05 - Anomaly Detection

SYS 4582/6018 | Spring 2019

05-anomaly.pdf

Contents

1 Anomaly Detection Intro

1.1 Required R Packages

We will be using the R packages of:

- `tidyverse` for data manipulation and visualization
- `mclust` for model-based clustering

```
library(mclust)        # install.packages("mclust")
library(tidyverse)
library(readxl)
```

1.2 Anomaly Detection

Anomaly Detection: The identification of unusual observations. Statistically, this means finding observations that come from a different distribution that the *normal* or *usual* observations.

1. **Goodness of Fit (GOF)**
   - Tests if data conform to a given distribution (or distributional family)
   - Use case: Failure of the first digits in a financial statement to conform to Benford's distribution may indicate fraud.
2. **Two-Sample Tests (A/B Testing)**
   - Tests if two datasets come from the same distribution
   - Often simplified to test if one group has a larger mean than the other
   - Use case: Determine if a new surgical technique leads to faster recovery times.
3. **Outlier Detection**
   - Tests if a single observation or small set of observations come from the same distribution as the rest of the data
   - Use case: Detect and correct data entry errors.
4. **Hotspot Detection**
   - Identification of regions that have *unusually* high density
   - Use case: add additional police patrols to regions and times that are experiencing an unusually high crime rate
5. **Outbreak Detection**
   - A sequential method that repeatedly tests for a change in an event (i.e., point process) distribution
   - Focus on determining *if* a change occurred, and then determining *when* it occurred
   - For outbreak detection, the changes of interest are those that conform to an expected *outbreak pattern*
   - Use case: quickly detect the presence of West Nile Virus from the *chief complaints* field of health records and initiate a rapid mosquito control spraying

1.3 Example #1: Benford's Distribution

State/Territory	Real or Faked Area (km^2)	
Afghanistan	645,807	796,467
Albania	28,748	9,943
Algeria	2,381,741	3,168,262
American Samoa	197	301
Andorra	464	577
Anguilla	96	82
Antigua and Barbuda	442	949
Argentina	2,777,409	4,021,545
Armenia	29,743	54,159
Aruba	193	367
Australia	7,682,557	6,563,132
Austria	83,858	64,154
Azerbaijan	86,530	71,661
Bahamas	13,962	9,125
Bahrain	694	755
Bangladesh	142,615	347,722
Barbados	431	818
Belgium	30,518	47,123
Belize	22,965	20,648
Benin	112,620	97,768
...	...	...

Table from Fewster (2009) A Simple Explanation of Benford's Law, *The American Statistician*, 63, 1, pp 26–32

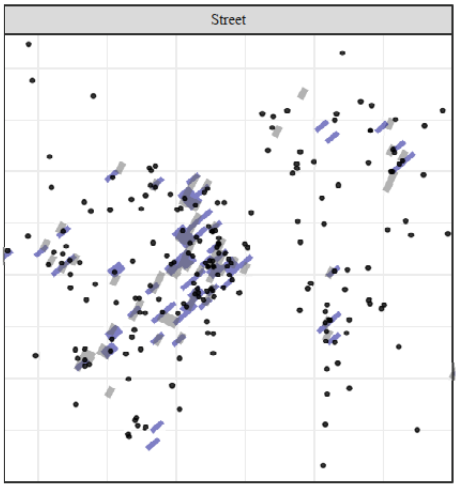
- Someone that fakes numbers, say on a financial statement, may be tempted to use a random number generator
  - But watch out for Benford's Law
- Note on terminology:
  - Law = probability distribution
  - Anomalous Numbers = Random numbers (no known relationship)
- Benford's PMF:

$$\Pr(\text{first digit} = x) = \log_{10}\left(1 + \frac{1}{x}\right) \quad \text{for } x = 1, 2, \ldots, 9$$

1.4 Example #2: Crime Hotspots

In 2017, the National Institute of Justice (NIJ) held a Crime Forecasting Challenge

- Predict the crime hotspots for 4 crime types (burglary, street crime, motor vehicle theft, all types) for 5 forecasting windows (1 week ahead, ..., 3 months ahead)



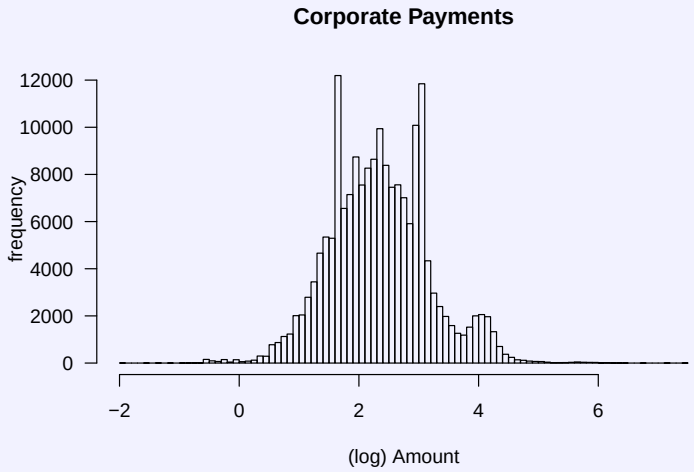
2 Goodness of Fit Testing

Tests if data conform to a given distribution (or distributional family)

Your Turn #1

Mark Nigrini provides a dataset of the 2010 payments from a division of a West Coast utility company https://www.nigrini.com/BenfordsLaw/CorporatePaymentsData.xlsx.

```
## Download xlsx file to local machine, then load into R:
## Notice that the data is in the "Data" sheet
library(readxl)
corp_payments = read_excel("../topics/anomaly/data/CorporatePaymentsData.xlsx",
                           sheet="Data") %>%
  filter(Amount > 0)   # only consider positive payments
```




```
#-- Extract the first digit
first = stringr::str_extract(corp_payments$Amount, "[1-9]") %>%
  factor(levels=1:9)
```

Does the first digit of the `Amount` column conform to Benford's distribution?

2.1 GOF Hypothesis Test

- Let $D = (X_1, X_2, \ldots, X_n)$ be the observed random variables (i.e., the data).
- Let $\mathcal{H}_0$ be the *null* hypothesis
- Choose a *test statistic* $T = T(X_1, \ldots, X_n)$ that is a function of the observed data
  - T is a *random variable*; it has a distribution.
  - Let $t = T(x_1, \ldots, x_n)$ be the *observed* value of the test statistic
  - Common to adjust the test statistic so that *extreme* means large values of T
- The p-value is *the probability that chance alone would produce a test statistic as extreme as the observed test statistic if the null hypothesis is true*
  - E.g., p-value $= \Pr(T \geq t | \mathcal{H}_0)$
- Think of T or the p-value as the *evidence against* the null hypothesis
  - Its common to set a threshold (e.g., p-value $\leq .05$) and *reject* the null hypothesis when this threshold is crossed.
  - This is a form of *outlier detection*. Reject null if t_{obs} is an *outlier*; that is t_{obs} is from a different distribution that what is specified in $\mathcal{H}_0$.

- To calculate a p-value, we need to know/estimate the *distribution* of $T|\mathcal{H}_0$!
  - Even if we don't know the distribution of T under the null, we can often approximate it using simulation (Monte Carlo)

2.1.1 Example: one sample t-test

- In 2012, the Obama administration issued new rules on the fuel efficiency requirements for new cars and trucks by 2025.
  - The fleetwise fuel efficiency requirement is 54.5 mpg
- Suppose a car maker in 2025 designed a car to get an average fuel efficiency of 54.5 mpg.
  - Also, they think the fuel efficiency will be Normally distributed (Gaussian)
- The government officials randomly tested $n = 9$ cars. They got a sample mean of $\bar{x} = 53.0$ and sample standard deviation of $s = 2.5$.

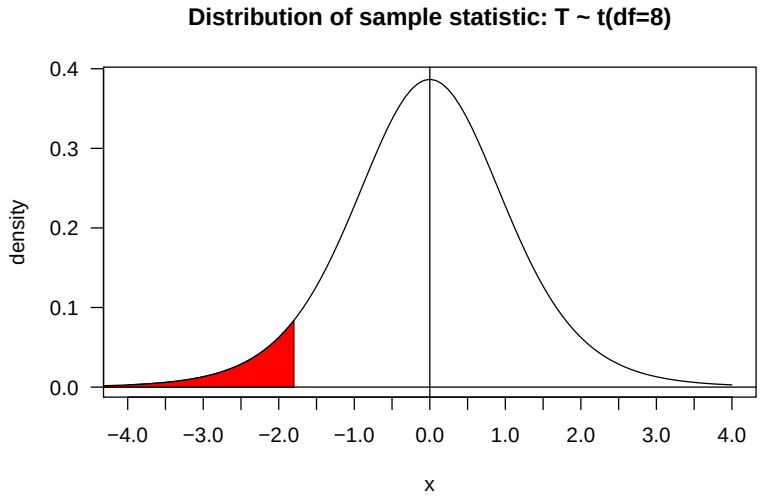
Your Turn #2

Is this enough evidence to conclude the car manufacturer failed to meet the requirements? Or can the results be attributed to chance fluctuation?

- Null Hypothesis:
  - the mpg come from a Normal distribution
  - mean of $\mu_0 = 54.5$
  - independent
  - standard deviation, σ is unknown
  - $X|\mathcal{H}_0 \overset{\text{iid}}{\sim} \mathcal{N}(\mu = 54.5, \sigma)$
- Choose the test statistic:

Test Statistic

- Under the null, T has a t-distribution with $df = n - 1$
  - Shoutout to William Sealy Gosset, a.k.a *Student*



- $\Pr(T \leq -1.8|\mathcal{H}_0) =$ `pt((53.0-54.5)/(2.5/3), df=8)` $= 0.055$

2.1.2 Alternative Hypotheses

- The choice of test statistic depends on the expected deviations from the null
  - That is, we can come up with *better* test statistic if we know what sort of deviations from $\mathcal{H}_0$ are expected.
  - *better* meaning more *power* to correctly reject the null

2.2 Test Statistics

- Going back to the original question about the corporate payments conforming to Benford's distribution, we can state the *null hypothesis* formally:

$$\mathcal{H}_0 : X \overset{\text{iid}}{\sim} Benf$$

- X is the *first* digit(s)
- $Benf$ stands for Benford's distribution for the first digit(s). This has pmf:

$$f(x) = \log_{10}\left(1 + \frac{1}{x}\right)$$

  - Note: there are no parameters to estimate!

- A generic *alternative hypothesis* is:

$$\mathcal{H}_1 : X \not\sim Benf$$

- R code for a Benford pmf

```
#-- pmf for Benford's distribution
dbenford <- function(x) log10(1 + 1/x)

#-- first digit
dbenford(1:9)
#> [1] 0.30103 0.17609 0.12494 0.09691 0.07918 0.06695 0.05799 0.05115 0.04576

#-- first two digits
expand.grid(first=1:9, second=0:9) %>%
  mutate(two = paste0(first, second) %>% as.integer) %>%
  mutate(f = dbenford(two)) %>%
  select(first, second, f) %>%
  spread(second, f) %>%
  knitr::kable(digits=3)
```

first	0	1	2	3	4	5	6	7	8	9
1	0.041	0.038	0.035	0.032	0.030	0.028	0.026	0.025	0.023	0.022
2	0.021	0.020	0.019	0.018	0.018	0.017	0.016	0.016	0.015	0.015
3	0.014	0.014	0.013	0.013	0.013	0.012	0.012	0.012	0.011	0.011
4	0.011	0.010	0.010	0.010	0.010	0.010	0.009	0.009	0.009	0.009
5	0.009	0.008	0.008	0.008	0.008	0.008	0.008	0.008	0.007	0.007
6	0.007	0.007	0.007	0.007	0.007	0.007	0.007	0.006	0.006	0.006
7	0.006	0.006	0.006	0.006	0.006	0.006	0.006	0.006	0.006	0.005
8	0.005	0.005	0.005	0.005	0.005	0.005	0.005	0.005	0.005	0.005
9	0.005	0.005	0.005	0.005	0.005	0.005	0.005	0.004	0.004	0.004

2.2.1 χ^2 Test Statistic

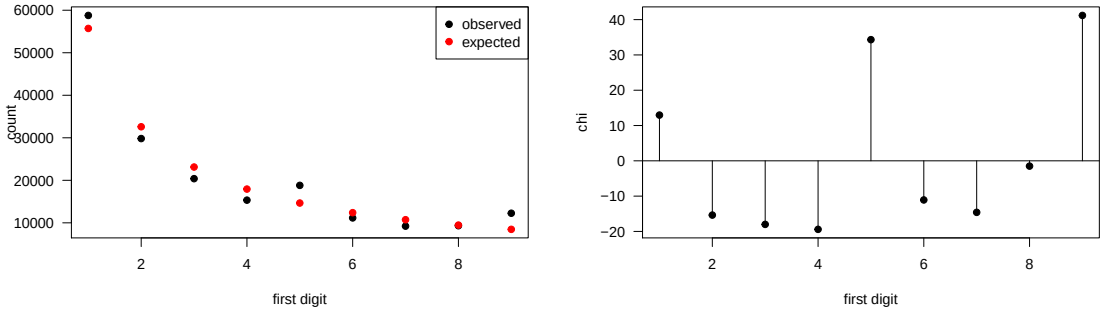
- The Pearson's χ^2 (chi-squared) test statistic is commonly used in goodness-of-fit testing.
- It requires the data to be *discrete* or *categorical*
  - Continuous data can be *binned*
- Test Statistic:

$$\chi^2 = \sum_{j=1}^{J} \frac{(y_j - E_j)^2}{E_j} = \sum_{j=1}^{J} \frac{(y_j - np_j)^2}{np_j}$$

  - J: number of categories or possible values
  - y_j: *observed* count in category j
  - E_j: *expected* count (i.e., under $\mathcal{H}_0$) in category j
  - Let $n = \sum_{j=1}^{J} y_j$ be the total number of observations
  - p_j: the proportion of events under the null (i.e., $\Pr(X = j|\mathcal{H}_0)$)
  - Asymptotically, χ^2 converges to a chi-squared distribution with $J - 1$ degrees of freedom
- R code for corporate payments data

```
#-- Get counts
Y = table(first) %>% as.integer    # ensure first is factor with properly ordered
n = length(first)                  # number of observations

#-- chi-squared
n = length(first)     # number of observations
E = n*dbenford(1:9)   # expected count vector
chi = (Y-E)/sqrt(E)   # vector of deviations
(chisq = sum(chi^2)) # chi-squared test statistic
#> [1] 4317
```



- Note: there is a build-in R function `chisq.test()` which does these calculations

```
chisq.test(Y, p=dbenford(1:9))$statistic
#> X-squared
#>      4317
```

2.2.2 Likelihood Ratio Test Statistic

- When an *alternative hypothesis* can be specified, the log-likelihood ratio test statistic is commonly used in goodness-of-fit testing.

- The general binary hypothesis formulation is:
  - $\mathcal{H}_0 : X \sim f_0(X)$ (null hypothesis)
  - $\mathcal{H}_1 : X \sim f_1(X)$ (alternative hypothesis)
- The *likelihood ratio* is:

$$\begin{aligned} LR &= \frac{f_1(X_1, \ldots, X_n)}{f_0(X_1, \ldots, X_n)} \\ &= \prod_{i=1}^{n} \frac{f_1(X_i)}{f_0(X_i)} \qquad \text{if } X\text{'s are iid} \end{aligned}$$

- The *log-likelihood ratio*, when the observations are iid, becomes:

$$\begin{aligned} llr &= \sum_{i=1}^{n} \log \frac{f_1(X_i)}{f_0(X_i)} \\ &= \sum_{i=1}^{n} \log f_1(X_i) - \sum_{i=1}^{n} \log f_0(X_i) \end{aligned}$$

- The hypotheses for the corporate payments data:
  - $\mathcal{H}_0 : X \overset{\text{iid}}{\sim} Benf$
  - $\mathcal{H}_1 : X \overset{\text{iid}}{\sim} Cat(p_1, p_2, ..., p_9)$ where $\{p_k\}$ do *not* match Benford's probabilities.
- There are many reasonable choices for setting $\mathcal{H}_1$ parameters $(p_1, \ldots, p_9)$
  - *Discrete Uniform*: $p_1 = \ldots = p_9 = 1/9$
  - *MLE*: $\hat{p}_k = y_j/n$

Your Turn #3

Write out the log-likelihood for the two alternatives.

Using MLE, the llr = 2031.83.

- Note: $2 \times llr$ has an asymptotic chi-squared distribution (same as the chi-squared test statistic).

2.3 Testing

- The two-test statistics, χ^2 and llr, provide evidence against $\mathcal{H}_0$.
- But how do we know if these values are *unusually* large? Perhaps by chance alone (i.e. the data sample we observed) the values are as large as they are.
- We can answer this with a solid probabilistic statement if we knew the *distribution of the test statistic under the null hypothesis*

  - We don't often know this exactly, but there are often good approximations that hold as the sample size grows (asymptotically).

- There are two primary options:

  1. Use an asymptotic distribution (e.g., the chi-squared distribution)
  2. Use Monte Carlo simulation

2.3.1 Monte Carlo Simulation

- If we can sample from the null hypothesis, then it becomes straightforward to estimate the distribution of *any* test statistic and consequently, p-values.

- **Monte Carlo based GOF Test**

  1. Calculate the test statistic for the original observation, t
  2. Generate M samples from the null distribution
     - $\{Y_1, \ldots, Y_M\}$
  3. For each sample, calculate the test statistic T^*
     - $\{T_1^*, \ldots, T_M^*\}$
  4. p-value = $\frac{1+\text{number of } T^*\text{'s greater than or equal to } t}{M+1}$

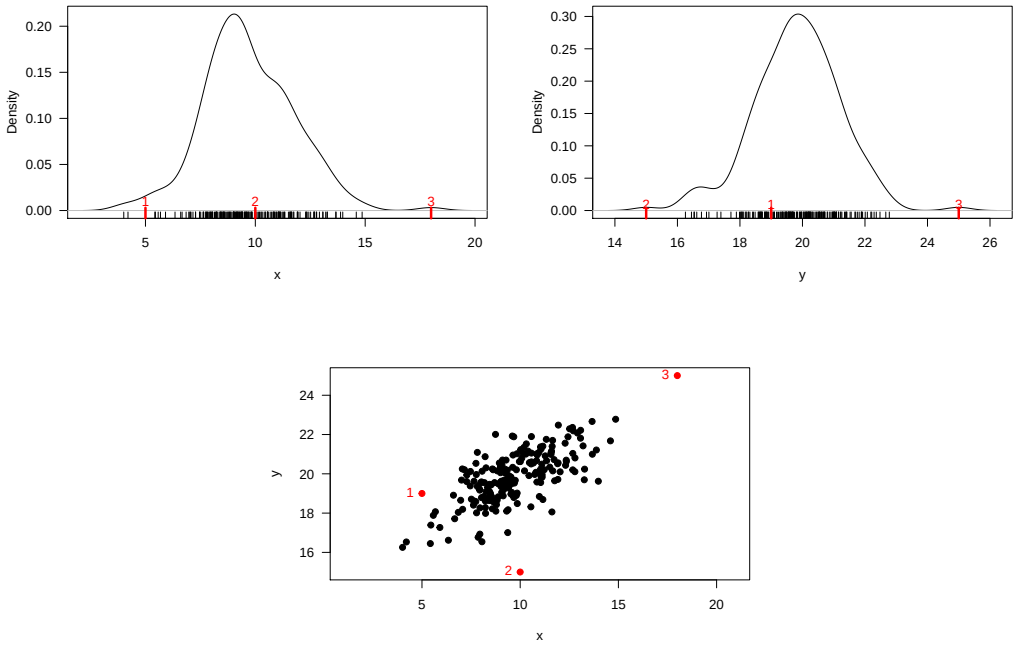
```r
#-- Monte Carlo based p-value
n = length(first)

M = 1000                              # number of simulations
stat.chisq = numeric(M)               # initialize statistic
for(m in 1:M){
  #- generate observation under the null of Benford
  y.sim = rmultinom(1, size=n, prob=dbenford(1:9))
  #- calculate test statistic
  stat.chisq[m] = chisq.test(y.sim, p=dbenford(1:9))$statistic
}

#- calculate p-values
(1 + sum(stat.chisq > chisq)) / (M+1)   # chi-square p-value
#> [1] 0.000999
```

3 Outlier Detection

- Outlier Detection tests if a single observation or small set of observations come from the same distribution as the rest of the data
- Are any of the *red* points outliers?



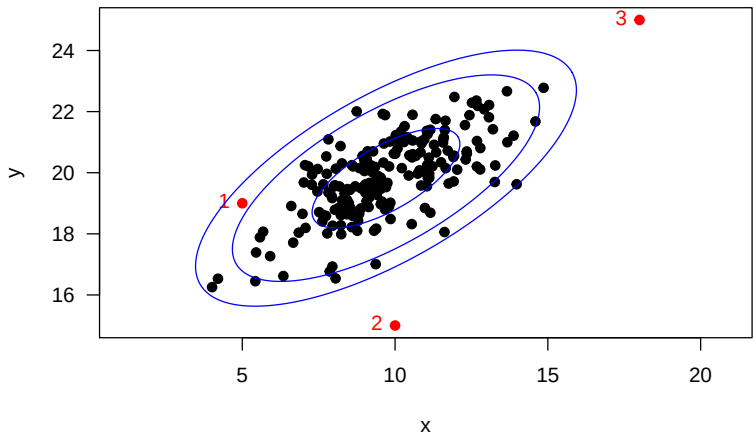
3.1 Distance based approach

- One approach, with strong connections to clustering, is to calculate the *distance* from an observation to the centroid
  - This assumes the "normal" observations are from a unimodal distribution
  - To allow for an ellipse shape (orientation) and different spreads in each dimension, use the squared *Mahalanobis Distance*

$$D_i^2 = (\mathbf{x_i} - \bar{\mathbf{x}})^\mathsf{T} \hat{\Sigma}^{-1} (\mathbf{x_i} - \bar{\mathbf{x}})$$

  - Note that $\bar{\mathbf{x}}$ and $\hat{\Sigma}$ are estimated from all of the data.
- Estimated Parameters:
  - $\bar{x} = (9.70, 19.82)$
  - $\hat{\Sigma} = (4.2171, 1.9482, 1.9482, 1.906)$

obs	x	y	Dsq
1	5	19	7.056
2	10	15	24.471
3	18	25	18.123



3.2 Likelihood Based Approach

- From the plots, it appears the a 2D Gaussian/Normal model could be a decent approximation to the distribution of the non-outlier observations
- We can use this to calculate the *log-likelihood* of observation i using the estimated parameters

Gaussian Log-Likelihood

$$\log L_i = \mathcal{N}(\mathbf{x}_i; \mu = \bar{x}, \Sigma = \hat{\Sigma})$$
$$= $$

- Notice that this is a function of the squared Mahalanobis distance!
- Robust estimation:
  - If indeed we have outliers, then these will be affecting out estimated parameters $\bar{x}$ and $\hat{\Sigma}$.
  - Robust estimation techniques can help limit the damage caused by the outliers
  - Another, more structured approach, is mixture models!

3.3 Mixture Model Approach

- We can go back to our mixture model formulation and propose that the outliers come from a different distribution than the normal observations
- In mixture formulation

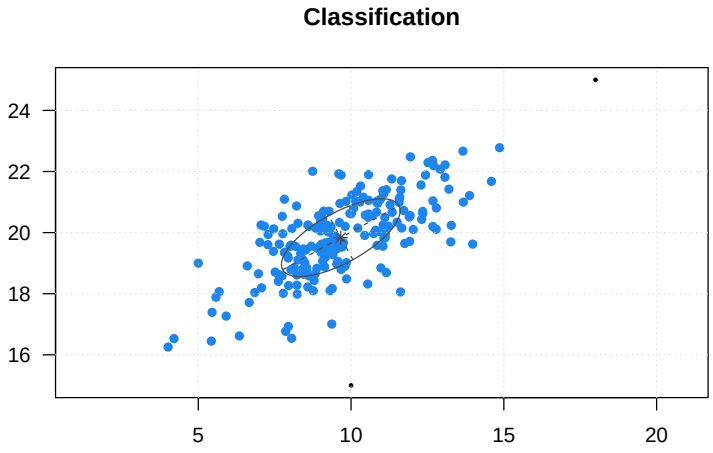
$$f(x) = \pi h(x) + (1 - \pi) g(x)$$

  - $h(x)$ is the pdf for the *outliers*
  - $g(x)$ is the pdf for the *normal observations*
  - π is the prior probability that an observation will be an outlier
  - $E[\text{number of outliers}] = n\pi$

- There are several options for the outlier distribution $h(x)$
  - The uniform distribution on the *minimum bounding box* is one simple approach
    - $h(x) = 1/V$, where V is the volume of the bounding box
    - $V = \prod_{j=1}^{p} (max(x_j) - min(x_j))$
- The `Mclust()` function in the R package `mclust` permits this formulation using the `initialization=list(noise=TRUE)` argument

```
#-- Fit mixture model with uniform noise
library(mclust)
mc = Mclust(X, initialization=list(noise=TRUE), verbose=FALSE)
summary(mc)
#> ----------------------------------------------------
#> Gaussian finite mixture model fitted by EM algorithm
#> ----------------------------------------------------
#>
#> Mclust EVE (ellipsoidal, equal volume and orientation) model with 1 component
#> and a noise term:
#>
#>  log.likelihood   n df   BIC   ICL
#>          -713.4 203  7 -1464 -1473
#>
#> Clustering table:
#>   1   0
#> 201   2
```

```
plot(mc, what="classification", asp=1, las=1)
grid()
```



4 Two-Sample Testing (A/B Testing)

4.1 Example (A/B Testing): Clinical Trials

A placebo-controlled randomized trial proposes to assess the effectiveness (i.e., cure rate) of Drug A in curing infants suffering from sepsis. A clinical trial of $n = 600$ infants using Drug A found that 40% were cured of sepsis while 36% of the $n = 1200$ infants on a placebo were cured.

Your Turn #4

1. Is Drug A better than the placebo?
2. How much better?

Let $p_1 = \Pr(\text{ cure } | \text{ Drug A })$ and $p_2 = \Pr(\text{ cure } | \text{ Placebo })$

- $\mathcal{H}_0 : p_1 = p_2$ or $p_1 - p_2 = 0$
- $\mathcal{H}_a : p_1 > p_2$ or $p_1 - p_2 > 0$

4.1.1 Simulation Based Testing

The 1800 patients were randomly assigned to the treatment (Drug A) or placebo group. It turned out that:

- of the $n_1 = 600$ given Drug A, $n_1\bar{p}_1 = 600(0.4) = 240$ were cured
- of the $n_2 = 1200$ given Drug A, $n_1\bar{p}_2 = 1200(0.36) = 432$ were cured
- of the $n_1 + n_2 = 1800$ patients, a total of 672 (37.3%) were cured

Under the null hypothesis, $\mathcal{H}_0 : p_1 = p_2$, there is no real difference in the cure rate between treatment and placebo. The observed difference is due **only** to the random assignment.

4.1.2 Permutation Test

We can see what the outcomes would have been if we used a different assignment into treatment and placebo.

- Regroup all patients
- Draw n_1 samples, at random, and calculate $\bar{p}_1^*$
- Use the remaining n_2 to calculate $\bar{p}_2^*$
- Calculate the test statistic $Z^* = (\bar{p}_1^* - \bar{p}_2^*)$

This is a possible outcome if the null hypothesis was actually true.

- If we repeat this procedure for all possible re-groupings, then we get the exact[1] distribution of the test statistic, if the null hypothesis was true.
- But, there are $\binom{1800}{600}$(*huge number*) possible regroupings (permutations)
- Monte Carlo simulation can be used to approximate this distribution
- Just repeat the re-grouping procedure many times (say 1000 or 10000)
  - gives a set of observed values under the null model
- The estimated p-value is the proportion of simulated test statistic values that are more extreme than the **observed** test statistic

[1] conditional on the data and study design

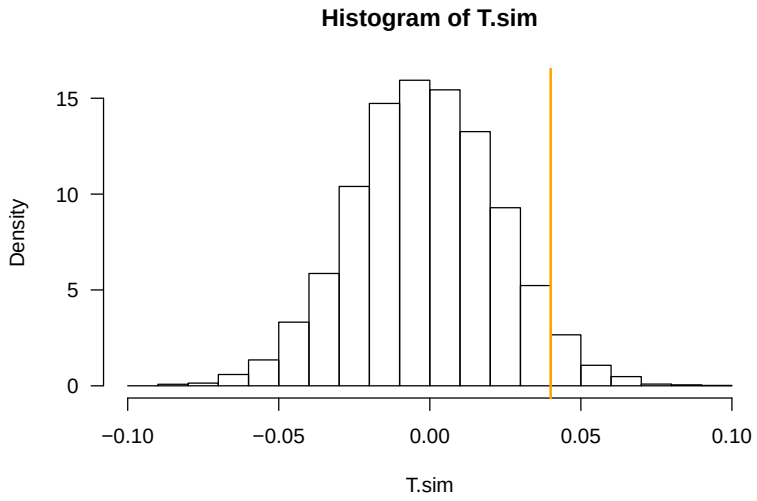
4.1.3 R Code

```
#- observed data
n1 = 600
p1 = 0.40
n2 = 1200
p2 = 0.36
p0 = (n1*p1 + n2*p2)/(n1+n2)  # average cure rate
T.obs = p1 - p2       # Test Statistic: observed difference
```

```
#- Simulation Data
n = n1 + n2             # number of patients
x = n1*p1 + n2*p2       # total number cured

#- Run Simulation
set.seed(100)                               # set seed for replication
nsim = 10000                                # of simulations
x1.sim = rhyper(nsim, m=x, n=n-x, k=n1)     # simulated # cured in pop 1
x2.sim = x - x1.sim                         # simulated # cured in pop 2
T.sim = x1.sim/n1 - x2.sim/n2               # simulated test statistics
```

```
#- plots
hist(T.sim, breaks=seq(-.1,.1,by=.01), freq=FALSE, las=1) # histogram
abline(v=T.obs,col="orange",lwd=2)        # add observed test statistic
```




```
#- p-value
(sum(T.sim >= T.obs) + 1) / (nsim +1)     # non-parametric p-value
#> [1] 0.05349
```

4.2 Friedman's Supervised Modeling Approach to Testing

- Worth reading: https://statweb.stanford.edu/~jhf/ftp/gof

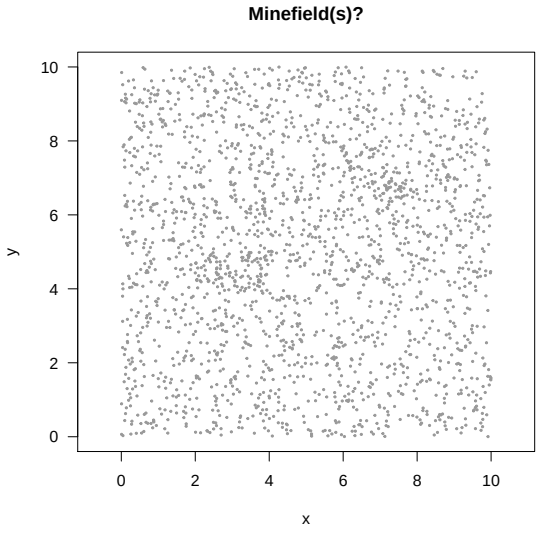
5 Hotspot Detection

Identification of regions that have *unusually* high density

5.1 Example: Land Mine and UXO Detection

Ground penetrating radar is able to detect the presence of land mines and unexploded ordinance (UXO). However, it also detects clutter; rocks and non-interesting metallic objects.

- The items of interest will appear as hotspots
- Uncertain shape and direction



5.2 Hotspot mixture model

$$f(x) = \pi_0 g(x) + \sum_{k=1}^{K} \pi_k f_k(x)$$

- $g(x)$ is the pdf for the *normal/background observations*
- π_0 is the prior probability that an observation comes from the background distribution
- $f_k(x)$ is the pdf for the kth *hotspot*
- π_k is the prior probability that an observation comes from the kth hotspot
- K is the number of hotspots
- $E[\text{number of observations in hotspots}] = n \sum_{k=1}^{K} \pi_k$

5.2.1 Considerations

- The success of this formulation will depend on the forms and restrictions imposed on the components
- If $g(x)$ is allowed too much flexibility, then no hotspots will be detected
- Restrictions should be put on $\sum_k \pi_k$ to prevent the hotspot components from being dominant
- The form of the hotspot component densities $f_k(x)$ should match the expected shapes if a hotspot were actually to occur.
  - E.g., if a hotspot represents a mine field, then the allowable shapes of f_k should match what is possible/probably in mine fields (prior info)
  - E.g., if we think the hotspots will be circular, then restrict f_k to have $\Sigma_k = \lambda_k I$

5.2.2 `mclust` R package

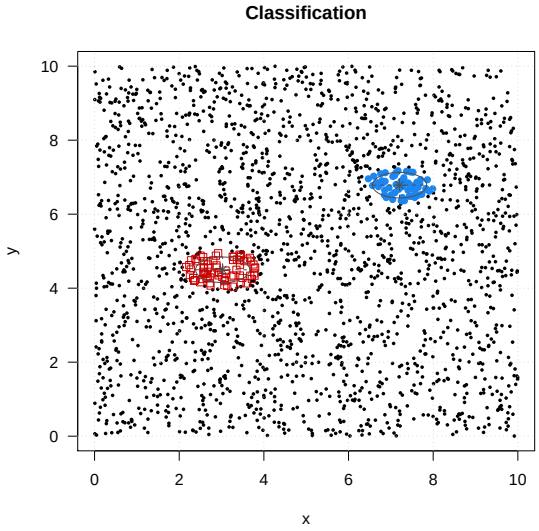
- The `Mclust()` function in the `mclust` R package can facilitate the hotspot mixture model *only if the background distribution is uniform*.
  - $g(x) = 1/V$, where V is the volume of the bounding box

$$V = \prod_{j=1}^{p} (max(x_j) - min(x_j))$$

  - This is the same setup as was used in outlier detection, but reversed (so to speak) in the sense that the we expect most observations to come from the background (uniform) component.
- Recall that `Mclust()` chooses the number of clusters and their form by optimizing BIC
  - If we want more control, we can specify the `G=`, `modelNames=`, or `prior=` arguments

```
# Note: X is the two column matrix of point coordinates
library(mclust)
Kmax = 4
mc = Mclust(X, G=1:Kmax,      # set of hotspots to consider
            initialization=list(noise=TRUE), # uniform background
            verbose=FALSE)    # don't show progress bar
summary(mc)
#> ----------------------------------------------------
#> Gaussian finite mixture model fitted by EM algorithm
#> ----------------------------------------------------
#>
#> Mclust EEI (diagonal, equal volume and shape) model with 2 components and a
#> noise term:
#>
#>  log.likelihood    n df    BIC    ICL
#>           -9820 2144  9 -19709 -19970
#>
#> Clustering table:
#>    1    2    0
#>   64   73 2007

plot(mc, what="classification", las=1)
grid()
```

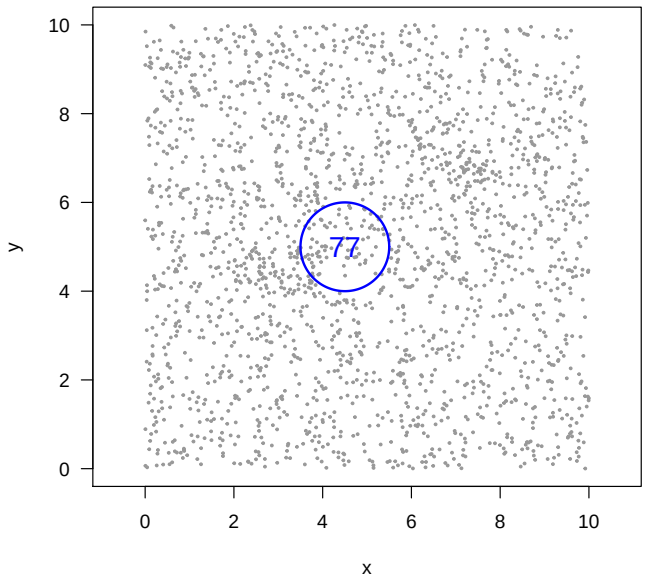


5.3 Spatial Scan Statistic

The spatial scan statistic is an approach to detect spatial *hotspots* (sometimes called *clusters*) that adjusts for *multiple testing*.

5.3.1 Set-up

- Let S be a spatial region (e.g., circle)
- Let $N(S)$ be the *number of events in* S



- **Model:** $N(S) \sim Pois(\lambda)$

$$\Pr(N(S) = y) = \frac{\lambda^y e^{-\lambda}}{y!}$$

- **Hypotheses:**

No Outbreak	$\mathcal{H}_0: \ \lambda(S) = \lambda_0$	
Outbreak in S	$\mathcal{H}_1(S): \ \lambda(S) = q\lambda_0$	where $q > 1$

Your Turn #5

Under $\mathcal{H}_1(S)$, given a count $N(S) = y$, what is the MLE for q, when λ_0 is *known*?

- **Test Statistic: D(S)**
  - Use Likelihood Ratio
  - Consider that λ_0 is *known*

D(S): Likelihood Ratio Test Statistic

5.3.2 Significance Testing

- Simulate N_0's according to $\mathcal{H}_0$. $N_0 \sim Pois(\lambda_0)$
- Calculate $D_0(S)$ for each simulation
- Estimated p-value is the proportion of times that the D_0's exceed y (observed)
- Example:
  - For the data in the plot, suppose $\lambda_0 = 63$ for region S (blue circle)
  - We observed $N(S) = 77$
  - $D(S) = (\frac{77}{63})^{77} \exp(63 - 77) = 4.2701$

$$\begin{aligned} \Pr(D(S) \geq 4.2701 | \mathcal{H}_0, \lambda_0 = 63) &= \Pr(N_0(S) \geq 77 | \lambda_0 = 63) \\ &= \text{R: 1-ppois(77-1, lambda=63)} \\ &= 0.0479 \end{aligned}$$

  - *Significance!* Are you ready to conclude that S is a *hotspot*?

5.3.3 Multiple Testing

- Before we jump the gun, remember that we want to *search the entire area* for hotspots.
- If we considered a different region S', but with same $\lambda_0 = 63$, what is the probability that we would get $D(S') \geq 4.2701$, even if S' is not a hotspot?

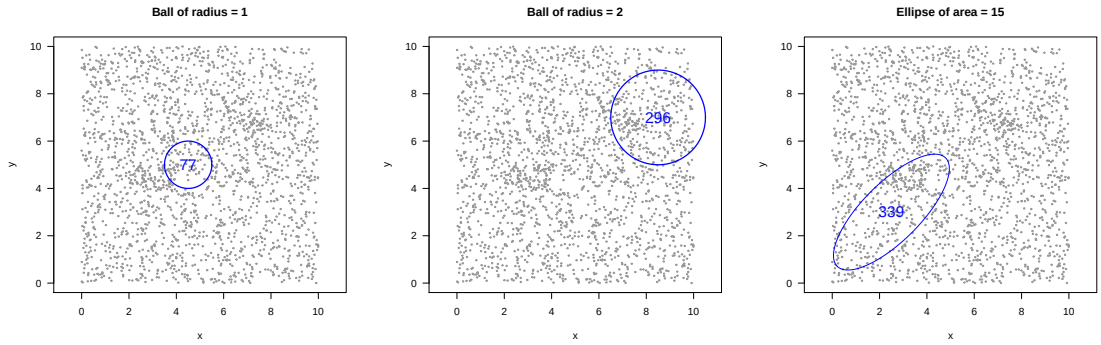
Your Turn #6

If we searched 5 non-overlapping regions $S_1, \ldots, S_5$ (all with $\lambda_0 = 63$), what is the probability that *at least one* will have $D_j(S) \geq 4.2701$, even if there are *no hotspots*?

- The probability of at least one *false alarm* starts to grow fast as the *number of tests* increases
- There are a few options:
  1. Bonferroni: Only reject $\mathcal{H}_0$ if p-value $\leq \alpha/K$, where K is the number of tests/comparisons
     - Can be too conservative if the tests are *correlated*
  2. Scan Statistics (i.e., only test significance of the *maximum* score)

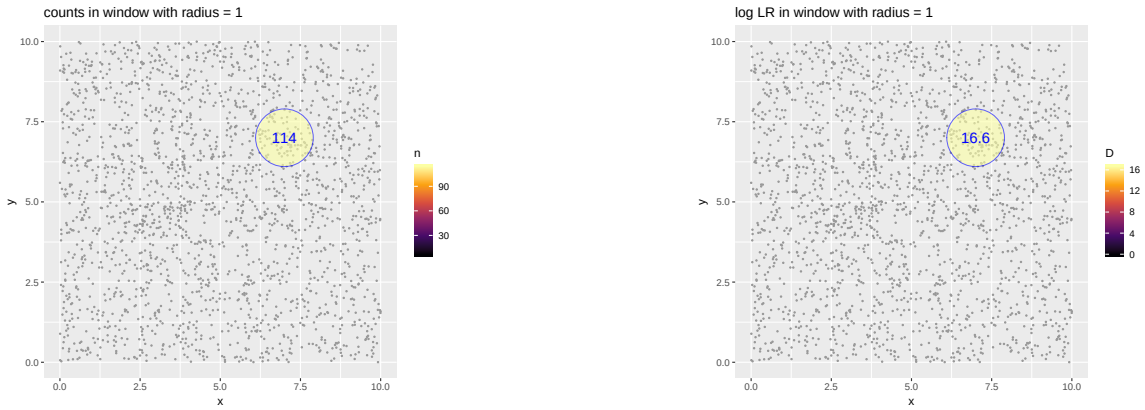
5.3.4 Spatial Search

- Let $\mathcal{S}$ be the *search set*, the set of spatial regions that will be searched
- The spatial regions may be restricted by *size* and/or *shape*. E.g.,
  - all balls/circles of radius r
  - All ellipses of area a that are not too thin



- *Scan* the search windows over all locations and record the counts, $N(S)$ and corresponding discrepancy scores $D(S)$.

Click on image to see animation

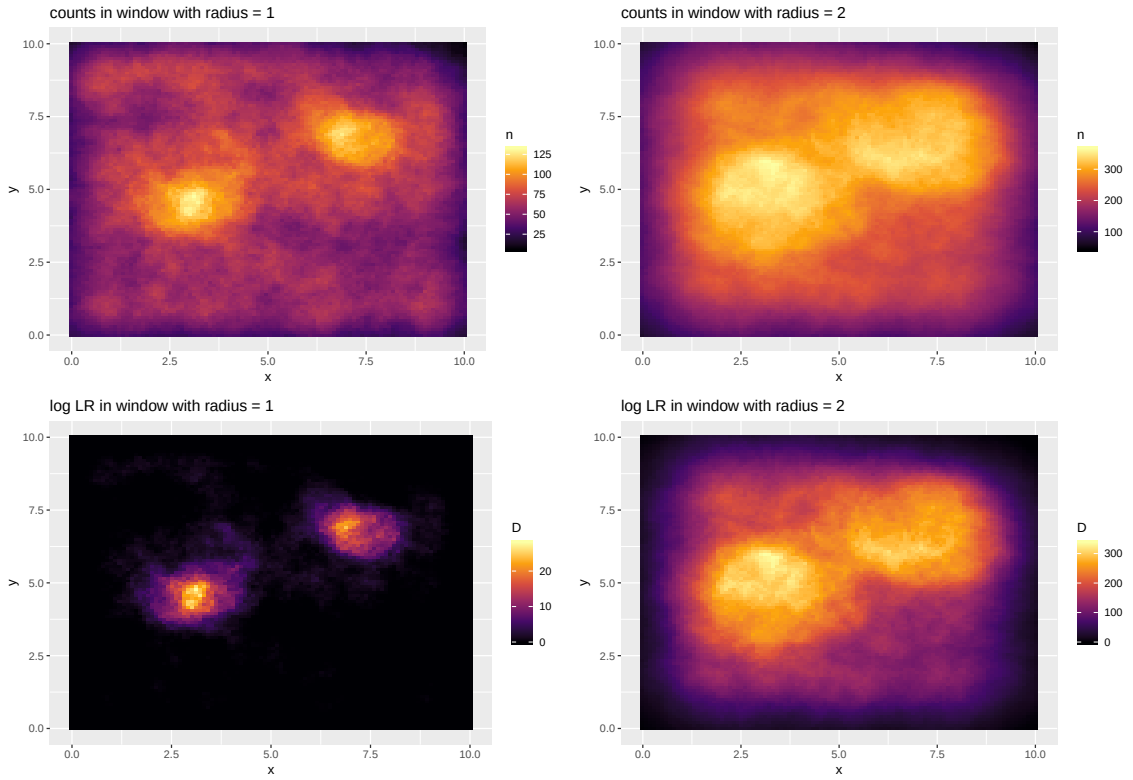


5.3.5 Scan Stat (Test Statistic)

- The *spatial scan stat* test statistic, for radius r, is the region that gives the maximum discrepancy

$$D_r^* = \arg\max_{S \in \mathcal{S}} D_r(S)$$

5.3.6 Results



- NOTE: There is strong connection to kernel density estimation! Do you see how we can build the search from special KDE functions?
- Maximum Scores (aka the *scan statistics*)

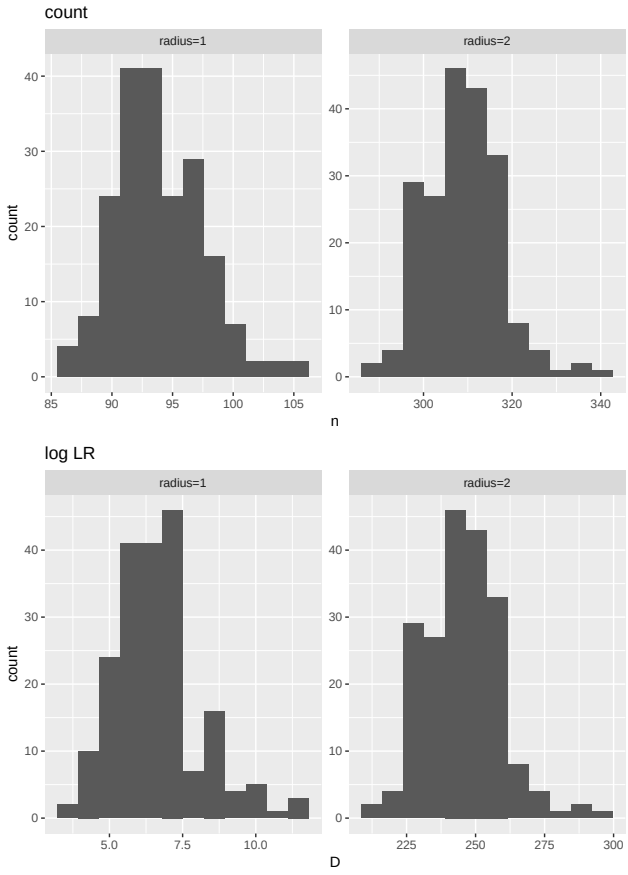
radius	x	y	n	D*
1	3.131	4.747	131	27.9
2	3.131	5.960	363	335.7

5.3.7 Monte Carlo Significance Testing

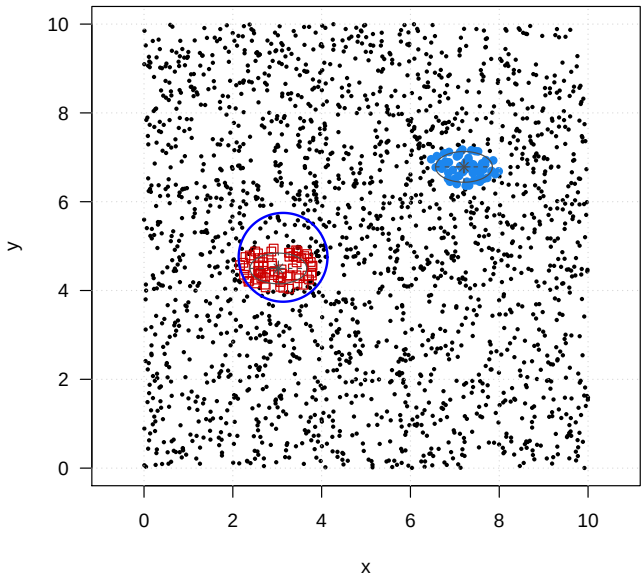
- Is one of the D^*'s large enough to conclude there is a *hotspot*?
- What radius should be used?
- To answer these questions, we need to find the distribution of D^* under the null hypothesis of *no hotspots*

1. **Simulate from the null hypothesis**
   - Our data came from a *spatial point process*
   - Simulate n events (n is the number of events we observed)
   - Density is $f(x) = \lambda_0(x) / \int \lambda_0(u)\, du$
2. **Search for Clusters**

3. **Record scan statistics**
4. **Redo M of times**
5. **Calculate p-values**
   - p-value $= \frac{1+\sum_m \mathbb{1}(D_m^* \geq D_{\text{obs}}^*)}{1+M}$



- No simulation exceeded our observed scores. Tiny p-values. There is likely *at least one hotspot*



5.4 Network Scan Statistic

- The *network scan statistic* is very much like the spatial scan statistic, the difference being the hotspots are a *connected set of nodes*

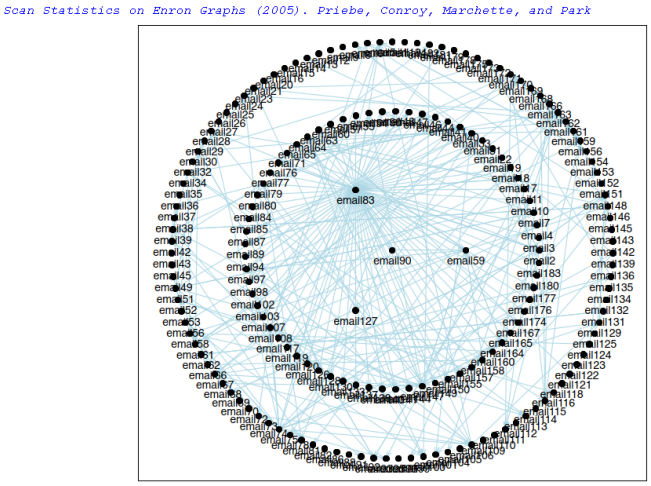


Figure 4. Plot of the 'detection' Enron email graph D_{132} (sans isolates) for which our scan statistic methodology detects an anomaly. The center vertex, *email*90, is $v^* = \arg\max_v \tilde{\Psi}_{2,132}$.

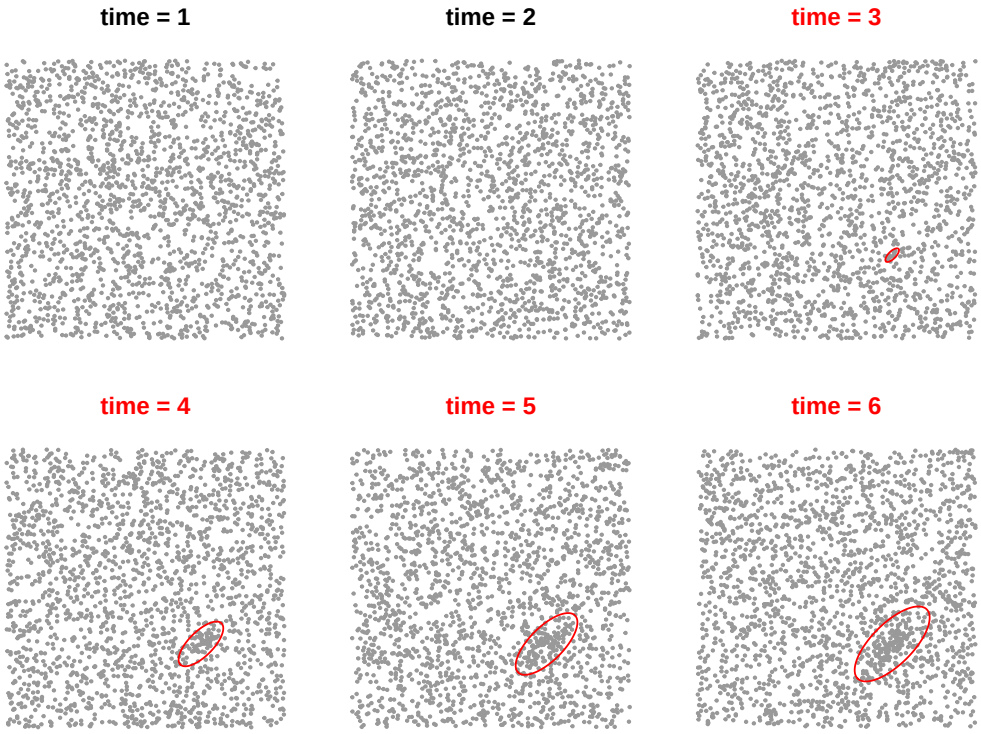
- The computational challenge of searching all reasonable hotspots is substantial

6 Outbreak Detection

- Consider having the spatial observations occurring over time.
  - E.g., daily, weekly, hourly
- Denote $N(t, S)$ as the number of events in spatial region S at time period t
- There is a time point τ when an outbreak starts

$$\mathcal{H}_0: \quad \lambda(t, S) = \lambda_0(t) \qquad \text{for all } t$$

$$\mathcal{H}_1(\tau, S): \quad \lambda(t, S) = \begin{cases} \lambda_0(t) & \text{for } t < \tau \\ q(t)\lambda_0(t) & \text{for } t \geq \tau \text{ and } q(t) > 1 \end{cases}$$



- One option is to compute $D*(t)$ over each time t and look for the time when there is an increase
  - Use Statistical Process Control (SPC, control charts)
- If the signal (i.e., outbreak) is small, it may take many time periods before there is enough evidence to detect the outbreak
  - Another option is to include time into the likelihood ratio test statistic
  - See: Kulldorff, M. (2001), Prospective time periodic geographical disease surveillance using a scan statistic. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 164: 61-72 for more details