

PM520: Advanced Statistical Computing

- Instructor: Paul Marjoram
 - Office: Soto 202V
 - Office Hours: Thursdays 2-3pm on Zoom
 - email: pmarjora@usc.edu
- Teaching Assistant: Yue Tu (yuetu@usc.edu)
 - Office Hours: 2-3pm Mondays
- Room:
 - Soto 117 (Zoom)
- Will post a copy of the slides (and any other useful material) to Github. Go to <https://github.com/PM520-Spring2022> to find them in a repository called "Lecture-Slides".

Course goals

- To gain skills in writing custom-built code to solve statistical problems
 - Get better at writing statistical analysis code
 - **Get better understanding of how algorithms/methods work**
 - **Introduce you to new methods**
 - Emphasis on simulation-based (Monte Carlo) methods

Our language of choice: R

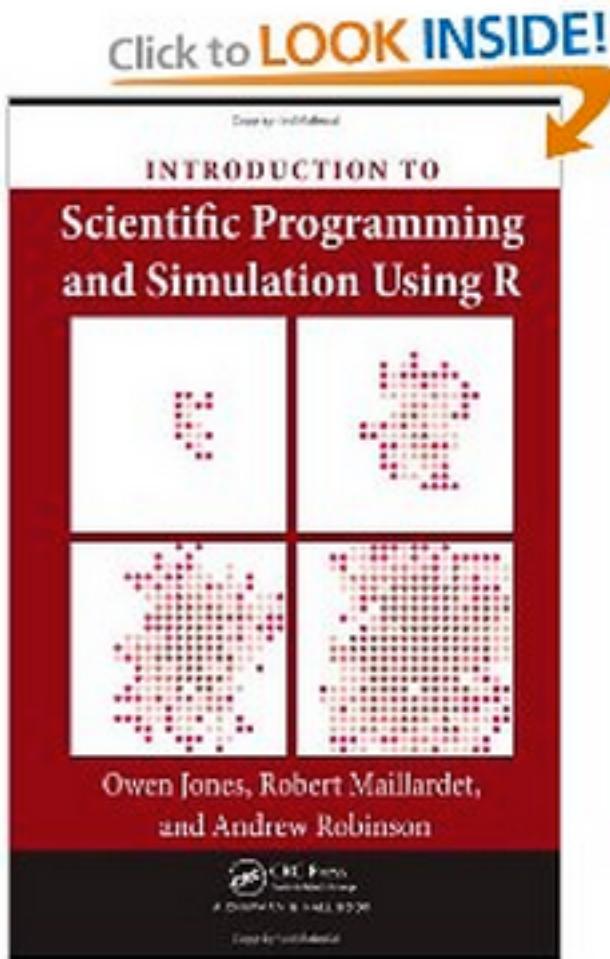
- R is a statistical programming language
- Merits:
 - It is free
 - It is becoming the ‘go to’ language for stats applications
 - Widely-used in genomics
 - High-level language with many built-in functions (regression, optimization, etc.)
- You can use other languages if you prefer (*but please don’t*)

What you will need

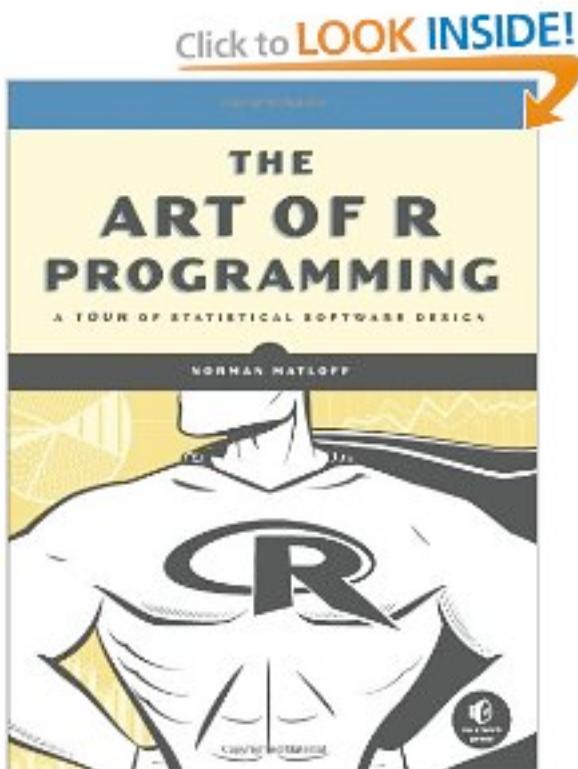
- A copy of R (cran.r-project.org)
- A laptop
- The texts (if you wish - but you don't need them)
- A tolerance for bug-hunting
- I recommend using the RStudio **Integrated Development Environment [IDE]**.
 - You can get RStudio from <http://www.rstudio.com>
- If you are having trouble getting R and an IDE up and running I can help during today's lab.

Suggested Texts - Course text

“Scientific Programming and Simulation Using R” – Owen Jones, Robert Maillardet and Andrew Robinson, CRC Press.



Suggested Texts - Other



[Norman Matloff](#)

No Starch Press

ISBN-10: 1593273843

ISBN-13: 978-1593273842

Online (free!) resources

- Mastering Software Development in R, 2017. Roger Peng, Sean Kross, Brooke Anderson. <https://bookdown.org/rdpeng/RProgDA/>
- R for Data Science, 2017 Garrett Grolemund and Hadley Wickham. <http://r4ds.had.co.nz/>
- Advanced R, 2014, Hadley Wickham, CRC press. <http://adv-r.had.co.nz/> (Second edition is available in print on June 28 <https://www.amazon.com/gp/product/0815384572/>, online here: <https://adv-r.hadley.nz/>)
- R Packages, 2015, Hadley Wickham, O'Reilly. <http://r-pkgs.had.co.nz/> (work on a second edition is in development here <https://r-pkgs.org/> starting as of 2019-02)
- **Advanced Statistical Computing, 2018 Roger Peng** (<https://bookdown.org/rdpeng/advstatcomp/>) (for PM520)

Tentative Course Outline

Weeks 1-2: Introduction – Random number generation and Monte Carlo Estimation. How to estimate things that you cannot calculate. Coin-tossing. Occupancy problems. Hypercubes. Golf balls (external material).

Week 2-3: More Monte Carlo Estimation: Estimating pi, Random variable simulation, Likelihood Estimation. Bayesian methods. Accept/Reject Algorithms (external material).

Week 4: Methods for finding function roots and fixed points. Math as art. (Chapter 10 of course text).

Week 5: Probability and Stochastic Simulation – Urn Models and Chinese Restaurants (Chapter 18 vol 1; **chapter 20, in the 2nd ed.**).

Week 6: Optimization and Regression (chapter 12)

Week 7-9: Markov Chain Monte Carlo [MCMC] Methods. Adaptive MCMC, Parallel Tempered-MCMC, Code-breaking (external material).

Week 10: Gibbs Sampling, Accept/Reject Algorithms (external material).

Week 11: Permutation tests, Numerical Integration and Importance Sampling (Chapters 19 and 22 (vol 1); **chaps. 21 and 24 (2nd ed.)**).

Week 12: Approximate Bayesian Computation, ABC-Rejection, ABC-MCMC (external material).

Week 13: Sequential Monte Carlo Methods, Regression-adjusted ABC (external material)

Week 14: Hidden Markov Models, Genetic Algorithms. (external material)

Week 15: E-M Algorithm, Boot-strapping

How the course will work

- Each class hours will have 3 components:
 - Part 1: Recap of last week's problems, results etc. (This is your chance to contribute and get feedback!)
 - Part 2: Lecture introduction to this week's problems. Explanation of related methods.
 - Part 3: Lab work to begin writing code for this week's problems.
 - Parts 2 and 3 will be mixed together.
- This isn't a course designed to teach you R itself, but Google is your friend. If you don't know R use one of the online texts, and be prepared to do some extra work outside of class. Use the class (i.e., us) as a chance to get help debugging.

How the course will be examined

- Every few weeks you will be introduced to an examinable assignment or two.
- You have two(ish) weeks to work on it.
- You then have to turn in a report written in Rmarkdown, explaining what you did and discussion your results. Your Rcode and should be included in that file, as should a file showing the output (.md, .html or .pdf)
- I may run the code on test datasets.
- There will be 5 of these examinable projects.
- **The last of these may also involve small-group presentations on exam day:**
 - Short presentation about a topic we haven't covered in the course: e.g. fancy MCMC versions, (e.g. Hamiltonian MCMC), Empirical Bayes methods, Machine Learning, ...), *or a related project from your own research.*

Grading: 80% projects + 20% final presentation

Proposed assignment schedule

- Assignment 1: Lecture 2; due Lecture 4;
- Assignment 2: Lecture 4; due Lecture 6;
- Assignment 3: Lecture 7; due Lecture 9;
- Assignment 4: Lecture 10; due Lecture 12;
- Assignment 5: Lecture 12; due end of semester.

How will the projects be graded

1. How well the algorithms perform on the test data.
 2. ‘Style’ - How well is the code explained and commented; How easy it is for me to follow what is going on?
-
- I will grade it in such a way so as not to disadvantage those of you who are new to R. So, no extra marks for using particularly clever R tricks (although I encourage you to do so if you can).
 - But I do want you to annotate your code and output. Present your results as a research report, with explanation of what you are doing and what the output shows.

What I would like from you:

- Your participation!
- Learn by doing, and then correcting your mistakes (i.e., bugs).
- Celebrate your mistakes! (And then correct them...)
- Share what you have learned, and your problems, with the rest of the class. (Make use of the class as a resource, **upload code so that we can run it, or debug it, or celebrate its wonderfulness, in class**).
- **I encourage you to be willing to show ‘work-in-progress’ to the class during the lab.**
- I will **not** be taking notes on who has made the most, worst, funniest,... mistakes. The **only** things that factor into your final grade are the projects write-ups that you turn in and the final group presentations.

Github

- Will be using GitHub to handle code sharing and submission: <https://github.com/PM520-Spring2022>
- You will receive invites to clone repositories there before each class. Those will be the repos we will use in class.
- My repositories there are public, yours will be private.
- *Guide to git/github:* <https://guides.github.com/introduction/git-handbook/>

Github

- Github repo structure:
 - “General-course-info”: some info re, Github.
 - “Lecture-slides”: I will post each week’s slides there (probably on the Friday before Monday’s class).
 - “Useful-Tips”: A place for you to put useful things you have learned that you think others might like to know.

Random questions

- Do we want to do introductions?
- Do we want to think about running of the class or lab in the courtyard once we get back to in-person teaching?
- How much experience have you had with R?
- How much experience have you had with Github?
- Anything else?

Analysis Method 1: Monte Carlo Estimation

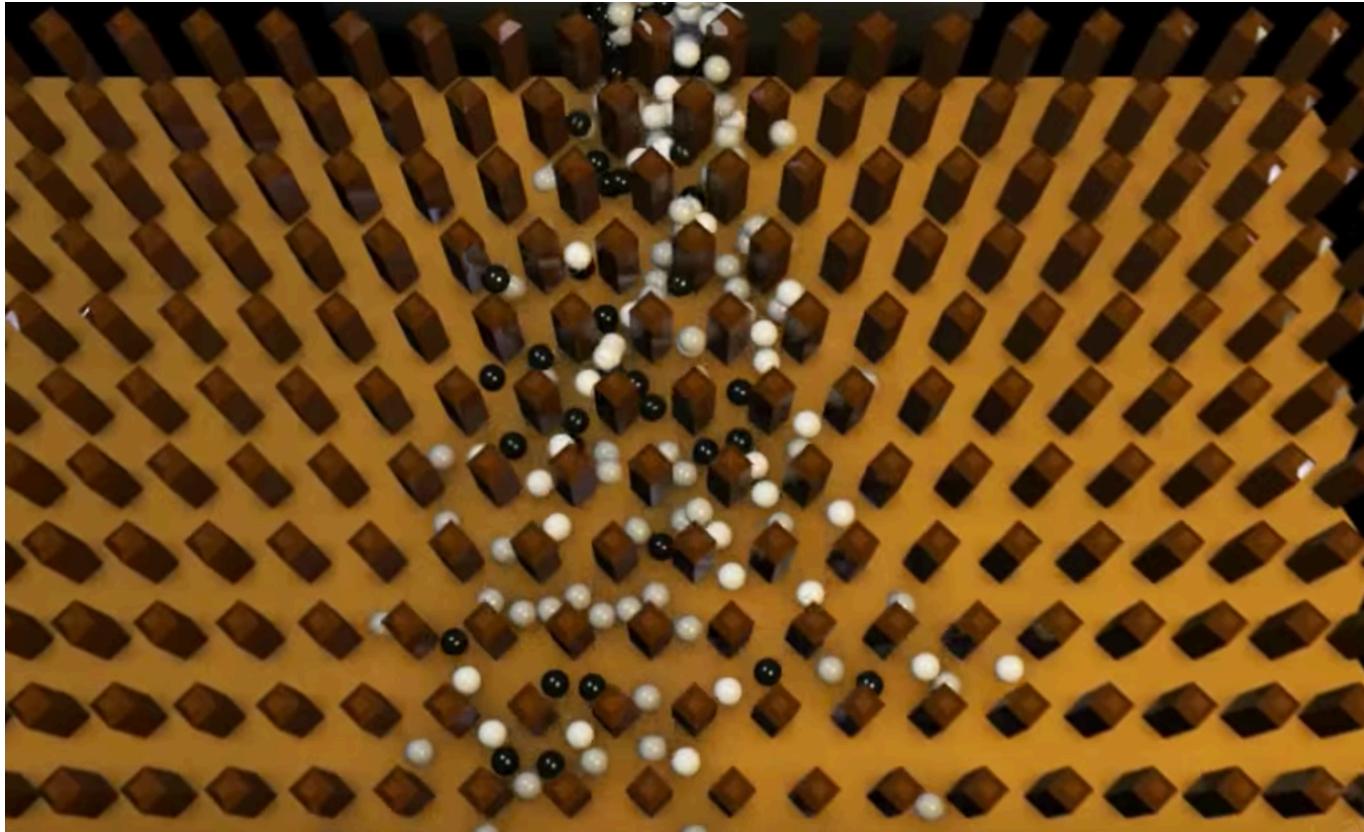
- Massive random simulation of process of interest.
- Estimate either:
 - the probability of an event, A , [i.e., $P(A)$] as the proportion of the simulations in which the event of interest occurs (“empirical estimate” of probability].
 - the expected value of a random variable by the average value it takes over a number of simulations
- Example 1: Toss coin repeatedly to estimate $P(\text{'heads'})$ as the proportion of tosses that come down as a ‘head’.
- Example 2: How many “Runs”
 - 011010100010110111001010

Monte Carlo



- “Monte Carlo simulation”.
- Massive simulation to derive empirical approximation to an object of interest.
- Estimate either:
 - the probability of an event, A , [i.e., $P(A)$] as the proportion of the simulations in which the event of interest occurs [“empirical estimate” of probability].
 - the expected value of a random variable by the average value it takes over a number of simulations

Galton Board



<https://www.youtube.com/watch?v=3m4bxse2JEQ>

<https://www.youtube.com/watch?v=UCmPmkHqHXk>

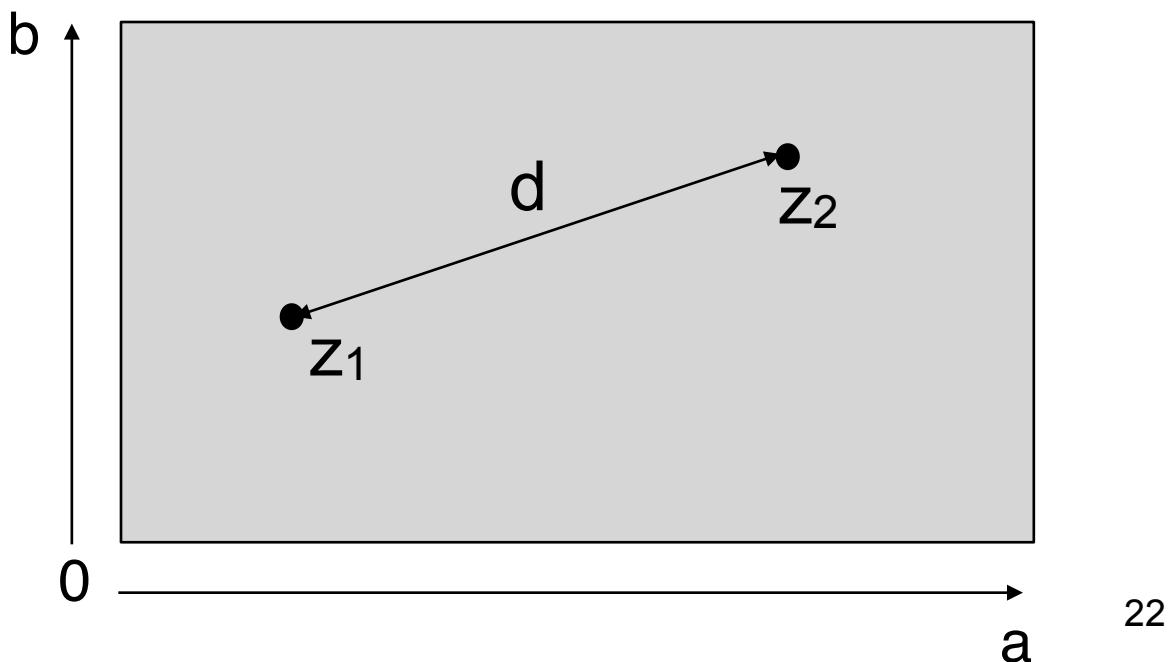
- Advantages of Monte Carlo simulation:
 - Simple.
 - Intuitive.
 - Simulation is easier than calculation (so can use complicated (i.e., realistic) models, for which exact calculation of solutions is impossible).
- Disadvantages:
 - *Estimates* the answer.
 - Inefficient, but has become practical due to rise in computational horse-power.

Random number generators

- e.g., Need to generate a random sequence of “Heads” (1s) and “Tails” (0s).
- A computer cannot truly do this.
- Instead we use a pseudorandom number generator. (For a nice intro see <https://www.youtube.com/watch?v=C82JyCmtKWg>).
- These produce a list of numbers that pass various tests of randomness (but are in fact deterministic).
- In R, we use a function called `runif(n, min = 0, max = 1)`.
- Random number generators use a ‘seed’. In R this is controlled via `set.seed()`.
- If you run a program twice with the same seed, then `runif()` (and other random functions) will produce exactly the same results. (Why would we want to be able to this?)

Monte Carlo Application 1: Distance between points

- Suppose we have a rectangle $[0,a] \times [0,b]$
- If we generate two points, z_1 and z_2 , randomly in the rectangle, what is the expected distance, d , between them?



Procedure

- Clone the code in the “Distance-Finder” repository using your Github classroom invite (received via Blackboard yesterday), [or using the methods we used in the Week0-Hello-World exercise].
- Commit your work regularly.
- When you are done, push it back to Github (using either method from “Hello-World”). **Create an issue with my ID (“@pmarjora” in it)** to let me know your code is ready for review. It might also work to use my ID in your commit message if you commit manually (probably not if you do it using Rstudio?)
- I will give you ~30 minutes to work on the problem and then review submitted answers or present my own solutions.

Procedure

- While online I will create breakout rooms that you can use to discuss the problem, your code, or your results, if you would like to do that.
- *Should groups be random or self-determined?*
- Yue and I are also available to help you. Please use that help! While online we will create breakout rooms for that purpose, so that we can deliver that feedback one-on-one without disturbing everyone else. Send us a private message if you'd like our help, comments, etc. (or we could just go and sit in our virtual offices?)
- We could randomly visit you in your groups as well. Would you like us to do that?

Pseudocode (DistanceFinder.Rmd on Github in repo Week1--InClassExercise1_Distance-Finder)

```
#How many simulations to run?  
NumberOfSims<-1000  
#What is the size of the grid?  
a<-10  
b<-5  
set.seed(123) # set the seed for the random number generator  
  
#Declare a variable to keep track of the sum of the distances across all simulations  
Distance<-0  
  
for (i in 1:NumberOfSims){  
  # generate z1 (so generate it's x and y coordinates)  
  z1<-cbind(runif(1,0,a),runif(1,0,b))  
  
  # generate z2 (likewise)  
  
  # Use Pythagoras' theorem to find the distance between them  
  
  # Add the distance to the variable Distance  
}  
  
# Divide Distance by NumberOfSims to calculate the average distance we observed across all sims  
AverageDistance<-Distance/NumberOfSims  
  
# report that number  
cat("\nOur estimate of the expected distance between the two points is ",AverageDistance)
```

Questions

- How many simulations should you run? (The program will return an answer for any value of NumberOfSims that you give it.)
- Answer:

$$\begin{aligned} E(d) = & \frac{1}{15} \left[\frac{a^3}{b^2} + \frac{b^3}{a^2} + \sqrt{a^2 + b^2} \left(3 - \frac{a^2}{b^2} - \frac{b^2}{a^2} \right) \right] \\ & + \frac{1}{6} \left[\frac{b^2}{a} \operatorname{arccosh} \left(\frac{\sqrt{a^2 + b^2}}{b} \right) + \frac{a^2}{b} \operatorname{arccosh} \left(\frac{\sqrt{a^2 + b^2}}{a} \right) \right] \end{aligned}$$

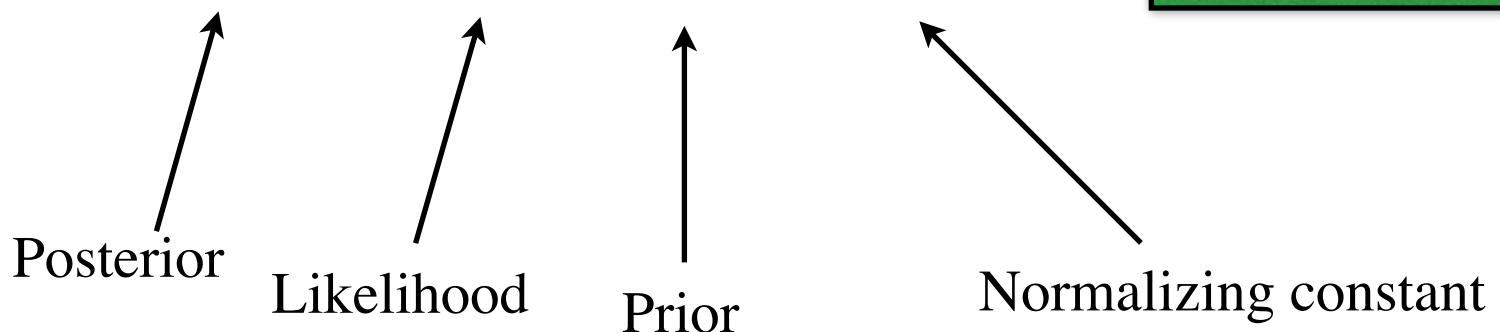
where

$$\operatorname{arccosh}(t) = \log(t + \sqrt{t^2 - 1})$$

So, why do we want to learn
statistical programming anyway?

What Bayesians do for a living

- Given data D,
- Model M,
- Parameter(s) θ .
- Wish to make inference re. $f(\theta|D)$.
- $f(\theta|D) = f(D|\theta) \pi(\theta) / P(D)$



Statistical evaluation of alternative models of human evolution

Nelson J. R. Fagundes^{†‡§}, Nicolas Ray[§], Mark Beaumont[¶], Samuel Neuenschwander^{§||}, Francisco M. Salzano^{†††}, Sandro L. Bonatto^{†,††}, and Laurent Excoffier^{§††}

[†]Laboratório de Biologia Genômica e Molecular, Faculdade de Biociências, Pontifícia Universidade Católica do Rio Grande do Sul (PUCRS), 90619-900 Porto Alegre, RS, Brazil; [‡]Departamento de Genética, Universidade Federal do Rio Grande do Sul, 91501-970 Porto Alegre, RS, Brazil; [§]Computational and Molecular Population Genetics (CMPG), Zoological Institute, University of Bern, CH-3012 Bern, Switzerland; [¶]School of Animal and Microbial Sciences, University of Reading, Reading RG6 6AJ, United Kingdom; and ^{||}Department of Ecology and Evolution, University of Lausanne, Biophore, CH-1015 Lausanne, Switzerland

Contributed by Francisco M. Salzano, August 31, 2007 (sent for review June 1, 2007)

An appropriate model of recent human evolution is not only important to understand our own history, but it is necessary to disentangle the effects of demography and selection on genetic diversity. Although most genetic data support the view that modern humans originated recently in Africa, it is still unclear if it completely replaced former members of the *Homo* genus, or if some interbreeding occurred during its range expansion. Several scenarios of modern human evolution have been proposed on the basis of molecular and paleontological data, but their likelihood has not been statistically assessed. Using DNA data from 50 nuclear loci,

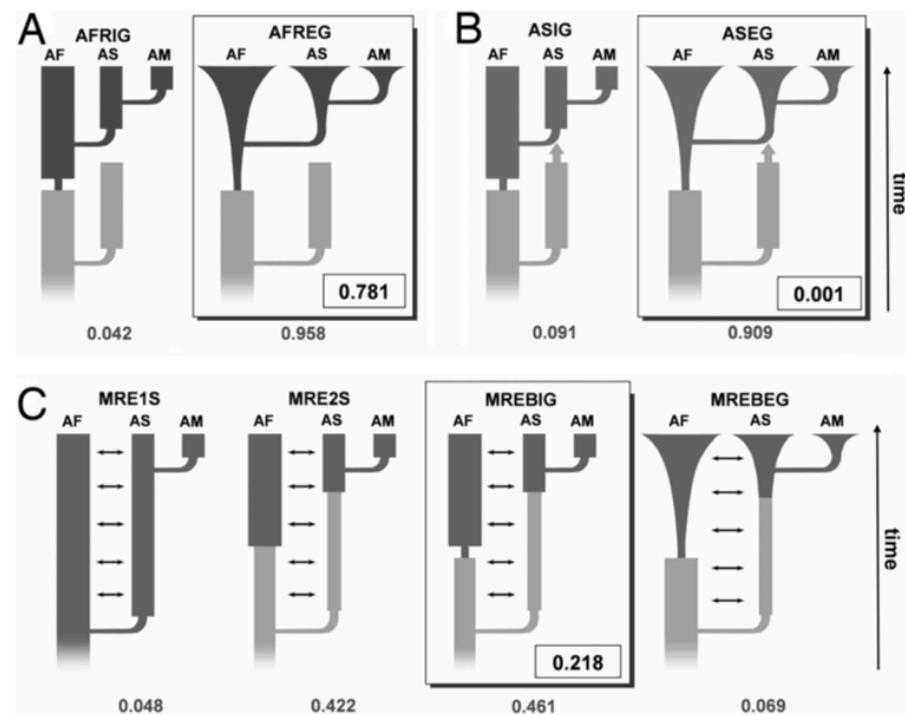
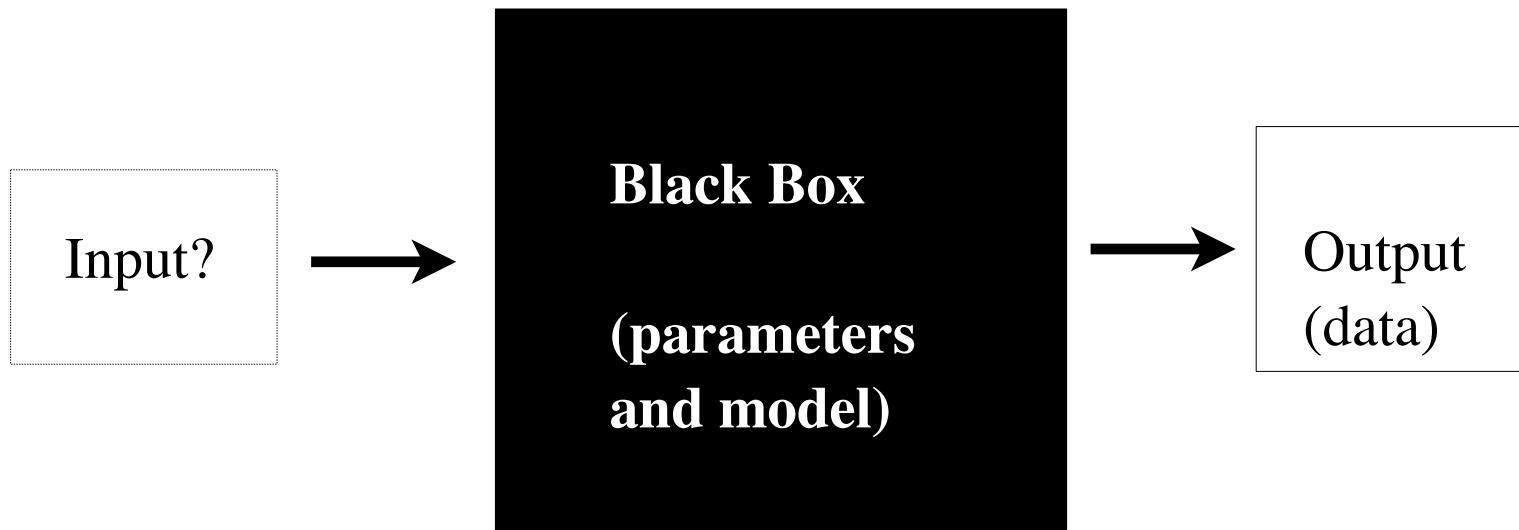


Fig. 1. Alternative scenarios of human evolution. (A) African replacement models: AFRIG, African replacement with instantaneous population growth; AFREG, African replacement with exponential population growth. (B) Assimilation models: ASIG, African replacement with gene flow; ASEG, African replacement with gene flow and exponential growth. (C) Migration models: MRE1S, migration between Africa and Asia; MRE2S, migration between Africa and America; MREBEG, migration between Africa, Europe, and America; MREBIG, migration between Africa, Europe, and India. Likelihood values are indicated in each panel.





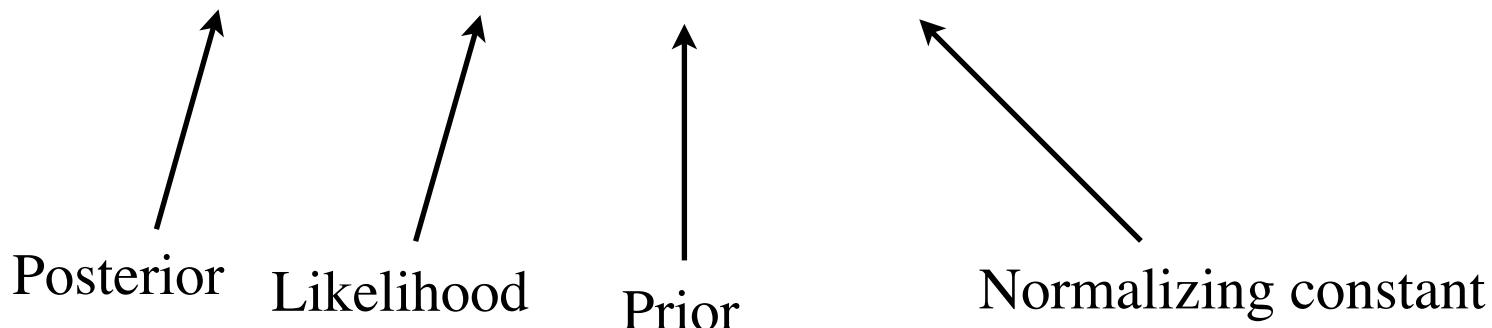
Reverse-engineering



Goal: What is in the box?

What we do for a living

- Given data D,
- Model M,
- Parameter(s) θ .
- Wish to make inference re. $f(\theta|D)$.
- $f(\theta|D) = f(D|\theta) \pi(\theta) / P(D)$



Intractability of likelihood

- Has led to greater use of computers!
- But, even then, tractability can remain an issue.
- Two possible responses:
 - Simplify the model so that it again becomes possible to calculate the likelihood
 - Keep the ‘realistic’ model, but simplify the analysis by approximation

Intractability of likelihood

- Has led to greater use of computers!
- But, even then, tractability can remain an issue.
- Two possible responses:
 - Simplify the model so that it again becomes possible to calculate the likelihood
 - Keep the ‘realistic’ model, but simplify the analysis by approximation

“Far better an approximate answer to the right question, which is often vague, than an exact answer to the wrong question, which can always be made precise.” (Tukey).

- Move to simulation-based (model-based) analysis

Monte Carlo



- “Monte Carlo simulation”.
- Massive simulation to derive empirical approximation to an object of interest.
- e.g. tossing a coin to estimate the probability of coming down “heads”.
- Spinning a roulette wheel to determine the probability of ‘red’.
- **Principle:** Estimate the probability/lielihood of an event by the proportion of iterations in which it occurs.

- Advantages of Monte Carlo simulation:
 - Simple.
 - Intuitive.
 - Simulation is easier than calculation (so can use complicated (i.e., realistic) models, for which exact calculation of solutions is impossible).
- Disadvantages:
 - *Estimates* the answer.
 - Inefficient, but has become practical due to rise in computational horse-power.

- A variety of methods have built upon these kinds of ideas:
 - Likelihood estimation
 - Markov Chains
 - Markov chain Monte Carlo
 - Hidden Markov Models
 - Accept/Reject algorithms
 - Importance Sampling
 - Approximate Bayesian Computation

Example Applications

Evolution, 2009 April • 63(4): 907–926

Biol Philos
DOI 10.1007/s10539-013-9391-1

The phylogeography debate and the epistemology of model-based evolutionary biology

Alfonso Arroyo-Santos · Mark E. Olson · Francisco Vergara-Silva

Received: 20 March 2012 / Accepted: 17 June 2013
© Springer Science+Business Media Dordrecht 2013

Abstract Phylogeography, a relatively new subdiscipline of evolutionary biology that attempts to unify the fields of phylogenetics and population biology in an explicit geographical context, has hosted in recent years a highly polarized debate related to the purported benefits and limitations that qualitative versus quantitative methods might contribute or impose on inferential processes in evolutionary biology. Here we present a friendly, non-technical introduction to the conflicting methods underlying the controversy, and exemplify it with a balanced selection of

¹School of Animal and Microbial Sciences, University of

Reading, Whiteknights, PO Box 229, Reading, RG6 6AU, UK

population genetics. We also examine

Animal behavior

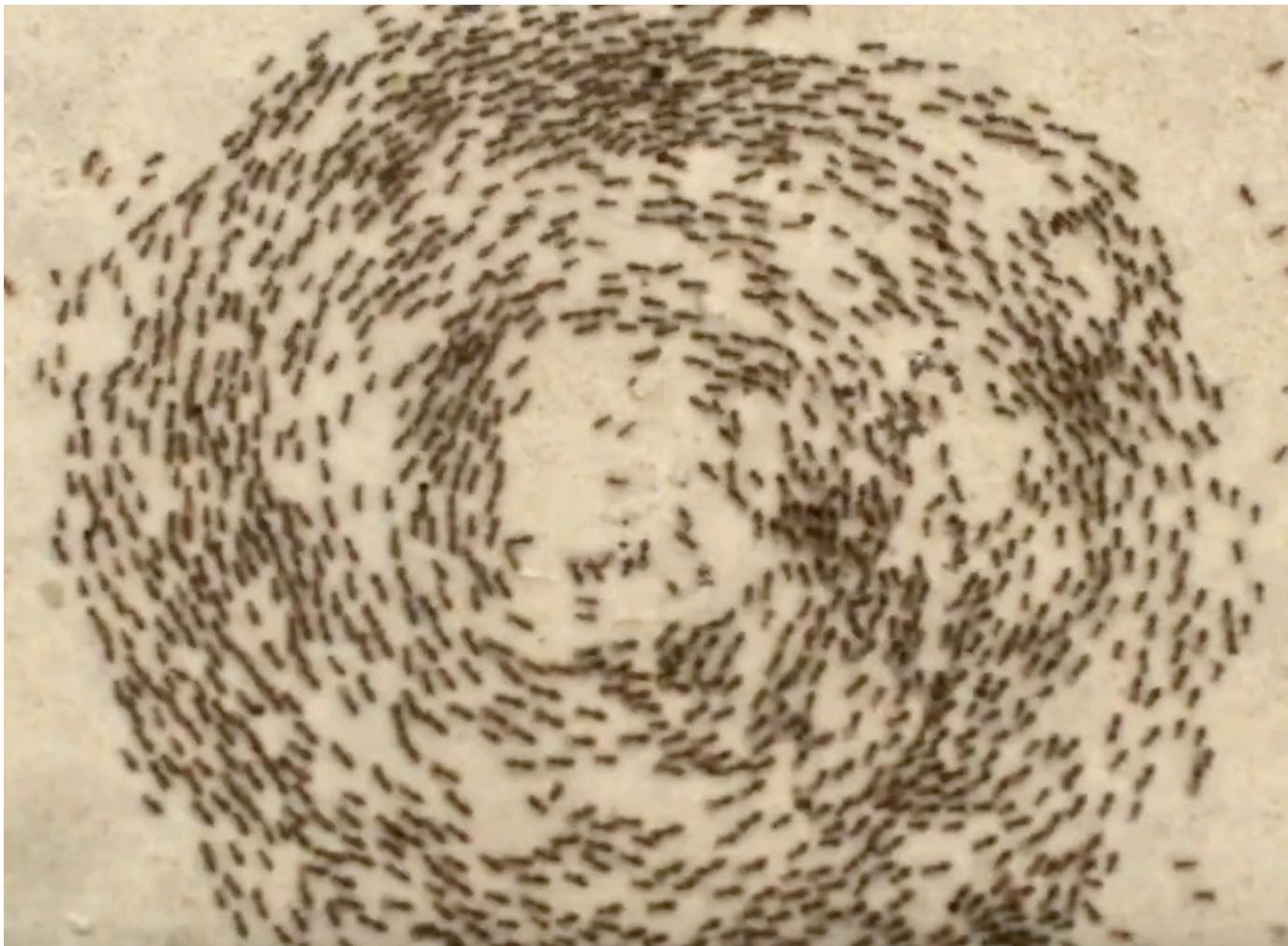
A wide-angle photograph capturing a massive flock of birds in flight against a hazy, light sky. The birds are concentrated in the upper half of the frame, appearing as a dense, swirling mass. Below them, the silhouette of bare trees stands in the foreground, and a dark, indistinct horizon line marks the base of the scene.

Tuna tornado



<https://www.youtube.com/watch?v=D6HdolsLMFg>

Ant mill



<https://www.youtube.com/watch?v=LEKwQxO4EZU>

Wisdom of crowds

- An aggregation of ‘not very smart’ individuals can make good decisions.
- In groups, individuals move more slowly but more often in the right direction. Result: Destination is reached more quickly.
- Counterexamples: Congress; IRAQ War
- Pre-requisites:
 - Independence of decisions (no peer pressure)
 - Diversity
 - Lack of bias





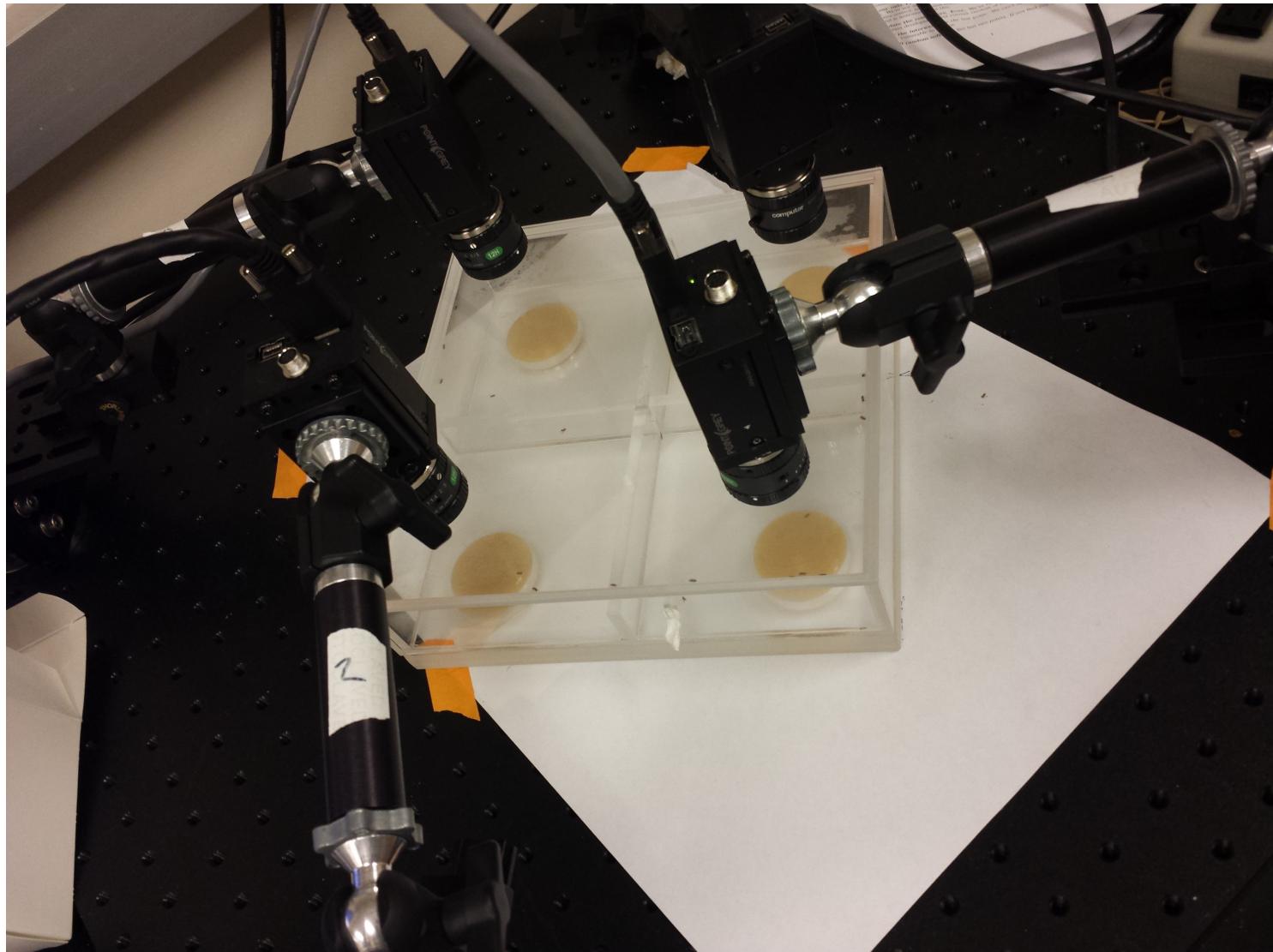
Mem Cogn (2011) 39:914–923
DOI 10.3758/s13421-010-0059-7

The wisdom of the crowd playing The Price Is Right

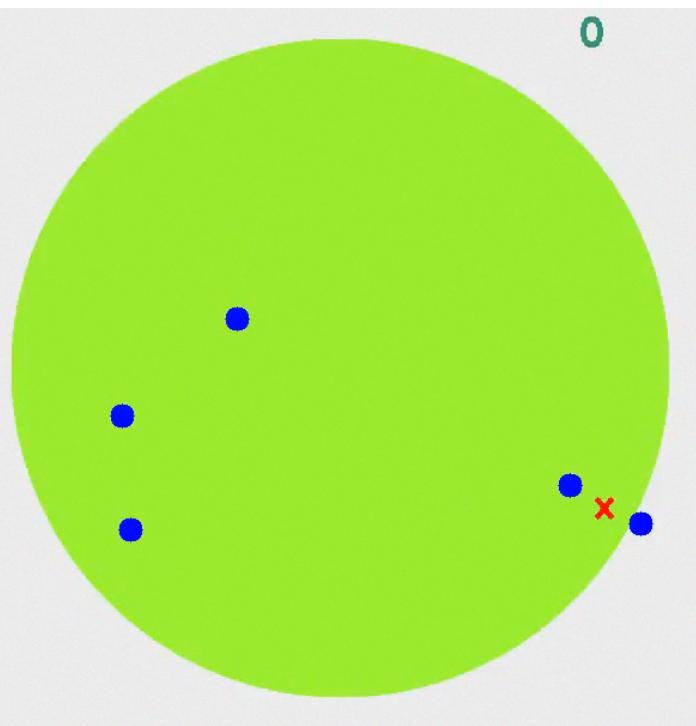
Michael D. Lee · Shunan Zhang · Jenny Shi

Abstract In The Price Is Right game show, players compete to win a prize, by placing bids on its price. We ask whether it is possible to achieve a “wisdom of the crowd” effect, by combining the bids to produce an aggregate price estimate that is superior to the estimates of individual players. Using data from the game show, we show that a wisdom of the crowd effect is possible, especially by using models of the decision-making processes involved in bidding. The key insight is that, because of the competitive nature of the game, what people bid is not necessarily the same as what they know. This means better estimates are formed by aggregating latent knowledge than by aggregating observed bids. We use our results to highlight the

Social behavior of *Drosophila*



Our data





Thriller Championships
Training Room
Home



Fruit Fly Fight Club

Video Highlights



[High Intensity](#)



[Mid Intensity](#)



[Low Intensity](#)

Most files are RealPlayer. See [real.com](#) to download RealPlayer for PC or Mac or [apple.com](#) for QuickTime.

THRILLER CHAMPIONSHIPS: Researchers bet on fruit fly fights to expose underlying biology of aggression

Round by round, move by move, video replay of 75 fruit fly fights reveals statistically significant patterns of normal fighting behavior

Male fruit flies that pick a fight are likely to win the battle. Losing fruit flies don't give up easily, even if it takes them longer to re-engage their foes after a particular bruising encounter. Most fruit fly fights are resolved before they escalate to intense physical contact.

These are some of the results from a study of 75 fruit fly fights staged at a neurobiology laboratory at Harvard Medical School (HMS) in Boston, Mass. USA. The researchers videotaped and analyzed the fights to learn more about aggression. Future studies in the lab will use genetic mutations to investigate the neurobiology of aggression and also explore the gene expression consequences of winners and losers. The first report in the April 16 *Proceedings of the National Academy of Sciences* establishes the pattern of normal fruit fly fights.

In the half-hour fights, the fruit flies averaged 27 encounters of about 11 seconds each. Each scuffle moved so fast that the researcher-referees needed slow-motion instant replay to score the fights. Harvard University undergraduates Selby Chen and Ann Yeelin Lee scored about 9,000 individual moves in 2,000 skirmishes. In case of doubt or disagreement, HMS neurobiology professor Edward Kravitz determined the final score.

Scoring System

High Intensity



Tussling: Both flies tumble over each other, sometimes leaving food surface

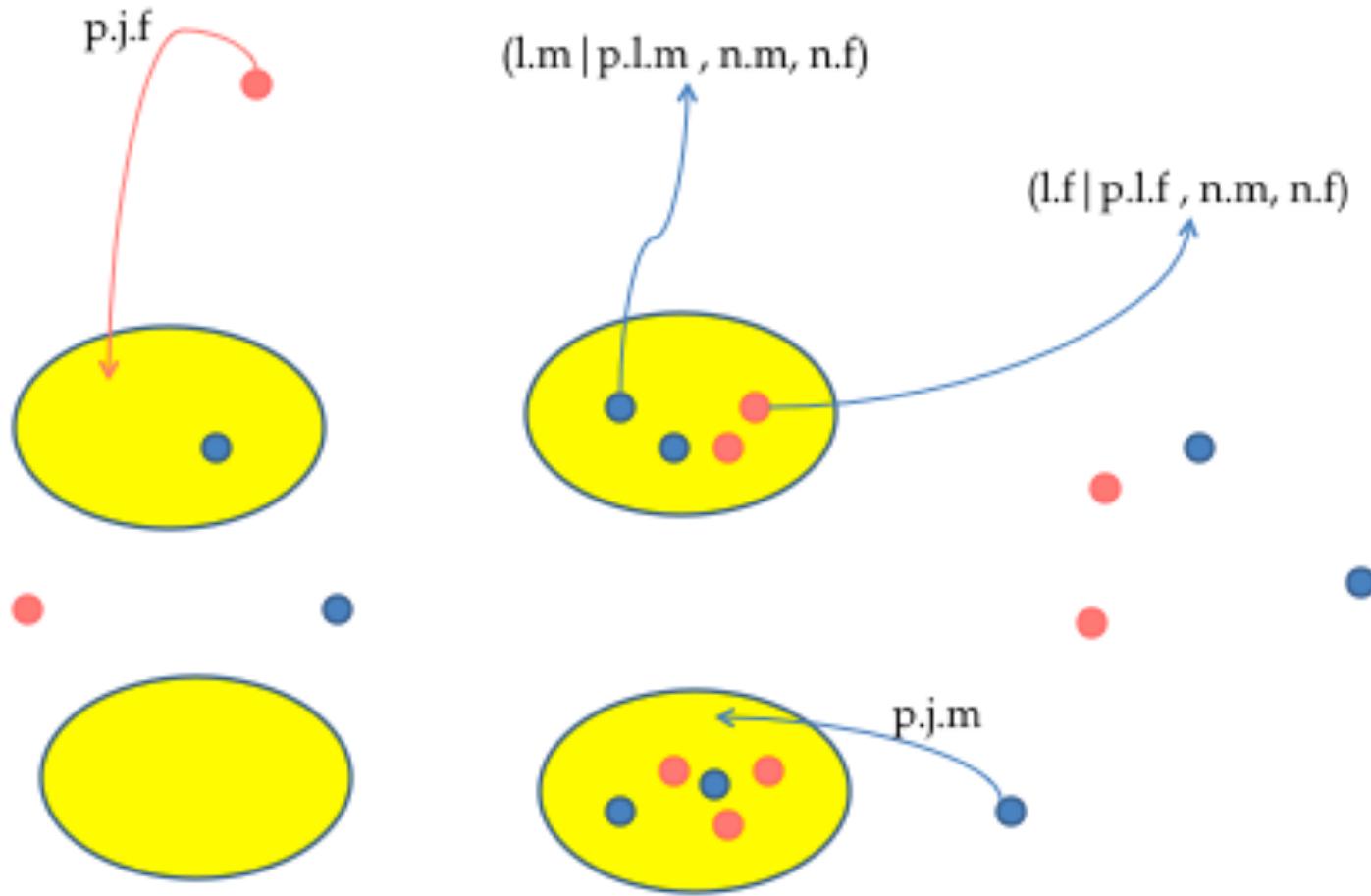


Boxing: Both rear up on hind legs and strike opponent with forelegs

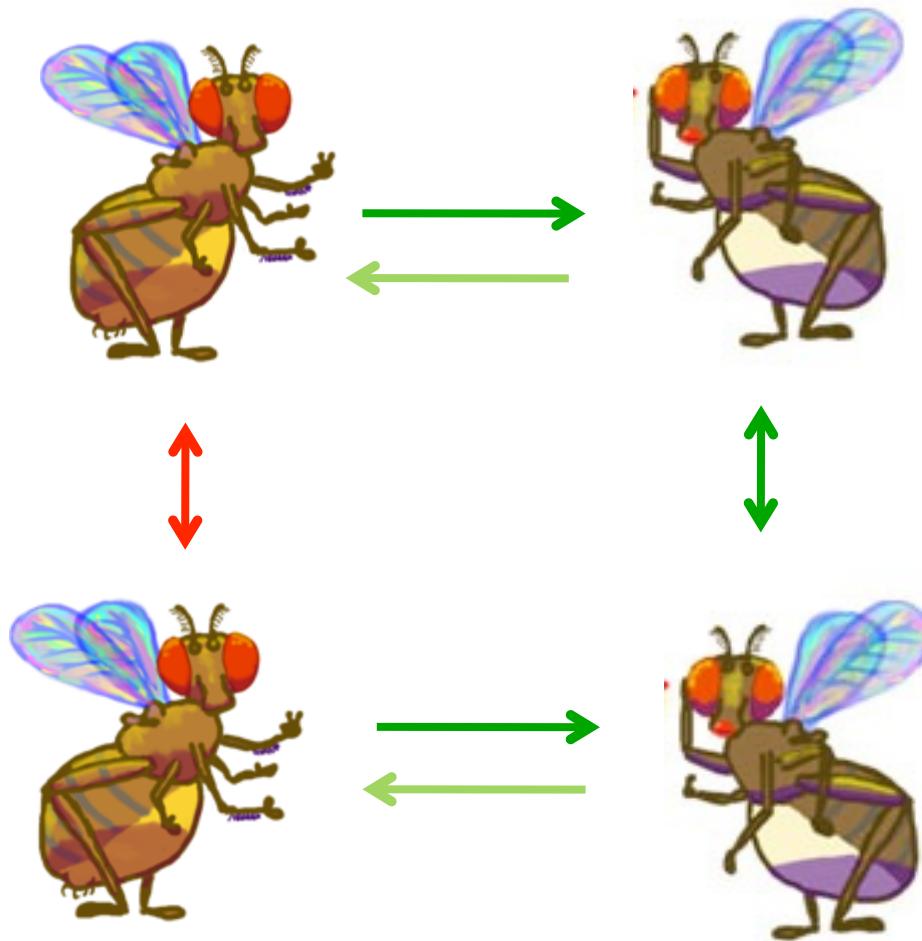


Holding: One grasps the other with forelegs and tries to immobilize it

Agent-based modeling

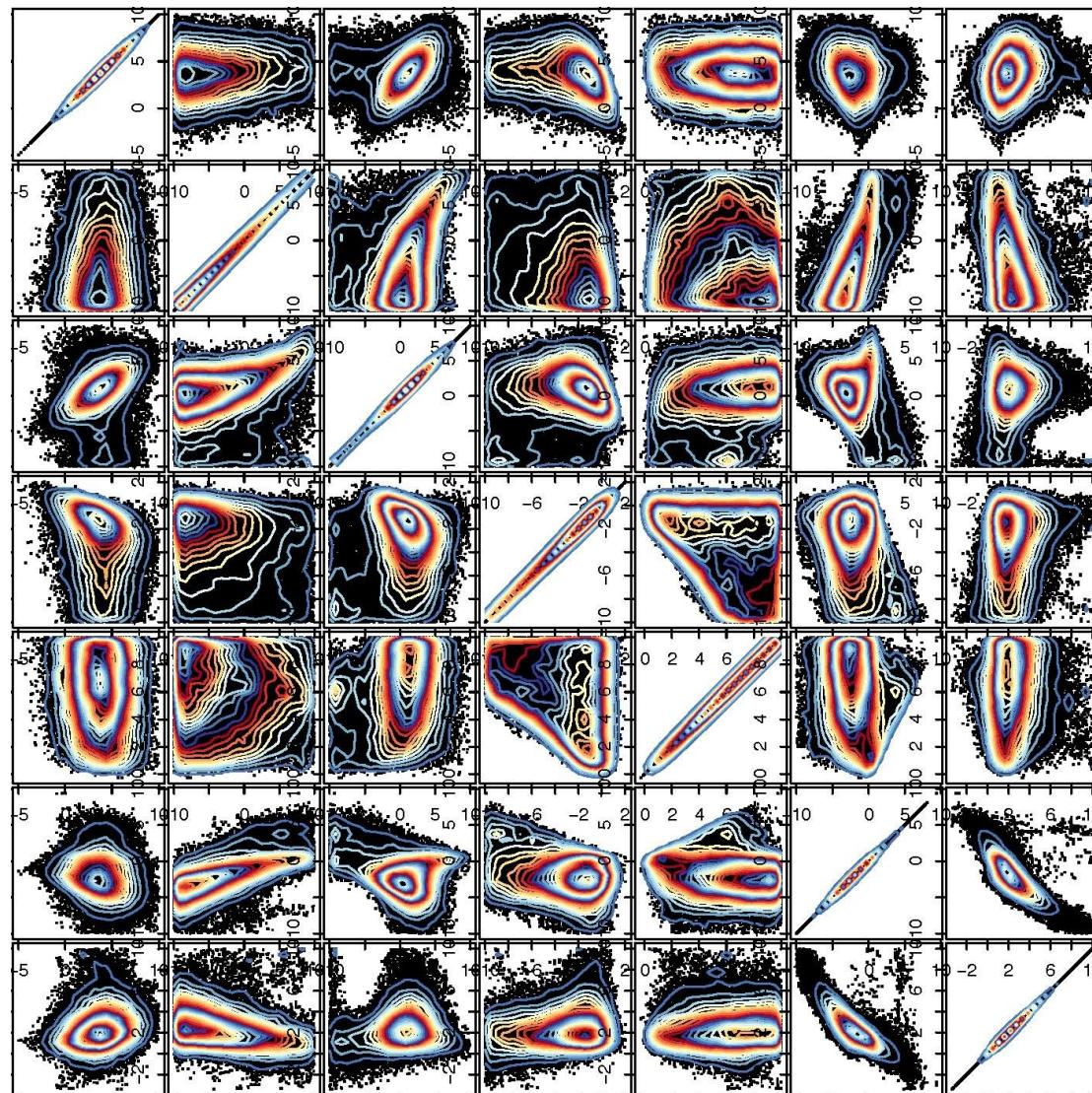


Results - Overview



Foley, Brad R., et al. "A Bayesian Approach to Social Structure Uncovers Cryptic Regulation of Group Dynamics in *Drosophila melanogaster*." *The American Naturalist* 185.6 (2015): 797-808.

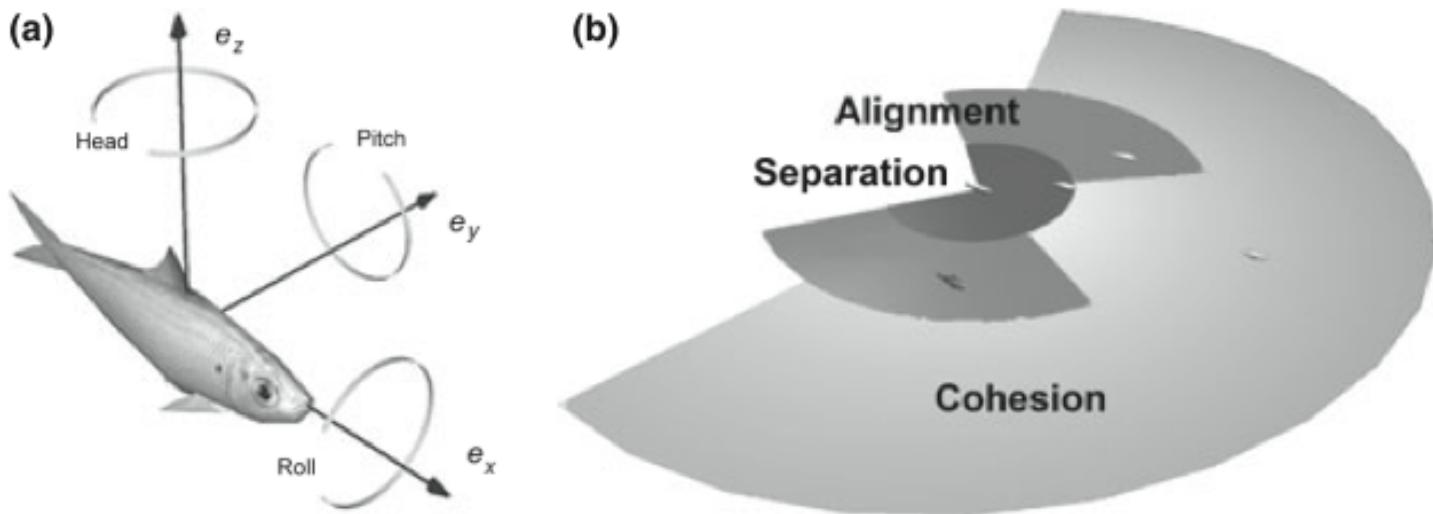
Results





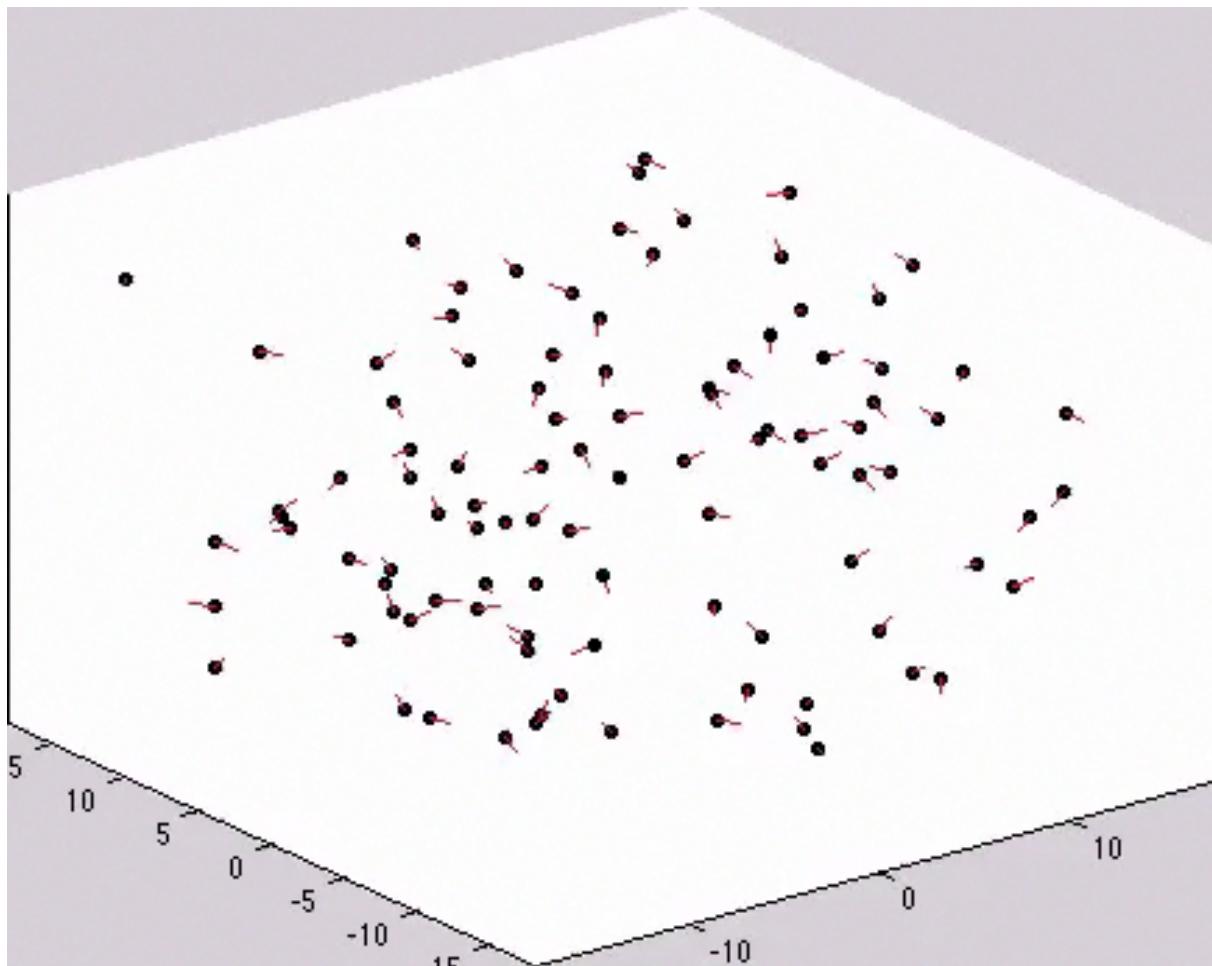
- Faria, et al. **A novel method for investigating the collective behaviour of fish: introducing 'Robofish'.** *Behavioral Ecology and Sociobiology*, 2010; DOI:

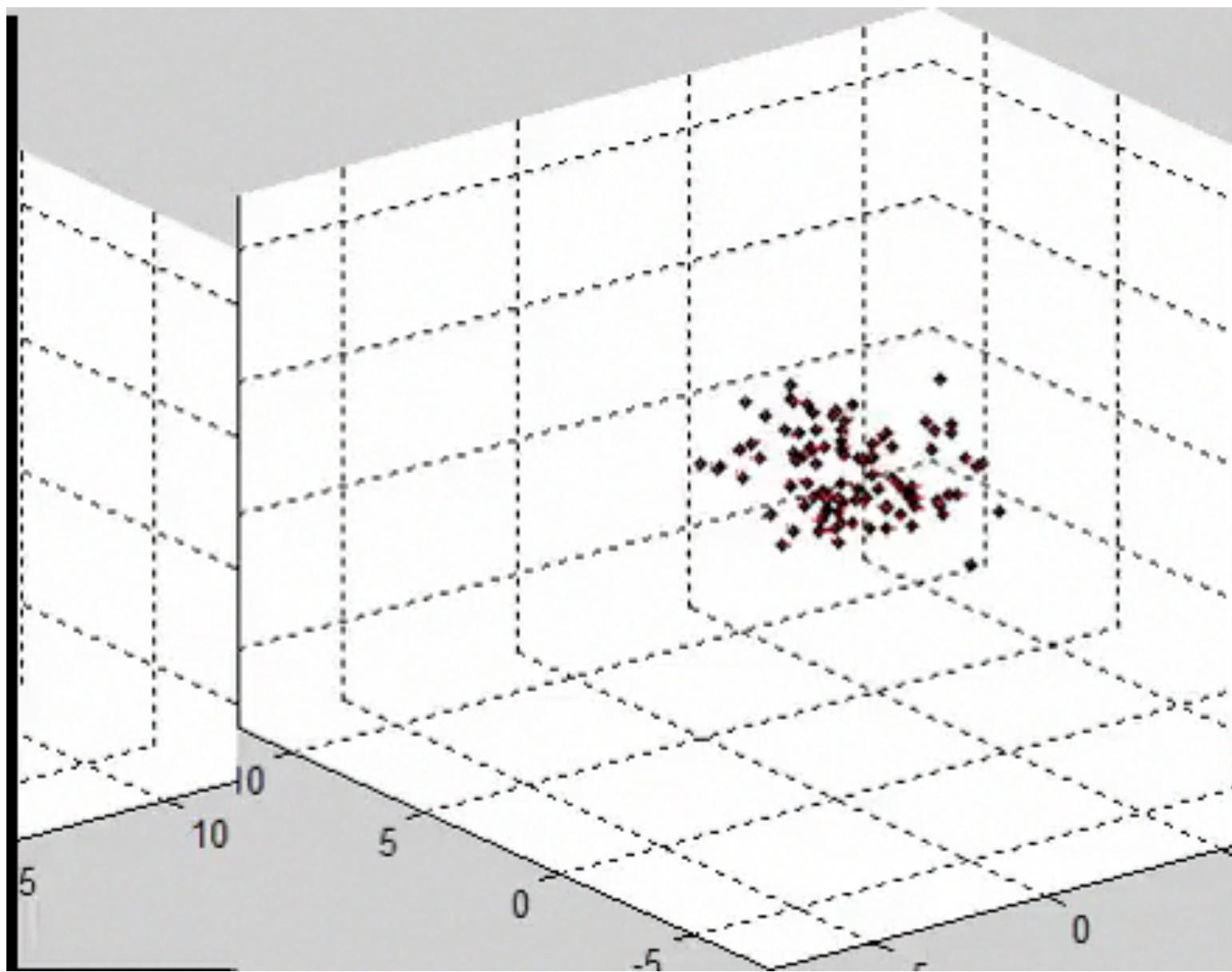
Agent-based model



Hemelrijk and Hildenbrandt.
Ethology 114 (2008) 245–254

Simulation-based analysis





Social Integration of Robots into Groups of Cockroaches to Control Self-Organized Choices

J. Halloy,^{1,*†} G. Sempo,^{1,*} G. Caprari,³ C. Rivault,² M. Asadpour,³ F. Tâche,³
I. Saïd,² V. Durier,² S. Canonge,¹ J. M. Amé,¹ C. Detrain,¹ N. Correll,⁴ A. Martinoli,⁴
F. Mondada,⁵ R. Siegwart,³ J. L. Deneubourg¹

Collective behavior based on self-organization has been shown in group-living animals from insects to vertebrates. These findings have stimulated engineers to investigate approaches for the coordination of autonomous multirobot systems based on self-organization. In this experimental study, we show collective decision-making by mixed groups of cockroaches and socially integrated autonomous robots, leading to shared shelter selection. Individuals, natural or artificial, are perceived as equivalent, and the collective decision emerges from nonlinear feedbacks based on local interactions. Even when in the minority, robots can modulate the collective decision-making process and produce a global pattern not observed in their absence. These results demonstrate the possibility of using intelligent autonomous devices to study and control self-organized behavioral patterns in group-living animals.

www.sciencemag.org SCIENCE VOL 318 16 NOVEMBER 2007

1155

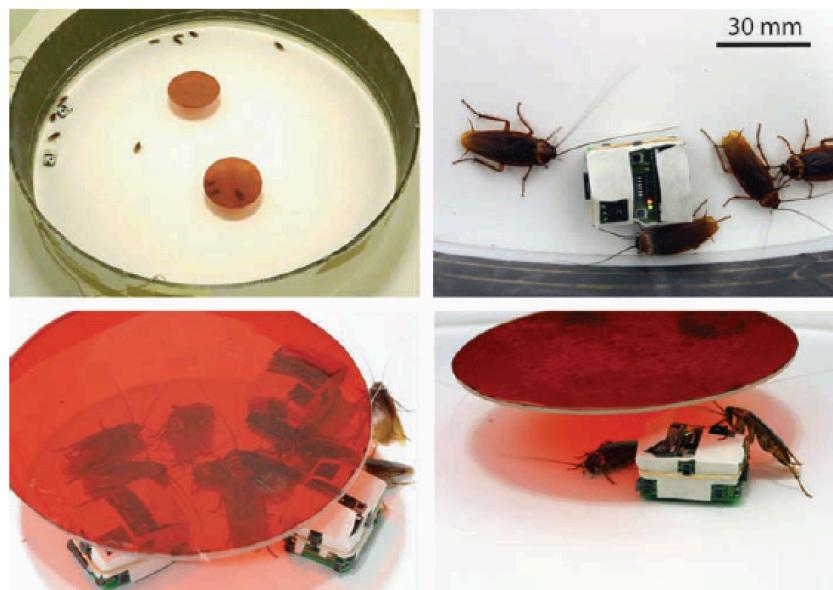
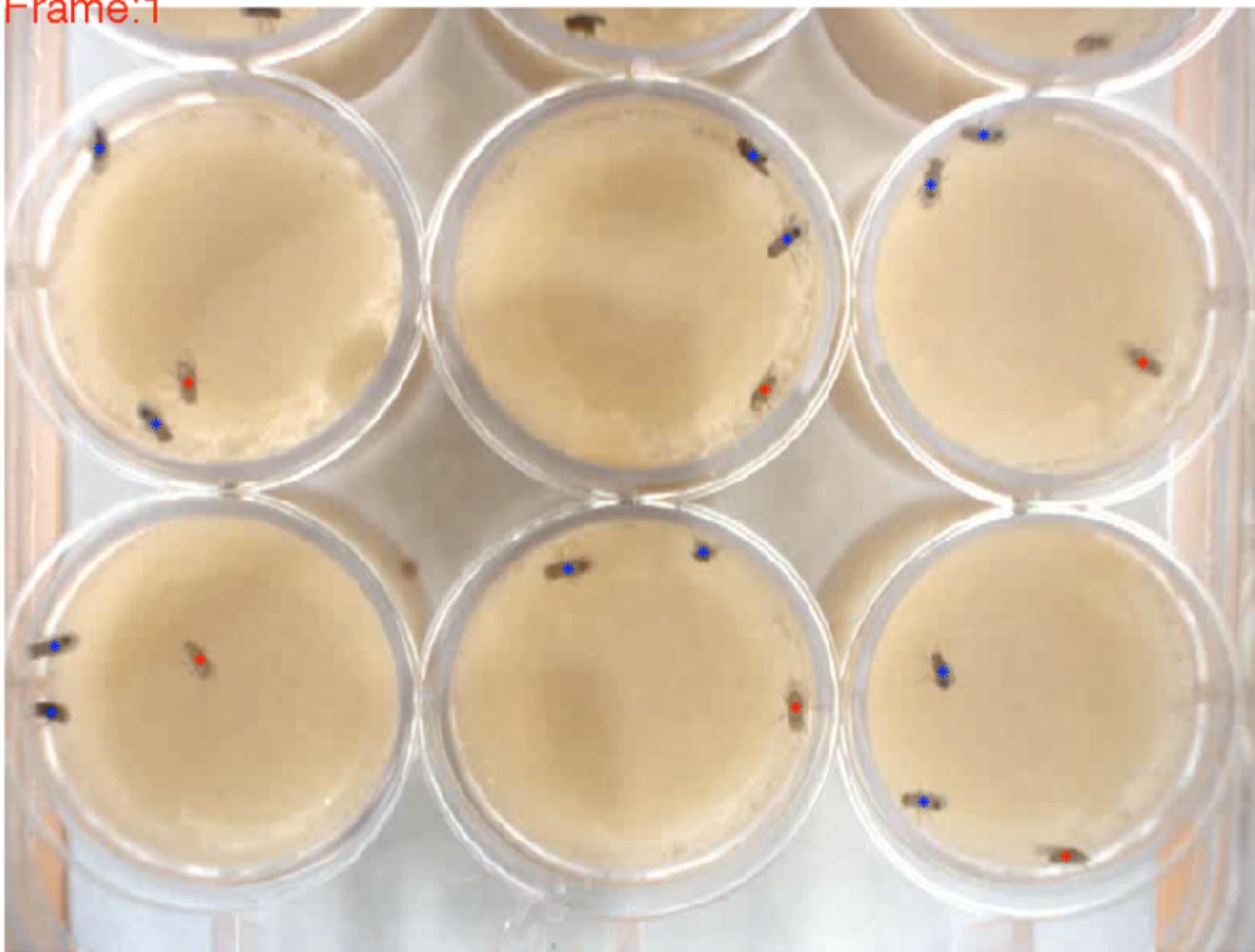


Fig. 1. Experimental setup showing the cockroaches (*Periplaneta americana*) and the robots. Two shelters (150 mm) made of plastic disks covered by red film filters are suspended (30 mm) above the floor of a circular arena (diameter 1 m). The darkness under the shelter is controlled by the number of layers of red film. Cockroaches aggregate under the shelters (18).

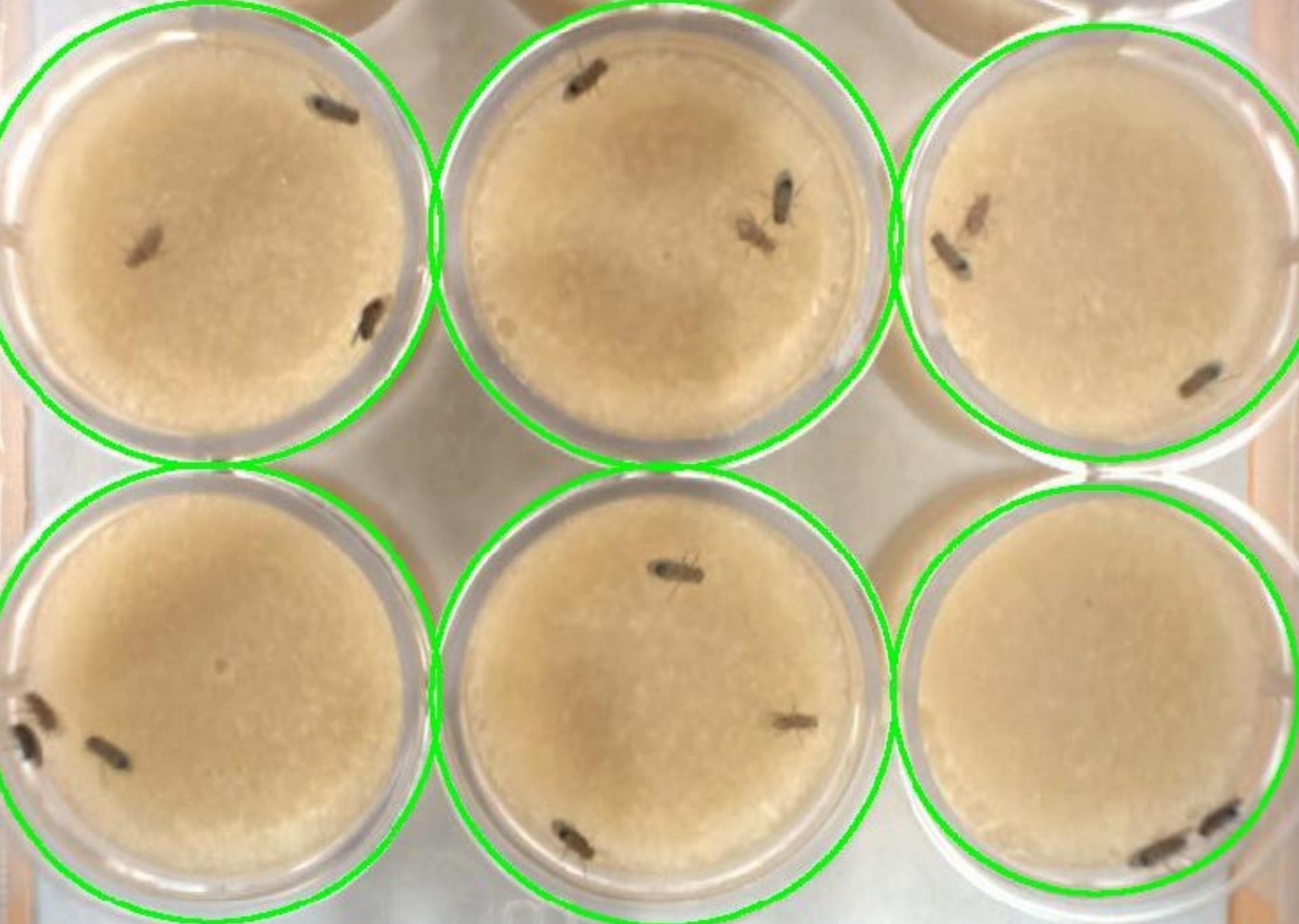
Flies as alcoholics

Frame:1

Tracking of the flies represented with stars



Goal: genetic determinants of alcohol resistance/addiction







Environment 1

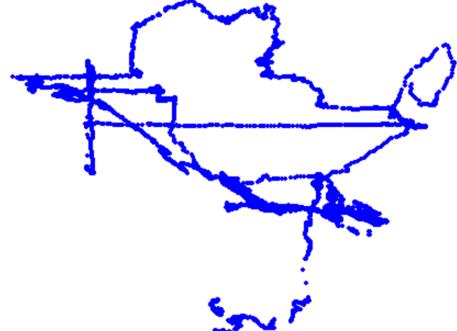
Environment 2

No Ethanol



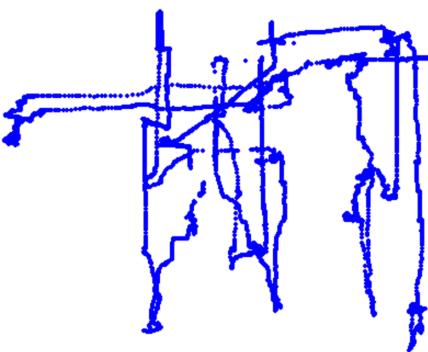
distance between male1 & female

No Ethanol



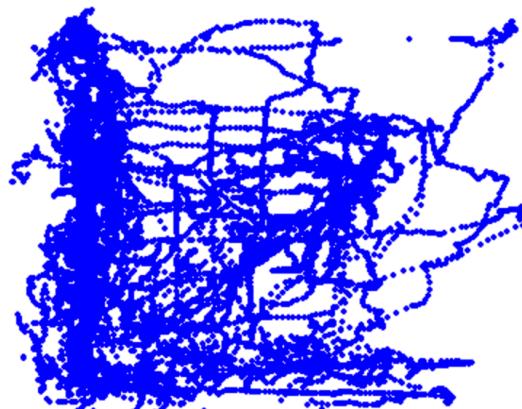
distance between male1 & female

Ethanol



distance between male1 & female

Ethanol



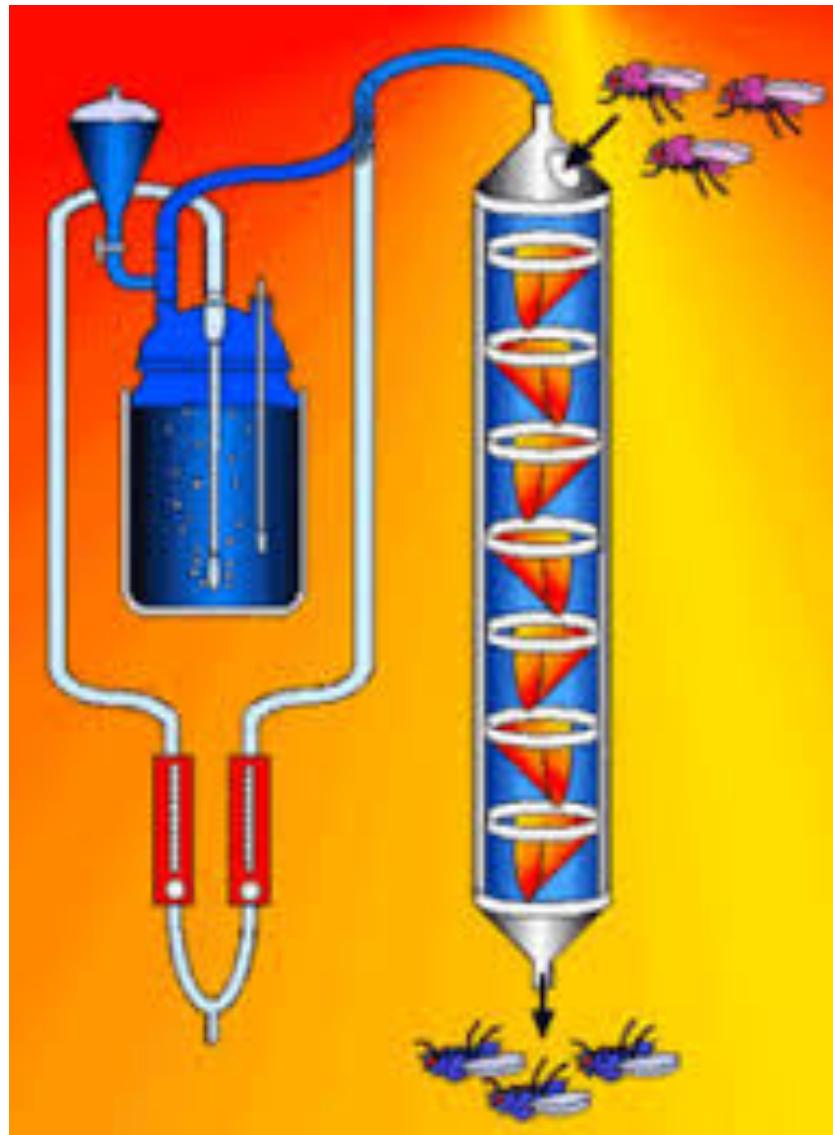
distance between male1 & female

Genotype 1

G x E

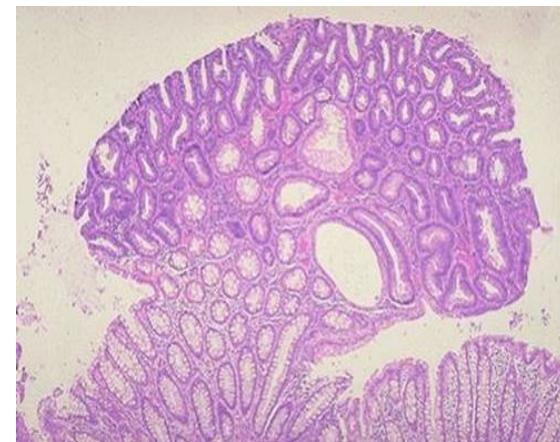
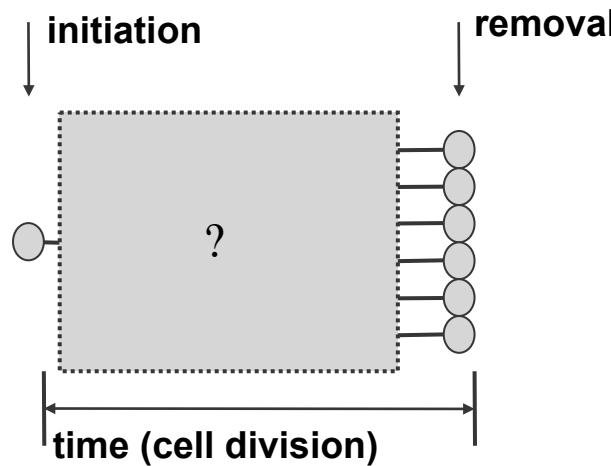
Genotype 2

The Inebriometer



Approximate Bayesian Computation applications: Tumor History

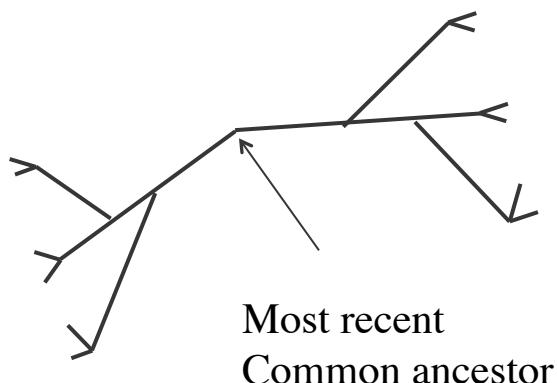
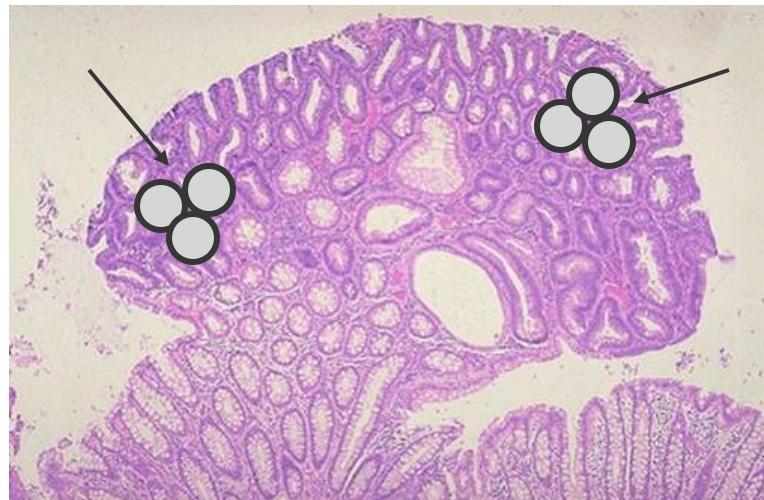
Cancer cell dynamics are not observed



Basic questions:

1. How many times has the tumor undergone cell division?
2. How many long-lived tumor cells (cancer stem cells) are there?
3. Is there an early burst of mutations?

Somatic Cell Trees

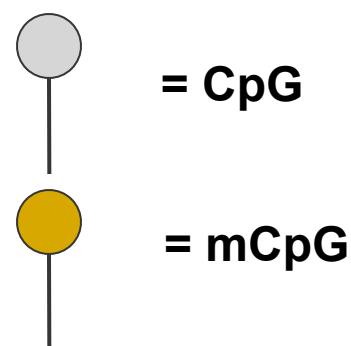
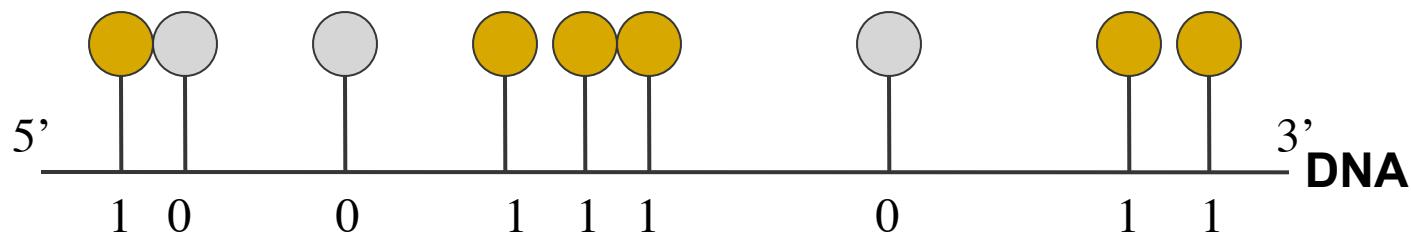


Epigenome comparisons can reconstruct genealogies from regions of a tumor

- DNA methylation patterns are inherited in cell division
- Replication errors introduce small differences between generations

Measuring DNA methylation

Bisulphite Genomic Sequencing



DNA Methylation

A chemical modification of DNA that can silence gene expression

Normal function

X-chromosome inactivation

Genetic imprinting

Abnormal function

Silence tumor suppressor genes in cancer

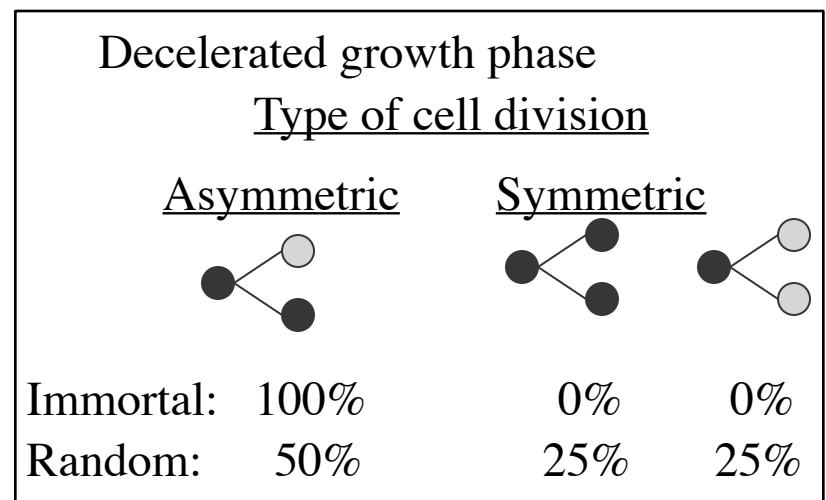
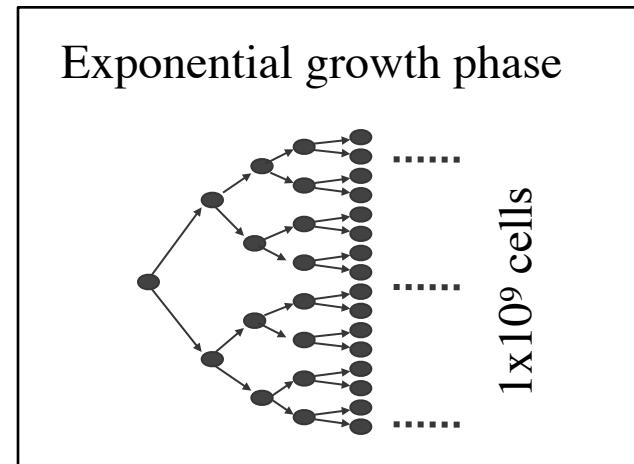
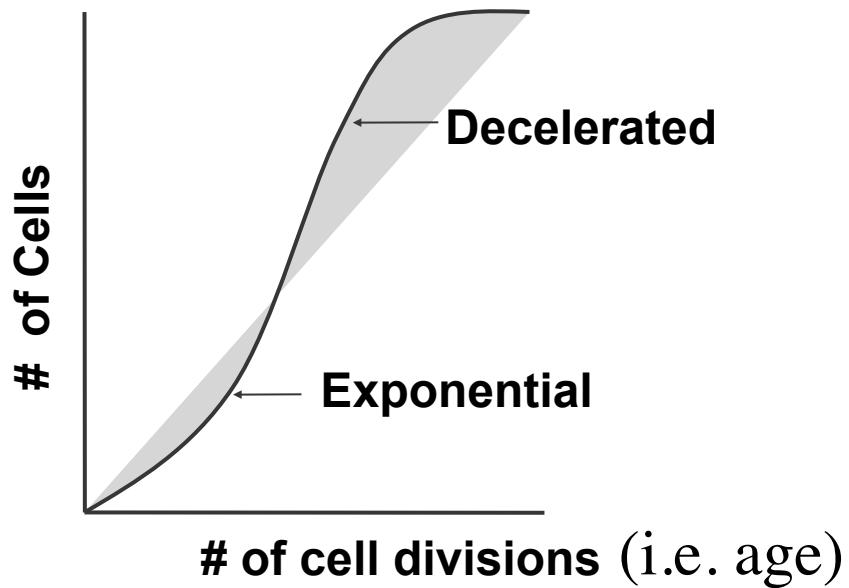
3 Key Features:

- inherited in cell division
- relatively stable
- ‘mutates’ at the right rate for our purpose



- The coloration of tortoise-shell (calico) cats is a visible manifestation of X-inactivation. The "black" and "orange" alleles of a fur coloration gene reside on the X chromosome. For any given patch of fur, the inactivation of an X chromosome that carries one gene results in the fur color of the other, active gene. That's why tortoise-shell cats are (almost always) female.

Tumor growth model



Mutation burst?

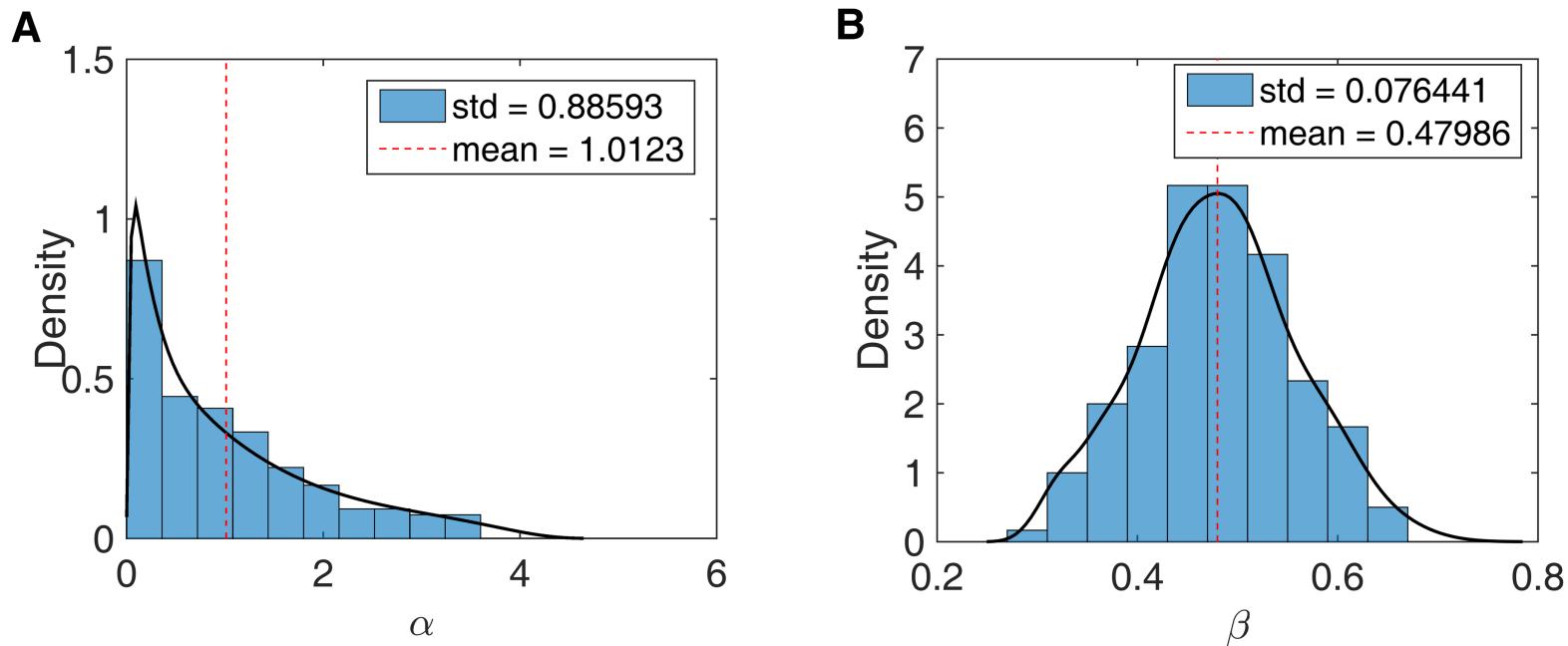
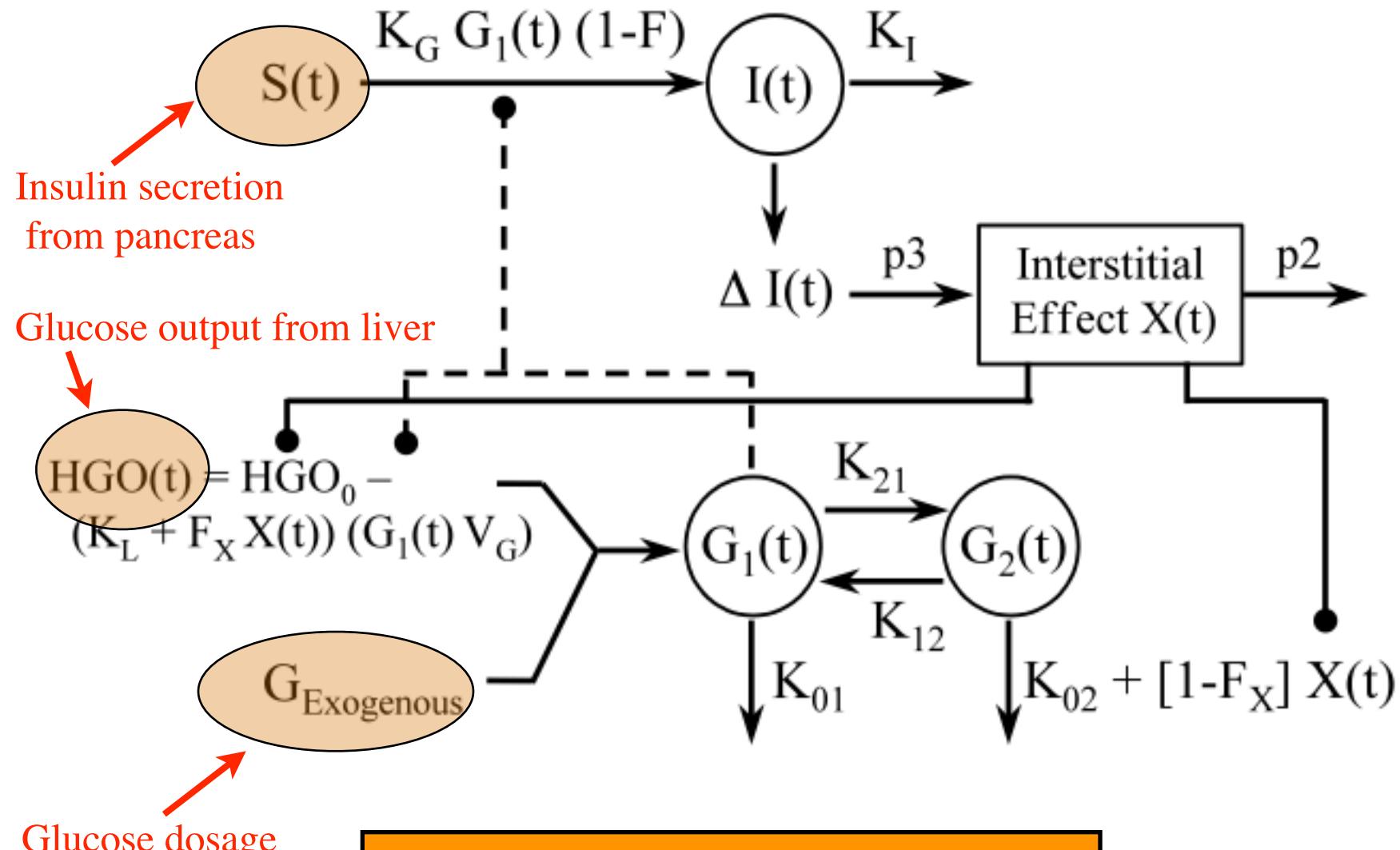


Fig 6. Posterior distributions of mutation rate for tumor U.

(A) mutation rate before gland formation. (B) mutation rate after gland formation. The dashed line indicates the mean of the posterior distribution.

Zhao, J., Salomon, M. P., Shibata, D., Curtis, C., Siegmund, K., & Marjoram, P. (2017). Early mutation bursts in colorectal tumors. *PLoS one*, 12(3), e0172516.

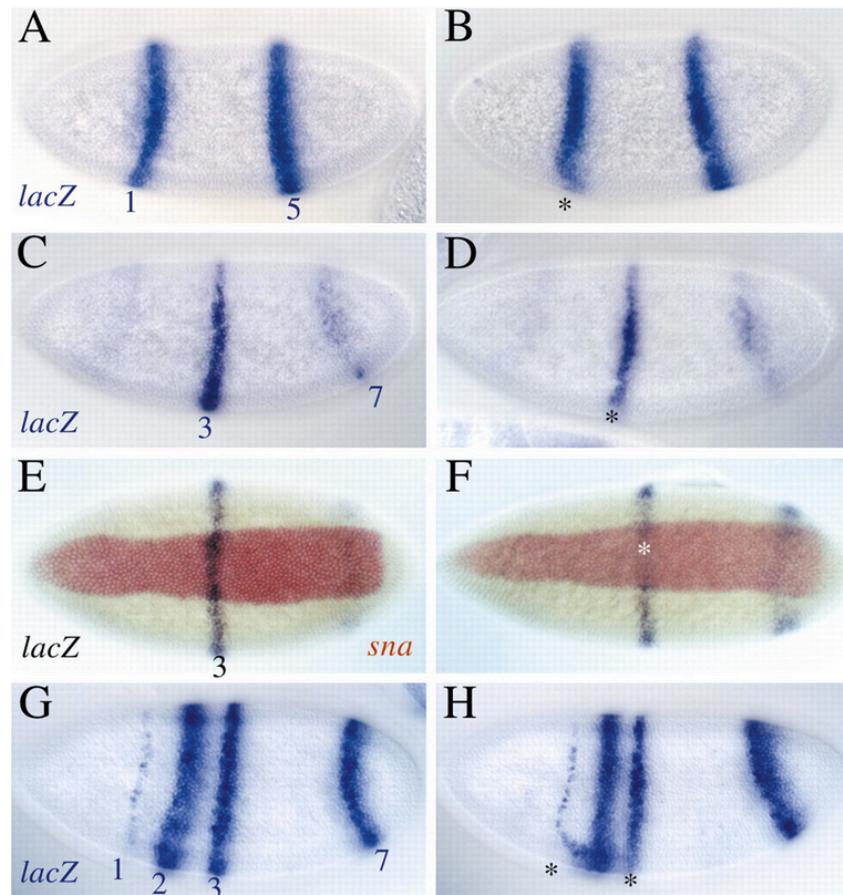
glucose-insulin regulatory feedback system in humans



Driven by a series of differential equations

Other pathway applications:

- Drosophila gap pathways [Nuzhdin]
- Exhaled nitric oxide levels [Eckel]



Monte Carlo application 2: Hypercubes

- Consider n-dimensional unit ‘cubes’
 - n=3 -> a cube. Coordinates= (x_1, x_2, x_3) [or (x, y, z)].
 - n=2 -> a square. Coordinates= (x_1, x_2) [or (x, y)].
 - n=1 -> line. Coordinates= (x_1) [or (x)].
 - n=4 -> 4-dimensional hypercube. Coordinates= (x_1, x_2, x_3, x_4) .
 - n=5 -> 5-dimensional hypercube. Coordinates= $(x_1, x_2, x_3, x_4, x_5)$.
 - n=N -> N-dimensional hypercube. Coordinates= $(x_1, x_2, x_3, x_4, \dots, x_N)$.
- Let’s suppose that all are unit cubes, so $0 \leq x_i \leq 1$, for all i.
- **Question: what proportion of the volume of an N-dimensional hypercube is within a distance of 0.1 of the surface?**

Question: If we sample a point ***uniformly at random*** from inside a hypercube, what is the probability that the point is within a distance of 0.1 of the surface?

If you finish quickly: what happens with spheres?

Pseudocode (HypercubesPseudocode.Rmd on Github)

```
set.seed(324) # set the seed for the random number generator to any number you like
NumberOfTrials<-500 # How many trials we will perform
TotalCountNearSurface<-numeric() # a vector that records how many were close to the surface for each dimension
MaxNumberOfDimensions<-10 # we will go up to this many dimensions
HowManyCloseToSurface<-0 # A variable to record how often the point is close to the surface in a given dimension
for (j in 1:MaxNumberOfDimensions) {
  HowManyCloseToSurface<-0 # reset your counter
  for (i in 1:NumberOfTrials) {
    # pick a point at random in the cube
    MyPoint<-runif(j,0,1)

    # check whether it is within 0.1 of the surface (how do you check this?)
    # if it is, increase the value of HowManyCloseToSurface by 1
  }

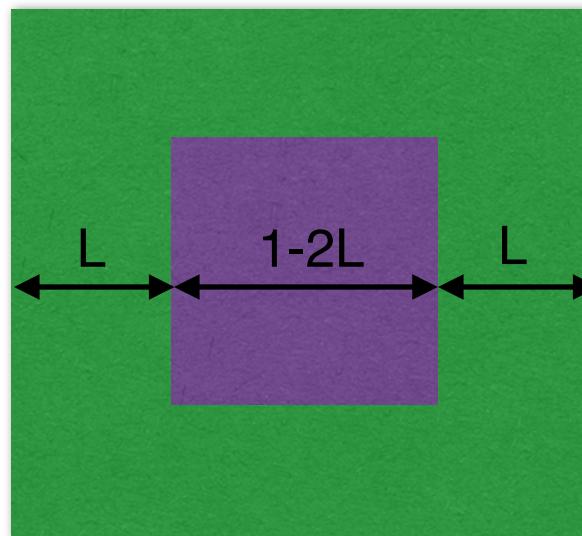
  # record the value of HowManyCloseToSurface, this will be your estimate of the prob. of being close to surface
  TotalCountNearSurface[j]<-HowManyCloseToSurface
  # output the proportion of the volume of the cube that is within 0.1 of the surface
  cat("\nEstimatedProb.:", TotalCountNearSurface[j]/NumberOfTrials)
}

# plot your estimates of the proportion of the volume that is within 0.1 of the surface (y-axis)
# against the number of dimensions the hypercube has (x-axis)

# what do you notice about this proportion as N increases?
# what value do you think it takes for very large N?
```

Hypercubes

- Answer:
 - The proportion of the volume that is within a distance of L of the surface of an N -dimensional hypercube with sides of length 1 is
$$1-(1-2L)^N \rightarrow 1 \text{ as } N \rightarrow \infty \text{ (for } 0 \leq L \leq 0.5\text{).}$$



Monte Carlo Application 3: coin tossing

- Suppose we simulate a sequence of 500 coin tosses. Define a ‘run’ as a sequence of all Heads (and of all Tails). Answer the following:
 - What is the distribution of the number of heads?
 - What is the distribution of the length of the run starting at the first toss?
 - What is the expected number and distribution of the number of runs in total?
 - What is the expected value and distribution of the length of the longest run?

Code

- I have uploaded an Rmd-script to Github (“Coin-tossing” repo) that answers most of these questions, but seems to have a bug (SequenceGenerator.Rmd).
- You can use this as the basis of your answer if you wish.

MC Application 3: Answers

- Suppose we simulate a sequence of n coin tosses. Define a ‘run’ as a sequence of all Heads (or all Tails). Answer the following:
 - What is the distribution of the number of heads?
 - $\text{Binomial}(n, 1/2)$
 - What is the distribution of the length of the run starting at the first toss?
 - $\text{Geometric}(0.5)$
 - What is the expected number and distribution of the number of runs in total?
 - $1 + \text{Binomial}(n-1, 1/2)$
 - What is the expected value and distribution of the length of the longest run?
 - ?

QQ plots

- Used to compare samples, X_1, \dots, X_n and Y_1, \dots, Y_n :
 - Order the data points in each sample from low to high, to get $X_{[1]}, \dots, X_{[n]}$ and $Y_{[1]}, \dots, Y_{[n]}$
 - Plot $X_{[1]}$ against $Y_{[1]}$, $X_{[2]}$ against $Y_{[2]}$, $X_{[3]}$ against $Y_{[3]}$, etc.
 - If the distributions are the same, you should see a straight line (for large samples)
 - ‘qqplot’ in R
- Can do the same with one sample and Normal random deviates (qnorm in R)
- Formal tests: Kruskal-Wallis test or ANOVA.

Distributions

- Binomial (n,p) ~ Result of n independent trials, each of which has probability p of being a success.
 - $P(X = m) = \binom{n}{m} p^m (1 - p)^{n-m}$
- Geometric(p) ~ number of successes before first failure, when each trial has prob. p of success.
 - $P(X=m) = p^m (1-p)$

Additional problem (no code provided): Occupancy problems

- Suppose we have n buckets and drop m balls independently into those buckets.
- Suppose ball i goes into bucket j with prob. $1/n$ for all i, j .
- *What is the distribution of the number of balls in bucket i (for any i)?*
- *What is the joint distribution of the number of balls in each bucket at the end?*
- *What is the distribution of the number of empty buckets?*
- *What is the distribution of the number of balls in the bucket with the most balls?*

Assignment1 (i.e. examinable): Randomization tests - golf balls

- Allan Rossman used to live along a golf course and collected the golf balls that landed in his yard. Most of these golf balls had a number on them.



- Question: What is the distribution of these numbers?
- In particular, are the numbers 1, 2, 3, and 4 equally likely?

Assignment1 : golf balls

- **Population:** Golf balls at that driving range
- Allan tallied the numbers on the first 500 golf balls that landed in his yard one summer.
- **Sample:** Those golf balls driven ~150 yards and sliced.

1	2	3	4
137	138	107	104

- There were 14 “others”, which we will ignore
- **Question:** What is the distribution of these numbers? In particular, are the numbers 1, 2, 3, and 4 equally likely?
- **Meta-Question:** How do we answer this question using the data?

Randomization test set-up

- **Null hypothesis:** Our default belief about the data.
 - Here, it is that the numbers on the population golf balls are uniformly distributed between 1 and 4.
- **Test statistic:** A single number that can be calculated from the data and used to test whether our null hypothesis is true.

1	2	3	4
137	138	107	104

- What test-statistic should we use?
- How should we conduct the test?

'Pseudocode' on Github

```
#some pseudocode for the golf ball problem
# it uses the maximum frequency as the test statistic

# Set up some global variables
NumberOfGolfBalls<-486 # how many golf balls were sampled in the observed
dataset
NumberOfSamples<-1000
HighestNumberSeen<-4

# Here's a function that returns a set of golf balls numbers from a uniform distribution
SampleGolfBalls<-function(HowMany,HighestNumber){
  SupposedPopulation<-1:HighestNumber
  Sample<-sample(SupposedPopulation,size=HowMany,replace=TRUE)
  return (Sample)
}
...

```

Points to consider

- Logic of a hypothesis test
 - 1. State Hypotheses
 - Null hypothesis must provide a model (for simulation/calculation)
 - 2. Calculate a Test Statistic
 - 3. Determine the p-value
 - 4. Interpret the p-value
- What makes a good test statistic? Compare several (even if you know the ‘book’ answer).
- Advantages to knowing the sampling distribution?
- Power against particular types of alternatives.

Nate Silver and the Seven Dwarves

- <https://fivethirtyeight.com/features/where-will-the-seven-dwarfs-sleep-tonight/>
- Solution: <https://fivethirtyeight.com/features/draw-a-circle-then-draw-a-triangle-now-solve-a-riddle/>

Variation: What if the youngest dwarf just chooses a bed at random (possibly the correct bed)?

END