Get your hands dirty

Kaie Kubjas

Geometric and Algebraic Combinatorics in Paris

June 27, 2019

“Statistics is a branch of mathematics working with data collection, organization, analysis, interpretation and presentation.”

Wikipedia

statistical population or statistical model **all log-concave distributions**

statistical population or statistical model

all log-concave distributions

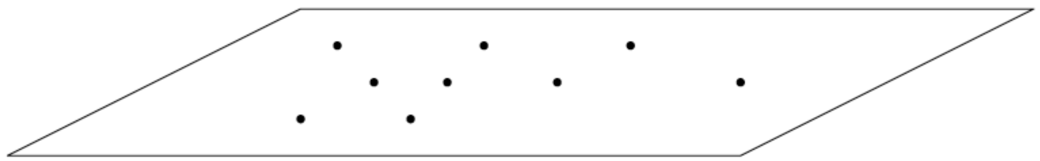
design of surveys and experiments

statistical population or statistical model

all log-concave distributions

design of surveys and experiments

sampling

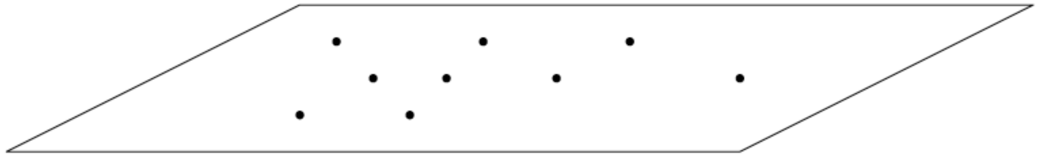


statistical population or statistical model

all log-concave distributions

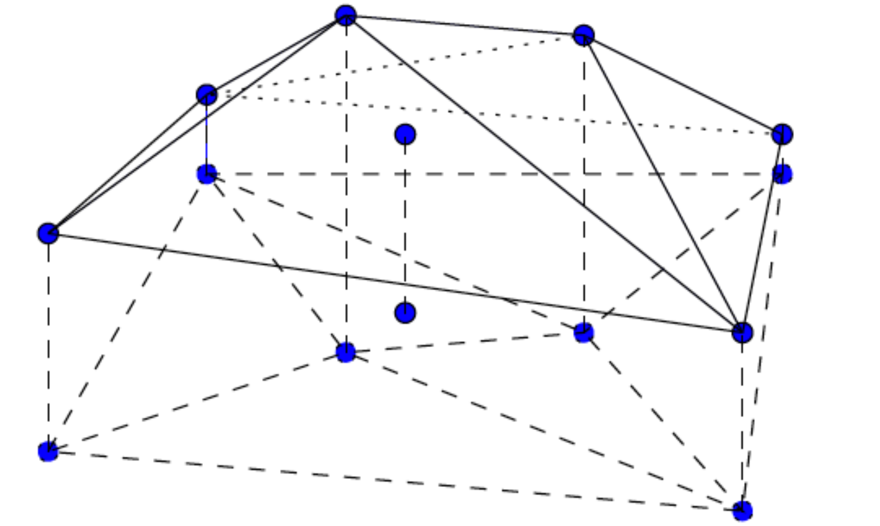
design of surveys and experiments

sampling



data analysis: descriptive and inferential statistics

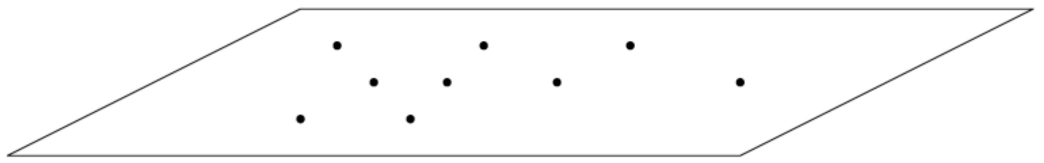
maximum likelihood estimation



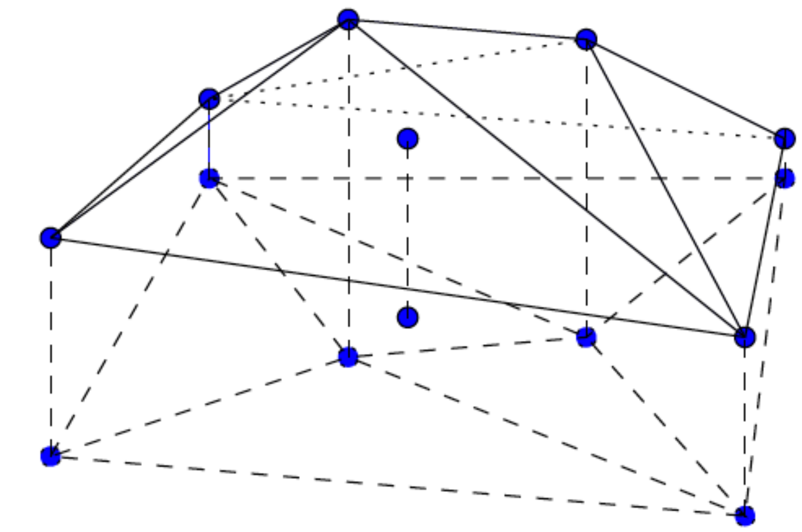
statistical population or statistical model **all log-concave distributions**

design of surveys and experiments

sampling



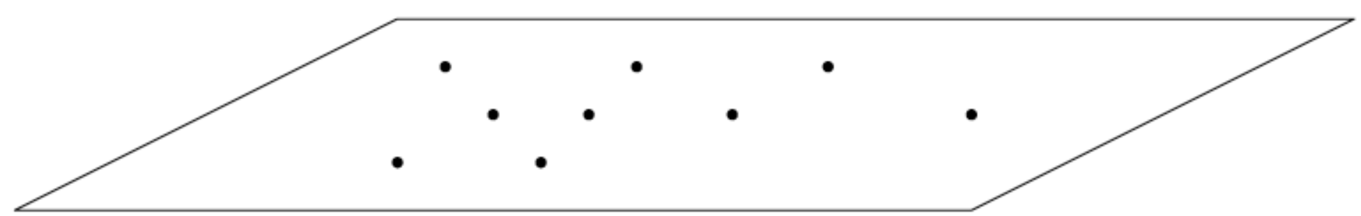
data analysis: descriptive and inferential statistics **maximum likelihood estimation**



R is the standard software for statistical computing

Log-concave density estimation

- Given data: points $X = \{x_1, \ldots, x_n\} \in \mathbb{R}^d$ with weights $w = (w_1, \ldots, w_n)$ where $w_1, \ldots, w_n \geq 0, \sum w_i = 1$



- Maximize the log-likelihood of the given sample (X, w) over all integrable functions $p : \mathbb{R}^d \to \mathbb{R}_{\geq 0}$ such that $\log(p)$ is concave and $\int_{\mathbb{R}^d} p(x)dx = 1.$

Log-concave density estimation

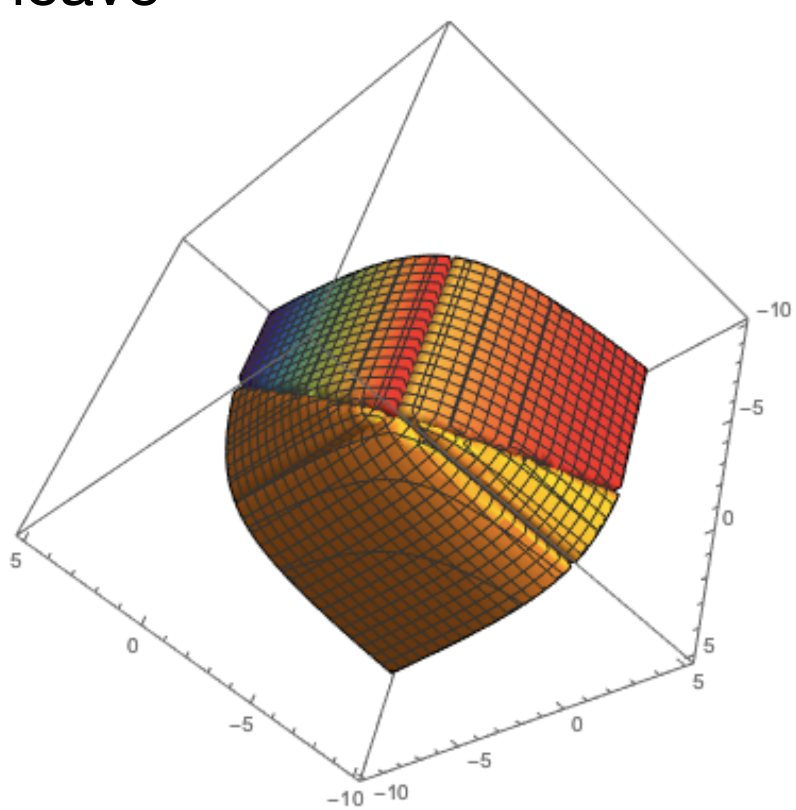
Two equivalent optimization problems:

$$\max \sum_{i=1}^{n} w_i \log(p(x_i))$$

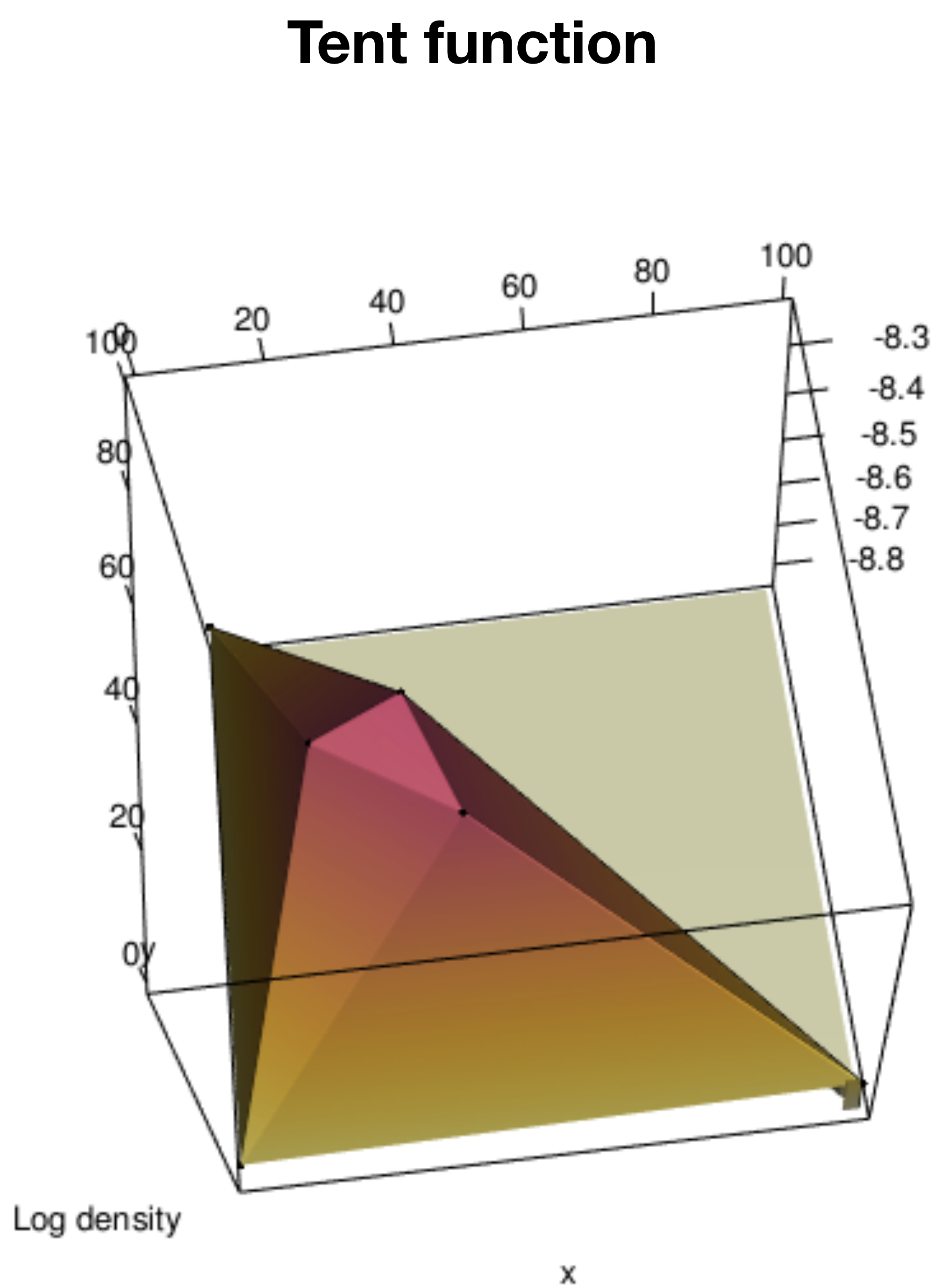
s.t p is density
and p is log-concave

$$\max_{y \in \mathbb{R}^n} \sum_{i=1}^{n} w_i y_i$$

s.t. $\int \exp(h_{X,y}(t))dt = 1$



Optimal solution



LogConcDEAD

Madeleine Cule, Robert Gramacy, Richard Samworth, Yining Chen

LogConcDEAD

Madeleine Cule, Robert Gramacy, Richard Samworth, Yining Chen

- first need to install the package

```
install.packages("LogConcDEAD")
library(LogConcDEAD)
```

LogConcDEAD

Madeleine Cule, Robert Gramacy, Richard Samworth, Yining Chen

- first need to install the package

```
install.packages("LogConcDEAD")
library(LogConcDEAD)
```

- input point configuration and weights

```
x <- matrix(c(0,0,100,0,0,100,22,37,43,22,36,41), ncol = 2, byrow = TRUE)
w <- 1/12 * c(2,2,2,1,4,1)
```

LogConcDEAD

Madeleine Cule, Robert Gramacy, Richard Samworth, Yining Chen

- first need to install the package

```
install.packages("LogConcDEAD")
library(LogConcDEAD)
```

- input point configuration and weights

```
x <- matrix(c(0,0,100,0,0,100,22,37,43,22,36,41), ncol = 2, byrow = TRUE)
w <- 1/12 * c(2,2,2,1,4,1)
```

- mlelcd computes the maximum likelihood estimator

```
lcd <- mlelcd(x,w)
```

LogConcDEAD

LogConcDEAD

- output gives heights of the MLE at sample points (log scale)

```
> lcd

 Log MLE at observations:
[1] -9.097136 -8.945554 -9.305969 -8.343196 -7.518446 -8.103404

 Number of Iterations:  58

 Number of Function Evaluations:  193
```

LogConcDEAD

- output gives heights of the MLE at sample points (log scale)

```
> lcd

 Log MLE at observations:
[1] -9.097136 -8.945554 -9.305969 -8.343196 -7.518446 -8.103404

 Number of Iterations:  58

 Number of Function Evaluations:  193
```

- additional information available about the output, for example lcd$triang, lcd$b, lcd$beta

```
> lcd$triang
     [,1] [,2] [,3]
[1,]    5    2    1
[2,]    5    3    1
[3,]    5    3    2
```

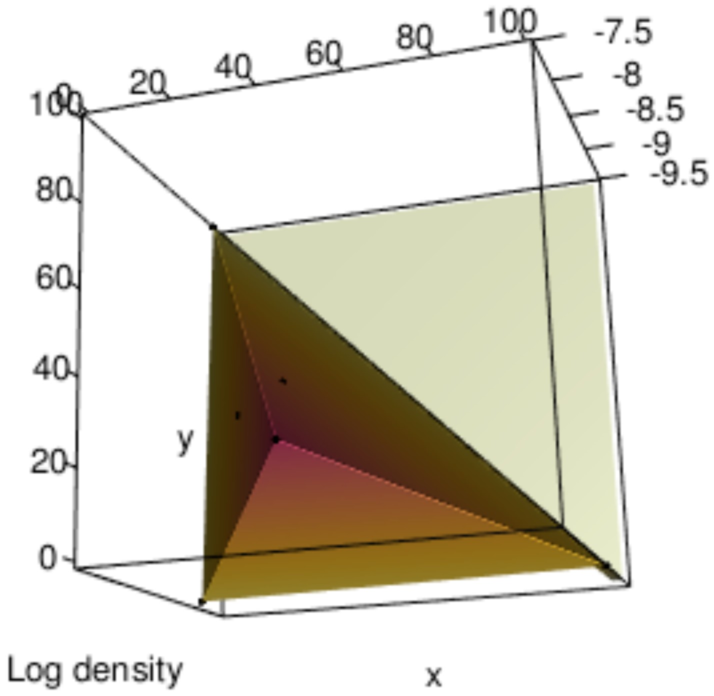
```
> lcd$b
             [,1]         [,2]
[1,]  0.001515821  0.068795889
[2,]  0.037782164 -0.002088327
[3,] -0.043039972 -0.046644120
```

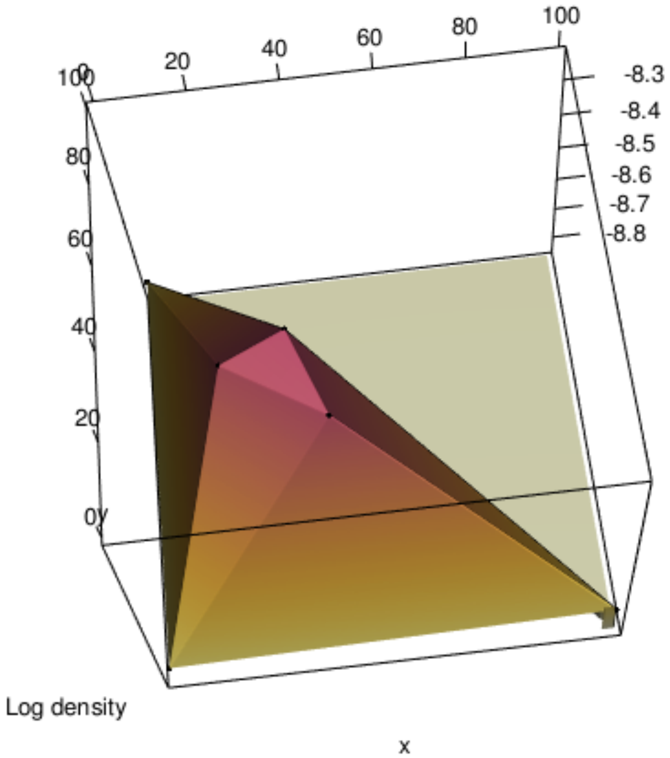
```
> lcd$beta
[1] 9.097136 9.097136 4.641557
```

LogConcDEAD

LogConcDEAD

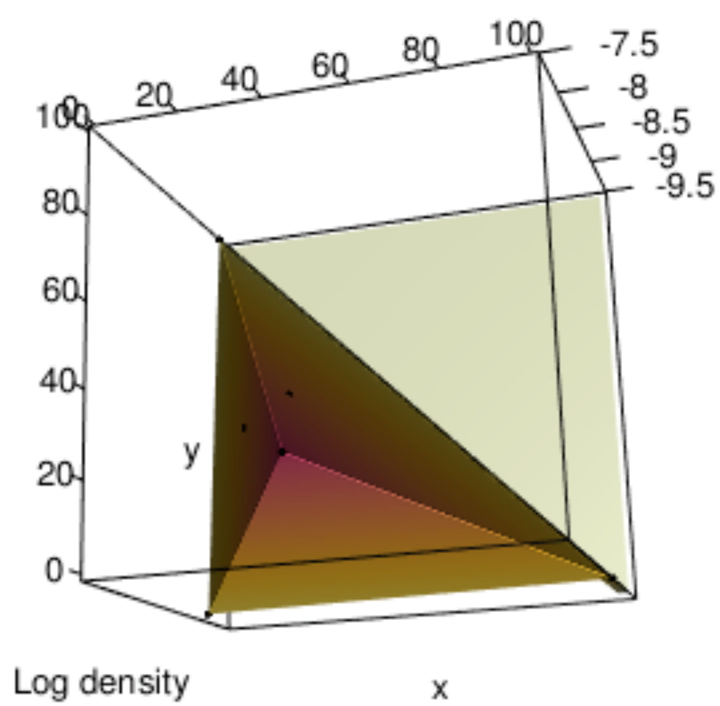
- plot.LogConcDEAD

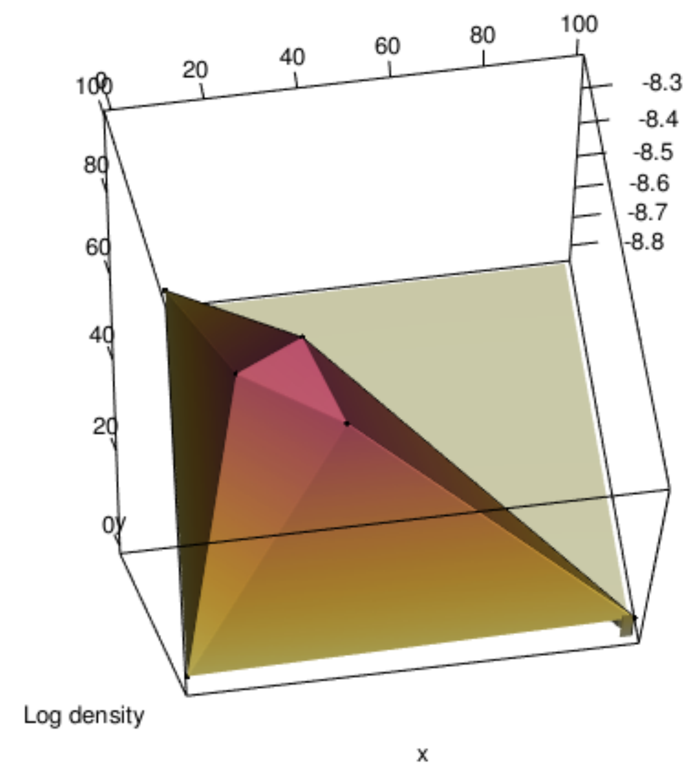




LogConcDEAD

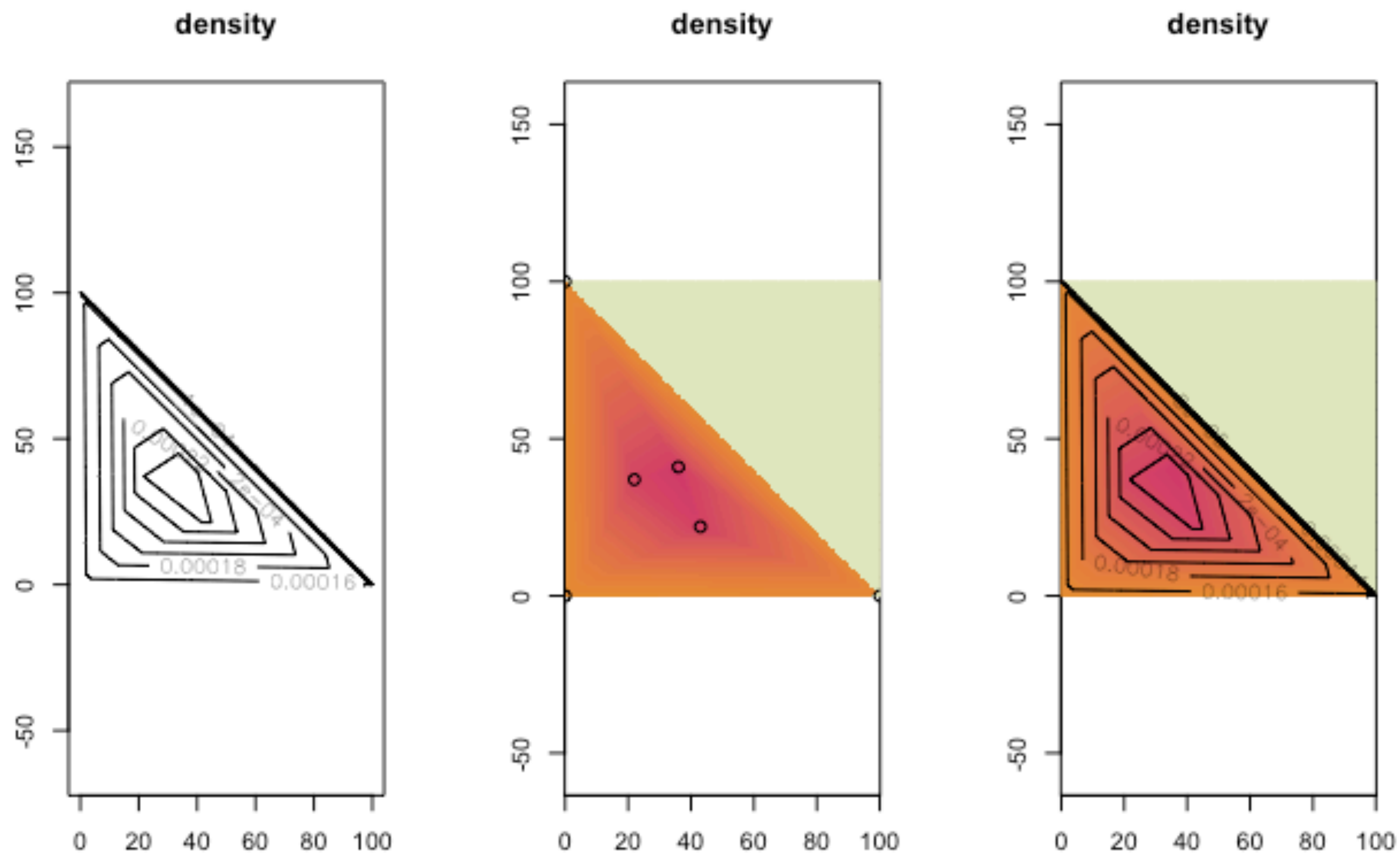
- plot.LogConcDEAD





- use help("plot.LogConcDEAD") to see all plotting options

- reference manuals and vignettes available for R packages


density
density
density
0.00018
0.00016
0.00018
0.00016

`plot.LogConcDEAD`	*Plot a log-concave maximum likelihood estimator*

Description

`plot` method for class `"LogConcDEAD"`. Plots of various types are available for 1- and 2-d data. For dimension greater than 1, plots of axis-aligned marginal density estimates are available.

Usage

```
## S3 method for class 'LogConcDEAD'
plot(x, uselog=FALSE, type="ic", addp=TRUE,
  drawlabels=TRUE, gridlen=400, g, marg, g.marg, main, xlab, ylab, ...)
```

Arguments

`x`	Object of class `"LogConcDEAD"` (typically output from `mlelcd`)
`uselog`	Scalar `logical`: should the plot be on the log scale?
`type`	Plot type: `"p"` perspective, `"c"` contour, `"i"` image, `ic` image and contour, `r` using **rgl** (the best!)
`addp`	Scalar `logical`: should the data points be plotted? (as black dots on the surface for $d \geq 2$; as circles for $d = 1$)
`drawlabels`	Scalar `logical`: should labels be added to contour lines? (only relevant for types `"ic"` and `"c"`)
`gridlen`	Integer scalar indicating the number of points at which the maximum likelihood estimator is evaluated in each dimension
`g`	(optional) a `matrix` of density estimate values (the result of a call to `interplcd`). If many plots of a single dataset are required, it may be quicker to compute the grid using `interplcd(x)` and pass the result to `plot`
`marg`	If non-`NULL`, this scalar integer determines which marginal should be plotted (should be between 1 and d)

LogConcDEAD

LogConcDEAD

- LogConcDEAD solves the optimization problem

$$\textbf{argmin}_{y \in \mathbb{R}^n} \sigma(y) = -\sum_{i=1}^{n} w_i y_i + \int_P e^{h_y(x)} dx$$

where P is the convex hull of X and h is the tent function

LogConcDEAD

- LogConcDEAD solves the optimization problem

$$\textbf{argmin}_{y \in \mathbb{R}^n} \sigma(y) = -\sum_{i=1}^{n} w_i y_i + \int_P e^{h_y(x)} dx$$

  where P is the convex hull of X and h is the tent function

- $\sigma(y)$ is convex but not differentiable

- subgradient methods for convex non-differentiable optimization problems

- LogConcDEAD implements r-algorithm by Shor for the particular problem in log-concave density estimation

LogConcDEAD

- The algorithm produces a sequence (y^t) satisfying

$$\sigma(y^t) \to \min_{y \in \mathbb{R}^n} \sigma(y)$$

  as $t \to \infty$.

- At each iteration, the algorithm requires the evaluation of $\sigma(y^t)$ and the subgradient $\partial\sigma(y^t)$ which determines the direction of the move to the next term y^{t+1} in the sequence.

- Their computation requires the evaluation of convex hulls and triangulations of certain finite sets of points (geometry package in R).

LogConcDEAD

- Works in any dimension
- In theory gives the correct solution
- Works with triangulations
- Relatively slow
- In practice solution up to some precision

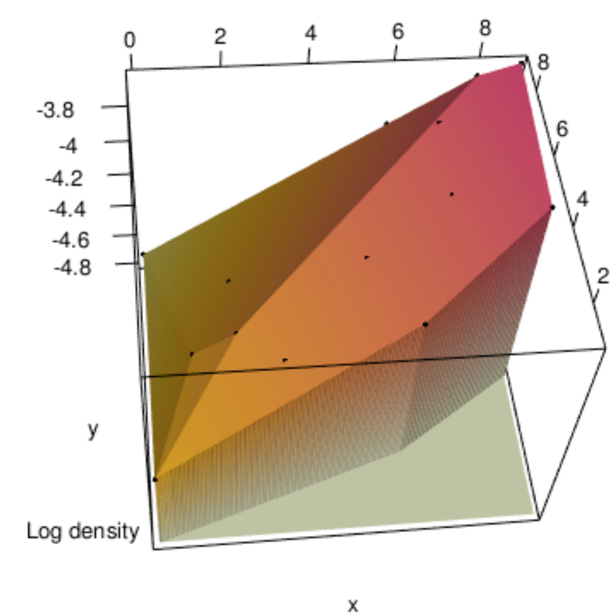
Problem 18

$$X = \{(9,9), (3,3), (0,9), (0,1), (6,3), (1,4), (7,8), (2,4), (8,9), (7,6), (6,9), (9,5), (2,6), (5,5)\}$$

```
> lcd$b
            [,1]        [,2]
 [1,] 0.19582201 -0.04887419
 [2,] 0.12874079 -0.02651378
 [3,] 0.06166755  0.01820171
 [4,] 0.06168464  0.01816418
 [5,] 0.06174931  0.01809952
 [6,] 0.12874090 -0.06229042
 [7,] 0.12874079 -0.06229026
 [8,] 0.06166378  0.01820548
 [9,] 0.06166401  0.01820544
[10,] 0.06166317  0.01820586
> lcd$beta
 [1] 4.274158 4.296518 4.341233 4.341053 4.340988 4.153412 4.153412 4.341237 4.341238 4.341235
```

```
> lcd$triang
      [,1] [,2] [,3]
 [1,]    6    4    3
 [2,]    6    8    4
 [3,]    7    8    4
 [4,]    7    9    1
 [5,]    7    9    8
 [6,]    6    9    3
 [7,]    6    9    8
 [8,]    5    7    4
 [9,]    5    7    1
[10,]    5   12    1
```

```
> lcd$bunique
           [,1]        [,2]
[1,] 0.19582201 -0.04887419
[2,] 0.12874079 -0.02651378
[3,] 0.06166755  0.01820171
[4,] 0.06168464  0.01816418
[5,] 0.06174931  0.01809952
[6,] 0.12874079 -0.06229026
[7,] 0.06166337  0.01820556
> lcd$betaunique
[1] 4.274158 4.296518 4.341233 4.341053 4.340988 4.153412 4.341235
```



logcondens

Kaspar Rufibach and Lutz Duembgen

- Works only in one dimension
- Main function is logConDens
- Better results compared to LogConcDead
- Results for 50 points drawn from the normal distribution:

logcondens

```
> lcd$knots
[1] -2.6030024 -1.8731591 -1.6035737  0.5049737  0.5474262  2.2948347
```

LogConcDEAD

```
> lcd2$betaunique
[1]  0.9934601  1.1225721  1.1225797  0.2851384  1.1225854  1.1225816 -1.8973259  0.2853703  0.2978494
```

FMLogConc

Fabian Rathke and Christoph Schnörr

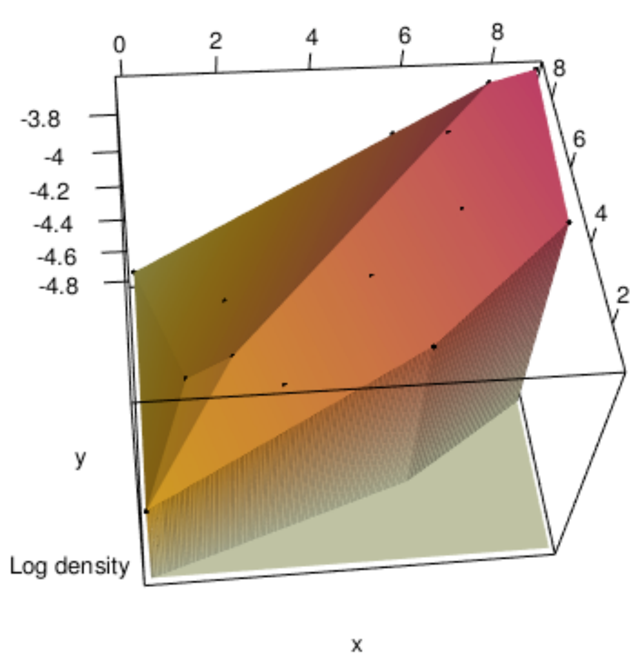
- Fast multivariate log-concave density estimation
- Faster than LogConcDEAD
- Main function is fmlcd
- No guarantee of returning the correct solution even theoretically

FMLogConc

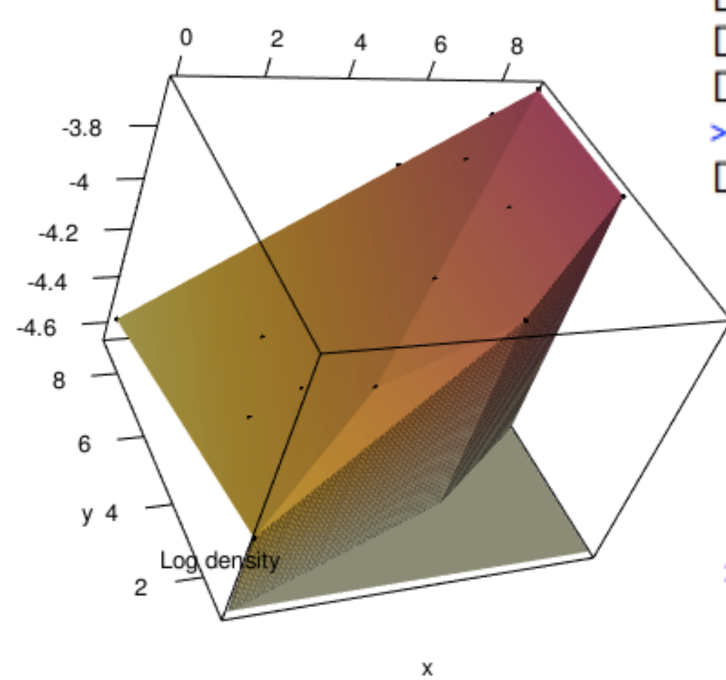
Problem 18

$X = \{(9,9), (3,3), (0,9), (0,1), (6,3), (1,4), (7,8), (2,4), (8,9), (7,6), (6,9), (9,5), (2,6), (5,5)\}$

LogConcDead



FMLogConc




```
> r$a
            [,1]         [,2]
[1,] -0.10354396  0.028688619
[2,] -0.10224575  0.027390413
[3,] -0.10215564  0.027289032
[4,] -0.07635489 -0.011412094
[5,] -0.07635489  0.001488281
[6,] -0.07635489  0.001488281
> r$b
[1] 4.313425 4.314723 4.314825 4.353526 4.314825 4.314825
```

```
> r$aSparse
             [,1]        [,2]
[1,] -0.076354887 -0.10354396
[2,]  0.001488281  0.02868862
> r$bSparse
[1] 4.314139 4.312740
```

FMLogConc

FMLogConc

- Smooth approximation of the objective function

$$\textbf{argmin} \sum_{i=1}^{n} \phi_\gamma(x_i) + \int_P \exp(-\phi_\gamma(x))dx$$

where

$$\phi_\gamma(x) := \gamma \log(\sum_{i=1}^{N} \exp(\frac{h_{y,i}(x)}{\gamma})) \textbf{ for } x \in P$$

$$\phi_\gamma(x) := \infty \textbf{ for } x \notin P$$

FMLogConc

- Smooth approximation of the objective function

$$\textbf{argmin} \sum_{i=1}^{n} \phi_\gamma(x_i) + \int_P \exp(-\phi_\gamma(x))dx$$

where

$$\phi_\gamma(x) := \gamma \log\left(\sum_{i=1}^{N} \exp\left(\frac{h_{y,i}(x)}{\gamma}\right)\right) \textbf{ for } x \in P$$

$$\phi_\gamma(x) := \infty \textbf{ for } x \notin P$$

- Then $\phi_\gamma(x) - \gamma \log N \leq \max\{h_{y,1}(x), \ldots, h_{y,N}(x)\} \leq \phi_\gamma(x)$ **for** $x \in P$

FMLogConc

- Smooth approximation of the objective function

$$\textbf{argmin} \sum_{i=1}^{n} \phi_\gamma(x_i) + \int_P \exp(-\phi_\gamma(x))dx$$

where

$$\phi_\gamma(x) := \gamma \log(\sum_{i=1}^{N} \exp(\frac{h_{y,i}(x)}{\gamma})) \textbf{ for } x \in P$$

$$\phi_\gamma(x) := \infty \textbf{ for } x \notin P$$

- Then $\phi_\gamma(x) - \gamma \log N \leq \max\{h_{y,1}(x), \ldots, h_{y,N}(x)\} \leq \phi_\gamma(x)$ **for** $x \in P$

- Solved using a quasi-Newton method

FMLogConc

- Smooth approximation of the objective function

$$\textbf{argmin} \sum_{i=1}^{n} \phi_\gamma(x_i) + \int_P \exp(-\phi_\gamma(x))dx$$

where

$$\phi_\gamma(x) := \gamma \log\left(\sum_{i=1}^{N} \exp\left(\frac{h_{y,i}(x)}{\gamma}\right)\right) \textbf{ for } x \in P$$

$$\phi_\gamma(x) := \infty \textbf{ for } x \notin P$$

- Then $\phi_\gamma(x) - \gamma \log N \leq \max\{h_{y,1}(x), \ldots, h_{y,N}(x)\} \leq \phi_\gamma(x)$ **for** $x \in P$

- Solved using a quasi-Newton method

- After each step of the algorithm some hyperplanes are removed

Theoretical CS

- Recently lots of interest in the complexity of log-concave density estimation
- Axelrod and Valiant "An Efficient Algorithm for High-Dimensional Log-Concave Maximum Likelihood"
  - Algorithm with runtime poly($n, d, \frac{1}{\epsilon}, r$) to compute a log-concave distribution whose log-likelihood is at most ϵ less than that of the MLE
- Diakonikolas, Sidiropoulos, and Stewart "Polynomial time algorithm for maximum likelihood estimation of multivariate log-concave densities"
  - Algorithm with runtime poly($n, d, \frac{1}{\epsilon}$) that with high probability computes a log-concave distribution whose log-likelihood is at most ϵ less than that of the MLE

Thank you!