert        6          4
> fisher.test(politics)

  Fisher's Exact Test for Count Data

data:  politics
p-value = 0.3698
alternative hypothesis: true odds ratio is not equal to
1
95 percent confidence interval:
 0.03005364 2.46429183
sample estimates:
odds ratio
  0.305415

> library(MASS)
> 
```

ANALYSIS OF CONTINGENCY TABLES IN R



```
> data(politics) # load the politics dataset
> politics
           Party
Personality Democrat Republican
  Introvert        3          7
  Extrovert        6          4
> fisher.test(politics)

  Fisher's Exact Test for Count Data

data:  politics
p-value = 0.3698
alternative hypothesis: true odds ratio is not equal to
1
95 percent confidence interval:
 0.03005364 2.46429183
sample estimates:
odds ratio
  0.305415

> library(MASS)
> 
```



```
  Extrovert        6          4
> fisher.test(politics)

  Fisher's Exact Test for Count Data

data:  politics
p-value = 0.3698
alternative hypothesis: true odds ratio is not equal to
1
95 percent confidence interval:
 0.03005364 2.46429183
sample estimates:
odds ratio
  0.305415

> library(MASS)
> loglm(~Personality + Party, data = politics)
Call:
loglm(formula = ~Personality + Party, data = politics)

Statistics:
                      X^2 df  P(> X^2)
Likelihood Ratio 1.848033  1 0.1740123
Pearson          1.818182  1 0.1775299
> 
```

GENERATING RANDOM TABLES

PROBLEM

What happens when the entries in the table are too small to be confident on asymptotic methods, but the number of tables with given row and column sums is too large to enumerate?

GENERATING RANDOM TABLES

PROBLEM

What happens when the entries in the table are too small to be confident on asymptotic methods, but the number of tables with given row and column sums is too large to enumerate?

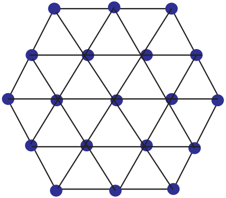
We would like to generate a sample of random tables from the set of all nonnegative integer table with given row and column sums.

				r_1
				r_2
				r_3
c_1	c_2	c_3	c_4	

RANDOM WALK

$$\begin{array}{|c|c|c||c|}\hline 2&2&2&6\\\hline 2&2&2&6\\\hline\hline 4&4&4&\\\hline\end{array} + \begin{array}{|c|c|c||c|}\hline 1&0&-1&0\\\hline -1&0&1&0\\\hline\hline 0&0&0&\\\hline\end{array} = \begin{array}{|c|c|c||c|}\hline 3&2&1&6\\\hline 1&2&3&6\\\hline\hline 4&4&4&\\\hline\end{array}$$

$$\begin{array}{|c|c|c||c|}\hline 3&2&1&6\\\hline 1&2&3&6\\\hline\hline 4&4&4&\\\hline\end{array} + \begin{array}{|c|c|c||c|}\hline 1&-1&0&0\\\hline -1&1&0&0\\\hline\hline 0&0&0&\\\hline\end{array} = \begin{array}{|c|c|c||c|}\hline 4&1&1&6\\\hline 0&3&3&6\\\hline\hline 4&4&4&\\\hline\end{array}$$



$$\left\{ \begin{pmatrix} 1 & -1 & 0 \\ -1 & 1 & 0 \end{pmatrix}, \begin{pmatrix} 1 & 0 & -1 \\ -1 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 0 & 1 & -1 \\ 0 & -1 & 1 \end{pmatrix} \right\}$$

allow for a connected random walk over these contingency tables.

CONNECTING LATTICE POINTS IN POLYTOPES

DEFINITION

- Let $A : \mathbb{Z}^n \to \mathbb{Z}^d$ a linear transformation and $b \in \mathbb{Z}^d$.
- $A^{-1}[b] := \{x \in \mathbb{N}^n \mid Ax = b\}$ (**fiber**)
- $\mathscr{B} \subset \ker_{\mathbb{Z}} A$

Let $A^{-1}[b]_{\mathscr{B}}$ be the **graph** with vertex set $A^{-1}[b]$ and edge set u — v for every u and v in $A^{-1}[b]$ such that $u - v \in \pm\mathscr{B}$. (**Markov graph**)

CONNECTING LATTICE POINTS IN POLYTOPES

DEFINITION

- Let $A : \mathbb{Z}^n \to \mathbb{Z}^d$ a linear transformation and $b \in \mathbb{Z}^d$.
- $A^{-1}[b] := \{x \in \mathbb{N}^n \mid Ax = b\}$ (**fiber**)
- $\mathscr{B} \subset \ker_{\mathbb{Z}} A$

Let $A^{-1}[b]_{\mathscr{B}}$ be the **graph** with vertex set $A^{-1}[b]$ and edge set $u — v$ for every u and v in $A^{-1}[b]$ such that $u - v \in \pm\mathscr{B}$. (**Markov graph**)

PROBLEM

Given A and b, find finite $\mathscr{B} \subset \ker_{\mathbb{Z}} A$ such that $A^{-1}[b]_{\mathscr{B}}$ is connected.

CONNECTING LATTICE POINTS IN POLYTOPES

DEFINITION

- Let $A : \mathbb{Z}^n \to \mathbb{Z}^d$ a linear transformation and $b \in \mathbb{Z}^d$.
- $A^{-1}[b] := \{x \in \mathbb{N}^n \mid Ax = b\}$ (**fiber**)
- $\mathscr{B} \subset \ker_{\mathbb{Z}} A$

Let $A^{-1}[b]_{\mathscr{B}}$ be the **graph** with vertex set $A^{-1}[b]$ and edge set $u — v$ for every u and v in $A^{-1}[b]$ such that $u - v \in \pm\mathscr{B}$. (**Markov graph**)

PROBLEM

Given A and b, find finite $\mathscr{B} \subset \ker_{\mathbb{Z}} A$ such that $A^{-1}[b]_{\mathscr{B}}$ is connected.

DEFINITION

If $\mathscr{B} \subset \ker_{\mathbb{Z}} A$ is a set such that $A^{-1}[b]_{\mathscr{B}}$ is connected for all b, then $\mathscr{B}$ is a **Markov basis** for A.

2-WAY TABLES

Let $A : \mathbb{Z}^{k_1 \times k_2} \to \mathbb{Z}^{k_1 + k_2}$ defined by

$$\begin{aligned} A(u) &= \left(u_{1+}, \ldots, u_{k_1+}; u_{+1}, \ldots, u_{+k_2}\right) \\ &= \text{vector of row and column sums of } u \end{aligned}$$

$\ker_{\mathbb{Z}}(A) = \left\{u \in \mathbb{Z}^{k_1 \times k_2} \mid \text{row and column sums of } u \text{ are } 0\right\}$

2-WAY TABLES

Let $A : \mathbb{Z}^{k_1 \times k_2} \to \mathbb{Z}^{k_1 + k_2}$ defined by

$$\begin{aligned} A(u) &= \left(u_{1+}, \ldots, u_{k_1+}; u_{+1}, \ldots, u_{+k_2}\right) \\ &= \text{vector of row and column sums of } u \end{aligned}$$

$\ker_{\mathbb{Z}}(A) = \left\{u \in \mathbb{Z}^{k_1 \times k_2} \mid \text{row and column sums of } u \text{ are } 0\right\}$

Markov basis consists of the $2\binom{k_1}{2}\binom{k_2}{2}$ moves like

$$\begin{pmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & -1 & 0 \\ -1 & 0 & 1 & 0 \end{pmatrix}$$

3-WAY TABLES

Let $A : \mathbb{Z}^{k_1 \times k_2 \times k_3} \to \mathbb{Z}^{k_1k_2 + k_1k_3 + k_2k_3}$ defined by

$$A(u) = \left(\left(\sum_{i_3} u_{i_1 i_2 i_3} \right)_{i_1, i_2} ; \left(\sum_{i_2} u_{i_1 i_2 i_3} \right)_{i_1, i_3} ; \left(\sum_{i_1} u_{i_1 i_2 i_3} \right)_{i_2, i_3} \right)$$

$= \text{all 2-way margins of the 3-way table } u$

3-WAY TABLES

Let $A : \mathbb{Z}^{k_1 \times k_2 \times k_3} \to \mathbb{Z}^{k_1 k_2 + k_1 k_3 + k_2 k_3}$ defined by

$$A(u) = \left(\left(\sum_{i_3} u_{i_1 i_2 i_3} \right)_{i_1, i_2} ; \left(\sum_{i_2} u_{i_1 i_2 i_3} \right)_{i_1, i_3} ; \left(\sum_{i_1} u_{i_1 i_2 i_3} \right)_{i_2, i_3} \right)$$

$$= \text{all 2-way margins of the 3-way table } u$$

Markov basis depends on k_1, k_2, k_3, contains moves like:

$$\begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix} \quad \begin{pmatrix} -1 & 1 \\ 1 & -1 \end{pmatrix}$$

but also non-obvious moves like:

$$\begin{pmatrix} 1 & -1 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} -1 & 1 & 0 \\ 0 & 0 & 0 \\ 1 & -1 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & -1 \\ -1 & 0 & 1 \end{pmatrix} \begin{pmatrix} 0 & -1 & 1 \\ 0 & 0 & 0 \\ 0 & 1 & -1 \end{pmatrix} \begin{pmatrix} 0 & 1 & -1 \\ 0 & -1 & 1 \\ 0 & 0 & 0 \end{pmatrix}$$

FUNDAMENTAL THEOREM OF MARKOV BASES

DEFINITION

Let $A : \mathbb{Z}^n \to \mathbb{Z}^d$. The **toric ideal** I_A is the ideal

$$\langle p^u - p^v \mid u, v \in \mathbb{N}^n, Au = Av \rangle \subset \mathbb{K}[p_1, \ldots, p_n],$$

where $p^u = p_1^{u_1} p_2^{u_2} \cdots p_n^{u_n}$.

FUNDAMENTAL THEOREM OF MARKOV BASES

DEFINITION

Let $A : \mathbb{Z}^n \to \mathbb{Z}^d$. The **toric ideal** I_A is the ideal

$$\langle p^u - p^v \mid u, v \in \mathbb{N}^n, Au = Av \rangle \subset \mathbb{K}[p_1, \ldots, p_n],$$

where $p^u = p_1^{u_1} p_2^{u_2} \cdots p_n^{u_n}$.

THEOREM (DIACONIS-STURMFELS 1998)

The set of moves $\mathscr{B} \subset \ker_{\mathbb{Z}} A$ is a ***Markov basis*** *for A if and only if the set of binomials $\{p^{b^+} - p^{b^-} \mid b \in \mathscr{B}\}$ generates I_A.*

FUNDAMENTAL THEOREM OF MARKOV BASES

DEFINITION

Let $A : \mathbb{Z}^n \to \mathbb{Z}^d$. The toric ideal I_A is the ideal

$$\langle p^u - p^v \mid u, v \in \mathbb{N}^n, Au = Av \rangle \subset \mathbb{K}[p_1, \ldots, p_n],$$

where $p^u = p_1^{u_1} p_2^{u_2} \cdots p_n^{u_n}$.

THEOREM (DIACONIS-STURMFELS 1998)

The set of moves $\mathscr{B} \subset \ker_{\mathbb{Z}} A$ is a ***Markov basis*** *for A if and only if the set of binomials $\{p^{b^+} - p^{b^-} \mid b \in \mathscr{B}\}$ generates I_A.*

$$\begin{pmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & -1 & 0 \\ -1 & 0 & 1 & 0 \end{pmatrix} \longrightarrow p_{21}p_{33} - p_{23}p_{31}$$

TORIC VARIETIES = LOG-LINEAR MODELS

DEFINITION

The variety $V_A = V(I_A)$ is a **toric variety**. The statistical model $\mathscr{M}_A = V(I_A) \cap \Delta_m$ is a **log-linear model**.

TORIC VARIETIES = LOG-LINEAR MODELS

DEFINITION

The variety $V_A = V(I_A)$ is a **toric variety**. The statistical model $\mathscr{M}_A = V(I_A) \cap \Delta_m$ is a **log-linear model**.

$$\mathscr{M}_A = \{p \in \Delta_m \mid \log p \in \operatorname{rowspan} A\}\,.$$

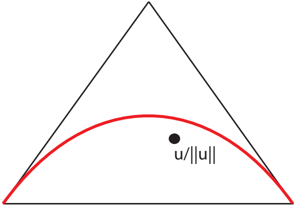
TORIC VARIETIES = LOG-LINEAR MODELS

DEFINITION

The variety $V_A = V(I_A)$ is a **toric variety**. The statistical model $\mathscr{M}_A = V(I_A) \cap \Delta_m$ is a **log-linear model**.

$$\mathscr{M}_A = \{p \in \Delta_m \mid \log p \in \text{rowspan}\, A\}\,.$$

Fisher's exact test: Does the data $\mathbf{u}$ fit the model $\mathscr{M}_A$?



ALGEBRAIC REPRESENTATION OF LOG-LINEAR MODELS

Traditional representation

$$\log \pi_{x_1 x_2} = \xi_0 + \xi^{(1)}_{x_1} + \xi^{(2)}_{x_2}$$

ALGEBRAIC REPRESENTATION OF LOG-LINEAR MODELS

Traditional representation

$$\log \pi_{x_1 x_2} = \xi_0 + \xi^{(1)}_{x_1} + \xi^{(2)}_{x_2}$$

$$\Updownarrow$$

Algebraic representation

$$\pi_{\boldsymbol{x}} = P\left[\boldsymbol{X} = \boldsymbol{x}\right] = \frac{1}{Z(\boldsymbol{\theta})}\boldsymbol{\theta}^{\boldsymbol{a}_{\boldsymbol{x}}}$$

$$\boldsymbol{A} = \begin{array}{c} \\ \theta_1 \\ \theta_2 \\ \theta_3 \\ \theta_4 \end{array} \begin{array}{c} \begin{array}{cccc} \pi_{11} & \pi_{12} & \pi_{21} & \pi_{22} \end{array} \\ \left[\begin{array}{cccc} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{array}\right] \end{array}$$

ALGEBRAIC REPRESENTATION OF LOG-LINEAR MODELS

Traditional representation

$$\log \pi_{x_1 x_2} = \xi_0 + \xi_{x_1}^{(1)} + \xi_{x_2}^{(2)}$$

$$\Updownarrow$$

Algebraic representation

$$\pi_{\boldsymbol{x}} = P\left[\boldsymbol{X} = \boldsymbol{x}\right] = \frac{1}{Z(\boldsymbol{\theta})}\boldsymbol{\theta}^{\boldsymbol{a}_{\boldsymbol{x}}}$$

$$\boldsymbol{A} = \boxed{\begin{array}{c|cccc|} & \pi_{11} & \pi_{12} & \pi_{21} & \pi_{22} \\ \theta_1 & 1 & 1 & 0 & 0 \\ \theta_2 & 0 & 0 & 1 & 1 \\ \theta_3 & 1 & 0 & 1 & 0 \\ \theta_4 & 0 & 1 & 0 & 1 \end{array}}$$

Compute the moves

ALGEBRAIC REPRESENTATION OF LOG-LINEAR MODELS

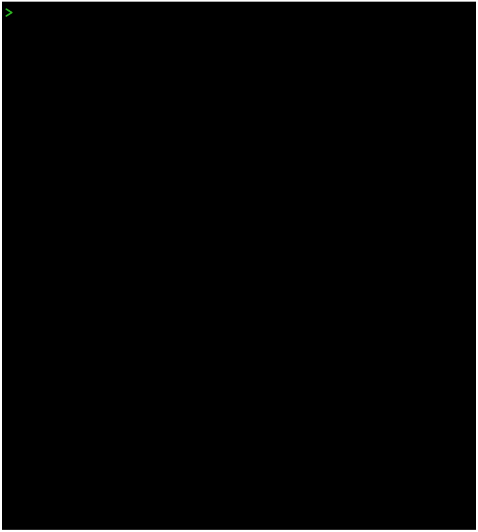
Algebraic representation

$$\pi_{\boldsymbol{x}} = P\left[\boldsymbol{X} = \boldsymbol{x}\right] = \frac{1}{Z(\boldsymbol{\theta})}\boldsymbol{\theta}^{\boldsymbol{a}_{\boldsymbol{x}}}$$

$$\boldsymbol{A} = \begin{array}{c} \\ \theta_1 \\ \theta_2 \\ \theta_3 \\ \theta_4 \end{array} \begin{array}{c} \begin{array}{cccc} \pi_{11} & \pi_{12} & \pi_{21} & \pi_{22} \end{array} \\ \left[\begin{array}{cccc} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{array}\right] \end{array}$$

Compute the moves

ALGEBRAIC REPRESENTATION OF LOG-LINEAR MODELS



```
>
```

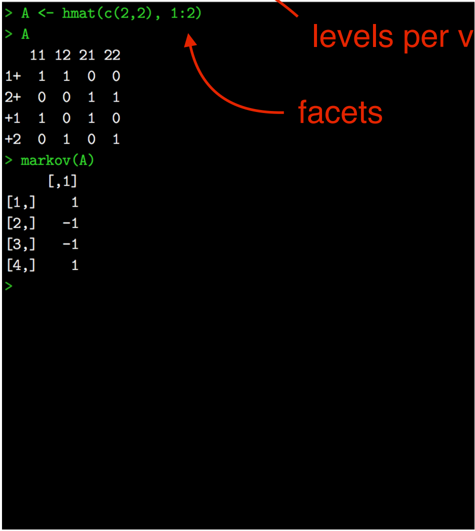
Algebraic representation

$$\pi_{\boldsymbol{x}} = P\left[\boldsymbol{X} = \boldsymbol{x}\right] = \frac{1}{Z(\boldsymbol{\theta})}\boldsymbol{\theta}^{\boldsymbol{a}_{\boldsymbol{x}}}$$

$$\boldsymbol{A} = \boxed{\begin{array}{c|cccc} & \pi_{11} & \pi_{12} & \pi_{21} & \pi_{22} \\ \hline \theta_1 & 1 & 1 & 0 & 0 \\ \theta_2 & 0 & 0 & 1 & 1 \\ \theta_3 & 1 & 0 & 1 & 0 \\ \theta_4 & 0 & 1 & 0 & 1 \end{array}}$$

Compute the moves

ALGEBRAIC REPRESENTATION OF LOG-LINEAR MODELS



```
> A <- hmat(c(2,2), 1:2)
> A
   11 12 21 22
1+  1  1  0  0
2+  0  0  1  1
+1  1  0  1  0
+2  0  1  0  1
> markov(A)
     [,1]
[1,]    1
[2,]   -1
[3,]   -1
[4,]    1
>
```

levels per variable

facets

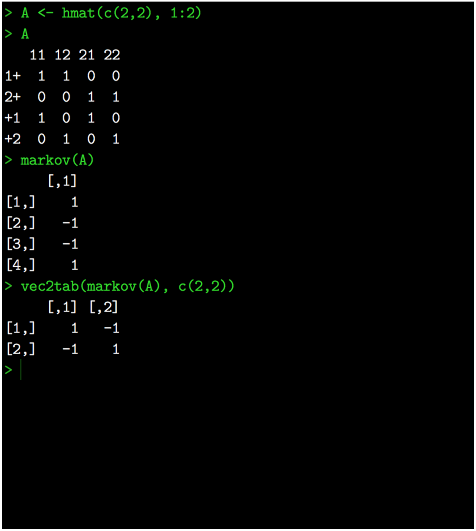
Algebraic representation

$$\pi_{\boldsymbol{x}} = P\left[\boldsymbol{X} = \boldsymbol{x}\right] = \frac{1}{Z(\boldsymbol{\theta})}\boldsymbol{\theta}^{\boldsymbol{a}_{\boldsymbol{x}}}$$

$$\boldsymbol{A} = \begin{array}{c} \\ \theta_1 \\ \theta_2 \\ \theta_3 \\ \theta_4 \end{array} \begin{array}{c} \begin{array}{cccc} \pi_{11} & \pi_{12} & \pi_{21} & \pi_{22} \end{array} \\ \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix} \end{array}$$

Compute the moves

ALGEBRAIC REPRESENTATION OF LOG-LINEAR MODELS



```
> A <- hmat(c(2,2), 1:2)
> A
   11 12 21 22
1+  1  1  0  0
2+  0  0  1  1
+1  1  0  1  0
+2  0  1  0  1
> markov(A)
     [,1]
[1,]    1
[2,]   -1
[3,]   -1
[4,]    1
> vec2tab(markov(A), c(2,2))
     [,1] [,2]
[1,]    1   -1
[2,]   -1    1
> 
```

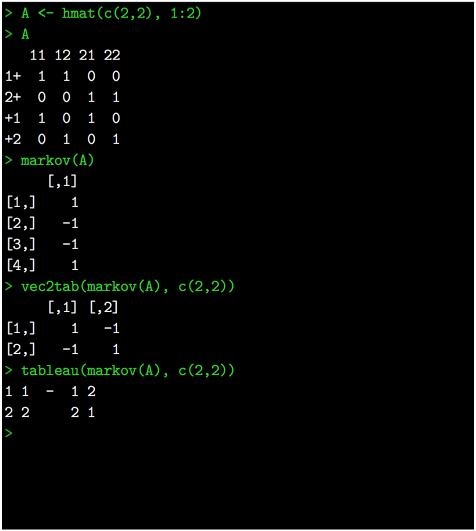
Algebraic representation

$$\pi_{\boldsymbol{x}} = P\left[\boldsymbol{X} = \boldsymbol{x}\right] = \frac{1}{Z(\boldsymbol{\theta})}\boldsymbol{\theta}^{\boldsymbol{a_x}}$$

$$\boldsymbol{A} = \begin{array}{c} \\ \theta_1 \\ \theta_2 \\ \theta_3 \\ \theta_4 \end{array} \begin{array}{cccc} \pi_{11} & \pi_{12} & \pi_{21} & \pi_{22} \\ \end{array} \left[\begin{array}{cccc} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{array}\right]$$

Compute the moves

ALGEBRAIC REPRESENTATION OF LOG-LINEAR MODELS



```
> A <- hmat(c(2,2), 1:2)
> A
   11 12 21 22
1+  1  1  0  0
2+  0  0  1  1
+1  1  0  1  0
+2  0  1  0  1
> markov(A)
     [,1]
[1,]    1
[2,]   -1
[3,]   -1
[4,]    1
> vec2tab(markov(A), c(2,2))
     [,1] [,2]
[1,]    1   -1
[2,]   -1    1
> tableau(markov(A), c(2,2))
1 1  -  1 2
2 2     2 1
>
```

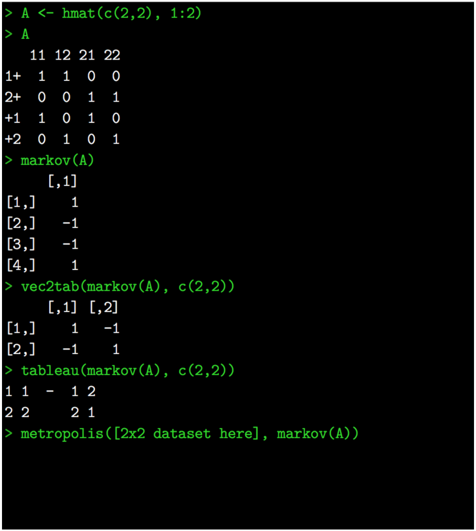
Algebraic representation

$$\pi_{\boldsymbol{x}} = P\left[\boldsymbol{X} = \boldsymbol{x}\right] = \frac{1}{Z(\boldsymbol{\theta})}\boldsymbol{\theta}^{\boldsymbol{a}_{\boldsymbol{x}}}$$

$$\boldsymbol{A} = \boxed{\begin{array}{c|cccc} & \pi_{11} & \pi_{12} & \pi_{21} & \pi_{22} \\ \hline \theta_1 & 1 & 1 & 0 & 0 \\ \theta_2 & 0 & 0 & 1 & 1 \\ \theta_3 & 1 & 0 & 1 & 0 \\ \theta_4 & 0 & 1 & 0 & 1 \end{array}}$$

Compute the moves

ALGEBRAIC REPRESENTATION OF LOG-LINEAR MODELS



```
> A <- hmat(c(2,2), 1:2)
> A
   11 12 21 22
1+  1  1  0  0
2+  0  0  1  1
+1  1  0  1  0
+2  0  1  0  1
> markov(A)
     [,1]
[1,]    1
[2,]   -1
[3,]   -1
[4,]    1
> vec2tab(markov(A), c(2,2))
     [,1] [,2]
[1,]    1   -1
[2,]   -1    1
> tableau(markov(A), c(2,2))
1 1  -  1 2
2 2     2 1
> metropolis([2x2 dataset here], markov(A))
```

Algebraic representation

$$\pi_{\boldsymbol{x}} = P\left[\boldsymbol{X} = \boldsymbol{x}\right] = \frac{1}{Z(\boldsymbol{\theta})}\boldsymbol{\theta}^{\boldsymbol{a_x}}$$

$$\boldsymbol{A} = \begin{array}{c} \\ \theta_1 \\ \theta_2 \\ \theta_3 \\ \theta_4 \end{array} \begin{array}{cccc} \pi_{11} & \pi_{12} & \pi_{21} & \pi_{22} \\ \end{array}\! \begin{array}{c} \\ \left[\begin{array}{cccc} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{array}\right] \end{array}$$

Compute the moves

TESTING INDEPENDENCE IN THE POLITICS DATASET

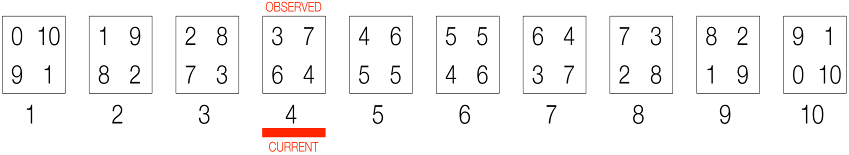
Consider the contingency table

		Political affiliation		
		Dem	Rep	
Personality	Introvert	3	7	10
	Extrovert	6	4	10
		9	11	20

Q : Are personality and political affiliation independent?

THE METROPOLIS ALGORITHM

Moves :  (This table/model only has one move.)



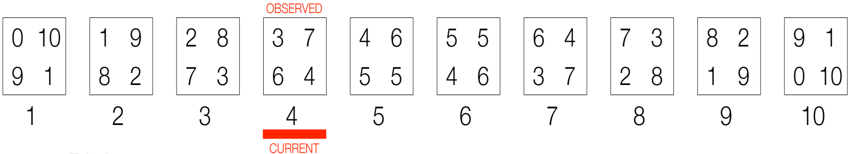
OBSERVED

$$\begin{array}{cccccccccc} \begin{bmatrix}0&10\\9&1\end{bmatrix} & \begin{bmatrix}1&9\\8&2\end{bmatrix} & \begin{bmatrix}2&8\\7&3\end{bmatrix} & \begin{bmatrix}3&7\\6&4\end{bmatrix} & \begin{bmatrix}4&6\\5&5\end{bmatrix} & \begin{bmatrix}5&5\\4&6\end{bmatrix} & \begin{bmatrix}6&4\\3&7\end{bmatrix} & \begin{bmatrix}7&3\\2&8\end{bmatrix} & \begin{bmatrix}8&2\\1&9\end{bmatrix} & \begin{bmatrix}9&1\\0&10\end{bmatrix} \\ 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \end{array}$$

CURRENT

THE METROPOLIS ALGORITHM

Moves : ↷ (This table/model only has one move.)



			OBSERVED						
0 10 9 1	1 9 8 2	2 8 7 3	3 7 6 4	4 6 5 5	5 5 4 6	6 4 3 7	7 3 2 8	8 2 1 9	9 1 0 10
1	2	3	4	5	6	7	8	9	10
			CURRENT						

1. Pick a move (here there's only one)

THE METROPOLIS ALGORITHM

Moves :

(This table/model only has one move.)

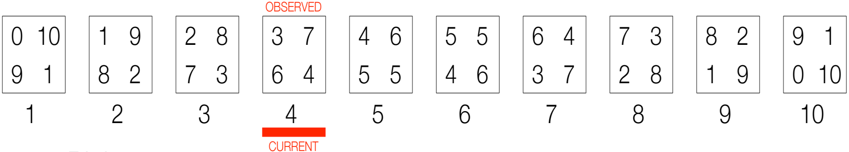
			OBSERVED						
$\begin{array}{cc} 0 & 10 \\ 9 & 1 \end{array}$	$\begin{array}{cc} 1 & 9 \\ 8 & 2 \end{array}$	$\begin{array}{cc} 2 & 8 \\ 7 & 3 \end{array}$	$\begin{array}{cc} 3 & 7 \\ 6 & 4 \end{array}$	$\begin{array}{cc} 4 & 6 \\ 5 & 5 \end{array}$	$\begin{array}{cc} 5 & 5 \\ 4 & 6 \end{array}$	$\begin{array}{cc} 6 & 4 \\ 3 & 7 \end{array}$	$\begin{array}{cc} 7 & 3 \\ 2 & 8 \end{array}$	$\begin{array}{cc} 8 & 2 \\ 1 & 9 \end{array}$	$\begin{array}{cc} 9 & 1 \\ 0 & 10 \end{array}$
1	2	3	4	5	6	7	8	9	10
			CURRENT						

1. Pick a move (here there's only one)
2. Pick a direction +/–

THE METROPOLIS ALGORITHM

Moves :

(This table/model only has one move.)



OBSERVED

0 10 9 1	1 9 8 2	2 8 7 3	3 7 6 4	4 6 5 5	5 5 4 6	6 4 3 7	7 3 2 8	8 2 1 9	9 1 0 10
1	2	3	4	5	6	7	8	9	10

CURRENT

1. Pick a move (here there's only one)
2. Pick a direction +/–
3. Move with a probability p (p is easy to compute; depends on the current and proposed state, but not the big sum)

THE METROPOLIS ALGORITHM

Moves :

(This table/model only has one move.)

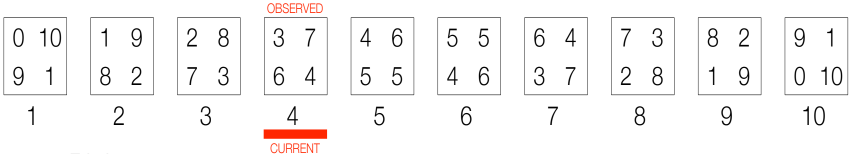
			OBSERVED						
0 10 9 1	1 9 8 2	2 8 7 3	3 7 6 4	4 6 5 5	5 5 4 6	6 4 3 7	7 3 2 8	8 2 1 9	9 1 0 10
1	2	3	4	5	6	7	8	9	10
			CURRENT						

1. Pick a move (here there's only one)
2. Pick a direction +/–
3. Move with a probability p (p is easy to compute; depends on the current and proposed state, but not the big sum)
4. Record your steps

4

THE METROPOLIS ALGORITHM

Moves :  (This table/model only has one move.)



			OBSERVED						
$\begin{array}{cc}0 & 10\\9 & 1\end{array}$	$\begin{array}{cc}1 & 9\\8 & 2\end{array}$	$\begin{array}{cc}2 & 8\\7 & 3\end{array}$	$\begin{array}{cc}3 & 7\\6 & 4\end{array}$	$\begin{array}{cc}4 & 6\\5 & 5\end{array}$	$\begin{array}{cc}5 & 5\\4 & 6\end{array}$	$\begin{array}{cc}6 & 4\\3 & 7\end{array}$	$\begin{array}{cc}7 & 3\\2 & 8\end{array}$	$\begin{array}{cc}8 & 2\\1 & 9\end{array}$	$\begin{array}{cc}9 & 1\\0 & 10\end{array}$
1	2	3	4	5	6	7	8	9	10
			CURRENT						

→ 1. Pick a move (here there's only one)

2. Pick a direction +/–

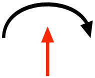
3. Move with a probability p (p is easy to compute; depends on the current and proposed state, but not the big sum)

4. Record your steps

4

THE METROPOLIS ALGORITHM

Moves :



(This table/model only has one move.)

$$\begin{array}{cccccccccc} & & & \text{OBSERVED} & & & & & & \\ \boxed{\begin{matrix}0 & 10\\ 9 & 1\end{matrix}} & \boxed{\begin{matrix}1 & 9\\ 8 & 2\end{matrix}} & \boxed{\begin{matrix}2 & 8\\ 7 & 3\end{matrix}} & \boxed{\begin{matrix}3 & 7\\ 6 & 4\end{matrix}} & \boxed{\begin{matrix}4 & 6\\ 5 & 5\end{matrix}} & \boxed{\begin{matrix}5 & 5\\ 4 & 6\end{matrix}} & \boxed{\begin{matrix}6 & 4\\ 3 & 7\end{matrix}} & \boxed{\begin{matrix}7 & 3\\ 2 & 8\end{matrix}} & \boxed{\begin{matrix}8 & 2\\ 1 & 9\end{matrix}} & \boxed{\begin{matrix}9 & 1\\ 0 & 10\end{matrix}} \\ 1 & 2 & 3 & \underline{4} & 5 & 6 & 7 & 8 & 9 & 10 \\ & & & \text{CURRENT} & & & & & & \end{array}$$

→ 1. Pick a move (here there's only one)

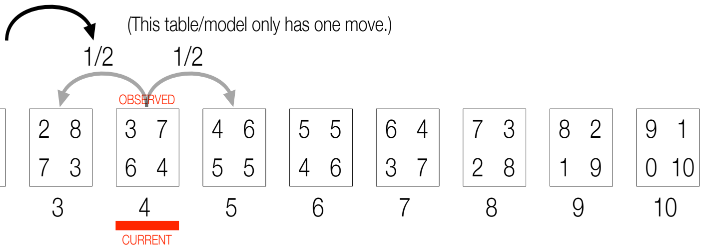
2. Pick a direction +/–

3. Move with a probability p (p is easy to compute; depends on the current and proposed state, but not the big sum)

4. Record your steps

4

THE METROPOLIS ALGORITHM



Moves : (This table/model only has one move.)

1/2 1/2

OBSERVED

0 10 9 1	1 9 8 2	2 8 7 3	3 7 6 4	4 6 5 5	5 5 4 6	6 4 3 7	7 3 2 8	8 2 1 9	9 1 0 10
1	2	3	4	5	6	7	8	9	10

CURRENT

1. Pick a move (here there's only one)

→ 2. Pick a direction +/–

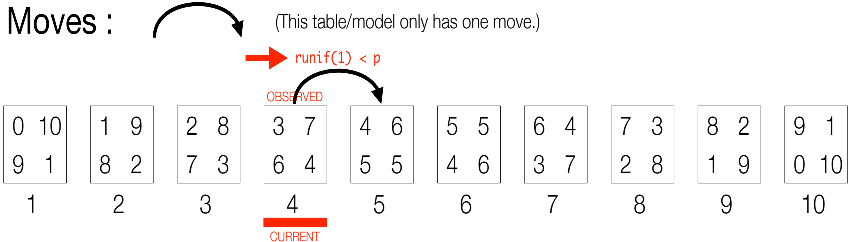
3. Move with a probability p (p is easy to compute; depends on the current and proposed state, but not the big sum)

4. Record your steps

4

THE METROPOLIS ALGORITHM

Moves :



(This table/model only has one move.)

`runif(1) < p`

OBSERVED

0 10	1 9	2 8	3 7	4 6	5 5	6 4	7 3	8 2	9 1
9 1	8 2	7 3	6 4	5 5	4 6	3 7	2 8	1 9	0 10
1	2	3	4	5	6	7	8	9	10

CURRENT

1. Pick a move (here there's only one)
2. Pick a direction +/–
3. Move with a probability p (p is easy to compute; depends on the current and proposed state, but not the big sum)
4. Record your steps

4

THE METROPOLIS ALGORITHM

Moves : ↷ (This table/model only has one move.)

			OBSERVED	✓					
$\begin{matrix}0 & 10\\ 9 & 1\end{matrix}$	$\begin{matrix}1 & 9\\ 8 & 2\end{matrix}$	$\begin{matrix}2 & 8\\ 7 & 3\end{matrix}$	$\begin{matrix}3 & 7\\ 6 & 4\end{matrix}$	$\begin{matrix}4 & 6\\ 5 & 5\end{matrix}$	$\begin{matrix}5 & 5\\ 4 & 6\end{matrix}$	$\begin{matrix}6 & 4\\ 3 & 7\end{matrix}$	$\begin{matrix}7 & 3\\ 2 & 8\end{matrix}$	$\begin{matrix}8 & 2\\ 1 & 9\end{matrix}$	$\begin{matrix}9 & 1\\ 0 & 10\end{matrix}$
1	2	3	4	5	6	7	8	9	10
				CURRENT					

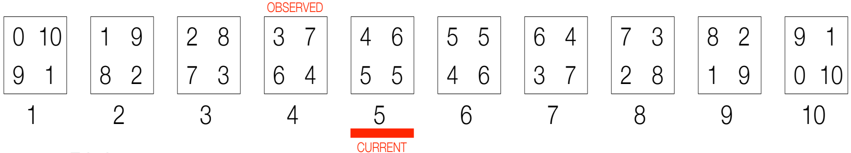
1. Pick a move (here there's only one)
2. Pick a direction +/–
3. Move with a probability p (p is easy to compute; depends on the current and proposed state, but not the big sum)
4. → Record your steps

4, 5

THE METROPOLIS ALGORITHM

Moves :

(This table/model only has one move.)



			OBSERVED						
0 10	1 9	2 8	3 7	4 6	5 5	6 4	7 3	8 2	9 1
9 1	8 2	7 3	6 4	5 5	4 6	3 7	2 8	1 9	0 10
1	2	3	4	5	6	7	8	9	10
				CURRENT					

→ 1. Pick a move (here there's only one)

2. Pick a direction +/–

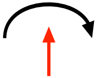
3. Move with a probability p (p is easy to compute; depends on the current and proposed state, but not the big sum)

4. Record your steps

4, 5

THE METROPOLIS ALGORITHM

Moves :



(This table/model only has one move.)

			OBSERVED						
0 10 9 1	1 9 8 2	2 8 7 3	3 7 6 4	4 6 5 5	5 5 4 6	6 4 3 7	7 3 2 8	8 2 1 9	9 1 0 10
1	2	3	4	5	6	7	8	9	10
				CURRENT					

→ 1. Pick a move (here there's only one)

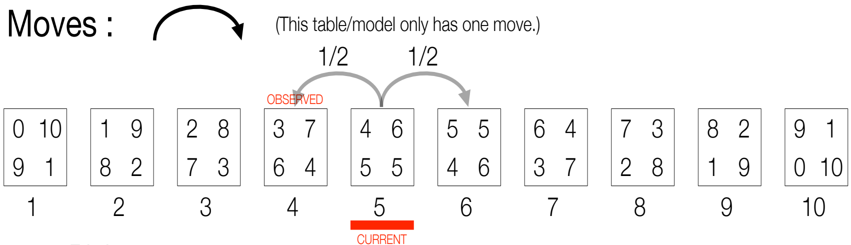
2. Pick a direction +/–

3. Move with a probability p (p is easy to compute; depends on the current and proposed state, but not the big sum)

4. Record your steps

4, 5

THE METROPOLIS ALGORITHM



Moves :

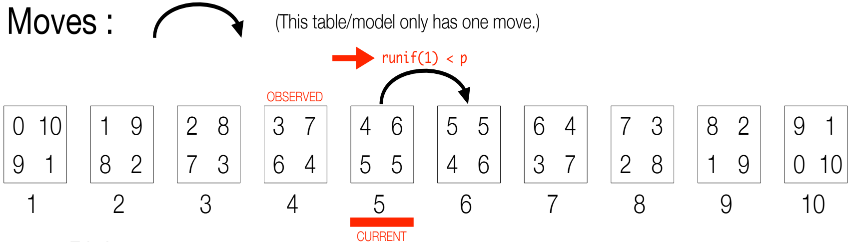
(This table/model only has one move.)

0 10 9 1	1 9 8 2	2 8 7 3	3 7 6 4	4 6 5 5	5 5 4 6	6 4 3 7	7 3 2 8	8 2 1 9	9 1 0 10
1	2	3	4	5	6	7	8	9	10

1. Pick a move (here there's only one)
2. Pick a direction +/–
3. Move with a probability p (p is easy to compute; depends on the current and proposed state, but not the big sum)
4. Record your steps

   4, 5

THE METROPOLIS ALGORITHM



Moves :

(This table/model only has one move.)

`runif(1) < p`

OBSERVED

0 10 9 1	1 9 8 2	2 8 7 3	3 7 6 4	4 6 5 5	5 5 4 6	6 4 3 7	7 3 2 8	8 2 1 9	9 1 0 10
1	2	3	4	5	6	7	8	9	10

CURRENT

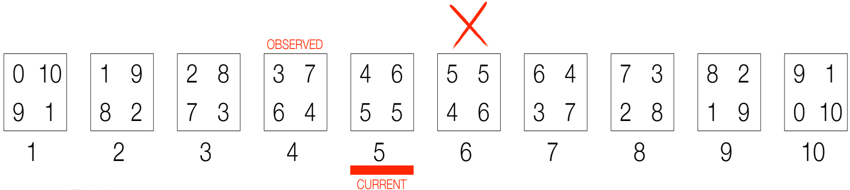
1. Pick a move (here there's only one)
2. Pick a direction +/–
3. Move with a probability p (p is easy to compute; depends on the current and proposed state, but not the big sum)
4. Record your steps

4, 5

THE METROPOLIS ALGORITHM

Moves :

(This table/model only has one move.)



OBSERVED: 4

CURRENT: 5

$$\underset{1}{\begin{bmatrix}0&10\\9&1\end{bmatrix}}\quad\underset{2}{\begin{bmatrix}1&9\\8&2\end{bmatrix}}\quad\underset{3}{\begin{bmatrix}2&8\\7&3\end{bmatrix}}\quad\underset{4}{\begin{bmatrix}3&7\\6&4\end{bmatrix}}\quad\underset{5}{\begin{bmatrix}4&6\\5&5\end{bmatrix}}\quad\underset{6}{\begin{bmatrix}5&5\\4&6\end{bmatrix}}\quad\underset{7}{\begin{bmatrix}6&4\\3&7\end{bmatrix}}\quad\underset{8}{\begin{bmatrix}7&3\\2&8\end{bmatrix}}\quad\underset{9}{\begin{bmatrix}8&2\\1&9\end{bmatrix}}\quad\underset{10}{\begin{bmatrix}9&1\\0&10\end{bmatrix}}$$

1. Pick a move (here there's only one)
2. Pick a direction +/–
3. Move with a probability p (p is easy to compute; depends on the current and proposed state, but not the big sum)
4. Record your steps

4, 5, 5

THE METROPOLIS ALGORITHM

Moves :

(This table/model only has one move.)

$$\begin{array}{cccccccccc}
 & & & \text{OBSERVED} & & & & & & \\
\boxed{\begin{matrix}0 & 10\\ 9 & 1\end{matrix}} & \boxed{\begin{matrix}1 & 9\\ 8 & 2\end{matrix}} & \boxed{\begin{matrix}2 & 8\\ 7 & 3\end{matrix}} & \boxed{\begin{matrix}3 & 7\\ 6 & 4\end{matrix}} & \boxed{\begin{matrix}4 & 6\\ 5 & 5\end{matrix}} & \boxed{\begin{matrix}5 & 5\\ 4 & 6\end{matrix}} & \boxed{\begin{matrix}6 & 4\\ 3 & 7\end{matrix}} & \boxed{\begin{matrix}7 & 3\\ 2 & 8\end{matrix}} & \boxed{\begin{matrix}8 & 2\\ 1 & 9\end{matrix}} & \boxed{\begin{matrix}9 & 1\\ 0 & 10\end{matrix}} \\
1 & 2 & 3 & 4 & \underline{5} & 6 & 7 & 8 & 9 & 10 \\
 & & & & \text{CURRENT} & & & & &
\end{array}$$

→ 1. Pick a move (here there's only one)

2. Pick a direction +/–

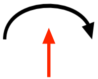
3. Move with a probability p (p is easy to compute; depends on the current and proposed state, but not the big sum)

4. Record your steps

4, 5, 5

THE METROPOLIS ALGORITHM

Moves :



(This table/model only has one move.)

			OBSERVED						
$\begin{matrix}0 & 10\\9 & 1\end{matrix}$	$\begin{matrix}1 & 9\\8 & 2\end{matrix}$	$\begin{matrix}2 & 8\\7 & 3\end{matrix}$	$\begin{matrix}3 & 7\\6 & 4\end{matrix}$	$\begin{matrix}4 & 6\\5 & 5\end{matrix}$	$\begin{matrix}5 & 5\\4 & 6\end{matrix}$	$\begin{matrix}6 & 4\\3 & 7\end{matrix}$	$\begin{matrix}7 & 3\\2 & 8\end{matrix}$	$\begin{matrix}8 & 2\\1 & 9\end{matrix}$	$\begin{matrix}9 & 1\\0 & 10\end{matrix}$
1	2	3	4	5	6	7	8	9	10
				CURRENT					

→ 1. Pick a move (here there's only one)

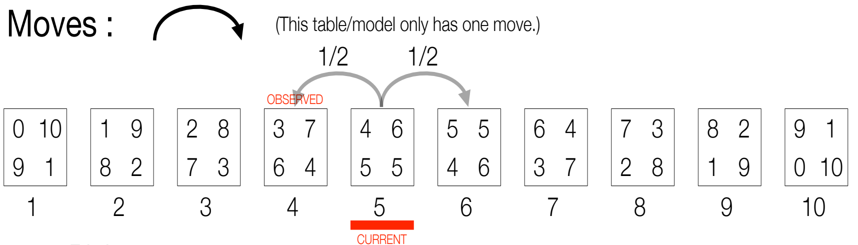
2. Pick a direction +/–

3. Move with a probability p (p is easy to compute; depends on the current and proposed state, but not the big sum)

4. Record your steps

4, 5, 5

THE METROPOLIS ALGORITHM



Moves : (This table/model only has one move.)

1/2 1/2

OBSERVED

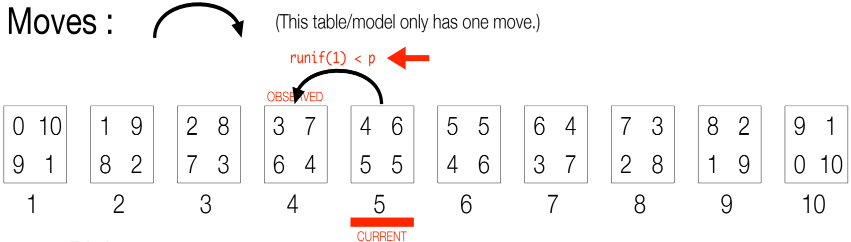
$\begin{matrix}0 & 10\\9 & 1\end{matrix}$	$\begin{matrix}1 & 9\\8 & 2\end{matrix}$	$\begin{matrix}2 & 8\\7 & 3\end{matrix}$	$\begin{matrix}3 & 7\\6 & 4\end{matrix}$	$\begin{matrix}4 & 6\\5 & 5\end{matrix}$	$\begin{matrix}5 & 5\\4 & 6\end{matrix}$	$\begin{matrix}6 & 4\\3 & 7\end{matrix}$	$\begin{matrix}7 & 3\\2 & 8\end{matrix}$	$\begin{matrix}8 & 2\\1 & 9\end{matrix}$	$\begin{matrix}9 & 1\\0 & 10\end{matrix}$
1	2	3	4	5	6	7	8	9	10

CURRENT

1. Pick a move (here there's only one)
2. Pick a direction +/–
3. Move with a probability p (p is easy to compute; depends on the current and proposed state, but not the big sum)
4. Record your steps

4, 5, 5

THE METROPOLIS ALGORITHM



Moves :

(This table/model only has one move.)

`runif(1) < p`

OBSERVED

0 10 9 1	1 9 8 2	2 8 7 3	3 7 6 4	4 6 5 5	5 5 4 6	6 4 3 7	7 3 2 8	8 2 1 9	9 1 0 10
1	2	3	4	5	6	7	8	9	10

CURRENT

1. Pick a move (here there's only one)
2. Pick a direction +/–
3. Move with a probability p (p is easy to compute; depends on the current and proposed state, but not the big sum)
4. Record your steps

4, 5, 5

THE METROPOLIS ALGORITHM

Moves :

(This table/model only has one move.)

			OBSERVED ✓						
$\begin{matrix}0&10\\9&1\end{matrix}$	$\begin{matrix}1&9\\8&2\end{matrix}$	$\begin{matrix}2&8\\7&3\end{matrix}$	$\begin{matrix}3&7\\6&4\end{matrix}$	$\begin{matrix}4&6\\5&5\end{matrix}$	$\begin{matrix}5&5\\4&6\end{matrix}$	$\begin{matrix}6&4\\3&7\end{matrix}$	$\begin{matrix}7&3\\2&8\end{matrix}$	$\begin{matrix}8&2\\1&9\end{matrix}$	$\begin{matrix}9&1\\0&10\end{matrix}$
1	2	3	4	5	6	7	8	9	10
			CURRENT						

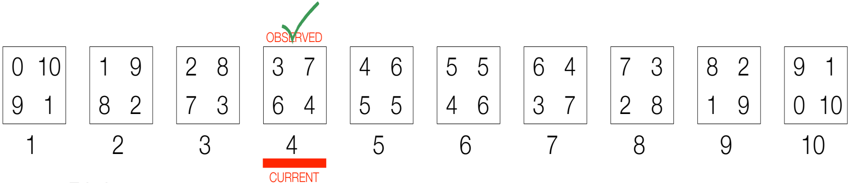
1. Pick a move (here there's only one)
2. Pick a direction +/–
3. Move with a probability p (p is easy to compute; depends on the current and proposed state, but not the big sum)
4. → Record your steps

4, 5, 5, 4

THE METROPOLIS ALGORITHM

Moves :

(This table/model only has one move.)



$\begin{matrix}0 & 10\\ 9 & 1\end{matrix}$	$\begin{matrix}1 & 9\\ 8 & 2\end{matrix}$	$\begin{matrix}2 & 8\\ 7 & 3\end{matrix}$	$\begin{matrix}3 & 7\\ 6 & 4\end{matrix}$	$\begin{matrix}4 & 6\\ 5 & 5\end{matrix}$	$\begin{matrix}5 & 5\\ 4 & 6\end{matrix}$	$\begin{matrix}6 & 4\\ 3 & 7\end{matrix}$	$\begin{matrix}7 & 3\\ 2 & 8\end{matrix}$	$\begin{matrix}8 & 2\\ 1 & 9\end{matrix}$	$\begin{matrix}9 & 1\\ 0 & 10\end{matrix}$
1	2	3	4	5	6	7	8	9	10

1. Pick a move (here there's only one)
2. Pick a direction +/–
3. Move with a probability p (p is easy to compute; depends on the current and proposed state, but not the big sum)
4. Record your steps

4, 5, 5, 4,

THE METROPOLIS ALGORITHM

Moves : ↷ (This table/model only has one move.)

$$\begin{array}{cccccccccc}
 & & & \checkmark\ \text{OBSERVED} & & & & & & \\
\boxed{\begin{matrix}0 & 10\\ 9 & 1\end{matrix}} & \boxed{\begin{matrix}1 & 9\\ 8 & 2\end{matrix}} & \boxed{\begin{matrix}2 & 8\\ 7 & 3\end{matrix}} & \boxed{\begin{matrix}3 & 7\\ 6 & 4\end{matrix}} & \boxed{\begin{matrix}4 & 6\\ 5 & 5\end{matrix}} & \boxed{\begin{matrix}5 & 5\\ 4 & 6\end{matrix}} & \boxed{\begin{matrix}6 & 4\\ 3 & 7\end{matrix}} & \boxed{\begin{matrix}7 & 3\\ 2 & 8\end{matrix}} & \boxed{\begin{matrix}8 & 2\\ 1 & 9\end{matrix}} & \boxed{\begin{matrix}9 & 1\\ 0 & 10\end{matrix}} \\
1 & 2 & 3 & \underline{4} & 5 & 6 & 7 & 8 & 9 & 10 \\
 & & & \text{CURRENT} & & & & & &
\end{array}$$

1. Pick a move (here there's only one)
2. Pick a direction +/–
3. Move with a probability p (p is easy to compute; depends on the current and proposed state, but not the big sum)
4. Record your steps

4, 5, 5, 4, 4, 5, 4, 7, 5, 7, 7, 5, 5, 6, 5, 6, 5, 4, 4, 4, 6,
7, 6, 6, 7, 5, 6, 5, 7, 5, 6, 6, 6, 5, 5, 6, 3, 5, 5, 4, 6,
5, 6, 4, 6, 4, 4, 3, 5, 5, 6, 5, 7, 7, 4, 5, 5, 5, 5, 6, 4, ...

THE METROPOLIS ALGORITHM

Moves : ↷ (This table/model only has one move.)

			OBSERVED						
$\begin{array}{cc} 0 & 10 \\ 9 & 1 \end{array}$	$\begin{array}{cc} 1 & 9 \\ 8 & 2 \end{array}$	$\begin{array}{cc} 2 & 8 \\ 7 & 3 \end{array}$	$\begin{array}{cc} 3 & 7 \\ 6 & 4 \end{array}$	$\begin{array}{cc} 4 & 6 \\ 5 & 5 \end{array}$	$\begin{array}{cc} 5 & 5 \\ 4 & 6 \end{array}$	$\begin{array}{cc} 6 & 4 \\ 3 & 7 \end{array}$	$\begin{array}{cc} 7 & 3 \\ 2 & 8 \end{array}$	$\begin{array}{cc} 8 & 2 \\ 1 & 9 \end{array}$	$\begin{array}{cc} 9 & 1 \\ 0 & 10 \end{array}$
1	2	3	4	5	6	7	8	9	10

THE METROPOLIS ALGORITHM

Moves :  (This table/model only has one move.)

			OBSERVED						
0 10 9 1	1 9 8 2	2 8 7 3	3 7 6 4	4 6 5 5	5 5 4 6	6 4 3 7	7 3 2 8	8 2 1 9	9 1 0 10
1	2	3	4	5	6	7	8	9	10
1	24	333	1470	3149	3173	1492	339	18	1

THE METROPOLIS ALGORITHM

Moves :

(This table/model only has one move.)

			OBSERVED						
0 10 9 1	1 9 8 2	2 8 7 3	3 7 6 4	4 6 5 5	5 5 4 6	6 4 3 7	7 3 2 8	8 2 1 9	9 1 0 10
1	2	3	4	5	6	7	8	9	10
1	24	333	1470	3149	3173	1492	339	18	1
-27.9	-24.1	-21.6	-20.1	-19.3	-19.3	-20.1	-21.6	-24.1	-27.9

Un-normalized log-likelihoods

THE METROPOLIS ALGORITHM

Moves :  (This table/model only has one move.)

			OBSERVED						
0 10 9 1	1 9 8 2	2 8 7 3	3 7 6 4	4 6 5 5	5 5 4 6	6 4 3 7	7 3 2 8	8 2 1 9	9 1 0 10
1	2	3	4	5	6	7	8	9	10
1	24	333	1470	3149	3173	1492	339	18	1
-27.9	-24.1	-21.6	-20.1	-19.3	-19.3	-20.1	-21.6	-24.1	-27.9

Un-normalized log-likelihoods = Same order as the probabilities

THE METROPOLIS ALGORITHM

Moves :

(This table/model only has one move.)

0 10 9 1	1 9 8 2	2 8 7 3	OBSERVED 3 7 6 4	4 6 5 5	5 5 4 6	6 4 3 7	7 3 2 8	8 2 1 9	9 1 0 10
1	2	3	4	5	6	7	8	9	10
1	24	333	1470	3149	3173	1492	339	18	1
-27.9	-24.1	-21.6	-20.1	-19.3	-19.3	-20.1	-21.6	-24.1	-27.9

Un-normalized log-likelihoods = Same order as the probabilities

THE METROPOLIS ALGORITHM

Moves :

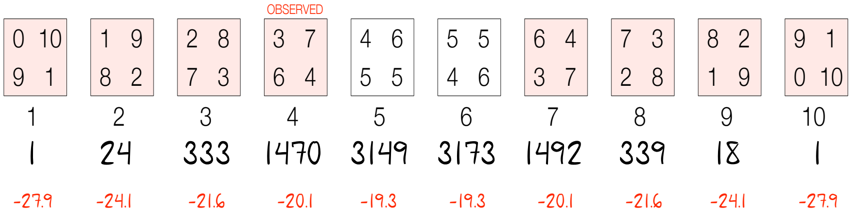
(This table/model only has one move.)

			OBSERVED						
0 10 9 1	1 9 8 2	2 8 7 3	3 7 6 4	4 6 5 5	5 5 4 6	6 4 3 7	7 3 2 8	8 2 1 9	9 1 0 10
1	2	3	4	5	6	7	8	9	10
1	24	333	1470	3149	3173	1492	339	18	1
-27.9	-24.1	-21.6	-20.1	-19.3	-19.3	-20.1	-21.6	-24.1	-27.9

Un-normalized log-likelihoods = Same order as the probabilities

p-value $\approx$ % of samples with un-normalized log-likelihoods $\leq$ observed table

THE METROPOLIS ALGORITHM



			OBSERVED						
0 10 9 1	1 9 8 2	2 8 7 3	3 7 6 4	4 6 5 5	5 5 4 6	6 4 3 7	7 3 2 8	8 2 1 9	9 1 0 10
1	2	3	4	5	6	7	8	9	10
1	24	333	1470	3149	3173	1492	339	18	1
-27.9	-24.1	-21.6	-20.1	-19.3	-19.3	-20.1	-21.6	-24.1	-27.9

p-value $\approx$ % of samples with un-normalized log-likelihoods $\leq$ observed table

THE METROPOLIS ALGORITHM

			OBSERVED						
$\begin{matrix}0 & 10\\ 9 & 1\end{matrix}$	$\begin{matrix}1 & 9\\ 8 & 2\end{matrix}$	$\begin{matrix}2 & 8\\ 7 & 3\end{matrix}$	$\begin{matrix}3 & 7\\ 6 & 4\end{matrix}$	$\begin{matrix}4 & 6\\ 5 & 5\end{matrix}$	$\begin{matrix}5 & 5\\ 4 & 6\end{matrix}$	$\begin{matrix}6 & 4\\ 3 & 7\end{matrix}$	$\begin{matrix}7 & 3\\ 2 & 8\end{matrix}$	$\begin{matrix}8 & 2\\ 1 & 9\end{matrix}$	$\begin{matrix}9 & 1\\ 0 & 10\end{matrix}$
1	2	3	4	5	6	7	8	9	10
1	24	333	1470	3149	3173	1492	339	18	1
-27.9	-24.1	-21.6	-20.1	-19.3	-19.3	-20.1	-21.6	-24.1	-27.9

p-value $\approx$ % of samples with un-normalized log-likelihoods $\leq$ observed table

$$\frac{1 + 24 + 333 + 1470 + 1492 + 339 + 18 + 1}{10000} = .3778$$

THE METROPOLIS ALGORITHM

			OBSERVED						
$\begin{matrix}0&10\\9&1\end{matrix}$	$\begin{matrix}1&9\\8&2\end{matrix}$	$\begin{matrix}2&8\\7&3\end{matrix}$	$\begin{matrix}3&7\\6&4\end{matrix}$	$\begin{matrix}4&6\\5&5\end{matrix}$	$\begin{matrix}5&5\\4&6\end{matrix}$	$\begin{matrix}6&4\\3&7\end{matrix}$	$\begin{matrix}7&3\\2&8\end{matrix}$	$\begin{matrix}8&2\\1&9\end{matrix}$	$\begin{matrix}9&1\\0&10\end{matrix}$
1	2	3	4	5	6	7	8	9	10
1	24	333	1470	3149	3173	1492	339	18	1
-27.9	-24.1	-21.6	-20.1	-19.3	-19.3	-20.1	-21.6	-24.1	-27.9

p-value $\approx$ % of samples with un-normalized log-likelihoods $\leq$ observed table

$$\frac{1 + 24 + 333 + 1470 + 1492 + 339 + 18 + 1}{10000} = .3778$$

From a previous slide...

2. Compute the p-value by summing the probabilities of the tables with smaller probabilities

$$p - \text{value} = .3698$$

THE METROPOLIS ALGORITHM

			OBSERVED						
$\begin{matrix}0&10\\9&1\end{matrix}$	$\begin{matrix}1&9\\8&2\end{matrix}$	$\begin{matrix}2&8\\7&3\end{matrix}$	$\begin{matrix}3&7\\6&4\end{matrix}$	$\begin{matrix}4&6\\5&5\end{matrix}$	$\begin{matrix}5&5\\4&6\end{matrix}$	$\begin{matrix}6&4\\3&7\end{matrix}$	$\begin{matrix}7&3\\2&8\end{matrix}$	$\begin{matrix}8&2\\1&9\end{matrix}$	$\begin{matrix}9&1\\0&10\end{matrix}$
1	2	3	4	5	6	7	8	9	10
1	24	333	1470	3149	3173	1492	339	18	1
-27.9	-24.1	-21.6	-20.1	-19.3	-19.3	-20.1	-21.6	-24.1	-27.9

p-value $\approx$ % of samples with un-normalized log-likelihoods $\leq$ observed table

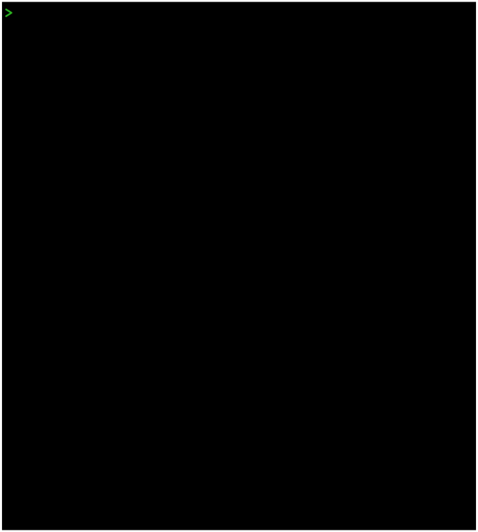
$$\frac{1 + 24 + 333 + 1470 + 1492 + 339 + 18 + 1}{10000} = \boxed{.3778}$$

From a previous slide...

2. Compute the p-value by summing the probabilities of the tables with smaller probabilities

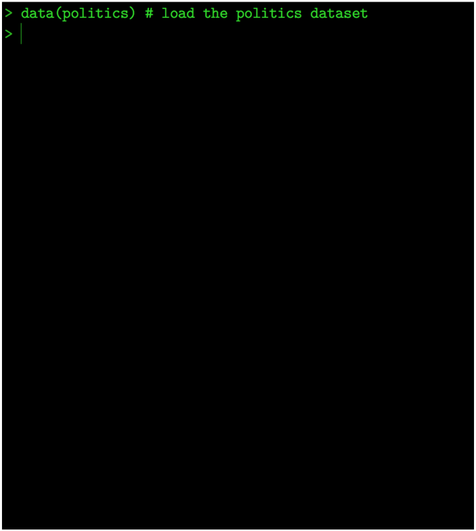
$\boxed{p - \text{value} = .3698}$ Equal to Monte Carlo error

HOW R AND ALGSTAT DO IT



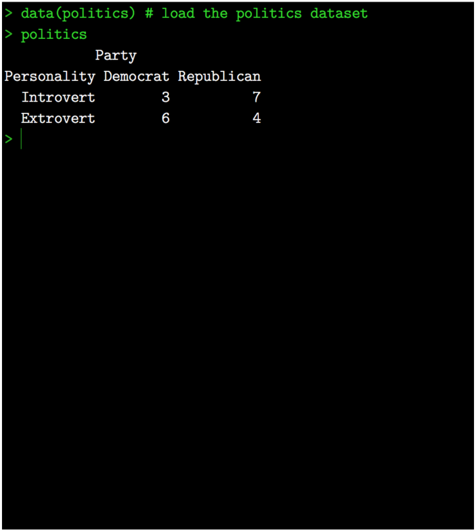
```
>
```

HOW R AND ALGSTAT DO IT



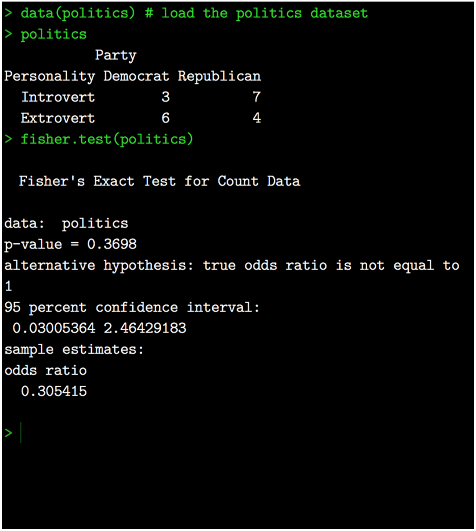
```
> data(politics) # load the politics dataset
> 
```

HOW R AND ALGSTAT DO IT



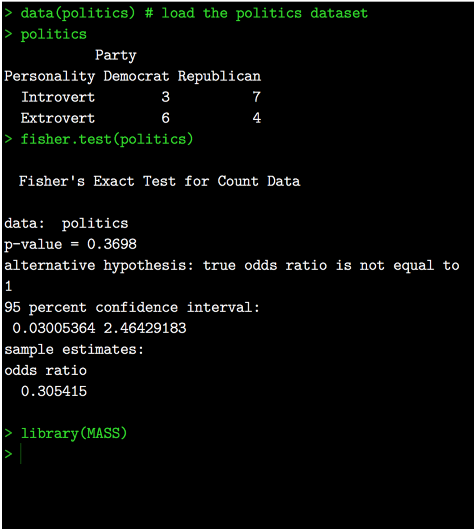
```
> data(politics) # load the politics dataset
> politics
          Party
Personality Democrat Republican
  Introvert        3          7
  Extrovert        6          4
> 
```

HOW R AND ALGSTAT DO IT



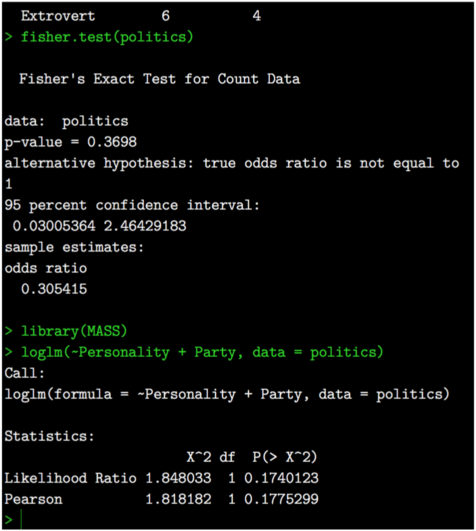
```
> data(politics) # load the politics dataset
> politics
           Party
Personality Democrat Republican
  Introvert        3          7
  Extrovert        6          4
> fisher.test(politics)

	Fisher's Exact Test for Count Data

data:  politics
p-value = 0.3698
alternative hypothesis: true odds ratio is not equal to
1
95 percent confidence interval:
 0.03005364 2.46429183
sample estimates:
odds ratio
  0.305415

> 
```

How R and algstat do it



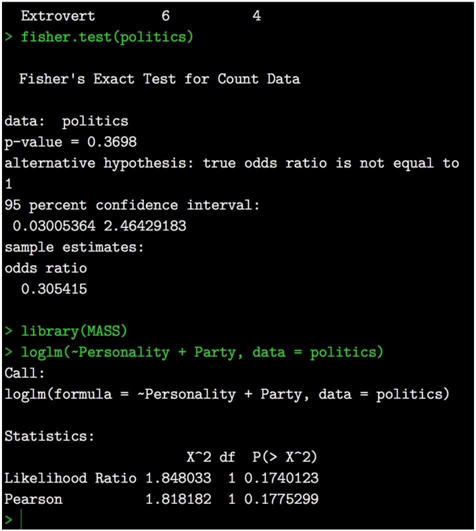
```
> data(politics) # load the politics dataset
> politics
           Party
Personality Democrat Republican
  Introvert        3          7
  Extrovert        6          4
> fisher.test(politics)

	Fisher's Exact Test for Count Data

data:  politics
p-value = 0.3698
alternative hypothesis: true odds ratio is not equal to
1
95 percent confidence interval:
 0.03005364 2.46429183
sample estimates:
odds ratio
  0.305415

> library(MASS)
> 
```

HOW R AND ALGSTAT DO IT

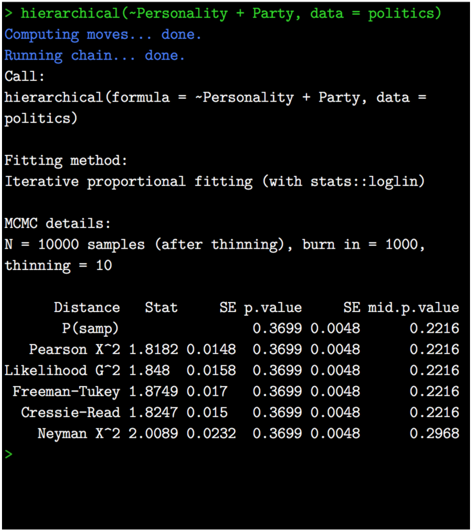


```
  Extrovert        6          4
> fisher.test(politics)

  Fisher's Exact Test for Count Data

data:  politics
p-value = 0.3698
alternative hypothesis: true odds ratio is not equal to
1
95 percent confidence interval:
 0.03005364 2.46429183
sample estimates:
odds ratio
  0.305415

> library(MASS)
> loglm(~Personality + Party, data = politics)
Call:
loglm(formula = ~Personality + Party, data = politics)

Statistics:
                      X^2 df  P(> X^2)
Likelihood Ratio 1.848033  1 0.1740123
Pearson          1.818182  1 0.1775299
> 
```

HOW R AND ALGSTAT DO IT

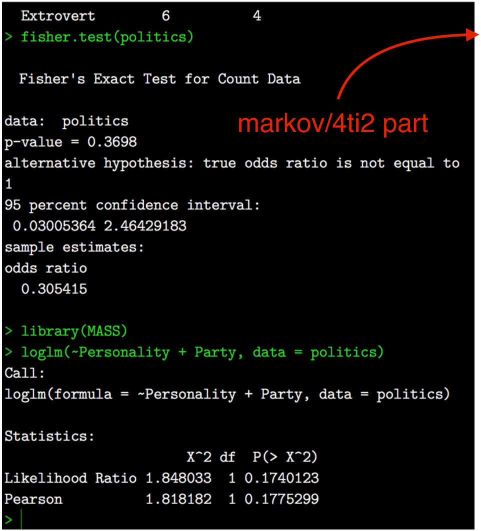


```
  Extrovert        6          4
> fisher.test(politics)

  Fisher's Exact Test for Count Data

data:  politics
p-value = 0.3698
alternative hypothesis: true odds ratio is not equal to
1
95 percent confidence interval:
 0.03005364 2.46429183
sample estimates:
odds ratio
  0.305415

> library(MASS)
> loglm(~Personality + Party, data = politics)
Call:
loglm(formula = ~Personality + Party, data = politics)

Statistics:
                      X^2 df  P(> X^2)
Likelihood Ratio 1.848033  1 0.1740123
Pearson          1.818182  1 0.1775299
>
```



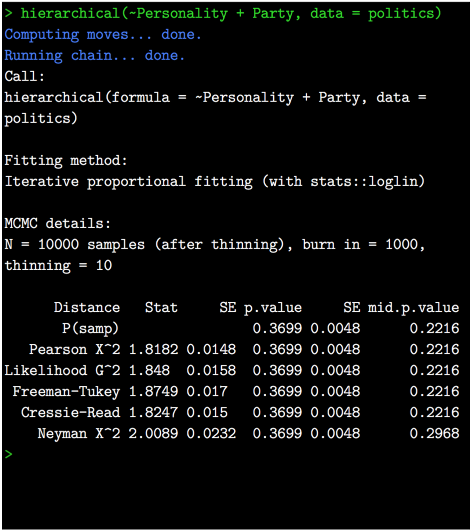
```
> hierarchical(~Personality + Party, data = politics)
Computing moves... done.
Running chain... done.
Call:
hierarchical(formula = ~Personality + Party, data =
politics)

Fitting method:
Iterative proportional fitting (with stats::loglin)

MCMC details:
N = 10000 samples (after thinning), burn in = 1000,
thinning = 10

      Distance   Stat     SE p.value     SE mid.p.value
       P(samp)               0.3699 0.0048       0.2216
   Pearson X^2 1.8182 0.0148 0.3699 0.0048       0.2216
Likelihood G^2 1.848  0.0158 0.3699 0.0048       0.2216
 Freeman-Tukey 1.8749 0.017  0.3699 0.0048       0.2216
   Cressie-Read 1.8247 0.015  0.3699 0.0048       0.2216
    Neyman X^2 2.0089 0.0232 0.3699 0.0048       0.2968
>
```

HOW R AND ALGSTAT DO IT



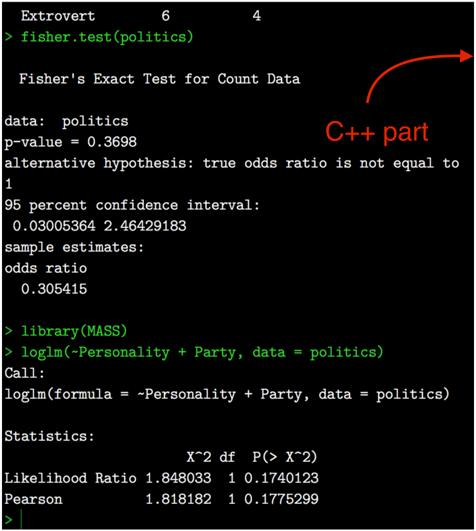
```
  Extrovert        6          4
> fisher.test(politics)

  Fisher's Exact Test for Count Data

data:  politics
p-value = 0.3698
alternative hypothesis: true odds ratio is not equal to
1
95 percent confidence interval:
 0.03005364 2.46429183
sample estimates:
odds ratio
  0.305415

> library(MASS)
> loglm(~Personality + Party, data = politics)
Call:
loglm(formula = ~Personality + Party, data = politics)

Statistics:
                      X^2 df  P(> X^2)
Likelihood Ratio 1.848033  1 0.1740123
Pearson          1.818182  1 0.1775299
>
```

markov/4ti2 part



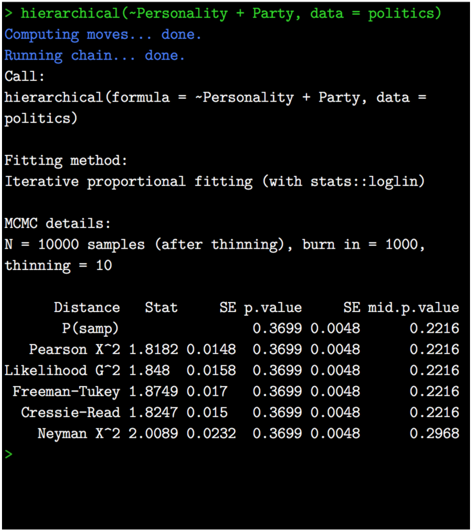
```
> hierarchical(~Personality + Party, data = politics)
Computing moves... done.
Running chain... done.
Call:
hierarchical(formula = ~Personality + Party, data =
politics)

Fitting method:
Iterative proportional fitting (with stats::loglin)

MCMC details:
N = 10000 samples (after thinning), burn in = 1000,
thinning = 10

      Distance   Stat     SE p.value     SE mid.p.value
       P(samp)               0.3699 0.0048      0.2216
   Pearson X^2 1.8182 0.0148 0.3699 0.0048      0.2216
Likelihood G^2 1.848  0.0158 0.3699 0.0048      0.2216
 Freeman-Tukey 1.8749 0.017  0.3699 0.0048      0.2216
  Cressie-Read 1.8247 0.015  0.3699 0.0048      0.2216
    Neyman X^2 2.0089 0.0232 0.3699 0.0048      0.2968
>
```

HOW R AND ALGSTAT DO IT



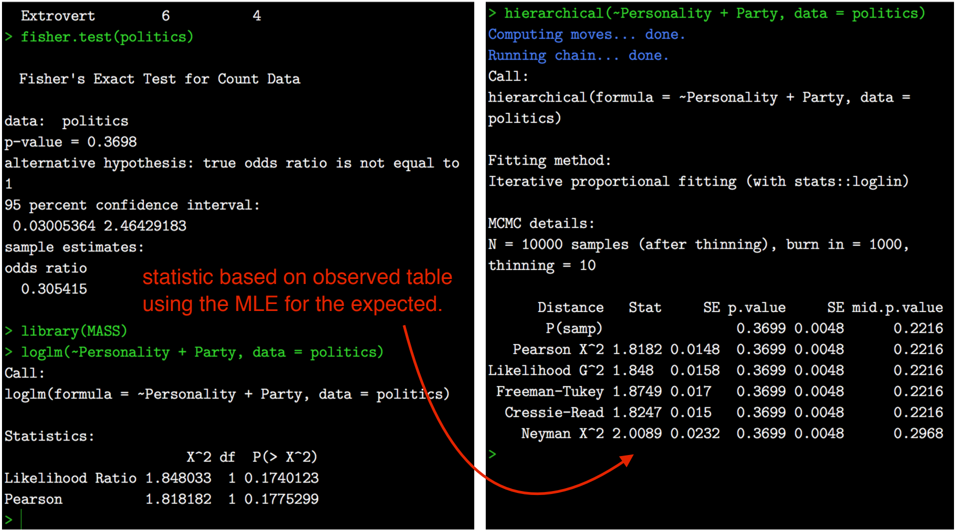
```
  Extrovert        6          4
> fisher.test(politics)

  Fisher's Exact Test for Count Data

data:  politics
p-value = 0.3698
alternative hypothesis: true odds ratio is not equal to
1
95 percent confidence interval:
 0.03005364 2.46429183
sample estimates:
odds ratio
  0.305415

> library(MASS)
> loglm(~Personality + Party, data = politics)
Call:
loglm(formula = ~Personality + Party, data = politics)

Statistics:
                      X^2 df  P(> X^2)
Likelihood Ratio 1.848033  1 0.1740123
Pearson          1.818182  1 0.1775299
>
```

C++ part



```
> hierarchical(~Personality + Party, data = politics)
Computing moves... done.
Running chain... done.
Call:
hierarchical(formula = ~Personality + Party, data =
politics)

Fitting method:
Iterative proportional fitting (with stats::loglin)

MCMC details:
N = 10000 samples (after thinning), burn in = 1000,
thinning = 10

      Distance   Stat     SE p.value     SE mid.p.value
       P(samp)                0.3699 0.0048      0.2216
   Pearson X^2 1.8182 0.0148  0.3699 0.0048      0.2216
Likelihood G^2 1.848  0.0158  0.3699 0.0048      0.2216
 Freeman-Tukey 1.8749 0.017   0.3699 0.0048      0.2216
  Cressie-Read 1.8247 0.015   0.3699 0.0048      0.2216
    Neyman X^2 2.0089 0.0232  0.3699 0.0048      0.2968
>
```

HOW R AND ALGSTAT DO IT

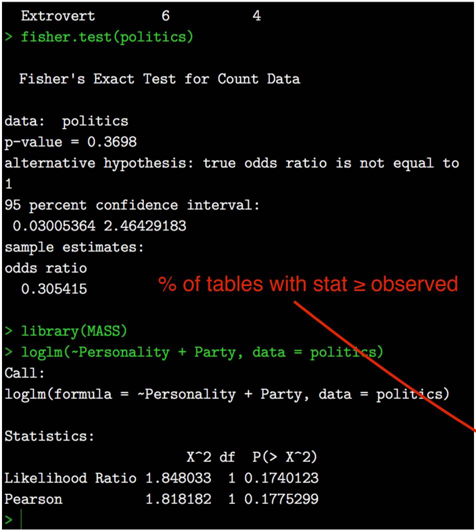


```
  Extrovert        6          4
> fisher.test(politics)

  Fisher's Exact Test for Count Data

data:  politics
p-value = 0.3698
alternative hypothesis: true odds ratio is not equal to
1
95 percent confidence interval:
 0.03005364 2.46429183
sample estimates:
odds ratio
  0.305415

> library(MASS)
> loglm(~Personality + Party, data = politics)
Call:
loglm(formula = ~Personality + Party, data = politics)

Statistics:
                      X^2 df  P(> X^2)
Likelihood Ratio 1.848033  1 0.1740123
Pearson          1.818182  1 0.1775299
>
```

statistic based on observed table using the MLE for the expected.

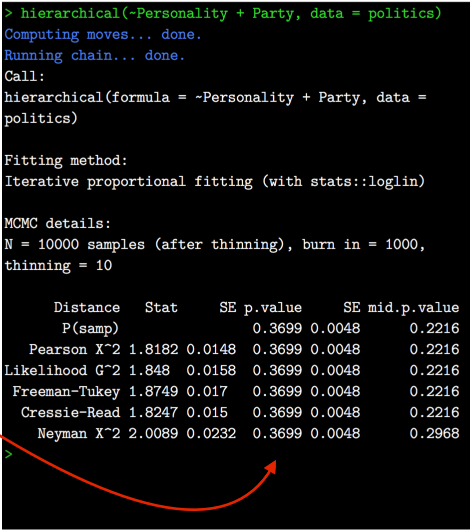
```
> hierarchical(~Personality + Party, data = politics)
Computing moves... done.
Running chain... done.
Call:
hierarchical(formula = ~Personality + Party, data =
politics)

Fitting method:
Iterative proportional fitting (with stats::loglin)

MCMC details:
N = 10000 samples (after thinning), burn in = 1000,
thinning = 10

      Distance   Stat     SE p.value     SE mid.p.value
       P(samp)               0.3699 0.0048      0.2216
   Pearson X^2 1.8182 0.0148 0.3699 0.0048      0.2216
Likelihood G^2 1.848  0.0158 0.3699 0.0048      0.2216
 Freeman-Tukey 1.8749 0.017  0.3699 0.0048      0.2216
  Cressie-Read 1.8247 0.015  0.3699 0.0048      0.2216
    Neyman X^2 2.0089 0.0232 0.3699 0.0048      0.2968
>
```

HOW R AND ALGSTAT DO IT



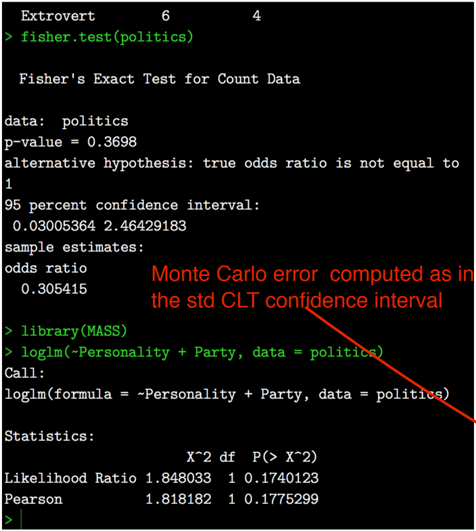
```
  Extrovert        6          4
> fisher.test(politics)

  Fisher's Exact Test for Count Data

data:  politics
p-value = 0.3698
alternative hypothesis: true odds ratio is not equal to
1
95 percent confidence interval:
 0.03005364 2.46429183
sample estimates:
odds ratio
  0.305415

> library(MASS)
> loglm(~Personality + Party, data = politics)
Call:
loglm(formula = ~Personality + Party, data = politics)

Statistics:
                      X^2 df  P(> X^2)
Likelihood Ratio 1.848033  1 0.1740123
Pearson          1.818182  1 0.1775299
>
```

% of tables with stat ≥ observed



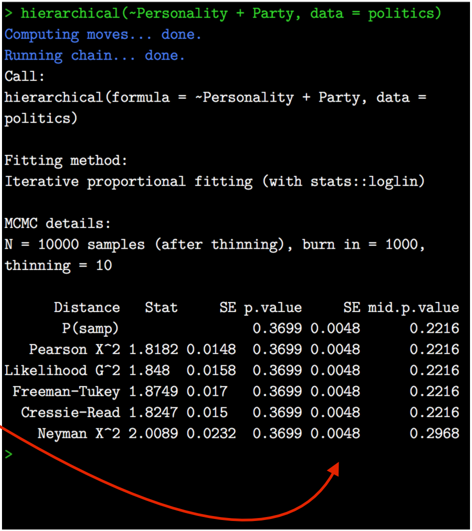
```
> hierarchical(~Personality + Party, data = politics)
Computing moves... done.
Running chain... done.
Call:
hierarchical(formula = ~Personality + Party, data =
politics)

Fitting method:
Iterative proportional fitting (with stats::loglin)

MCMC details:
N = 10000 samples (after thinning), burn in = 1000,
thinning = 10

      Distance   Stat     SE p.value     SE mid.p.value
       P(samp)               0.3699 0.0048      0.2216
   Pearson X^2 1.8182 0.0148 0.3699 0.0048      0.2216
Likelihood G^2 1.848  0.0158 0.3699 0.0048      0.2216
 Freeman-Tukey 1.8749 0.017  0.3699 0.0048      0.2216
  Cressie-Read 1.8247 0.015  0.3699 0.0048      0.2216
    Neyman X^2 2.0089 0.0232 0.3699 0.0048      0.2968
>
```

HOW R AND ALGSTAT DO IT



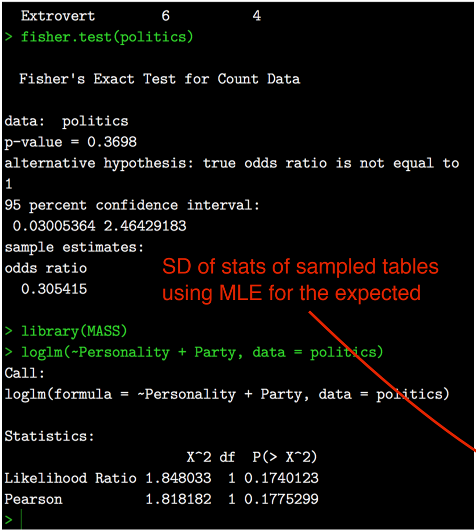
```
  Extrovert        6          4
> fisher.test(politics)

  Fisher's Exact Test for Count Data

data:  politics
p-value = 0.3698
alternative hypothesis: true odds ratio is not equal to
1
95 percent confidence interval:
 0.03005364 2.46429183
sample estimates:
odds ratio
  0.305415

> library(MASS)
> loglm(~Personality + Party, data = politics)
Call:
loglm(formula = ~Personality + Party, data = politics)

Statistics:
                      X^2 df  P(> X^2)
Likelihood Ratio 1.848033  1 0.1740123
Pearson          1.818182  1 0.1775299
>
```

Monte Carlo error computed as in the std CLT confidence interval



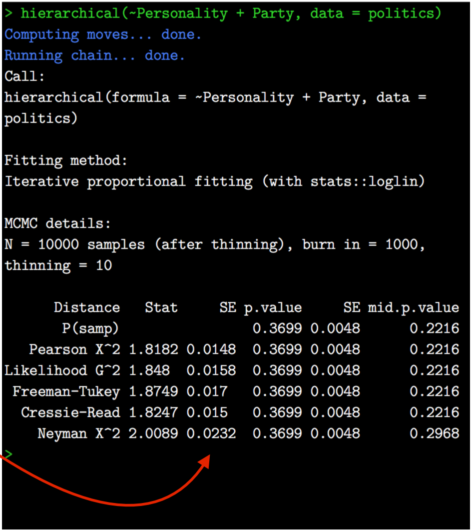
```
> hierarchical(~Personality + Party, data = politics)
Computing moves... done.
Running chain... done.
Call:
hierarchical(formula = ~Personality + Party, data =
politics)

Fitting method:
Iterative proportional fitting (with stats::loglin)

MCMC details:
N = 10000 samples (after thinning), burn in = 1000,
thinning = 10

      Distance   Stat     SE p.value     SE mid.p.value
       P(samp)               0.3699 0.0048      0.2216
   Pearson X^2 1.8182 0.0148 0.3699 0.0048      0.2216
Likelihood G^2 1.848  0.0158 0.3699 0.0048      0.2216
 Freeman-Tukey 1.8749 0.017  0.3699 0.0048      0.2216
  Cressie-Read 1.8247 0.015  0.3699 0.0048      0.2216
    Neyman X^2 2.0089 0.0232 0.3699 0.0048      0.2968
>
```

HOW R AND ALGSTAT DO IT



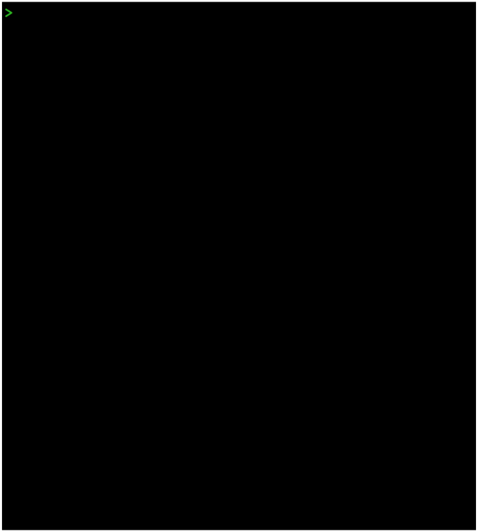
```
  Extrovert        6          4
> fisher.test(politics)

  Fisher's Exact Test for Count Data

data:  politics
p-value = 0.3698
alternative hypothesis: true odds ratio is not equal to
1
95 percent confidence interval:
 0.03005364 2.46429183
sample estimates:
odds ratio
  0.305415

> library(MASS)
> loglm(~Personality + Party, data = politics)
Call:
loglm(formula = ~Personality + Party, data = politics)

Statistics:
                      X^2 df  P(> X^2)
Likelihood Ratio 1.848033  1 0.1740123
Pearson          1.818182  1 0.1775299
>
```

SD of stats of sampled tables using MLE for the expected

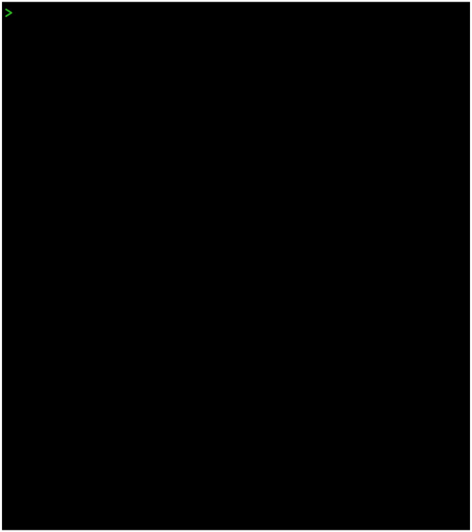


```
> hierarchical(~Personality + Party, data = politics)
Computing moves... done.
Running chain... done.
Call:
hierarchical(formula = ~Personality + Party, data =
politics)

Fitting method:
Iterative proportional fitting (with stats::loglin)

MCMC details:
N = 10000 samples (after thinning), burn in = 1000,
thinning = 10

      Distance   Stat     SE p.value     SE mid.p.value
       P(samp)               0.3699 0.0048      0.2216
   Pearson X^2 1.8182 0.0148 0.3699 0.0048      0.2216
Likelihood G^2 1.848  0.0158 0.3699 0.0048      0.2216
 Freeman-Tukey 1.8749 0.017  0.3699 0.0048      0.2216
  Cressie-Read 1.8247 0.015  0.3699 0.0048      0.2216
    Neyman X^2 2.0089 0.0232 0.3699 0.0048      0.2968
>
```

MODEL FITTING



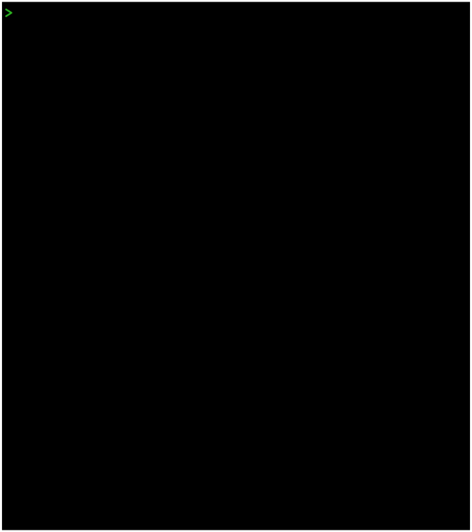
```
>
```



```
>
```

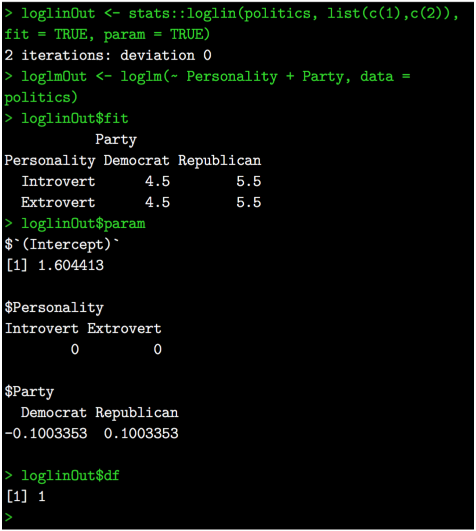
MODEL FITTING

```
> loglinOut <- stats::loglin(politics, list(c(1),c(2)),
fit = TRUE, param = TRUE)
2 iterations: deviation 0
> loglmOut <- loglm(~ Personality + Party, data =
politics)
> 
```

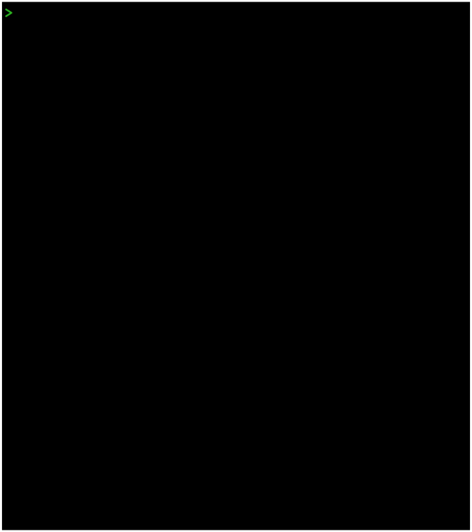


```
>
```

MODEL FITTING

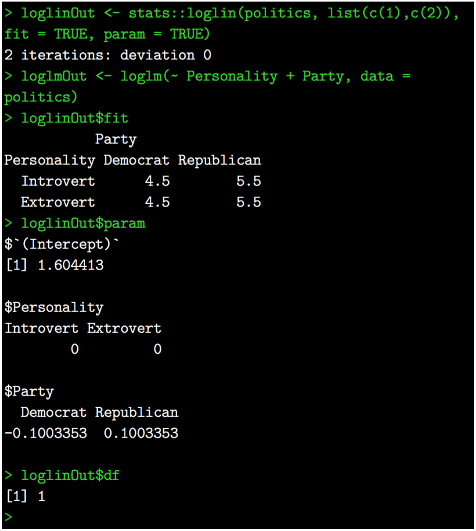


```
> loglinOut <- stats::loglin(politics, list(c(1),c(2)),
fit = TRUE, param = TRUE)
2 iterations: deviation 0
> loglmOut <- loglm(~ Personality + Party, data =
politics)
> loglinOut$fit
           Party
Personality Democrat Republican
  Introvert      4.5        5.5
  Extrovert      4.5        5.5
> loglinOut$param
$`(Intercept)`
[1] 1.604413

$Personality
Introvert Extrovert
        0         0

$Party
  Democrat Republican
-0.1003353  0.1003353

> loglinOut$df
[1] 1
>
```

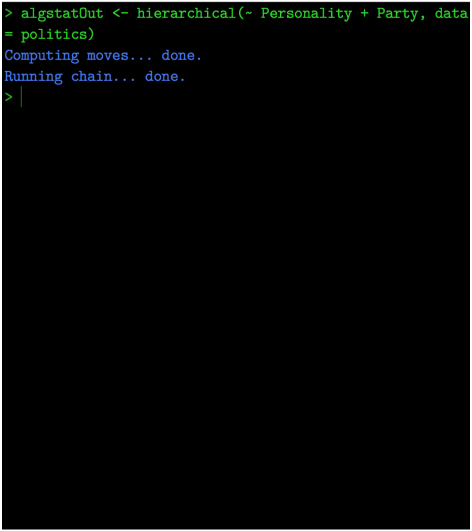


```
>
```

MODEL FITTING

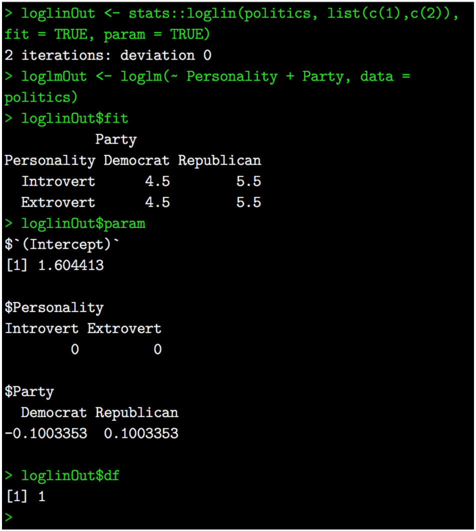


```
> loglinOut <- stats::loglin(politics, list(c(1),c(2)),
fit = TRUE, param = TRUE)
2 iterations: deviation 0
> loglmOut <- loglm(~ Personality + Party, data =
politics)
> loglinOut$fit
           Party
Personality Democrat Republican
  Introvert      4.5        5.5
  Extrovert      4.5        5.5
> loglinOut$param
$`(Intercept)`
[1] 1.604413

$Personality
Introvert Extrovert
        0         0

$Party
  Democrat Republican
-0.1003353  0.1003353

> loglinOut$df
[1] 1
>
```

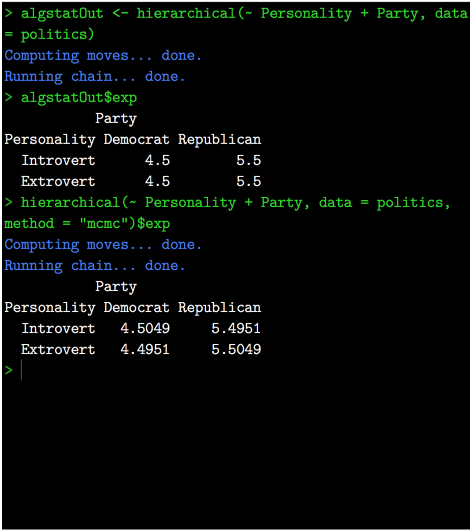


```
> algstatOut <- hierarchical(~ Personality + Party, data
= politics)
Computing moves... done.
Running chain... done.
>
```

MODEL FITTING

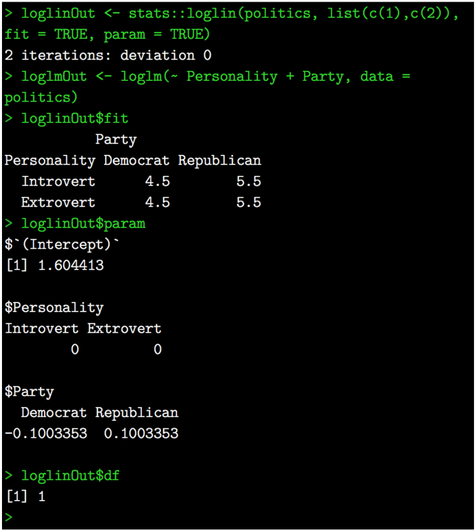


```
> loglinOut <- stats::loglin(politics, list(c(1),c(2)),
fit = TRUE, param = TRUE)
2 iterations: deviation 0
> loglmOut <- loglm(~ Personality + Party, data =
politics)
> loglinOut$fit
           Party
Personality Democrat Republican
  Introvert      4.5        5.5
  Extrovert      4.5        5.5
> loglinOut$param
$`(Intercept)`
[1] 1.604413

$Personality
Introvert Extrovert
        0         0

$Party
  Democrat Republican
-0.1003353  0.1003353

> loglinOut$df
[1] 1
>
```

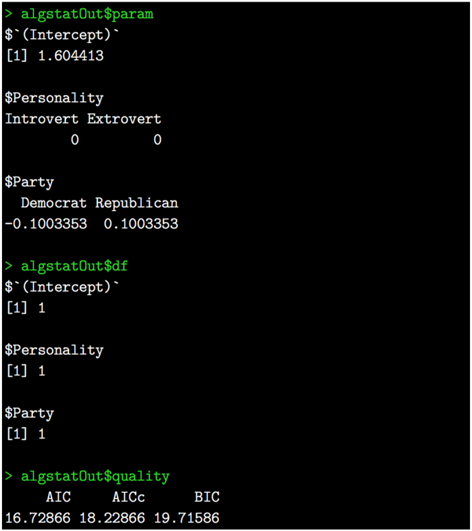


```
> algstatOut <- hierarchical(~ Personality + Party, data
= politics)
Computing moves... done.
Running chain... done.
> algstatOut$exp
           Party
Personality Democrat Republican
  Introvert      4.5        5.5
  Extrovert      4.5        5.5
> hierarchical(~ Personality + Party, data = politics,
method = "mcmc")$exp
Computing moves... done.
Running chain... done.
           Party
Personality Democrat Republican
  Introvert   4.5049     5.4951
  Extrovert   4.4951     5.5049
>
```

MODEL FITTING

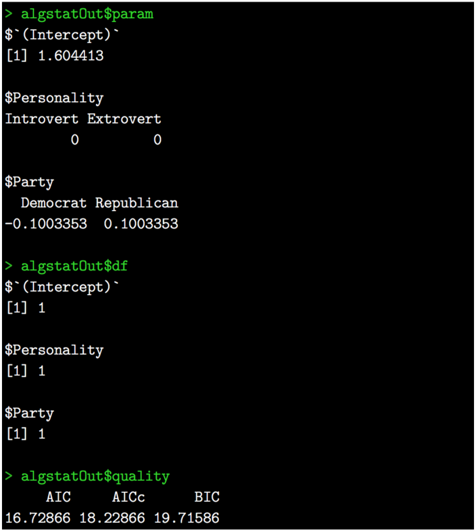


```
> loglinOut <- stats::loglin(politics, list(c(1),c(2)),
fit = TRUE, param = TRUE)
2 iterations: deviation 0
> loglmOut <- loglm(~ Personality + Party, data =
politics)
> loglinOut$fit
           Party
Personality Democrat Republican
  Introvert      4.5        5.5
  Extrovert      4.5        5.5
> loglinOut$param
$`(Intercept)`
[1] 1.604413

$Personality
Introvert Extrovert
        0         0

$Party
  Democrat Republican
-0.1003353  0.1003353

> loglinOut$df
[1] 1
>
```



```
> algstatOut$param
$`(Intercept)`
[1] 1.604413

$Personality
Introvert Extrovert
        0         0

$Party
  Democrat Republican
-0.1003353  0.1003353

> algstatOut$df
$`(Intercept)`
[1] 1

$Personality
[1] 1

$Party
[1] 1

> algstatOut$quality
     AIC     AICc      BIC
16.72866 18.22866 19.71586
```

MODEL FITTING



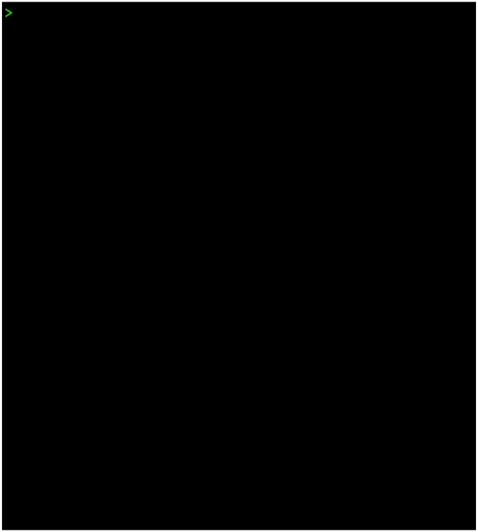
```
> algstatOut$param
$`(Intercept)`
[1] 1.604413

$Personality
Introvert Extrovert
        0         0

$Party
  Democrat Republican
-0.1003353  0.1003353

> algstatOut$df
$`(Intercept)`
[1] 1

$Personality
[1] 1

$Party
[1] 1

> algstatOut$quality
     AIC     AICc      BIC
16.72866 18.22866 19.71586
```

Project 1:

A. Use **hierarchical** to find a log-linear model that may seem to fit the dataset **drugs**.

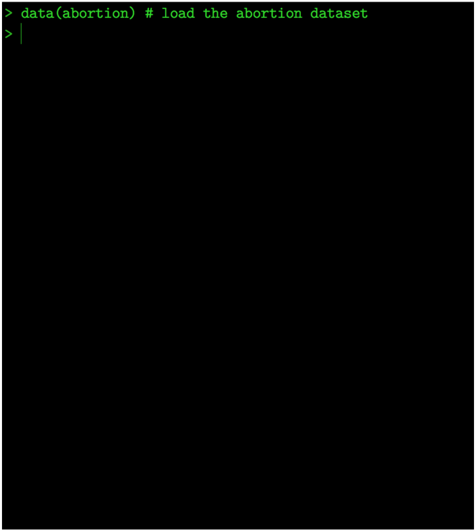
B. Investigate the dataset **haberman** under the no 3-way interaction model.

NO 3-WAY INTERACTION MODEL



```
>
```

NO 3-WAY INTERACTION MODEL



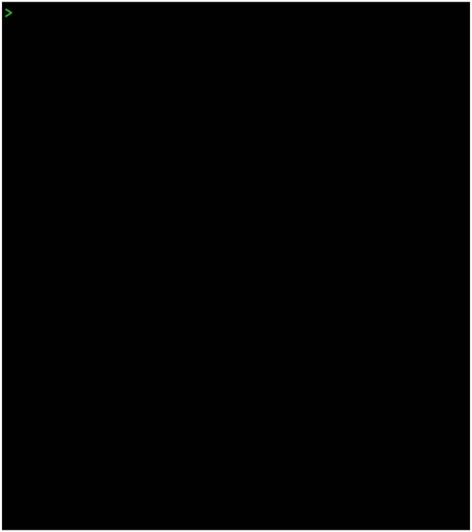
```
> data(abortion) # load the abortion dataset
> 
```

No 3-way interaction model

```
> abortion
, , Denomination = Northern Protestant

         Abortion
Education Positive Mixed Negative
   Low           9    16       41
   Medium       85    52      105
   High         77    30       38

, , Denomination = Southern Protestant

         Abortion
Education Positive Mixed Negative
   Low           8     8       46
   Medium       35    29       54
   High         37    15       22

, , Denomination = Catholic

         Abortion
Education Positive Mixed Negative
   Low          11    14       38
   Medium       47    35      115
   High         25    21       42
```

NO 3-WAY INTERACTION MODEL

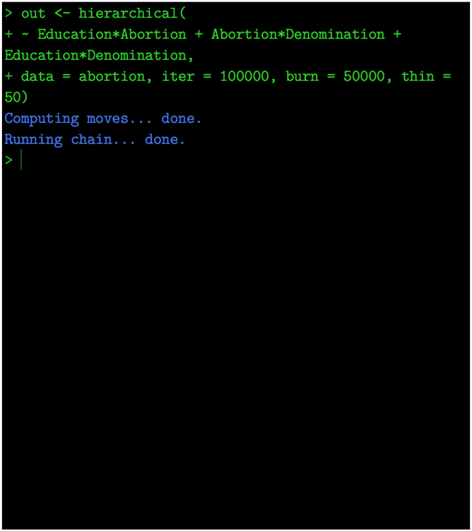
```
> abortion
, , Denomination = Northern Protestant

         Abortion
Education Positive Mixed Negative
   Low           9    16       41
   Medium       85    52      105
   High         77    30       38

, , Denomination = Southern Protestant

         Abortion
Education Positive Mixed Negative
   Low           8     8       46
   Medium       35    29       54
   High         37    15       22

, , Denomination = Catholic

         Abortion
Education Positive Mixed Negative
   Low          11    14       38
   Medium       47    35      115
   High         25    21       42
```



```
>
```

NO 3-WAY INTERACTION MODEL

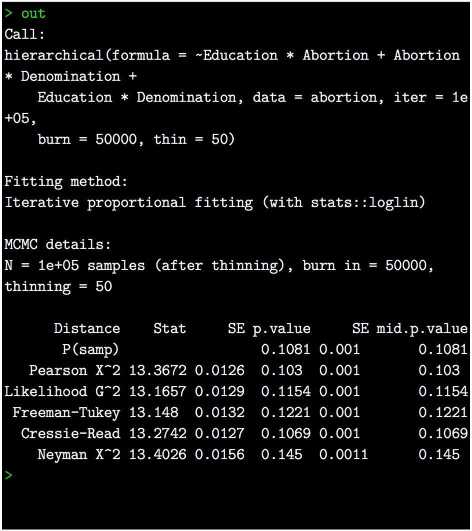
```
> abortion
, , Denomination = Northern Protestant

        Abortion
Education Positive Mixed Negative
   Low           9    16       41
   Medium       85    52      105
   High         77    30       38

, , Denomination = Southern Protestant

        Abortion
Education Positive Mixed Negative
   Low           8     8       46
   Medium       35    29       54
   High         37    15       22

, , Denomination = Catholic

        Abortion
Education Positive Mixed Negative
   Low          11    14       38
   Medium       47    35      115
   High         25    21       42
```



```
> out <- hierarchical(
+ ~ Education*Abortion + Abortion*Denomination +
Education*Denomination,
+ data = abortion, iter = 100000, burn = 50000, thin =
50)
Computing moves... done.
Running chain... done.
>
```

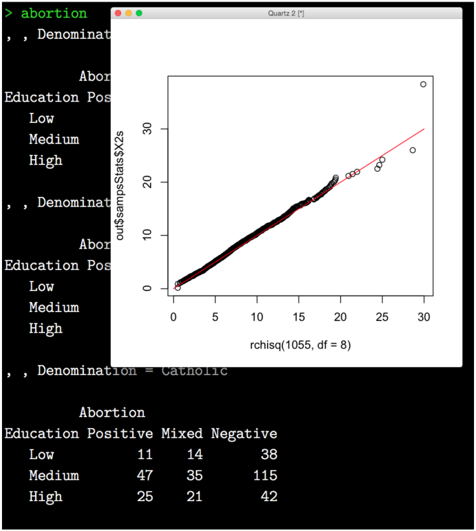
NO 3-WAY INTERACTION MODEL

```
> abortion
, , Denomination = Northern Protestant

         Abortion
Education Positive Mixed Negative
   Low           9    16       41
   Medium       85    52      105
   High         77    30       38

, , Denomination = Southern Protestant

         Abortion
Education Positive Mixed Negative
   Low           8     8       46
   Medium       35    29       54
   High         37    15       22

, , Denomination = Catholic

         Abortion
Education Positive Mixed Negative
   Low          11    14       38
   Medium       47    35      115
   High         25    21       42
```

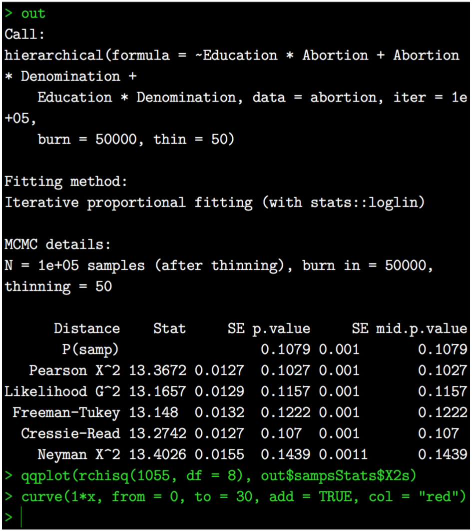


```
> out
Call:
hierarchical(formula = ~Education * Abortion + Abortion
* Denomination +
    Education * Denomination, data = abortion, iter = 1e
+05,
    burn = 50000, thin = 50)

Fitting method:
Iterative proportional fitting (with stats::loglin)

MCMC details:
N = 1e+05 samples (after thinning), burn in = 50000,
thinning = 50

      Distance    Stat     SE p.value     SE mid.p.value
       P(samp)                 0.1081 0.001       0.1081
   Pearson X^2 13.3672 0.0126  0.103  0.001       0.103
Likelihood G^2 13.1657 0.0129  0.1154 0.001       0.1154
 Freeman-Tukey 13.148  0.0132  0.1221 0.001       0.1221
   Cressie-Read 13.2742 0.0127  0.1069 0.001       0.1069
    Neyman X^2 13.4026 0.0156  0.145  0.0011      0.145
>
```

NO 3-WAY INTERACTION MODEL

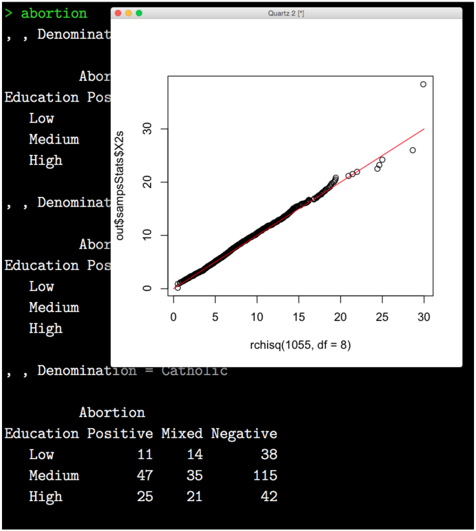


```
> abortion
, , Denominat

         Abor
Education Pos
   Low
   Medium
   High

, , Denominat

         Abor
Education Pos
   Low
   Medium
   High

, , Denomination = Catholic

         Abortion
Education Positive Mixed Negative
   Low          11    14       38
   Medium       47    35      115
   High         25    21       42
```



```
> out
Call:
hierarchical(formula = ~Education * Abortion + Abortion
* Denomination +
    Education * Denomination, data = abortion, iter = 1e
+05,
    burn = 50000, thin = 50)

Fitting method:
Iterative proportional fitting (with stats::loglin)

MCMC details:
N = 1e+05 samples (after thinning), burn in = 50000,
thinning = 50

      Distance    Stat     SE p.value     SE mid.p.value
       P(samp)                 0.1079 0.001       0.1079
   Pearson X^2 13.3672 0.0127  0.1027 0.001       0.1027
Likelihood G^2 13.1657 0.0129  0.1157 0.001       0.1157
 Freeman-Tukey 13.148  0.0132  0.1222 0.001       0.1222
  Cressie-Read 13.2742 0.0127  0.107  0.001       0.107
    Neyman X^2 13.4026 0.0155  0.1439 0.0011      0.1439
> qqplot(rchisq(1055, df = 8), out$sampsStats$X2s)
> curve(1*x, from = 0, to = 30, add = TRUE, col = "red")
>
```

NO 3-WAY INTERACTION MODEL



```
> abortion
, , Denominat

         Abor
Education Pos
   Low
   Medium
   High

, , Denominat

         Abor
Education Pos
   Low
   Medium
   High

, , Denomination = Catholic

         Abortion
Education Positive Mixed Negative
   Low          11    14       38
   Medium       47    35      115
   High         25    21       42
```

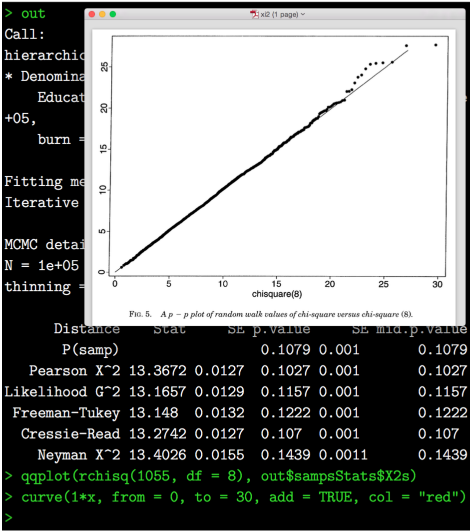
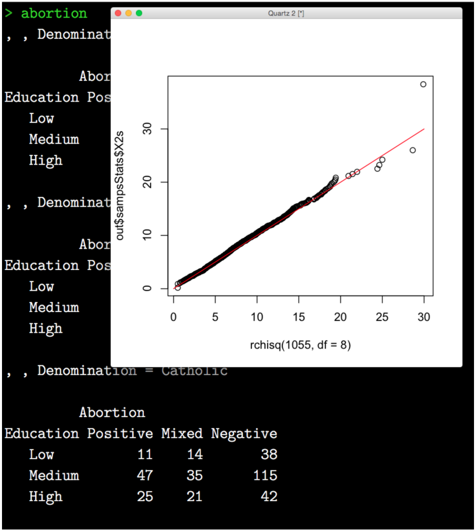


FIG. 5. *A p − p plot of random walk values of chi-square versus chi-square (8).*

```
> out
Call:
hierarchic
* Denomina
    Educat
+05,
    burn =

Fitting me
Iterative

MCMC detai
N = 1e+05
thinning =

      Distance    Stat      SE p.value     SE mid.p.value
       P(samp)                  0.1079 0.001       0.1079
   Pearson X^2 13.3672 0.0127  0.1027 0.001       0.1027
Likelihood G^2 13.1657 0.0129  0.1157 0.001       0.1157
 Freeman-Tukey 13.148  0.0132  0.1222 0.001       0.1222
  Cressie-Read 13.2742 0.0127  0.107   0.001       0.107
    Neyman X^2 13.4026 0.0155  0.1439 0.0011      0.1439
> qqplot(rchisq(1055, df = 8), out$sampsStats$X2s)
> curve(1*x, from = 0, to = 30, add = TRUE, col = "red")
>
```

NO 3-WAY INTERACTION MODEL

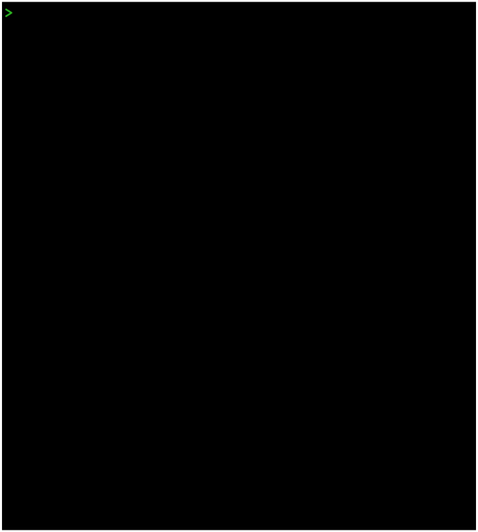


```
> abortion
, , Denominat

         Abor
Education Pos
   Low
   Medium
   High

, , Denominat

         Abor
Education Pos
   Low
   Medium
   High

, , Denomination = Catholic

         Abortion
Education Positive Mixed Negative
   Low          11    14       38
   Medium       47    35      115
   High         25    21       42
```

Project 2:

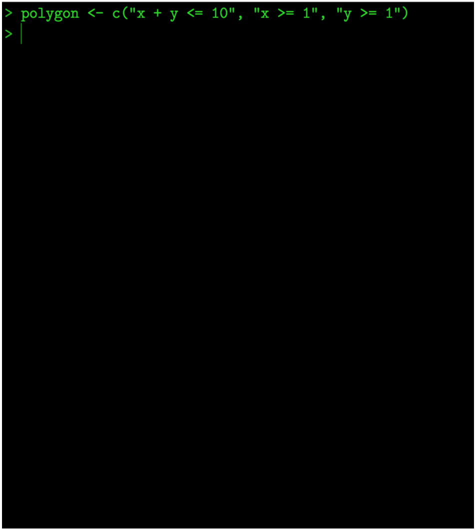
Use **algstat** to recover your favorite result in the seminal paper by Diaconis and Sturmfels: `https://projecteuclid.org/download/pdf_1/euclid.aos/1030563990`
Or any other of your favorite articles or books such as Lectures on Algbraic Statistics [Chapter 1]. `https://math.berkeley.edu/~bernd/owl.pdf`

LATTE FUNCTIONALITY



```
>
```

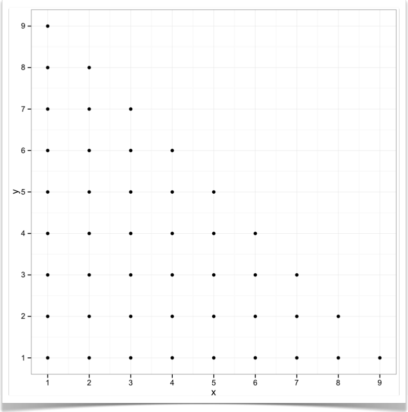
LATTE FUNCTIONALITY



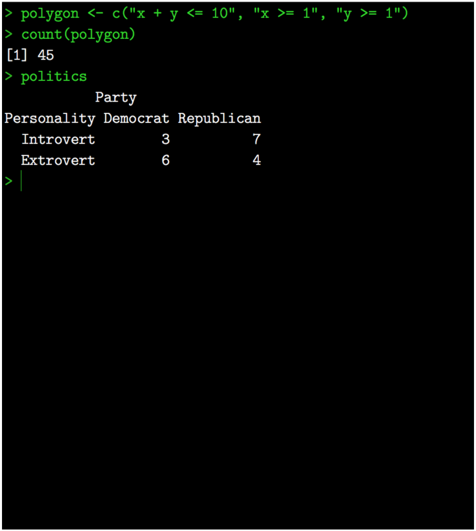
```
> polygon <- c("x + y <= 10", "x >= 1", "y >= 1")
> 
```

LATTE FUNCTIONALITY

```
> polygon <- c("x + y <= 10", "x >= 1", "y >= 1")
> count(polygon)
[1] 45
>
```

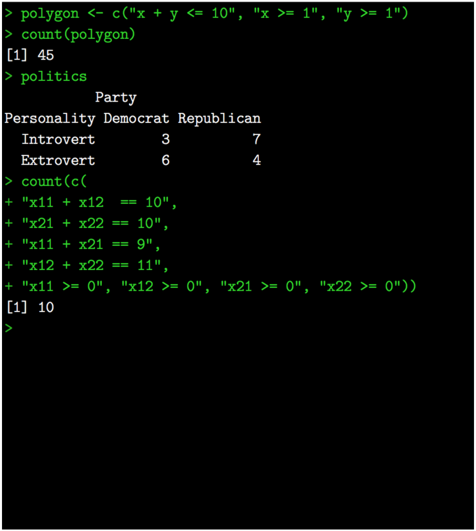


LATTE FUNCTIONALITY



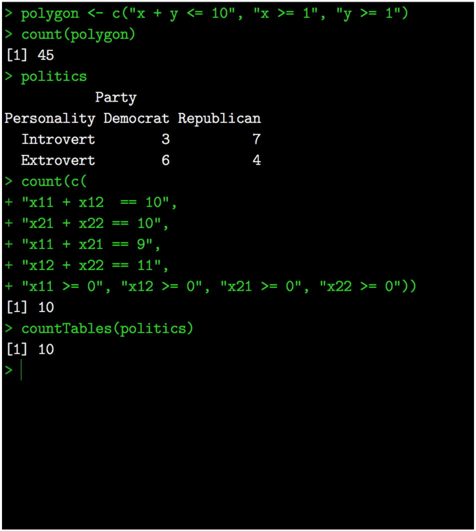
```
> polygon <- c("x + y <= 10", "x >= 1", "y >= 1")
> count(polygon)
[1] 45
> politics
           Party
Personality Democrat Republican
  Introvert        3          7
  Extrovert        6          4
>
```

LATTE FUNCTIONALITY



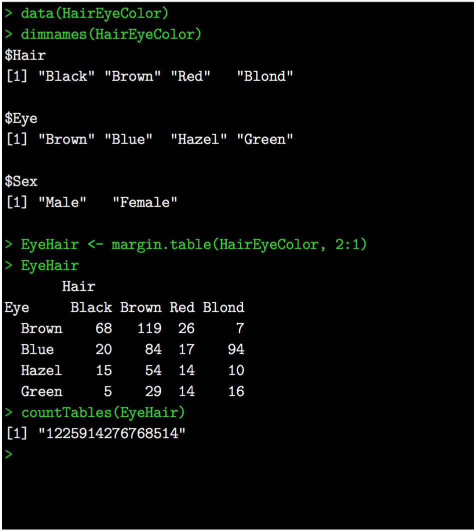
```
> polygon <- c("x + y <= 10", "x >= 1", "y >= 1")
> count(polygon)
[1] 45
> politics
           Party
Personality Democrat Republican
  Introvert        3          7
  Extrovert        6          4
> count(c(
+ "x11 + x12  == 10",
+ "x21 + x22 == 10",
+ "x11 + x21 == 9",
+ "x12 + x22 == 11",
+ "x11 >= 0", "x12 >= 0", "x21 >= 0", "x22 >= 0"))
[1] 10
>
```

LATTE FUNCTIONALITY



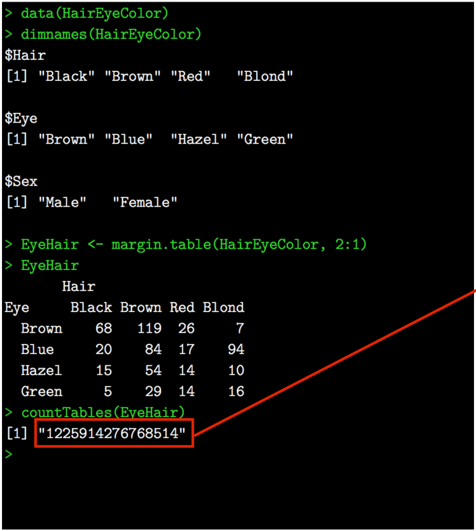
```
> polygon <- c("x + y <= 10", "x >= 1", "y >= 1")
> count(polygon)
[1] 45
> politics
           Party
Personality Democrat Republican
  Introvert        3          7
  Extrovert        6          4
> count(c(
+ "x11 + x12  == 10",
+ "x21 + x22 == 10",
+ "x11 + x21 == 9",
+ "x12 + x22 == 11",
+ "x11 >= 0", "x12 >= 0", "x21 >= 0", "x22 >= 0"))
[1] 10
> countTables(politics)
[1] 10
>
```

LATTE FUNCTIONALITY

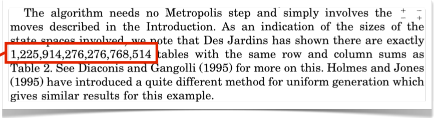


```
> data(HairEyeColor)
> dimnames(HairEyeColor)
$Hair
[1] "Black" "Brown" "Red"   "Blond"

$Eye
[1] "Brown" "Blue"  "Hazel" "Green"

$Sex
[1] "Male"   "Female"

> EyeHair <- margin.table(HairEyeColor, 2:1)
> EyeHair
       Hair
Eye     Black Brown Red Blond
  Brown    68   119  26     7
  Blue     20    84  17    94
  Hazel    15    54  14    10
  Green     5    29  14    16
> countTables(EyeHair)
[1] "1225914276768514"
>
```

LATTE FUNCTIONALITY



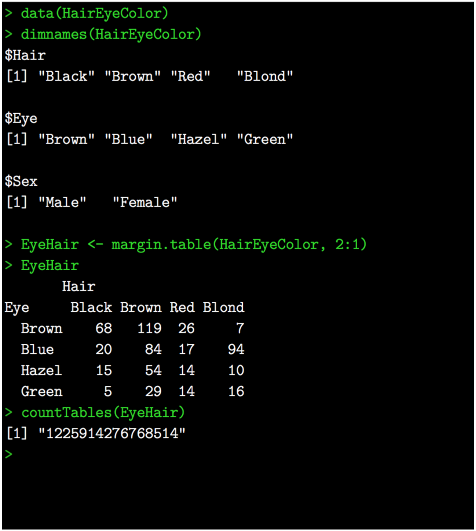
```
> data(HairEyeColor)
> dimnames(HairEyeColor)
$Hair
[1] "Black" "Brown" "Red"   "Blond"

$Eye
[1] "Brown" "Blue"  "Hazel" "Green"

$Sex
[1] "Male"   "Female"

> EyeHair <- margin.table(HairEyeColor, 2:1)
> EyeHair
       Hair
Eye     Black Brown Red Blond
  Brown    68   119  26     7
  Blue     20    84  17    94
  Hazel    15    54  14    10
  Green     5    29  14    16
> countTables(EyeHair)
[1] "1225914276768514"
>
```



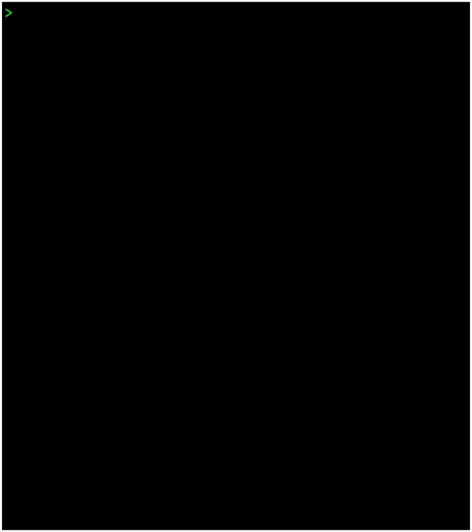
The algorithm needs no Metropolis step and simply involves the $\begin{smallmatrix}+&-\\-&+\end{smallmatrix}$ moves described in the Introduction. As an indication of the sizes of the state spaces involved, we note that Des Jardins has shown there are exactly 1,225,914,276,276,768,514 tables with the same row and column sums as Table 2. See Diaconis and Gangolli (1995) for more on this. Holmes and Jones (1995) have introduced a quite different method for uniform generation which gives similar results for this example.

Diaconis and Sturmfels (1998)

MIXING TIMES

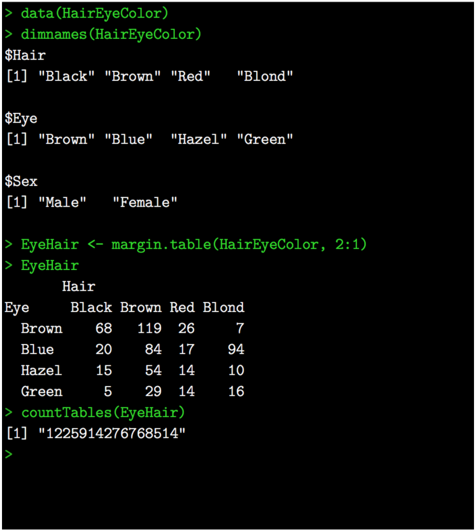


```
> data(HairEyeColor)
> dimnames(HairEyeColor)
$Hair
[1] "Black" "Brown" "Red"   "Blond"

$Eye
[1] "Brown" "Blue"  "Hazel" "Green"

$Sex
[1] "Male"   "Female"

> EyeHair <- margin.table(HairEyeColor, 2:1)
> EyeHair
       Hair
Eye     Black Brown Red Blond
  Brown    68   119  26     7
  Blue     20    84  17    94
  Hazel    15    54  14    10
  Green     5    29  14    16
> countTables(EyeHair)
[1] "1225914276768514"
>
```

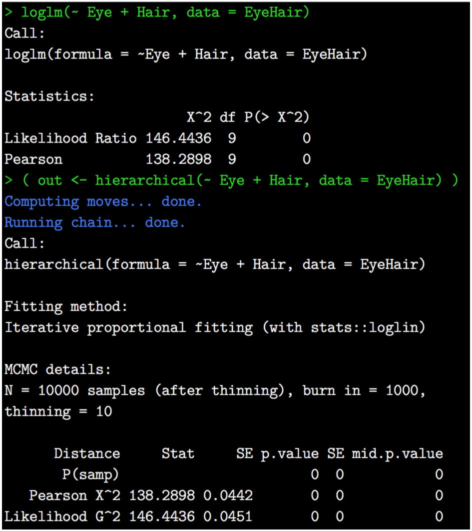


```
>
```

MIXING TIMES

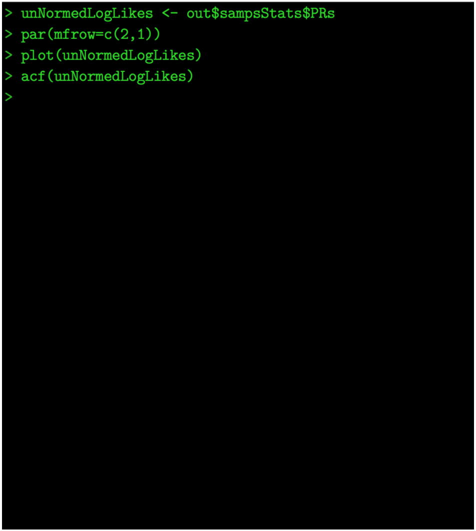


```
> data(HairEyeColor)
> dimnames(HairEyeColor)
$Hair
[1] "Black" "Brown" "Red"   "Blond"

$Eye
[1] "Brown" "Blue"  "Hazel" "Green"

$Sex
[1] "Male"   "Female"

> EyeHair <- margin.table(HairEyeColor, 2:1)
> EyeHair
       Hair
Eye     Black Brown Red Blond
  Brown    68   119  26     7
  Blue     20    84  17    94
  Hazel    15    54  14    10
  Green     5    29  14    16
> countTables(EyeHair)
[1] "1225914276768514"
>
```

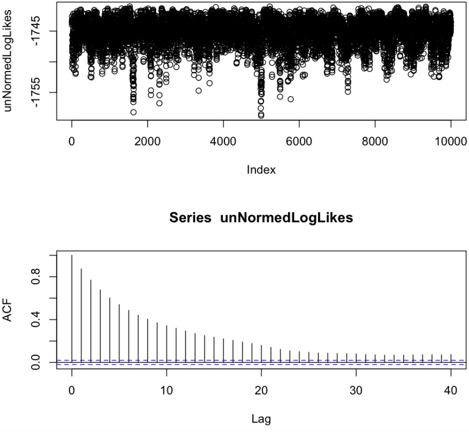


```
> loglm(~ Eye + Hair, data = EyeHair)
Call:
loglm(formula = ~Eye + Hair, data = EyeHair)

Statistics:
                      X^2 df P(> X^2)
Likelihood Ratio 146.4436  9        0
Pearson          138.2898  9        0
> ( out <- hierarchical(~ Eye + Hair, data = EyeHair) )
Computing moves... done.
Running chain... done.
Call:
hierarchical(formula = ~Eye + Hair, data = EyeHair)

Fitting method:
Iterative proportional fitting (with stats::loglin)

MCMC details:
N = 10000 samples (after thinning), burn in = 1000,
thinning = 10

      Distance     Stat     SE p.value SE mid.p.value
       P(samp)                       0  0           0
   Pearson X^2 138.2898 0.0442       0  0           0
Likelihood G^2 146.4436 0.0451       0  0           0
```

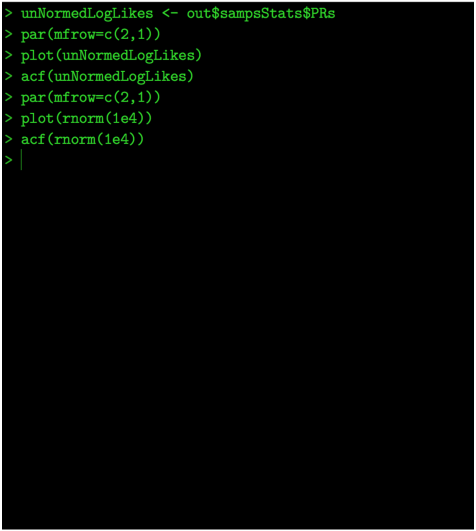
MIXING TIMES



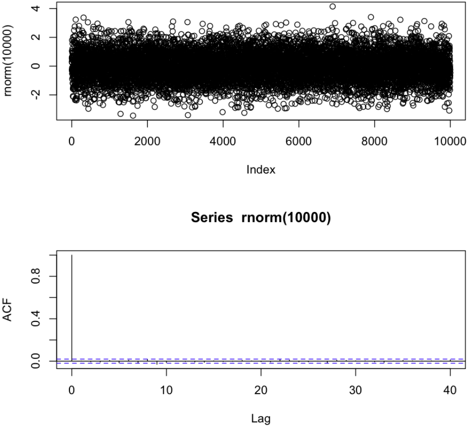
```
> unNormedLogLikes <- out$sampsStats$PRs
> par(mfrow=c(2,1))
> plot(unNormedLogLikes)
> acf(unNormedLogLikes)
>
```



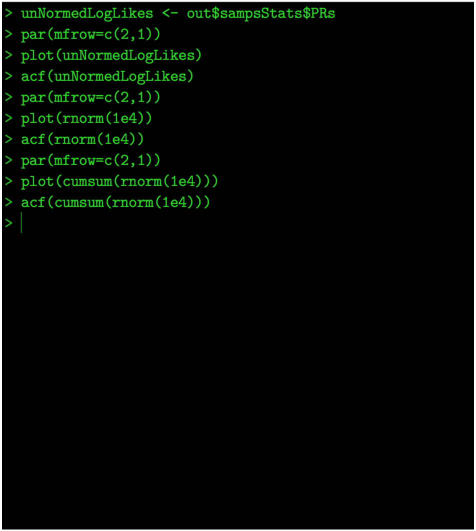
MIXING TIMES



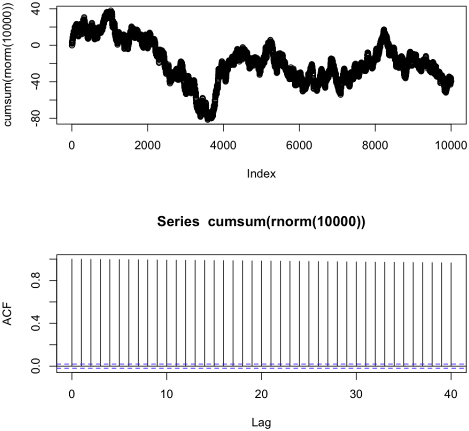
```
> unNormedLogLikes <- out$sampsStats$PRs
> par(mfrow=c(2,1))
> plot(unNormedLogLikes)
> acf(unNormedLogLikes)
> par(mfrow=c(2,1))
> plot(rnorm(1e4))
> acf(rnorm(1e4))
> 
```



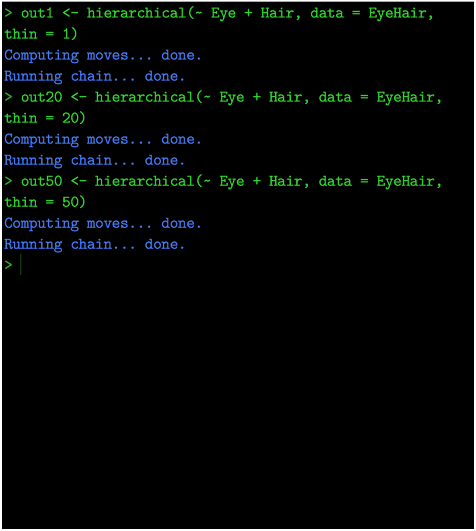
MIXING TIMES



```
> unNormedLogLikes <- out$sampsStats$PRs
> par(mfrow=c(2,1))
> plot(unNormedLogLikes)
> acf(unNormedLogLikes)
> par(mfrow=c(2,1))
> plot(rnorm(1e4))
> acf(rnorm(1e4))
> par(mfrow=c(2,1))
> plot(cumsum(rnorm(1e4)))
> acf(cumsum(rnorm(1e4)))
> 
```

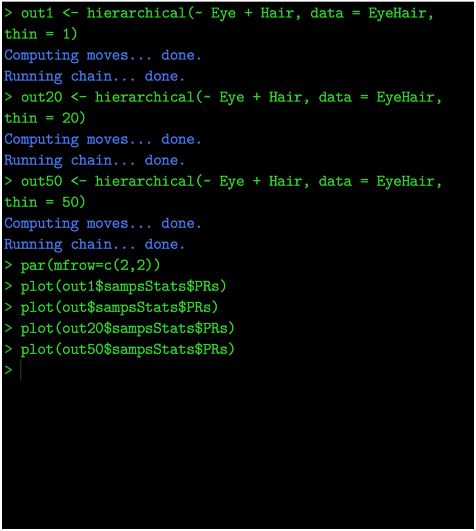


MIXING TIMES

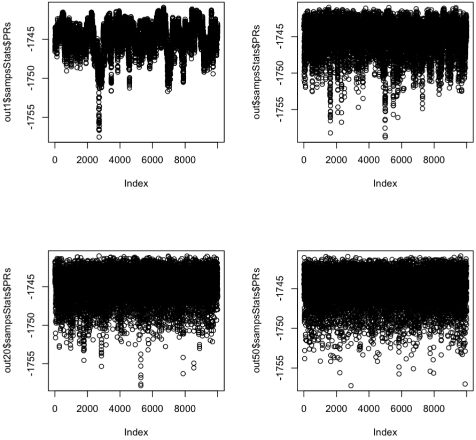


```
> out1 <- hierarchical(~ Eye + Hair, data = EyeHair,
thin = 1)
Computing moves... done.
Running chain... done.
> out20 <- hierarchical(~ Eye + Hair, data = EyeHair,
thin = 20)
Computing moves... done.
Running chain... done.
> out50 <- hierarchical(~ Eye + Hair, data = EyeHair,
thin = 50)
Computing moves... done.
Running chain... done.
>
```

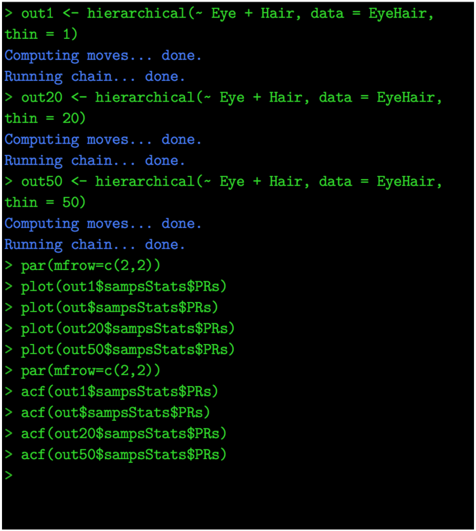
MIXING TIMES



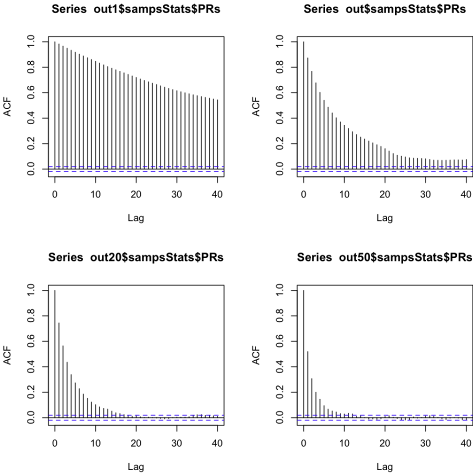
```
> out1 <- hierarchical(~ Eye + Hair, data = EyeHair,
thin = 1)
Computing moves... done.
Running chain... done.
> out20 <- hierarchical(~ Eye + Hair, data = EyeHair,
thin = 20)
Computing moves... done.
Running chain... done.
> out50 <- hierarchical(~ Eye + Hair, data = EyeHair,
thin = 50)
Computing moves... done.
Running chain... done.
> par(mfrow=c(2,2))
> plot(out1$sampsStats$PRs)
> plot(out$sampsStats$PRs)
> plot(out20$sampsStats$PRs)
> plot(out50$sampsStats$PRs)
> |
```



MIXING TIMES



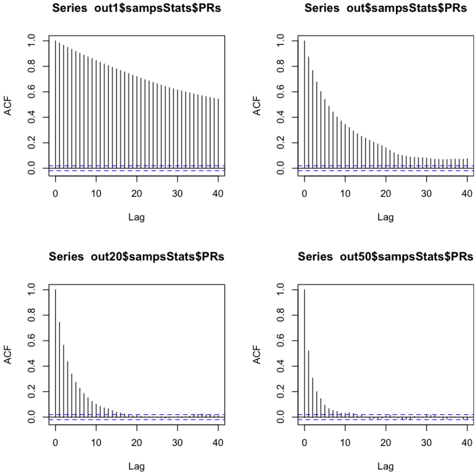
```
> out1 <- hierarchical(~ Eye + Hair, data = EyeHair,
thin = 1)
Computing moves... done.
Running chain... done.
> out20 <- hierarchical(~ Eye + Hair, data = EyeHair,
thin = 20)
Computing moves... done.
Running chain... done.
> out50 <- hierarchical(~ Eye + Hair, data = EyeHair,
thin = 50)
Computing moves... done.
Running chain... done.
> par(mfrow=c(2,2))
> plot(out1$sampsStats$PRs)
> plot(out$sampsStats$PRs)
> plot(out20$sampsStats$PRs)
> plot(out50$sampsStats$PRs)
> par(mfrow=c(2,2))
> acf(out1$sampsStats$PRs)
> acf(out$sampsStats$PRs)
> acf(out20$sampsStats$PRs)
> acf(out50$sampsStats$PRs)
>
```



MIXING TIMES

Project 3:

Analize the mixing times for the MCMC of your favorite dataset and log-linear model.



CONTACT

Thank you!

lgarcia@shsu.edu

`http://www.shsu.edu/~ldg005`