Explicitly defined Population Attributes

Explicitly defined Population Attributes

> "Statistics is: the fun of finding patterns in data; the pleasure of making discoveries; the import of deep philosophical questions; the power to shed light on important decisions, and the ability to guide decisions..... in business, science, government, medicine, industry..." Professor David Hand

Population attribute

- **Variates** are characteristics of each unit in the population and can take numerical or categorical values.
  - The values of variates typically differ from unit to unit.
- **Population attributes** are summaries describing characteristics of the population.
  - Formally an attribute is a function applied to the entire population and determined through the variate values on individual units.
- Attributes can be numerical or graphical.
  - A scatterplot (using the population) is an attribute.
  - The coefficients of the least squares line fitted to this scatterplot
  - and the residual variation around the line are numerical attributes.
- A clear specification of the attributes of interest can resolve many issues. Lord's paradox, as presented by Hand (1994), is easily resolved by noting that it involves two different attributes. See our discussion to Hand.
  - David. J. Hand. "Deconstructing statistical questions.", J.R. Statist. Soc. A, 157, 1994.

Location Attributes

- the population total:
$$a(\mathcal{P}) = \sum_{u \in \mathcal{P}} y_u$$
- the population average:
$$a(\mathcal{P}) = \overline{y} = \frac{1}{N} \sum_{u \in \mathcal{P}} y_u$$
- various counts over the population
$$a(\mathcal{P}) = \sum_{u \in \mathcal{P}} I_A(y_u)$$
where $I_A(y)$ is the indicator function
$$I_A(y) = \begin{cases} 1 & y \in A \\ 0 & y \notin A \end{cases}$$
- the population proportion
$$a(\mathcal{P}) = \frac{1}{N} \sum_{u \in \mathcal{P}} I_A(y_u)$$

minimum
maximum
median
mid-range : $a(\mathcal{P}) = \frac{1}{2}(\max_{u \in \mathcal{P}} y_u + \min_{u \in \mathcal{P}} y_u)$
are also location type attributes

Variability Attributes

- the population variance:

$$a(\mathcal{P}) = \frac{1}{N}\sum_{u\in\mathcal{P}}(y_u - \overline{y})^2$$

- the population standard deviation:

$$a(\mathcal{P}) = SD_{\mathcal{P}}(y) = \sqrt{\frac{\sum_{u\in\mathcal{P}}(y_u - \overline{y})^2}{N}}$$

- Coefficient of variation:

$$a(\mathcal{P}) = \frac{SD_{\mathcal{P}}(y)}{\overline{y}}$$

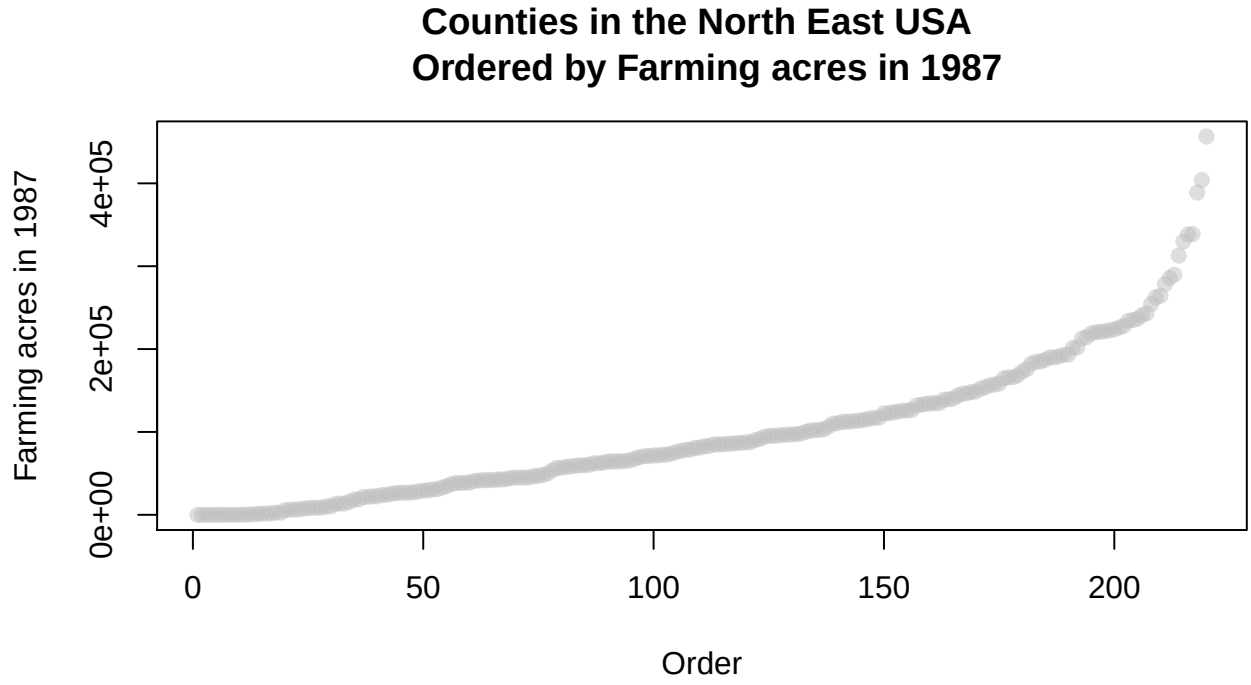
The Order Statistics

- Population attributes can also be an indexed collection of values,

$$y_{(1)} \leq y_{(2)} \leq \cdots \leq y_{(N)}$$

  which are the ordered values (including ties) of the variate values $y_u \in \mathcal{P}$

- An example,



Location Attributes based on Order Statistics

- the population minimum:

$$a(\mathcal{P}) = \min_{u\in\mathcal{P}} y_u = y_{(1)}$$

- the population max:
$$a(\mathcal{P}) = \max_{u \in \mathcal{P}} y_u = y_{(N)}$$
- the population mid-range:
$$a(\mathcal{P}) = \frac{1}{2}\left[\max_{u \in \mathcal{P}} y_u + \min_{u \in \mathcal{P}} y_u\right] = \frac{y_{(1)} + y_{(N)}}{2}$$
- the population median:
$$a(\mathcal{P}) = \text{median}_{u \in \mathcal{P}} y_u = \begin{cases} y_{([N+1]/2)} & \text{if } N \text{ is odd} \\ \frac{y_{(N/2)} + y_{(N/2+1)}}{2} & \text{if } N \text{ is even} \end{cases}$$
- the population quantiles: Q_1, Q_2, Q_3 where
  - Q_1 is 25^{th} ordered value, percentile or the first quantile,
  - Q_2 is the median and
  - Q_3 is 75^{th} ordered value, percentile or third quantiles.
- the population mid-hinge:
$$a(\mathcal{P}) = \frac{Q_1 + Q_3}{2}$$
- the population trimean (a resistant measure of central tendency):
$$a(\mathcal{P}) = \frac{Q_1 + 2Q_2 + Q_3}{4}$$

Variability Attributes based on Order Statistics

- the population range:
$$a(\mathcal{P}) = \max_{u \in \mathcal{P}} y_u - \min_{u \in \mathcal{P}} y_u$$
- the population interquantile range IQR:
$$a(\mathcal{P}) = Q_3 - Q_1$$
where Q_1 and Q_3 are 25^{th} and 75^{th} percentiles or the first and third quantiles.
- Median Absolute Deviation (MAD), the median of the absolute differences of the y_u and the median, is
$$a(\mathcal{P}) = \text{median}_{u \in \mathcal{P}} \left|y_u - \text{median}_{u \in \mathcal{P}} y_u\right|$$

Population Skewness Attributes

- Pearson's moment coefficient of Skewness:
$$a(\mathcal{P}) = \frac{\frac{1}{N}\sum_{u \in \mathcal{P}} \left(y_u - \overline{y}\right)^3}{\left[SD_{\mathcal{P}}(y)\right]^3}$$
- Pearson's second skewness coefficient (median skewness) given by
$$a(\mathcal{P}) = 3 \times \left(\overline{y} - \text{median}_{u \in \mathcal{P}} y_u\right) / SD_{\mathcal{P}}(y)$$
- Bowley's measure of skewness based on the quantiles
$$a(\mathcal{P}) = \frac{(Q_3 + Q_1)/2 - Q_2}{(Q_3 - Q_1)/2} = \frac{\text{midhinge} - \text{median}_{u \in \mathcal{P}} y_u}{\text{IQR}}$$

Agriculture Data Example

A number of population attributes can be calculated via `summary(agpop)`

```
summary(agpop)
```

```
##                county         state         acres92
##  WASHINGTON COUNTY:  30   TX     : 254   Min.   :    -99
##  JEFFERSON COUNTY :  25   GA     : 159   1st Qu.:  80903
##  FRANKLIN COUNTY  :  24   KY     : 120   Median : 191648
##  JACKSON COUNTY   :  23   MO     : 114   Mean   : 306677
##  LINCOLN COUNTY   :  23   KS     : 105   3rd Qu.: 366886
##  MADISON COUNTY   :  19   IL     : 102   Max.   :7229585
##  (Other)          :2934   (Other):2224
##     acres87           acres82           farms92          farms87
##  Min.   :    -99   Min.   :    -99   Min.   :   0.0   Min.   :   0.0
##  1st Qu.:  86236   1st Qu.:  96397   1st Qu.: 295.0   1st Qu.: 318.5
##  Median : 199864   Median : 207292   Median : 521.0   Median : 572.0
##  Mean   : 313016   Mean   : 320194   Mean   : 625.5   Mean   : 678.3
##  3rd Qu.: 372224   3rd Qu.: 377065   3rd Qu.: 838.0   3rd Qu.: 921.0
##  Max.   :7687460   Max.   :7313958   Max.   :7021.0   Max.   :7590.0
##
##     farms82          largef92         largef87         largef82
##  Min.   :   0.0   Min.   :  0.00   Min.   :  0.00   Min.   :  0.00
##  1st Qu.: 345.0   1st Qu.:  8.00   1st Qu.:  8.00   1st Qu.:  8.00
##  Median : 616.0   Median : 30.00   Median : 27.00   Median : 25.00
##  Mean   : 728.1   Mean   : 56.18   Mean   : 54.86   Mean   : 52.62
##  3rd Qu.: 991.0   3rd Qu.: 75.00   3rd Qu.: 70.00   3rd Qu.: 65.00
##  Max.   :7394.0   Max.   :579.00   Max.   :596.00   Max.   :546.00
##
##     smallf92          smallf87          smallf82       region
##  Min.   :   0.00   Min.   :   0.00   Min.   :   0.00   NC:1054
##  1st Qu.:  13.00   1st Qu.:  17.00   1st Qu.:  16.00   NE: 220
##  Median :  29.00   Median :  35.00   Median :  34.00   S :1382
##  Mean   :  54.09   Mean   :  59.54   Mean   :  60.97   W : 422
##  3rd Qu.:  59.00   3rd Qu.:  67.00   3rd Qu.:  67.00
##  Max.   :4298.00   Max.   :3654.00   Max.   :3522.00
##
```

Agriculture Data Example Cont. 1

- The first two variates (`county` and `state`) are categorical,
  - the first most frequent values are shown
- The last variate `region`, which takes only four different values (`NC`, `NE`, `S`, `W`) so each count appears.

```
##                county         state      region
##  WASHINGTON COUNTY:  30   TX     : 254   NC:1054
##  JEFFERSON COUNTY :  25   GA     : 159   NE: 220
##  FRANKLIN COUNTY  :  24   KY     : 120   S :1382
##  JACKSON COUNTY   :  23   MO     : 114   W : 422
##  LINCOLN COUNTY   :  23   KS     : 105
##  MADISON COUNTY   :  19   IL     : 102
##  (Other)          :2934   (Other):2224
```

Agriculture Data Example Cont. 2

- The remaining variates are numeric and so summary
  - the average (or `Mean`),
  - the minimum and maximum,
  - the first and third quartiles
  - and the median.

```
##     acres92           acres87           acres82           farms92
##  Min.   :    -99   Min.   :    -99   Min.   :    -99   Min.   :   0.0
##  1st Qu.:  80903   1st Qu.:  86236   1st Qu.:  96397   1st Qu.: 295.0
##  Median : 191648   Median : 199864   Median : 207292   Median : 521.0
##  Mean   : 306677   Mean   : 313016   Mean   : 320194   Mean   : 625.5
##  3rd Qu.: 366886   3rd Qu.: 372224   3rd Qu.: 377065   3rd Qu.: 838.0
##  Max.   :7229585   Max.   :7687460   Max.   :7313958   Max.   :7021.0
##     farms87           farms82           largef92          largef87
##  Min.   :   0.0   Min.   :   0.0   Min.   :  0.00   Min.   :  0.00
##  1st Qu.: 318.5   1st Qu.: 345.0   1st Qu.:  8.00   1st Qu.:  8.00
##  Median : 572.0   Median : 616.0   Median : 30.00   Median : 27.00
##  Mean   : 678.3   Mean   : 728.1   Mean   : 56.18   Mean   : 54.86
##  3rd Qu.: 921.0   3rd Qu.: 991.0   3rd Qu.: 75.00   3rd Qu.: 70.00
##  Max.   :7590.0   Max.   :7394.0   Max.   :579.00   Max.   :596.00
##     largef82          smallf92          smallf87          smallf82
##  Min.   :  0.00   Min.   :   0.00   Min.   :   0.00   Min.   :   0.00
##  1st Qu.:  8.00   1st Qu.:  13.00   1st Qu.:  17.00   1st Qu.:  16.00
##  Median : 25.00   Median :  29.00   Median :  35.00   Median :  34.00
##  Mean   : 52.62   Mean   :  54.09   Mean   :  59.54   Mean   :  60.97
##  3rd Qu.: 65.00   3rd Qu.:  59.00   3rd Qu.:  67.00   3rd Qu.:  67.00
##  Max.   :546.00   Max.   :4298.00   Max.   :3654.00   Max.   :3522.00
```

Agriculture Data Example Cont. 3

- Looking at the number of acres devoted to farms (i.e. `acres92`, `acres87`, `acres82`) reveals something curious
  - the minimum of each is `-99` which is a strange value for the number of acres!
  - No acreage should be less than zero.
  - Missing data are encoded as `-99` in this data set.
  - These should be replaced by `NA` which is the standard representation for missing data in `R`.

```
##     acres92           acres87           acres82
##  Min.   :    -99   Min.   :    -99   Min.   :    -99
##  1st Qu.:  80903   1st Qu.:  86236   1st Qu.:  96397
##  Median : 191648   Median : 199864   Median : 207292
##  Mean   : 306677   Mean   : 313016   Mean   : 320194
##  3rd Qu.: 366886   3rd Qu.: 372224   3rd Qu.: 377065
##  Max.   :7229585   Max.   :7687460   Max.   :7313958
```

Agriculture Data Example Cont. 4

Encoding the missing values as `NA`. The summary of these variates will now reflect the changes.

```r
summary(agpop[,c("acres92", "acres87", "acres82")])
```

```
##     acres92           acres87           acres82
##  Min.   :      0   Min.   :      0   Min.   :      0
##  1st Qu.:  82446   1st Qu.:  87530   1st Qu.:  97835
##  Median : 193688   Median : 201728   Median : 209222
##  Mean   : 308582   Mean   : 315374   Mean   : 321973
##  3rd Qu.: 368482   3rd Qu.: 374576   3rd Qu.: 379172
##  Max.   :7229585   Max.   :7687460   Max.   :7313958
##  NA's   :19        NA's   :23        NA's   :17
```

NAs in R

- **Note** that many programs in `R` accommodate missing data in `NA`s and do something appropriate (typically they omit them).
  - For your own code, either you must consider what to do with `NA`s or ensure that the data do not have any `NA`s.
  - For example, the function `na.omit(...)` will remove rows which contain an `NA` from a data set. For other possibilities see `help("na.omit")`.

Invariance and equivariance of attributes

- Often variate values are reported in some unit of measurement:
  - for example a length measurement in metres, millimetres, yards, or miles;
  - a weight measurement in kilograms, grams, or pounds; a temperature measure in degrees Celsius, degrees Kelvin, or degrees Fahrenheit; or
  - a liquid volume in imperial gallons, US gallons, or litres.

- For any attribute of a population that is a function of measured variates y_u given in some measurement units,
  - it is of interest to understand how that attribute changes (or not) to changes in the units of measurement.

When units of measurement change

- Sometimes only the **scale** of measurement is changed:
  - 1 yard = 3 feet; 1 mile = 5280 feet; 1 metre = 1000 mm;
  - 1 imperial gallon = 4.54609 litres; 1 US gallon = 0.832674 imperial gallon;
  - 1 kilogram = 1000 grams = 2.20462 pounds.
- Sometimes only the **location** of the zero for that measurement is changed:
  - absolute zero is 0° Kelvin or 273° Celsius
  - 1 Celsius degree = 1 Kelvin degree (no change in scale of measurement)
- Sometimes both **location** and **scale** change:

  - water freezes at 0° Celsius = 32° Fahrenheit (location change)
  - 1 Celsius degree = 1.8 Fahrenheit degrees (scale change)

- Sometimes the change involves more than just a change in location and/or scale of measurement:
  - fuel economy might be reported in miles per gallon (US or Imperial) or litres per hundred kilometres.
  - the Richter scale for earthquakes is a logarithmic measure of the amplitude of seismic waves

Location

- For any attribute measured in some measurement units,
  - it is of interest to understand how that attribute changes (or not) to changes in the units of measurement.

For an attribute

$$a(\mathcal{P}) = a(y_1, \ldots, y_N)$$

we say that for any $m > 0$ and $b \in \mathbb{R}$, that the attribute is

- **location invariant** if
$$a(y_1 + b, \ldots, y_N + b) = a(y_1, \ldots, y_N)$$
- **location equivariant** if
$$a(y_1 + b, \ldots, y_N + b) = a(y_1, \ldots, y_N) + b$$

Example

- the population average is location-scale equivariant, let

$$y_u = m \times x_u + b$$

then

$$\overline{y} = \frac{1}{N} \sum_{u \in \mathcal{P}} y_u = \frac{1}{N} \sum_{u \in \mathcal{P}} (m \times x_u + b) = m\overline{x} + b$$

- **Exercise** show that
  - the population median is location-scale equivariant;
  - the ratio of the population average to the population median is scale invariant but is neither location invariant nor location equivariant.

Scale and Location

For an attribute $a(\mathcal{P}) = a(y_1, \ldots, y_N)$ we say that for any $m > 0$ and $b \in \mathbb{R}$, that the attribute is

- **scale invariant** if
$$a(m \times y_1, \ldots, m \times y_N) = a(y_1, \ldots, y_N)$$
- **scale equivariant** if
$$a(m \times y_1, \ldots, m \times y_N) = m \times a(y_1, \ldots, y_N)$$

- **location-scale invariant** if it is both location invariant and scale invariant, i.e.
$$a(m \times y_1 + b, \ldots, m \times y_N + b) = a(y_1, \ldots, y_N)$$
- **location-scale equivariant** if it is both location equivariant and scale equivariant, i.e.
$$a(m \times y_1 + b, \ldots, m \times y_N + b) = m \times a(y_1, \ldots, y_N) + b$$

Replication

- Another invariance/equivariance property of interest for population attributes is **replication invariance** and **replication equivariance**.
  - If a population $\mathcal{P}$ is duplicated $k-1$ times how does the attribute change on this new population denoted by $\mathcal{P}^k$.

- The attribute $a(\mathcal{P})$ is
  - **replication invariant** whenever $a(\mathcal{P}^k) = a(\mathcal{P})$ and
  - **replication equivariant** whenever $a(\mathcal{P}^k) = k \times a(\mathcal{P})$.

- **Verify**
  - Replication invariant population attributes include the average, the median, and the inter-quartile range.
  - A replication equivariant attribute is the population total $\sum_{u \in \mathcal{P}} y_u$ for any variate y.

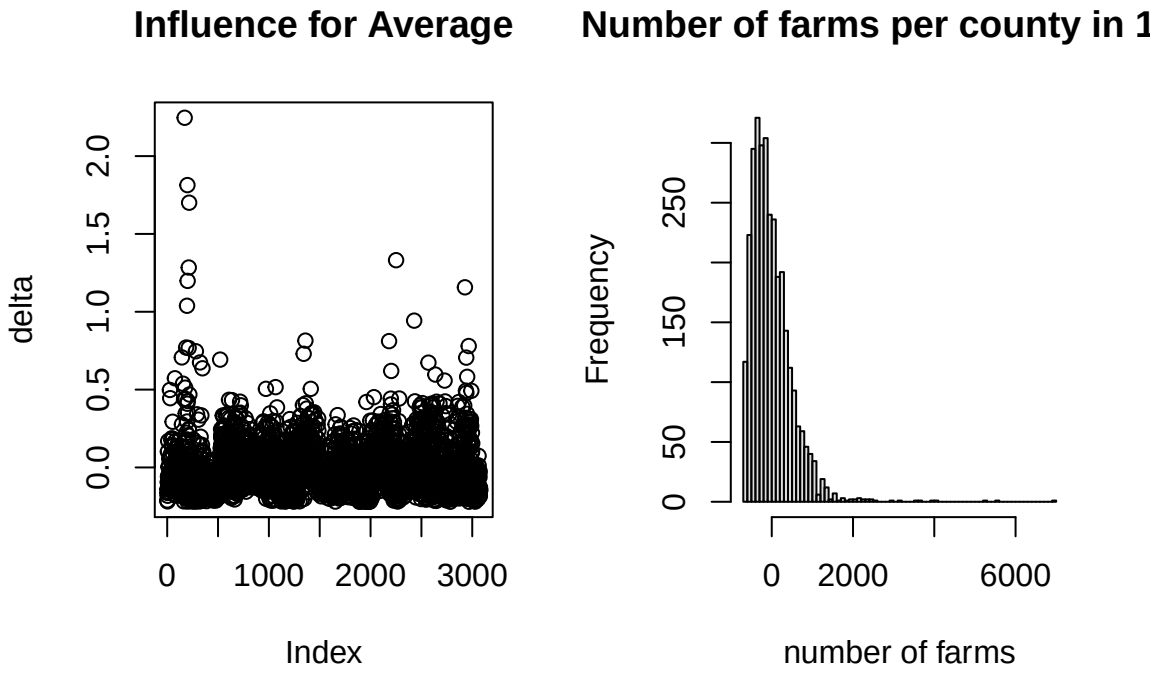
Influence and sensitivity curves

- One important characteristics of an population attribute is its sensitivity to the value of the variate on a single unit in the population.
  - We could quantify this with
$$\Delta(a, u) = a(y_1, \ldots, y_{u-1}, y_u, y_{u+1}, \ldots, y_N) - a(y_1, \ldots, y_{u-1}, y_{u+1}, \ldots, y_N)$$

- Ideally, no one unit's value should have greater influence than any other.
- If a unit had larger influence than the rest.
  - This requires further investigation as it might be in error
  - or it might be the most interesting unit in the population.

Influence - Example

```
y = agpop$farms87
N = length(y)
delta = sum(y)/N - (sum(y)-y)/(N-1)
SC    = N*delta

par(mfrow=c(1,2))
plot(delta, main="Influence for Average" )
hist(SC, col=adjustcolor("grey", alpha = 0.5),
     main="Number of farms per county in 1987",
     xlab="number of farms",
     breaks=100
     )
```



Sensitivity Curve

- Some attributes may be less sensitive to the effect if individual values.
  - To examine the effect of the value of this one unit, we take a population of size $N-1$ and add the value y and then we define the **sensitivity curve** of an attribute as

$$\begin{aligned} SC(y\ ;\ a(\mathcal{P})) &= \frac{a(y_1,\ldots,y_{N-1},y)-a(y_1,\ldots,y_{N-1})}{\frac{1}{N}} \\ &= N\,[\ a(y_1,\ldots,y_{N-1},y)-a(y_1,\ldots,y_{N-1})\ ] \end{aligned}$$

- Then can plot the **sensitivity curve** as a function of y (i.e. for y_N),
  - the sensitivity curve gives a scaled measure of the effect that a single variate value y has on the value of a population attribute $a(\mathcal{P})$.

- We can explore the sensitivity curve of any of the above attributes. These can be determined **mathematically** in general, but can also be determined **computationally** for any particular population.

Example: Arithmetic average

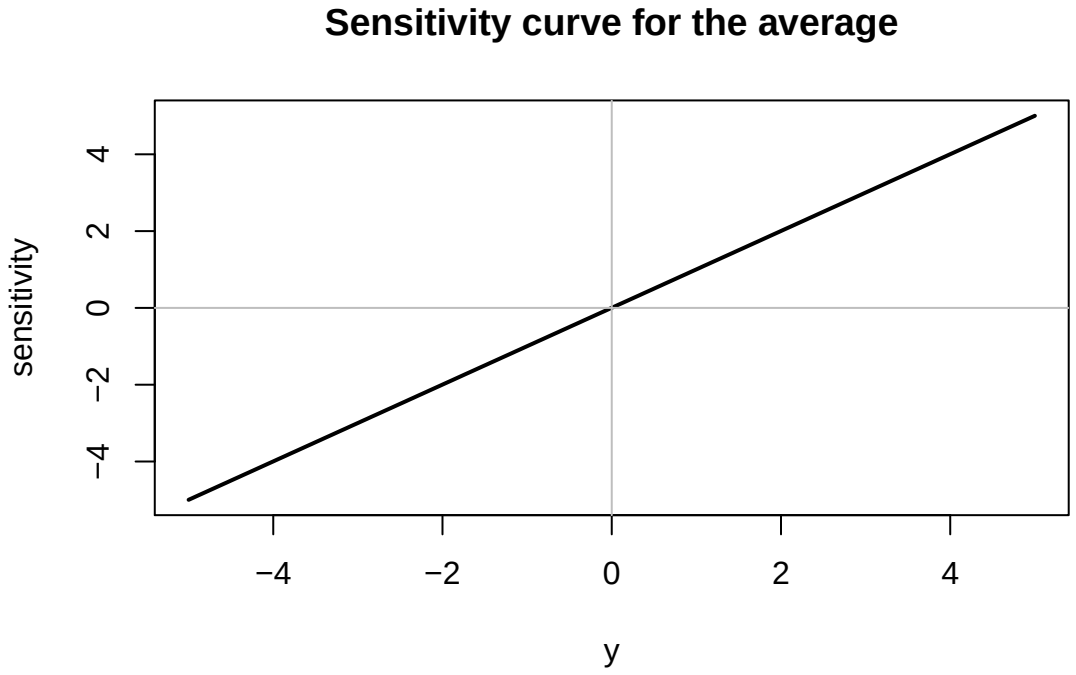
$$a(y_1, \ldots, y_N) = \frac{1}{N} \sum_{i=1}^{N} y_i = \overline{y}$$

- Derive the sentivity curve.

Example: Arithmetic average, SC Plot

We found the sentivity curve to be $SC(y) = y - \overline{y}_{N-1}$

```
set.seed(341)
ys <- rnorm(1000-1)
y <- seq(-5,5, length.out=200)

plot(y, y -  mean(ys), type="l", lwd = 2,
     main="Sensitivity curve for the average",
     ylab="sensitivity")
abline(h=0, v=0, col="grey")
```



- **Note** that the sensitivity curve here gets higher (or lower) without bound as $y \to \infty$ (or as $y \to -\infty$).
  - A single observation can change the average by a huge (even infinite) amount.
  - Averages may not be the best choice for a population attribute representing the location of a population.

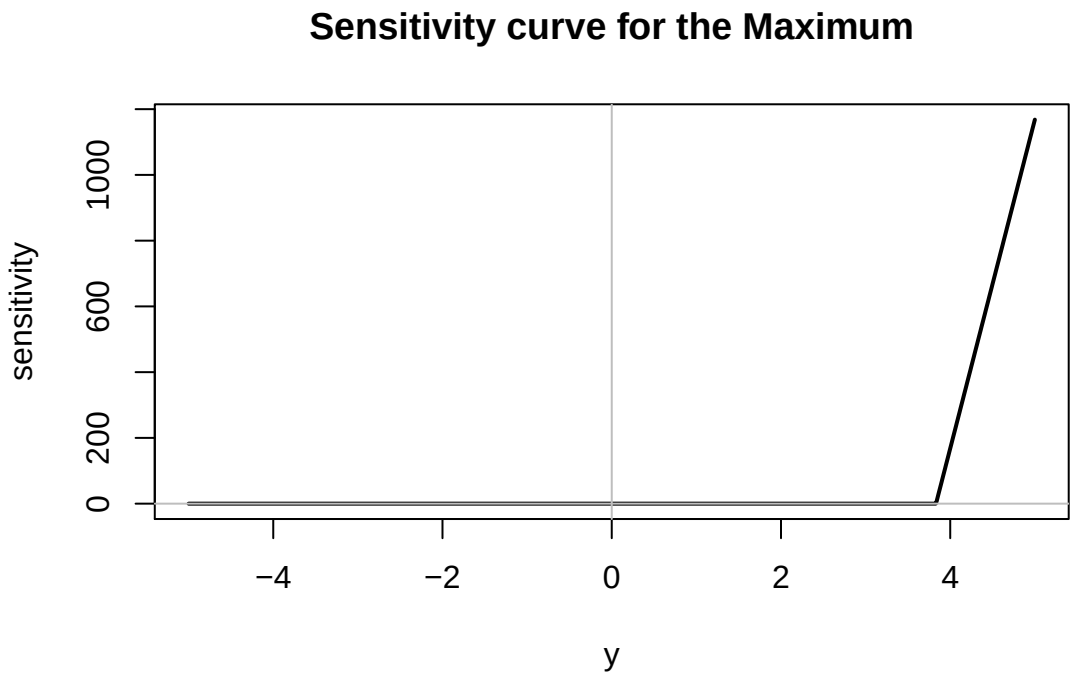
Example: Maximum

$$a(y_1, \ldots, y_N) = \max\{y_1, \ldots, y_N\} = y_{(N)}$$

- Derive the sentivity curve.

Maximum - Sentivity Curve

```
y <- seq(-5,5, length.out=1000)

sc = function(y.pop, y, attr, ...) {
  N <- length(y.pop) +1
  Map(function(y) {  N*(attr(c(y,y.pop),...) - attr(y.pop,...))  } ,y   )
  }

plot(y, sc(ys, y, max), type="l", lwd = 2,
     main="Sensitivity curve for the Maximum",
     ylab="sensitivity")
abline(h=0, v=0, col="grey")
```



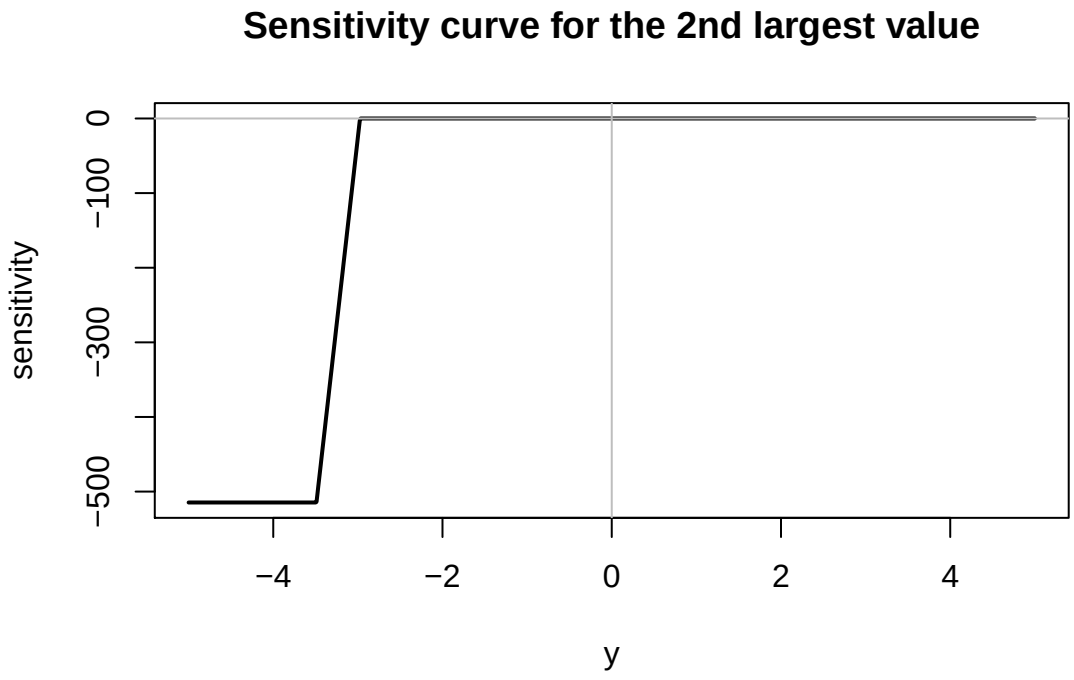
Example: 2^{nd} Order Statistic

$$a(y_1, \ldots, y_N) = y_{(2)}$$

- Derive the sentivity curve.

2^{nd} Order Statistic - Sensitivity Curve

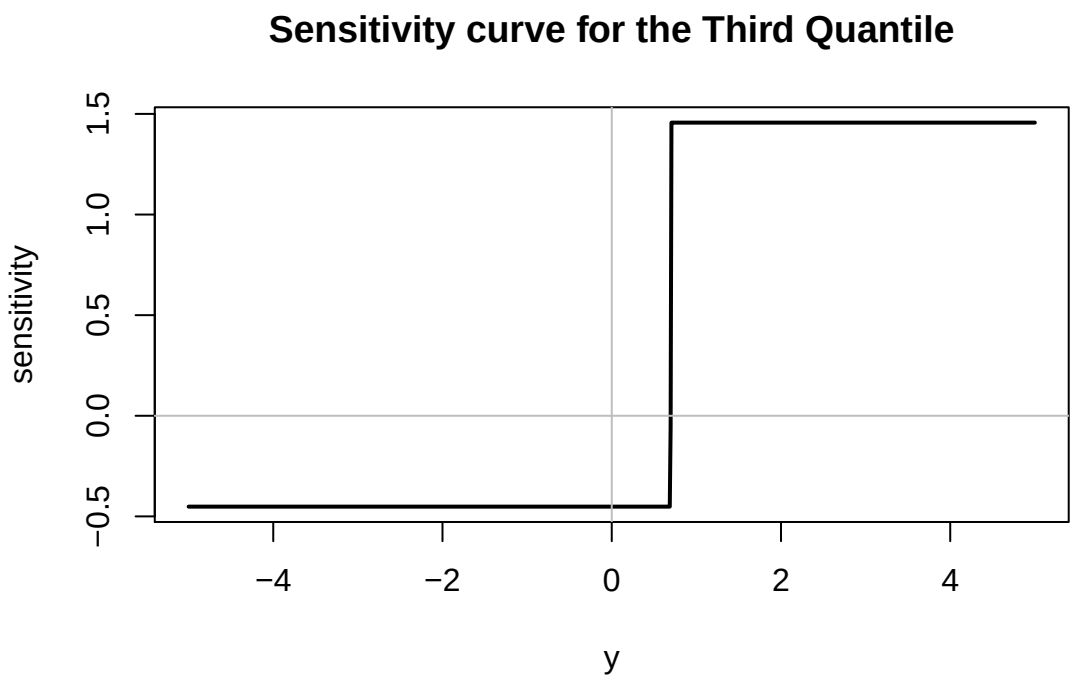
```
order.stat <- function(pop, k=1) {  sort(pop)[k]  }

plot(y, sc(ys, y, order.stat, k=2), type="l", lwd = 2,
     main="Sensitivity curve for the 2nd largest value",
     ylab="sensitivity")
abline(h=0, v=0, col="grey")
```



- **Excerise**
  - Describe how the plot changes with the value of k.
  - Derive the sensitivity for the k^{th} order statistic, i.e. $y_{(k)}$
  - Derive the sensitivity for the 2^{th} or k^{th} largest value, i.e. $y_{(N-1)}$ or $y_{(N-k+1)}$

Third Quantile - Sensitivity Curve

```
plot(y, sc(ys, y, quantile, p=0.75), type="l", lwd = 2,
     main="Sensitivity curve for the Third Quantile",
     ylab="sensitivity")
abline(h=0, v=0, col="grey")
```



Example: Median

Suppose $T_N(y_1, \ldots, y_N)$ is the median and that N is odd, that is $N = 2m + 1$. The ordered values are

$$y_{(1)} \leq y_{(2)} \leq \cdots \leq y_{(N)}$$

and then the median is

$$T_{N-1}(y_1, \ldots, y_{N-1}) = \frac{1}{2}\left(y_{(m)} + y_{(m+1)}\right).$$

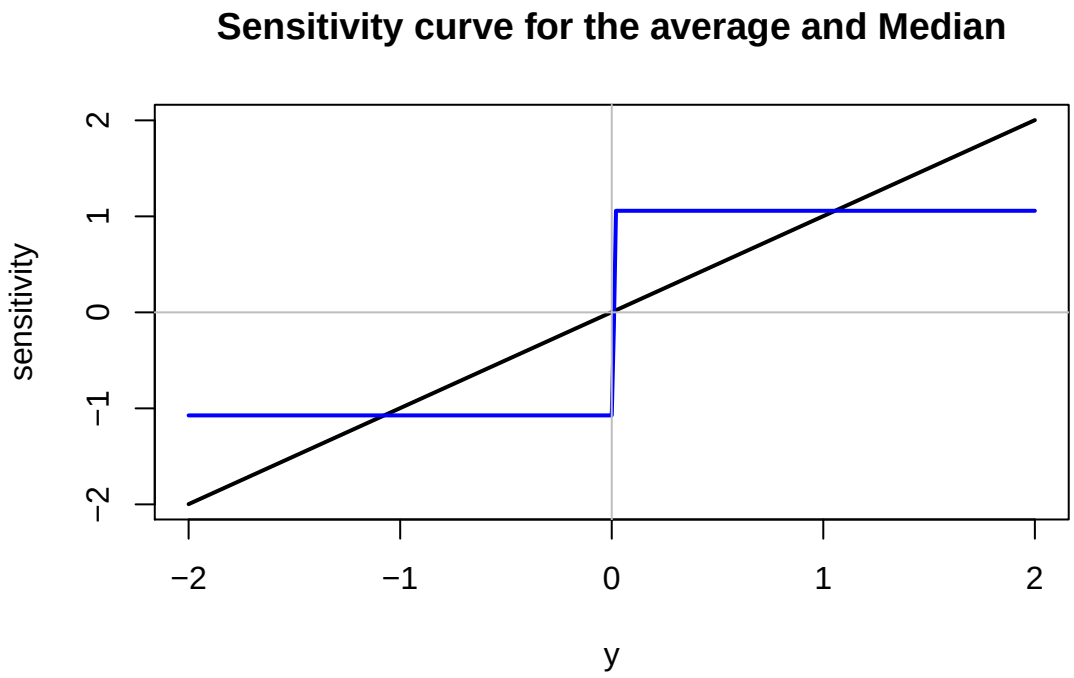
The sensitivity curve for the median is

$$SC(y) = \begin{cases} -\frac{N}{2}\left(y_{(m+1)} - y_{(m)}\right) & \text{if} \quad y < y_{(m)} \\ \frac{N}{2}\left(2y - y_{(m+1)} - y_{(m)}\right) & \text{if} \quad y_{(m)} \leq y \leq y_{(m+1)} \\ \frac{N}{2}\left(y_{(m+1)} - y_{(m)}\right) & \text{if} \quad y > y_{(m+1)} \end{cases}$$

which looks like a negative constant when $y < y_{(m)}$, is a positive constant at $y_{(m+1)} < y$, and is a simple straight line with positive slope when y is between $y_{(m)}$ and $y_{(m+1)}$.

As with the arithmetic average we can draw the sensitivity curve now for the median for any particular sample.

Example: Median & Average SC Plot



- Unlike the arithmetic average, the sensitivity curve for the median is at least bounded.
  - A single observation cannot change the median by very much.
  - This makes the median a very interesting population attribute for the location of a variate.

Breakdown point

- Another measure of robustness has been introduced called the **breakdown point**.
  - It gives an assessment of just how large a proportion of the data might be contaminated before the statistic breaks down.

```
x = rnorm(5)
x
```

```
## [1] 0.8307343 1.8328102 0.3049508 0.4303002 1.1113573
```

```
c(mean(x), median(x))
```

```
## [1] 0.9020306 0.8307343
```

```
# then if change the first value to infinity
y = x
y[1] = Inf
c(mean(y), median(y))
```

```
## [1]      Inf 1.111357
```

```r
# and the difference is

c(mean(y), median(y)) - c(mean(x), median(x))
```

```
## [1]       Inf 0.2806229
```

Breakdown point

- The breakdown point of a statistic is the largest possible fraction of the observations can be changed to something very big (infinte) and have the error still be relatively small (finite).
- Examples, the breakpoint for
  - the average is $1/N$ or zero asymptotically,
  - and the median is $1/2$, that is half of the data has to go to infinity before the median breaks down.
- Attributes with high breakdown points are called **resistant**.

Graphical attributes

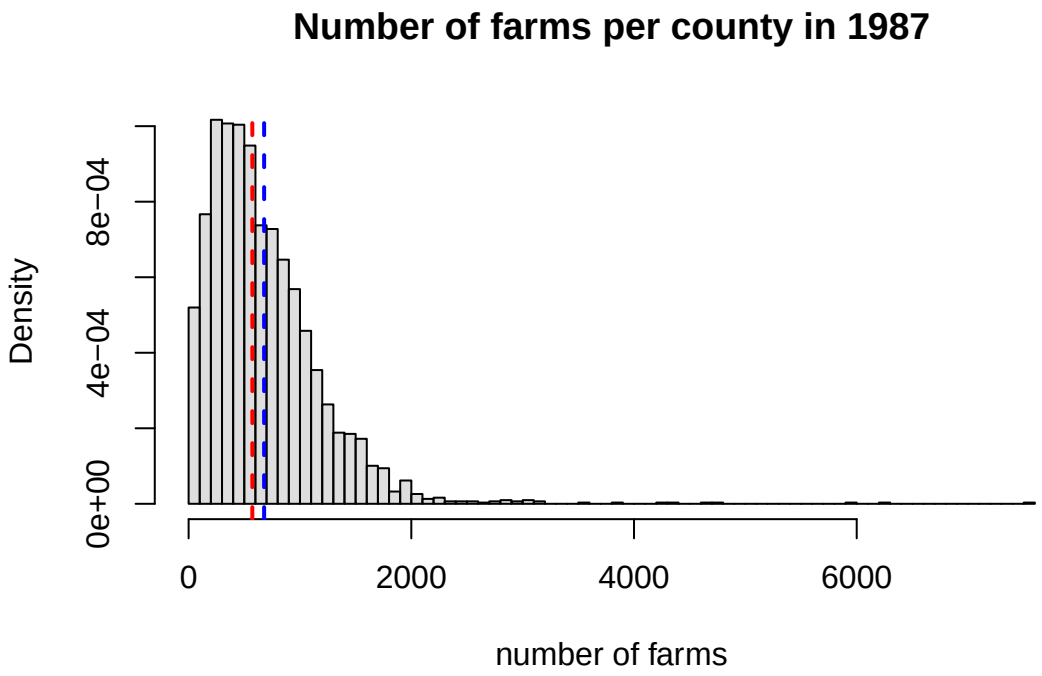
Population attributes can also be entirely graphical as in

- histograms of y_u values
- bar plots of y_u values
- scatterplots of pairs (x_u, y_u)
- scatterplots of quantiles and ranks of y_u.

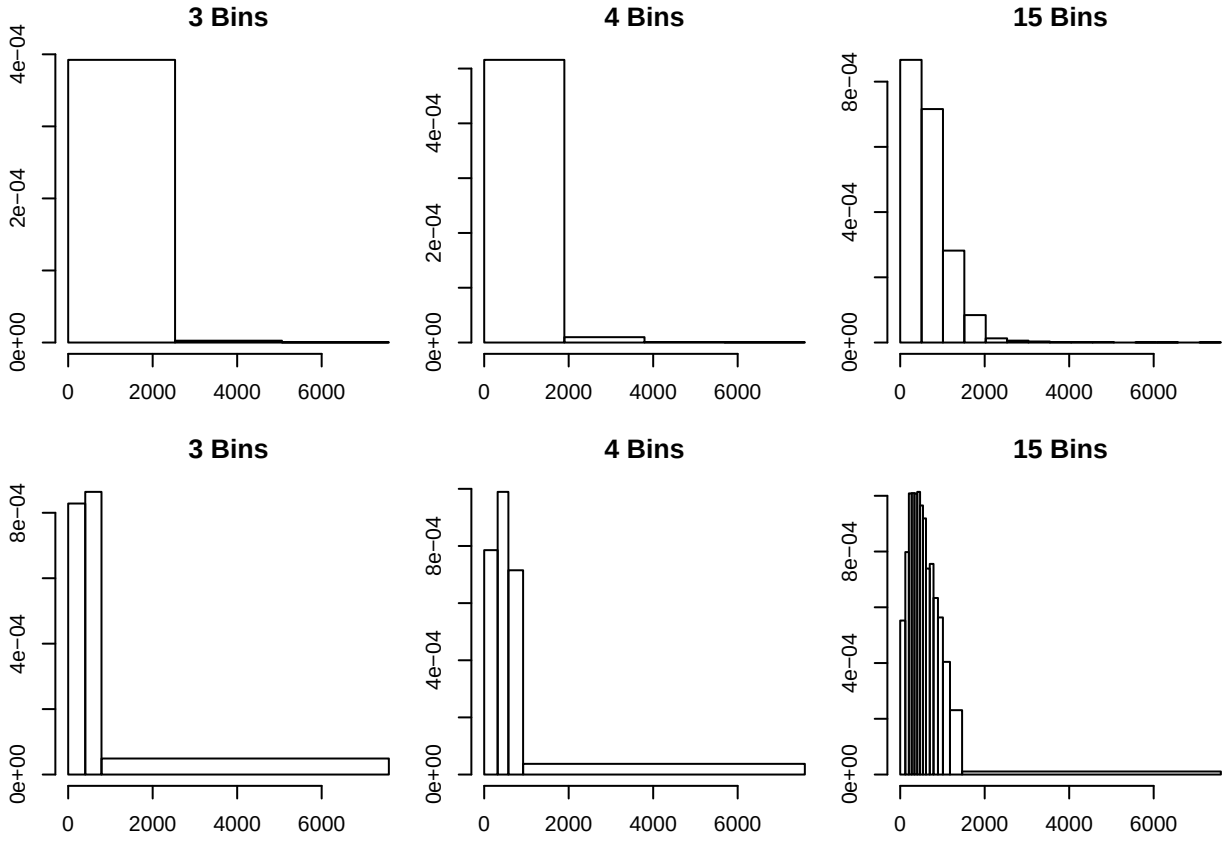
Histograms

- Consider measurements $\{y_1, y_2, \ldots, y_N\}$ on a variate y.
  - Partition the range of y into k non-overlapping intervals, called **bins**, $I_j = [a_{j1}, a_j)$, $j = 1, 2, \ldots, k$ and then calculate for $j = 1, \ldots, k$.
- We can define bins two ways.
  - The bins of equal size.
  - The bins with equal number of elements but varying size.

```r
hist(agpop$farms87, col=adjustcolor("grey", alpha = 0.5),
     main="Number of farms per county in 1987",
     xlab="number of farms",
     breaks=100, prob=TRUE )
abline(v=c(mean(agpop$farms87), median(agpop$farms87) ),
       col=c("blue","red"),lwd=2,lty=2)
```

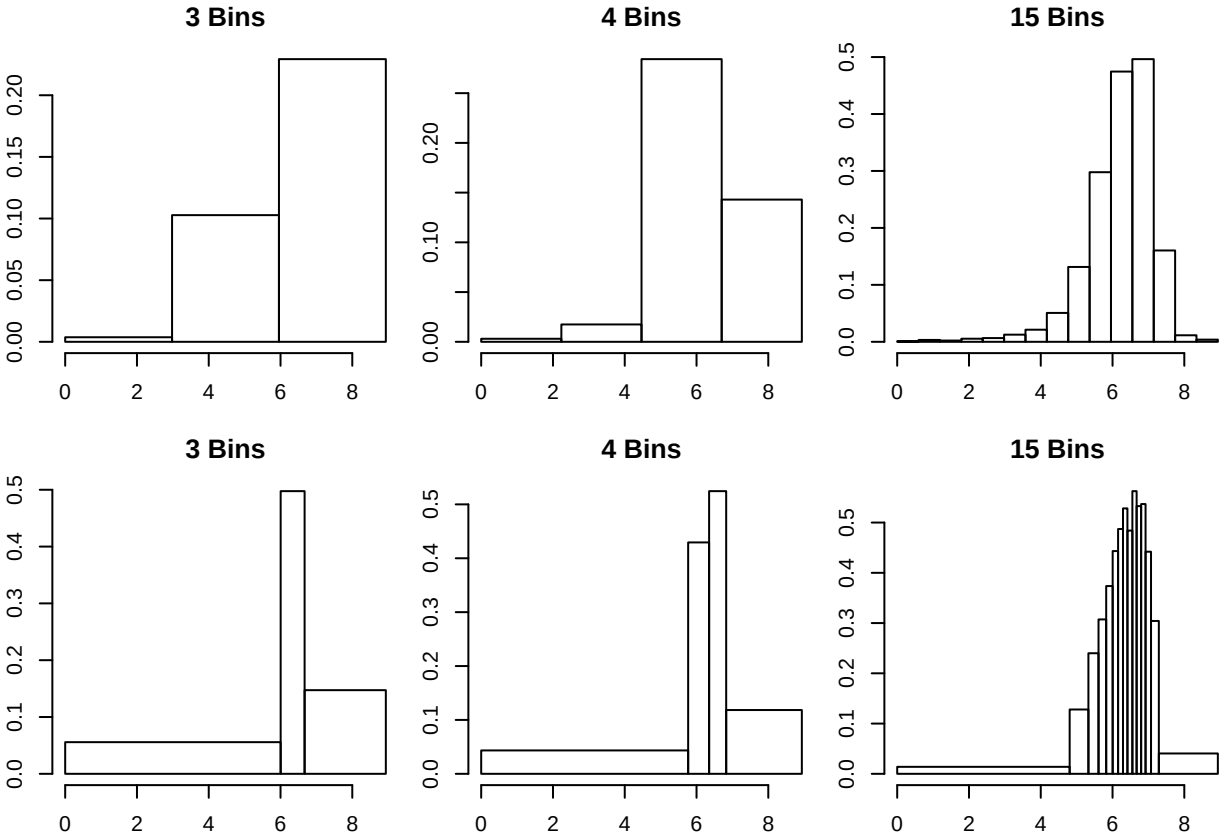


Histograms with varying bins



Transformated Data

- Histograms of `log(agpop$farms87+1)`



Rules for the Number of Bins

- Sturges rule

$$\text{the number of bins should be } = \lceil \log_2(n) + 1 \rceil$$

- Freedman–Diaconis rule

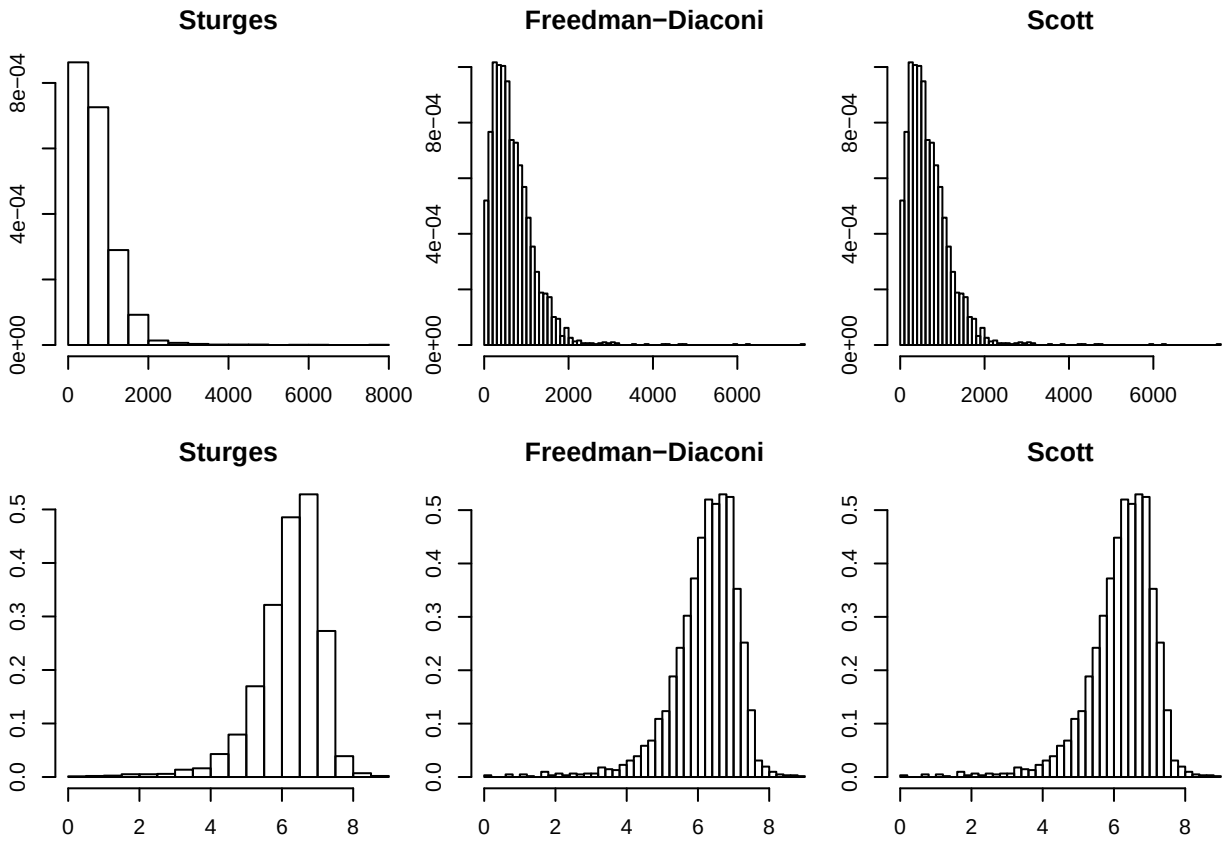
$$\text{Bin size} = 2\frac{\text{IQR}(x)}{n^{1/3}}$$

- Scott's rule

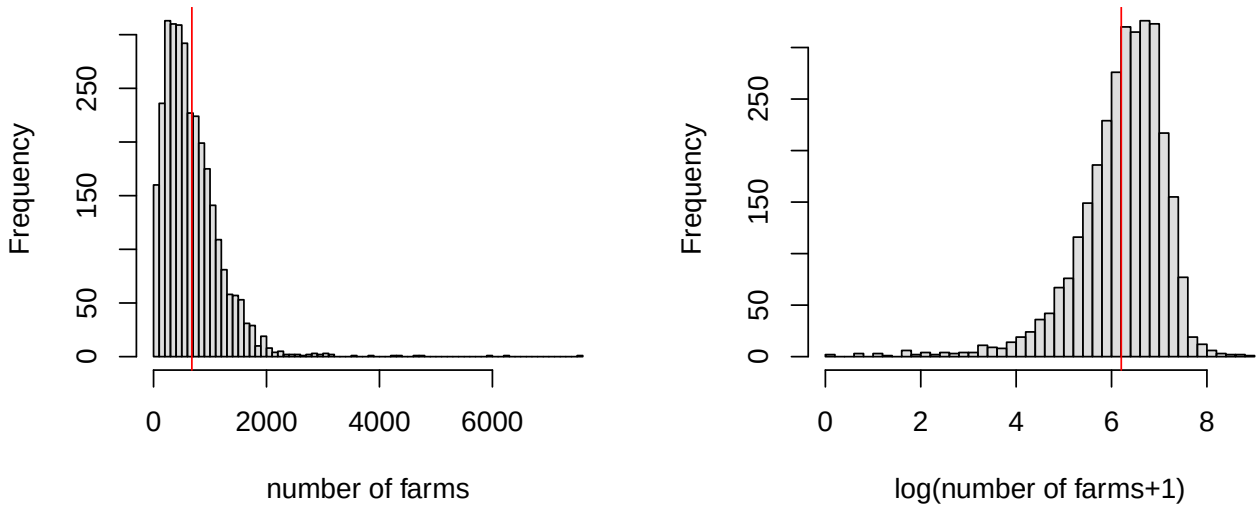
$$\text{Bin size} = 3.5\frac{\sigma}{n^{1/3}}$$

Application of the Rules

- Histograms using varying bin rules. The
  - first row is `Number of farms` and
  - second row is `log(Number of farms+1)`.



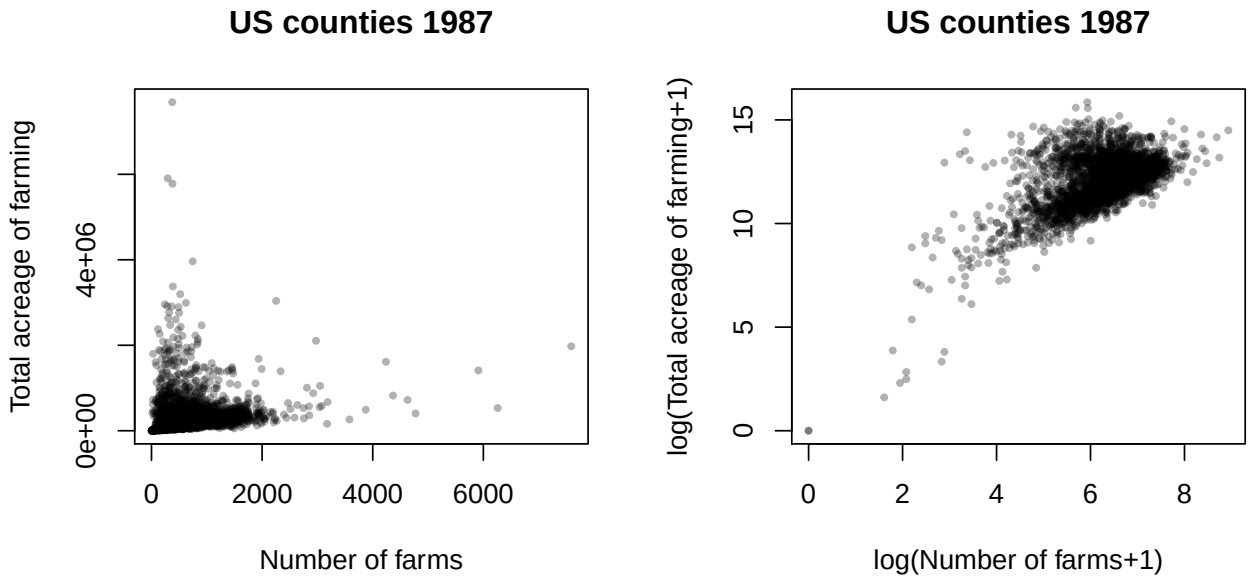
Orginal vs. Transformed Data



- In which plot is the average a better measure of the center?

Scatter Plots

```r
par(mfrow=c(1,2))
plot(agpop$farms87, agpop$acres87, pch = 19, cex=0.5,
     col=adjustcolor("black", alpha = 0.3),
     xlab = "Number of farms", ylab = "Total acreage of farming",
     main = "US counties 1987")

plot( log(agpop$farms87+1), log(agpop$acres87+1), pch = 19, cex=0.5,
     col=adjustcolor("black", alpha = 0.3),
     xlab = "log(Number of farms+1)", ylab = "log(Total acreage of farming+1)",
     main = "US counties 1987")
```



Power Transformations

For any variate y, it is sometimes helpful to re-express the values in a non-linear way via a transformation $T(y)$ so that on the re-expressed scale attributes are easier to define, to understand, or simply to determine.

- A common used re-expression when $y > 0$ is the family of **power transformations** which is indexed by a power α. The general form is

$$T_\alpha(y) = \begin{cases} y^\alpha & \alpha \neq 0 \\ \log(y) & \alpha = 0 \end{cases}$$

- These transformations are monotonic, in the sense that

$$y_u < y_v \iff T_\alpha(y_u) < T_\alpha(y_v),$$

  that is they preserve the order of the variate values associated with the units u and v.

  – What does change, often dramatically, is the relative positions of the variate values.

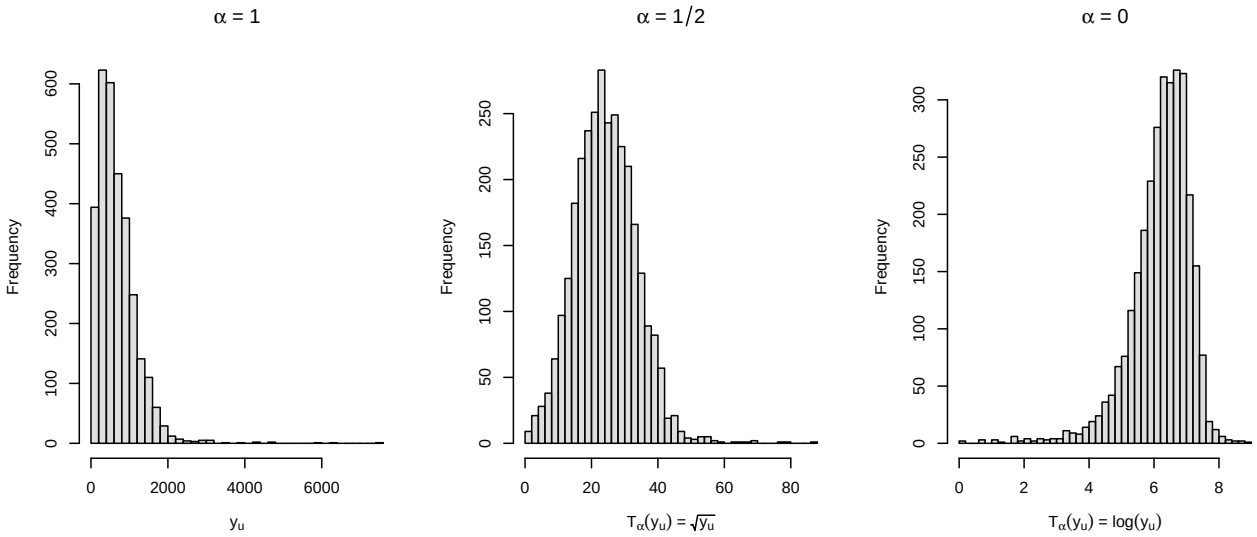


Figure 1: Effect of power transformation on number of farms in 1987

Power Transformation Example

Power Transformations

- If $y > 0$, the family of **power transformations** which is indexed by a power α

$$T_\alpha(y) = \begin{cases} y^\alpha & \alpha \neq 0 \\ \log(y) & \alpha = 0 \end{cases}$$

- **Note** the primarily purpose of the transformation is in changing the symmetry of the histogram,
  - a more convenient mathmatical form is

$$T_\alpha(y) = \frac{y^\alpha - 1}{\alpha} \quad \forall\ \alpha.$$

R code power transformation

```
powerfun <- function(x, alpha) {
  if(sum(x <= 0) > 1) stop("x must be positive")
  if (alpha == 0)
    log(x)
  else
    (x^alpha-1)/alpha
}
```

A more computationally efficient with minmial calculational errors, the following implementation is preferred:

```
powerfun <- function(x, alpha) {
  if(sum(x <= 0) > 1) stop("x must be positive")
  if (alpha == 0)
    log(x)
```

```
  else if (alpha > 0) {
    x^alpha
  } else -x^alpha
}
```

Power Transformation Example

Which α?

- α can take any real value in principle,
  - but in practice, we restrict the values to a small set.
  - The powers should be restricted to those which are easily interpretable.
  - John Tukey suggested (Tukey 1977) imagining that the set of powers were arranged in a "ladder" with the smallest powers on the bottom and the largest on the top.

Now one simply moves "up" or "down" on Tukey's ladder of powers to arrive at a re-expression that achieves the desired effect on the data values.

alpha	ladder
...	up
2	|
1	<- original values
1/2	|
1/3	|
0	|
-1/3	|
-1/2	|
-1	|
-2	|
...	down

How to pick α?

Two different, but related, effects are often of interest.

- First, producing more symmetric looking histogram.
- Second, in the case of two variates x and y, we might like to have each variate re-expressed separately so that the all pairs of the values might be well described as being roughly linearly related.
  - That is, imagine (for all $u \in \mathcal{P}$) a scatterplot of all pairs (x_u, y_u).
  - Of interest is whether there are powers α_x and α_y for each such that the scatterplot of the re-expressed pairs $(T_{\alpha_x}(x), T_{\alpha_y}(y))$ lie more nearly on a straight line.

Fortunately, for each of these effects there is a corresponding bump rule that indicates the direction (up or down) to move on Tukey's ladder to achieve it.

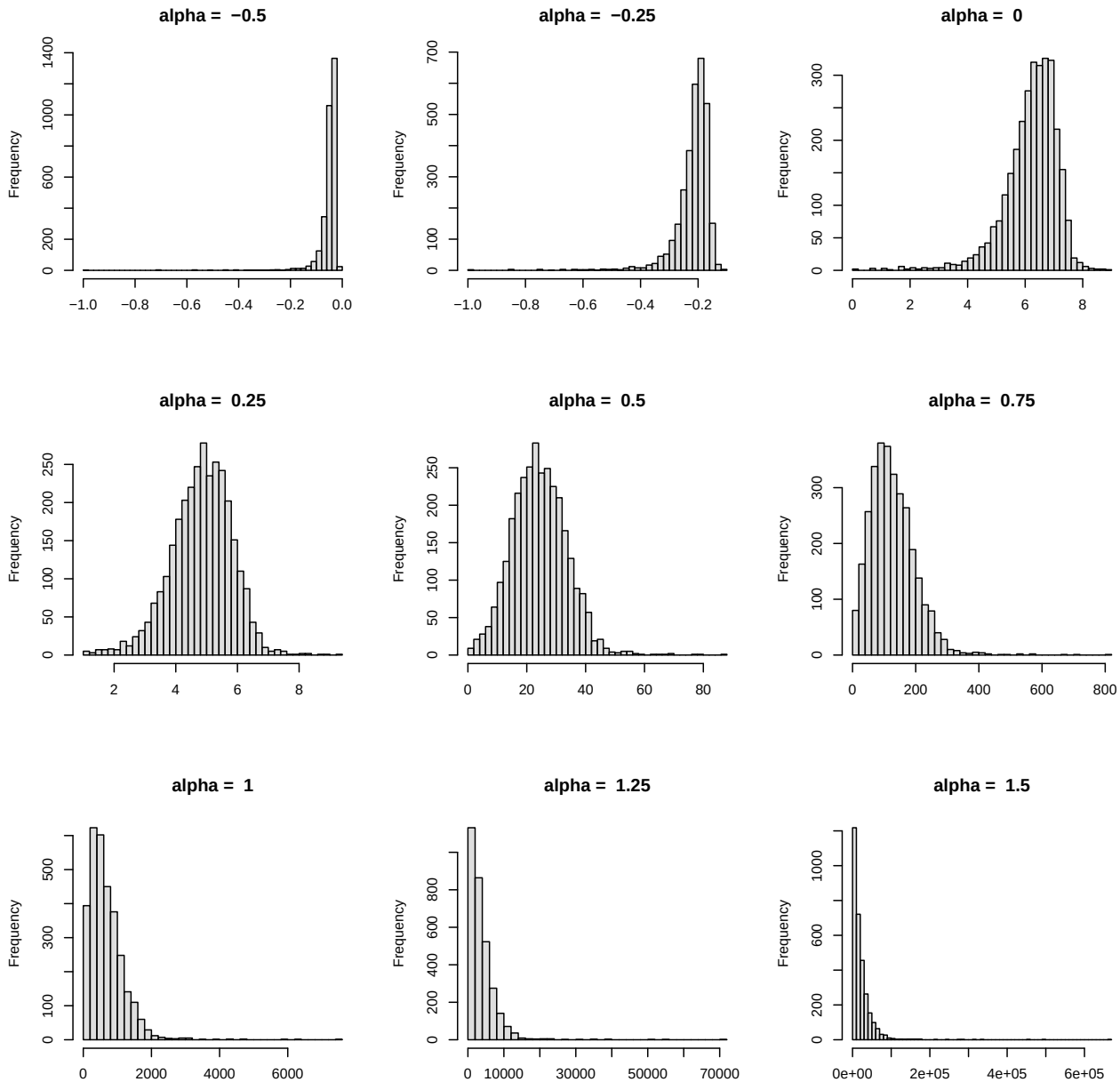


Figure 2: Effect of power transformation on number of farms in 1987

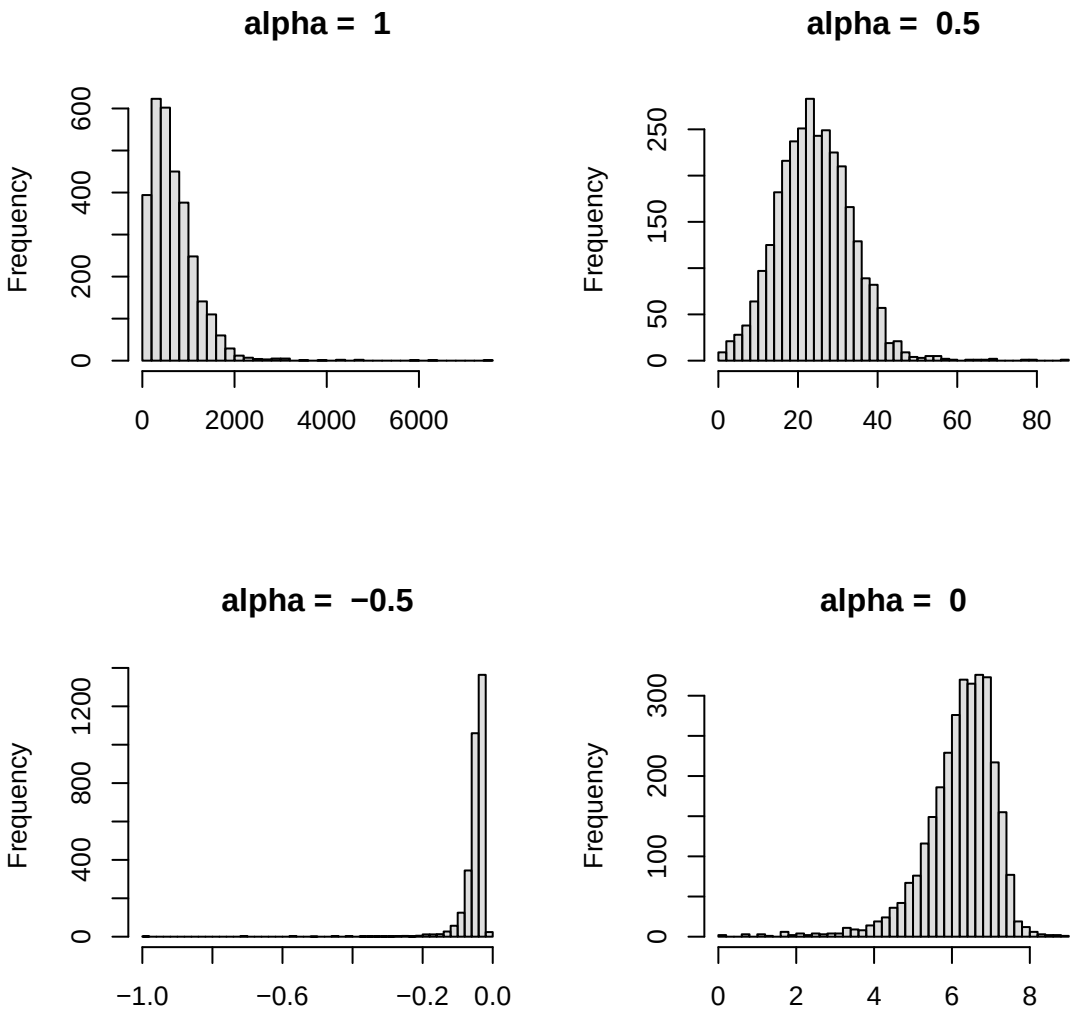


Figure 3: Effect of power transformation on number of farms in 1987

Bump rule 1: Making histograms more symmetric

- The rule is that the location of the "bump" in the histogram (where the points are concentrated) tells you which way to "move" on the ladder.
  - If the bump is on "lower" values, then move the power "lower" on the ladder;
  - if it is on the "higher" values, then move the power "higher" on the ladder.

Bump rule 1: Making histograms more symmetric

- The rule is that the location of the "bump" in the histogram (where the points are concentrated) tells you which way to "move" on the ladder.
  - If the bump is on "lower" values, then move the power "lower" on the ladder;
  - if it is on the "higher" values, then move the power "higher" on the ladder.

Straightening scatterplots

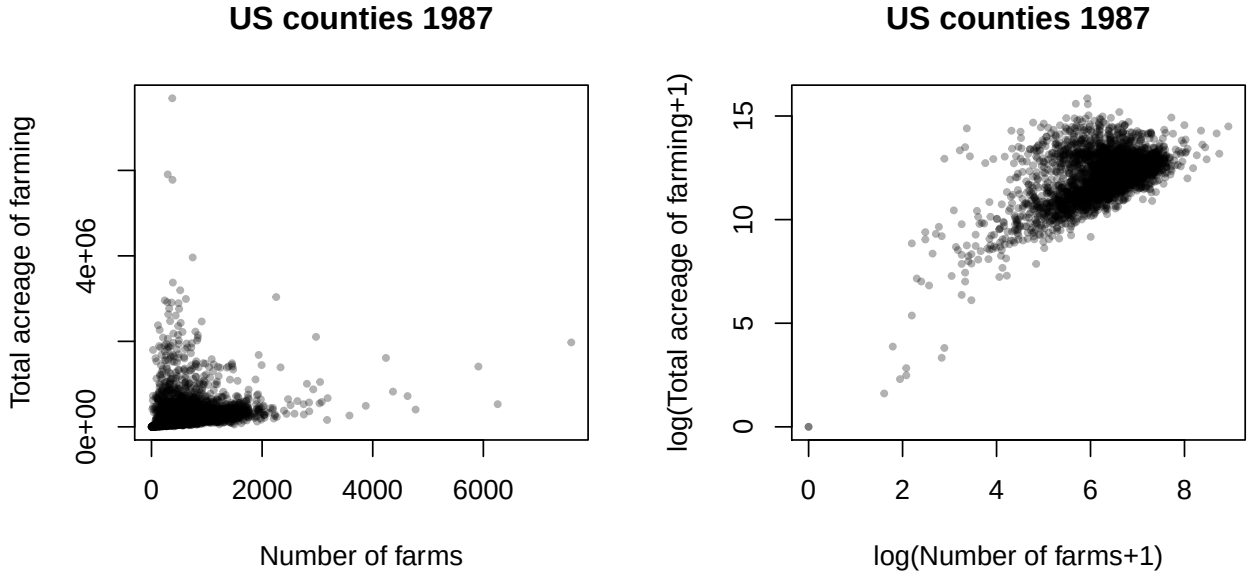
A scatterplot of (x_u, y_u) for $u \in \mathcal{P}$ may be "straightened" by applying (possibly) different power transforms to each coordinate to give a new (hopefully straighter looking) scatterplot of the re-expressed data $(T_{\alpha_x}(x_u), T_{\alpha_y}(y_u))$.

- Because each of the coordinates has its own power transformation there will be two different ladders of transformation
  - the x ladder and
  - the y ladder.
- As with histograms, there is a "bump" to tell you how to move on the ladder.
  - In this case, the "bump" corresponds to the curvature appearing in the scatterplot.
  - This is only approximate in practice but reduces to one of four different possibilities.

Bump rule 2: Agriculture Data - A

- Number of farms versus Total acreage of farming

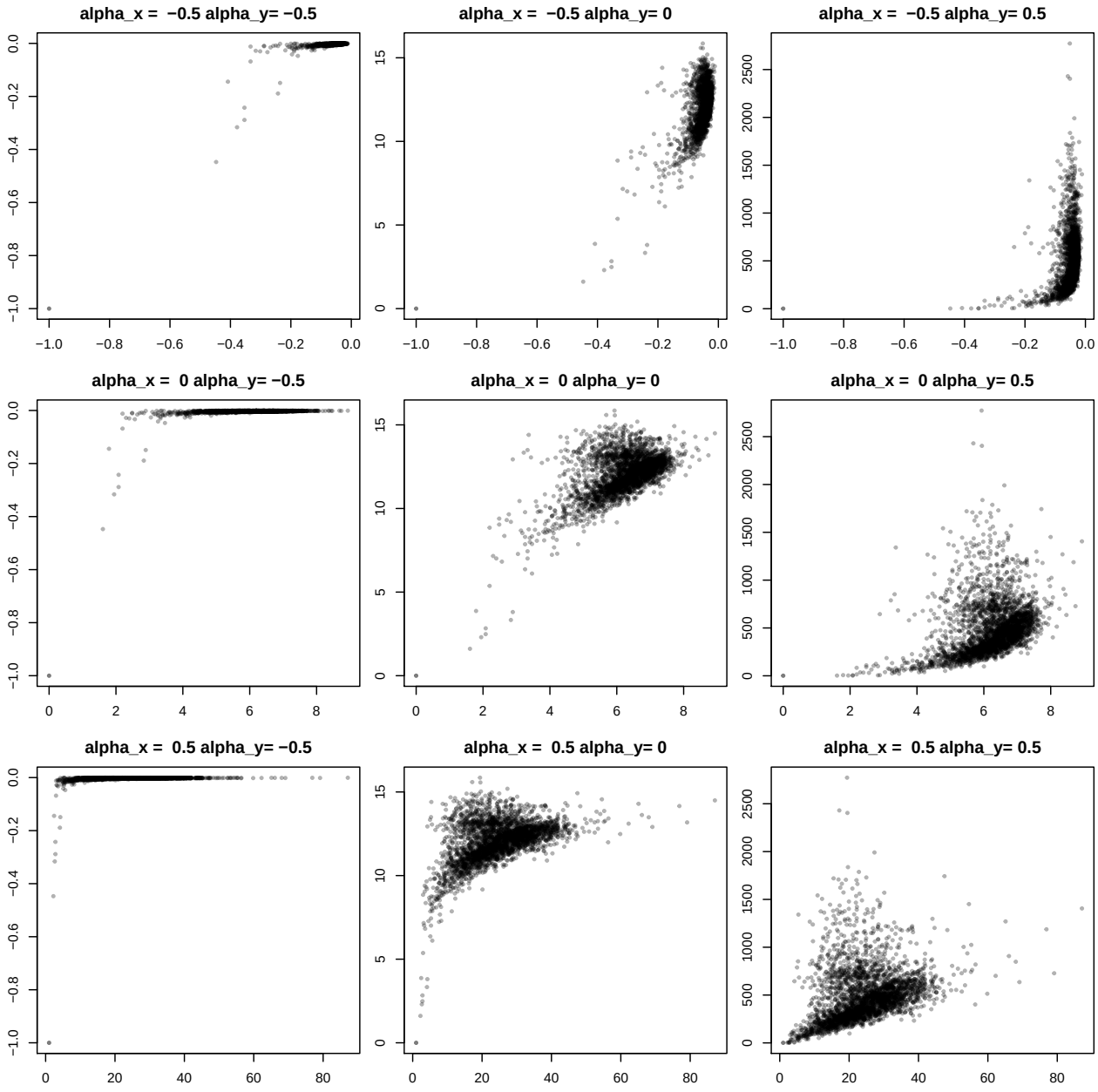
```
par(mfrow=c(1,2))
plot( agpop$farms87, agpop$acres87, pch = 19, cex=0.5,
     col=adjustcolor("black", alpha = 0.3),
     xlab = "Number of farms", ylab = "Total acreage of farming",
     main = "US counties 1987")

plot( log(agpop$farms87+1), log(agpop$acres87+1), pch = 19, cex=0.5,
     col=adjustcolor("black", alpha = 0.3),
     xlab = "log(Number of farms+1)", ylab = "log(Total acreage of farming+1)",
     main = "US counties 1987")
```



Bump rule 2: Agriculture Data - B

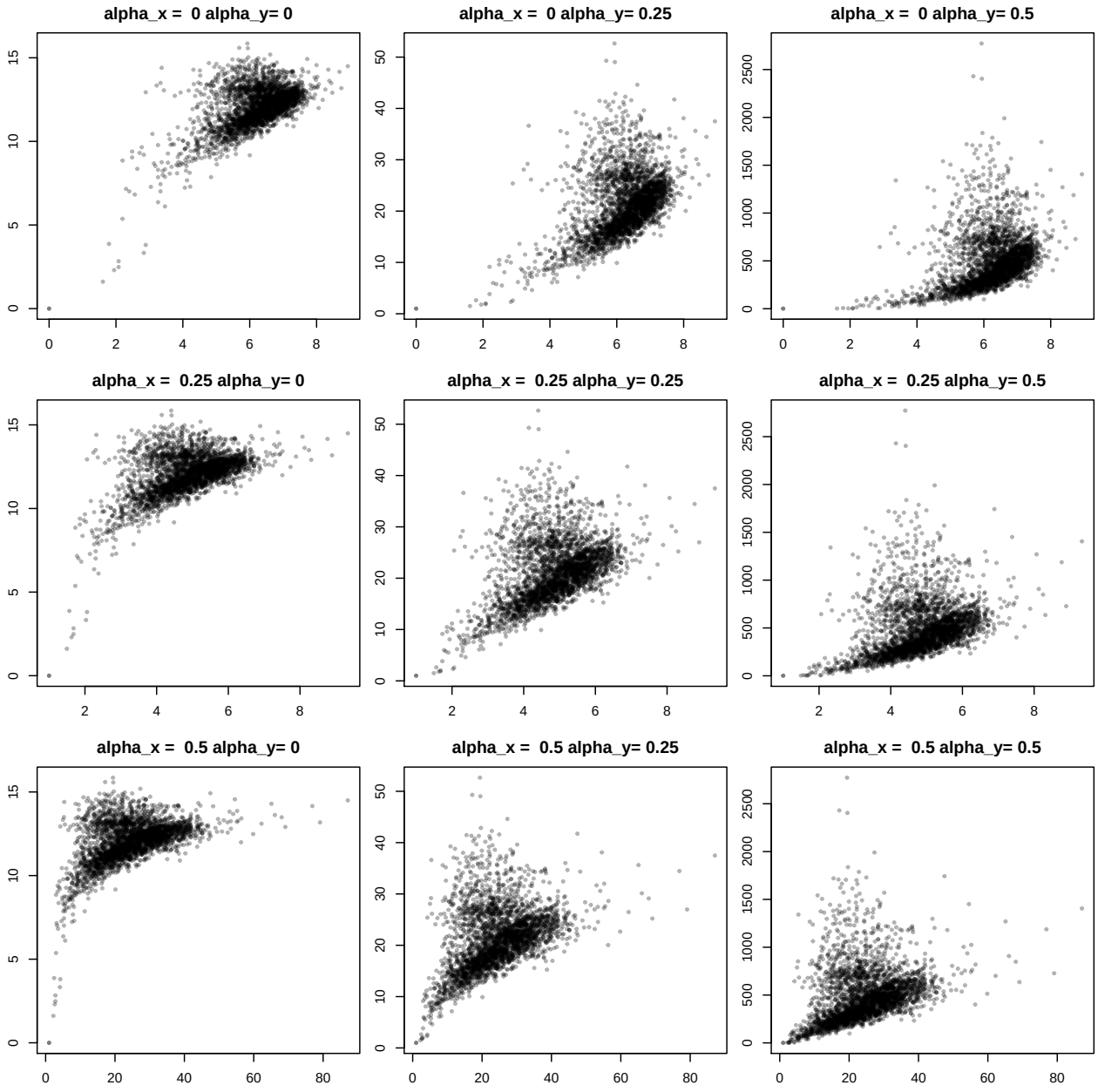
- (x) Number of farms versus (y) Total acreage of farming

```
par(mfrow=c(3,3), mar=2.5*c(1,1,1,0.1))
a = rep(c(-1/2,0,1/2),each=3)
b = rep(c(-1/2,0,1/2),times=3)
subdata = agpop[,c('farms87', 'acres87')]
subdata  = na.omit(subdata )

for (i in 1:9) {
plot( powerfun(subdata$farms87+1, a[i]), powerfun(subdata$acres87+1, b[i]), pch = 19, cex=0.5,
     col=adjustcolor("black", alpha = 0.3), xlab = "", ylab = "",
     main = paste('alpha_x = ', a[i], 'alpha_y=', b[i] ) )
}
```



Bump rule 2: Agriculture Data - C

- (x) Number of farms versus (y) Total acreage of farming



Bump rule 2: Straightening scatterplots

Bump rule 2: Agriculture Data Using loon

```
### This requires that the loon package be installed.
### install.package("loon") will install the package from CRAN
### (requires R >= 3.4)
###
library(loon)
###
power <- function(x, y,
                  linkingGroup="linkingGroup",
```

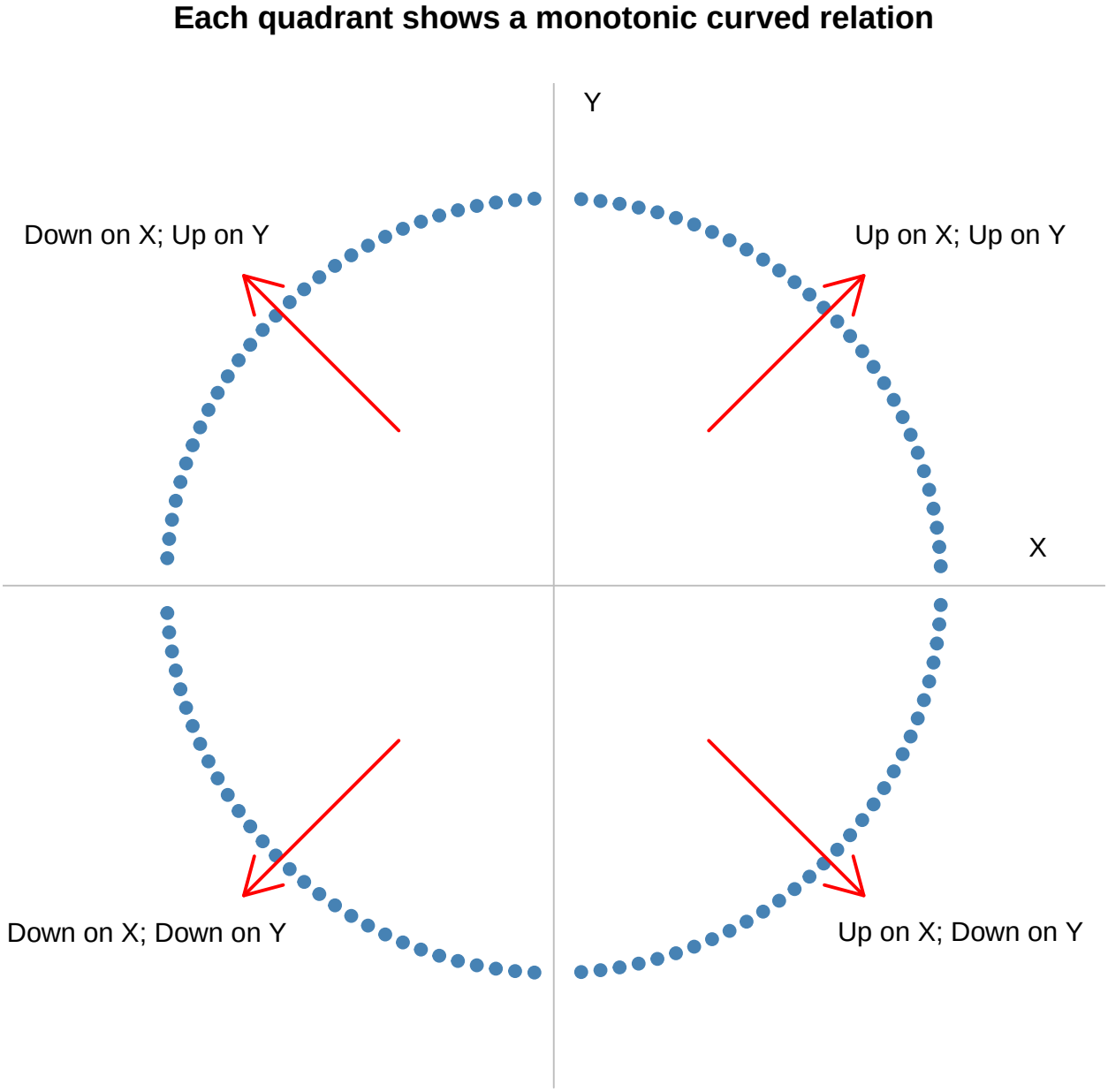


Figure 4: Bump rule for straightening scatterplots

```
                from=-5, to=5, ...) {
## Create histograms
histX <- l_hist(x, linkingGroup = linkingGroup,
                yshows="density")
histY <- l_hist(y, linkingGroup = linkingGroup,
                yshows="density", swapAxes = TRUE
                )
## Now we build an interactive scatterplot
## with sliders for power transformations
## on each of x and y
tt <- tktoplevel()
tktitle(tt) <- "Power Transformation"
p <- l_plot(x=x, y=y, parent=tt,
                linkingGroup=linkingGroup,
            ...)
## Alpha values
alpha_x <- tclVar('1')
alpha_y <- tclVar('1')
## Sliders to change the alphas
sx <- tkscale(tt, orient='horizontal',
              variable=alpha_x,
              from=from, to=to, resolution=0.1)
sy <- tkscale(tt, orient='vertical',
              variable=alpha_y,
              from=to, to=from, resolution=0.1)
## Laying out the pieces in one window
tkgrid(sy, row=0, column=0, sticky="ns")
tkgrid(p, row=0, column=1, sticky="nswe")
tkgrid(sx, row=1, column=1, sticky="we")
tkgrid.columnconfigure(tt, 1, weight=1)
tkgrid.rowconfigure(tt, 0, weight=1)

## This function redraws the plots with the alphas
## from the slider values whenever it is called.
##
update <- function(...) {
  ### get transformed x and y
  transformedX <- powerfun(x, as.numeric(tclvalue(alpha_x)))
  transformedY <- powerfun(y, as.numeric(tclvalue(alpha_y)))

  ## First the scatterplot
  l_configure(p,
              x = transformedX,
              y = transformedY)
  l_scaleto_world(p)
  ## Now the histograms
  l_configure(histX, x = transformedX)
  l_scaleto_world(histX)
  l_configure(histY, x = transformedY)
  l_scaleto_world(histY)
}
## Set the function update to be called
## whenever the slider values are changed
```

```
  tkconfigure(sx, command=update)
  tkconfigure(sy, command=update)
  ## Return the scatterplot if assigned
  invisible(p)
}

###
### Here's an example using the mammals data set
### from the MASS packages
p <- with(agpop,
          power(farms87+1, acres87+1,
                xlabel="# farms" ,
                ylabel="acres",
                title=
                  "",
                linkingGroup = "agpop",
                itemLabel=rownames(agpop),
                showItemLabels=TRUE)
)
```

Bump rule 2: Using loon Mammals

```
### This requires that the loon package be installed.
### install.package("loon") will install the package from CRAN
### (requires R >= 3.4)
###
library(loon)
###
power <- function(x, y,
                  linkingGroup="linkingGroup",
                  from=-5, to=5, ...) {
  ## Create histograms
  histX <- l_hist(x, linkingGroup = linkingGroup,
                  yshows="density")
  histY <- l_hist(y, linkingGroup = linkingGroup,
                  yshows="density", swapAxes = TRUE
                  )
  ## Now we build an interactive scatterplot
  ## with sliders for power transformations
  ## on each of x and y
  tt <- tktoplevel()
  tktitle(tt) <- "Power Transformation"
  p <- l_plot(x=x, y=y, parent=tt,
                  linkingGroup=linkingGroup,
              ...)
  ## Alpha values
  alpha_x <- tclVar('1')
  alpha_y <- tclVar('1')
  ## Sliders to change the alphas
  sx <- tkscale(tt, orient='horizontal',
                variable=alpha_x,
                from=from, to=to, resolution=0.1)
```

```
  sy <- tkscale(tt, orient='vertical',
                variable=alpha_y,
                from=to, to=from, resolution=0.1)
  ## Laying out the pieces in one window
  tkgrid(sy, row=0, column=0, sticky="ns")
  tkgrid(p, row=0, column=1, sticky="nswe")
  tkgrid(sx, row=1, column=1, sticky="we")
  tkgrid.columnconfigure(tt, 1, weight=1)
  tkgrid.rowconfigure(tt, 0, weight=1)

  ## This function redraws the plots with the alphas
  ## from the slider values whenever it is called.
  ##
  update <- function(...) {
    ### get transformed x and y
    transformedX <- powerfun(x, as.numeric(tclvalue(alpha_x)))
    transformedY <- powerfun(y, as.numeric(tclvalue(alpha_y)))

    ## First the scatterplot
    l_configure(p,
                x = transformedX,
                y = transformedY)
    l_scaleto_world(p)
    ## Now the histograms
    l_configure(histX, x = transformedX)
    l_scaleto_world(histX)
    l_configure(histY, x = transformedY)
    l_scaleto_world(histY)
  }
  ## Set the function update to be called
  ## whenever the slider values are changed
  tkconfigure(sx, command=update)
  tkconfigure(sy, command=update)
  ## Return the scatterplot if assigned
  invisible(p)
}

###
### Here's an example using the mammals data set
### from the MASS packages
library(MASS)
data("mammals")
p <- with(mammals,
          power(body, brain,
                xlabel="body weight",
                ylabel="brain weight",
                title=
                  "Brain and Body Weights for 62 Species of Land Mammals",
                linkingGroup = "mammals",
                itemLabel=rownames(mammals),
                showItemLabels=TRUE)
)
```

Order and Rank Statistics

- Population attributes can also be an indexed collection of values.
  - For example, consider the following different attributes
- Recall the order statistic

$$y_{(1)} \leq y_{(2)} \leq \cdots \leq y_{(N)}$$

  which are the ordered values (including ties) of the variate values $y_u \in \mathcal{P}$
- the rank statistic

$$r_1, r_2, \ldots, r_N$$

  - These are the **ranks** of the variate values $y_1, y_2, \ldots, y_N$ from the $y_u \in \mathcal{P}$.
  - For example, if $y_i = y_{(k)}$ then y_i is the kth smallest value and so y_i has rank $r_i = k$.
  - this means that

$$y_{(u)} = y_{r_u} \quad \forall \ \ u \in \mathcal{P}$$

Example

```
y <-  c(3, 1, 4, 22, 12)
### The order statistic
yordered <- sort(y)
yordered
```

```
## [1]  1  3  4 12 22
```

```
# The rank statistic (Note, no ties to worry about)
yrank <- rank(y)
yrank
```

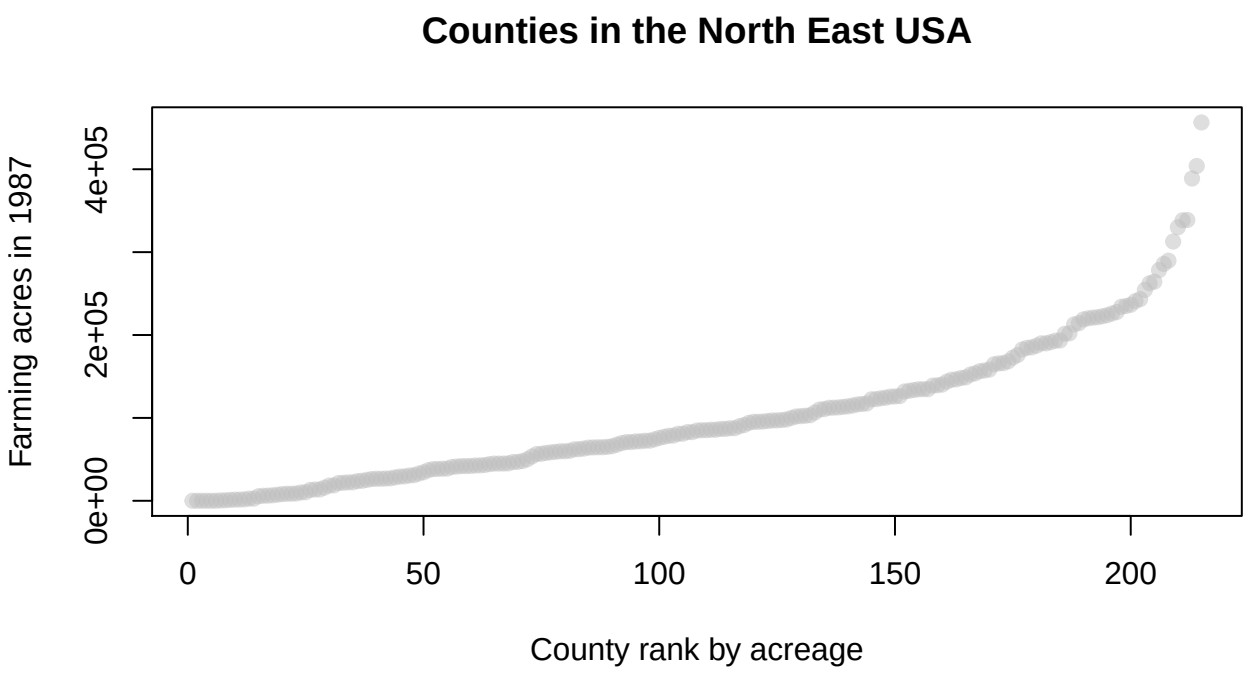
```
## [1] 2 1 3 5 4
```

```
### The connection between them
y[yrank] == yordered
```

```
## [1] TRUE TRUE TRUE TRUE TRUE
```

Graphical Example

- These two attributes are often combined as a single graphical attribute by plotting the pairs (r_u, y_u) (or equivalently $(u, y_{(u)})$ for all $u \in \mathcal{P}$).
  - For example, variate `acres87` from the agricultural census data



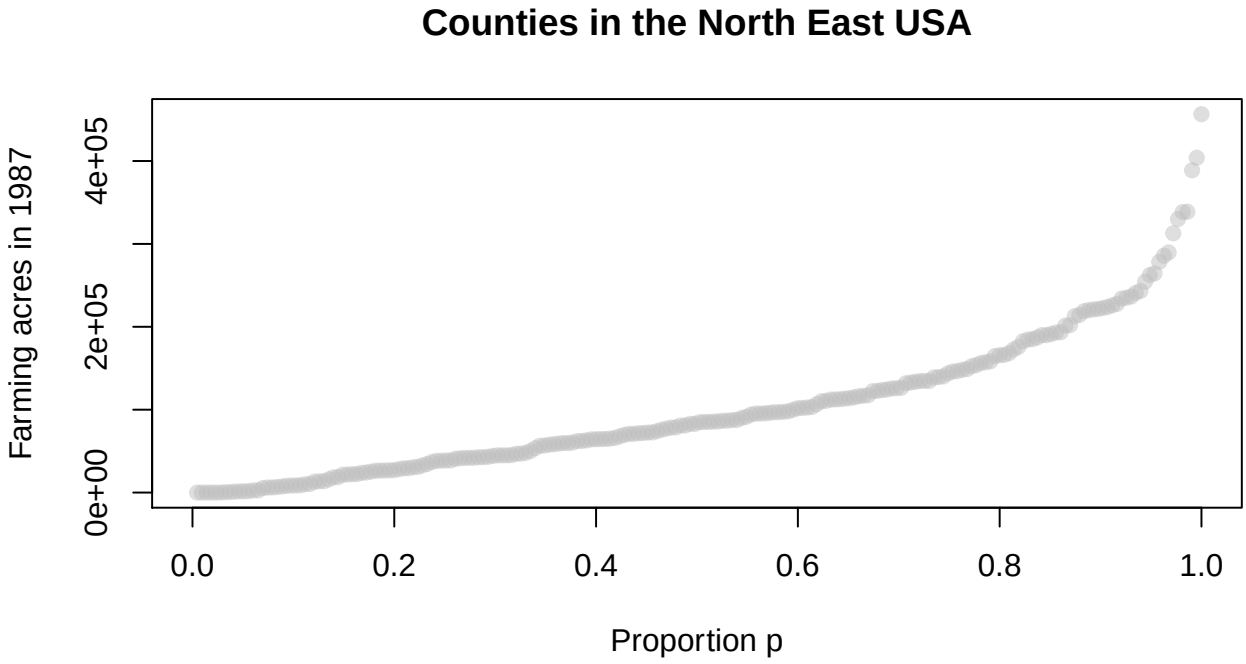
- **Notes**
  - the height at any point tells the location of the value of y
  - horizontal location identifies where in the order of the variate values that unit appears
  - the plot is monotonically non-decreasing from left to right.
  - flat spots indicate tied values of y; nearly flat spots are counties where the number of acres under farming are nearly the same.
  - rapidly rising spots are counties which, though near each other in order (rank), are very different in the actual values of y (acreage).
  - the slope of the curve indicates the spread

Quantiles

- Rather than use rank, it can be more convenient to use the proportion of units in the population having a smaller value of y.
  - That is, instead of plotting the pairs (r_u, y_u), we could equivalently plot the pairs (p_u, y_u) where

$$p_u = \frac{r_u}{N}$$

    is the proportion of the units $i \in \mathcal{P}$ whose value $y_i \leq y_u$



Quantiles - Location

- In this form, the plotted points are denoted

$$(p, Q_y(p))$$

  for $p \in \{\frac{1}{N}, \frac{2}{N}, \ldots, 1\}$ and $Q_y(p)$ is the pth **quantile** of y:

$$Q_y(p) = y_{(N \times p)}.$$

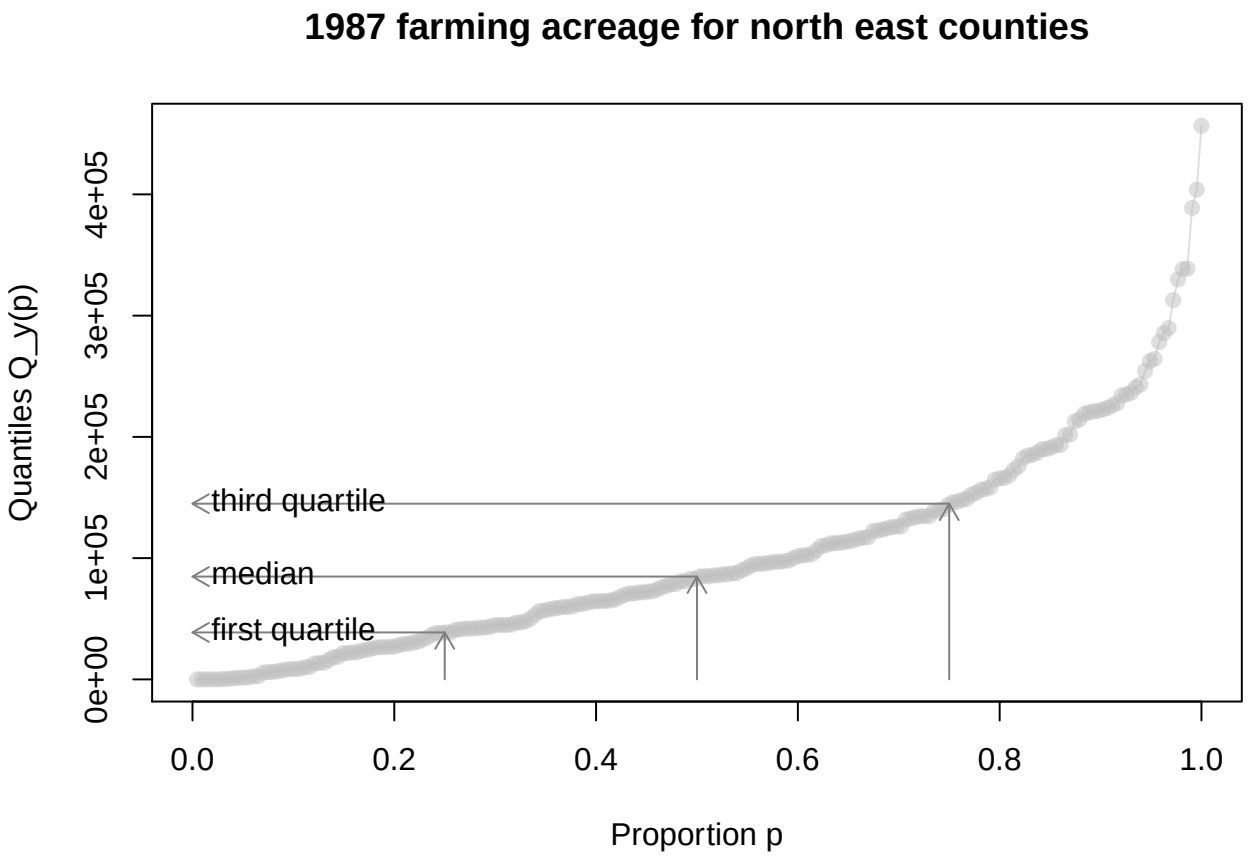
  - the $Q_y(p)$ is the **quantile function** of y for all $p \in [\frac{1}{N}, 1]$.

- The quantile function for y is a population attribute which in turn can be used to generate a number of other interesting population attributes.
  - For example, any quantile $Q_y(p)$ for any p locates the variate values in the population, and is called a **measure of location**.

- Although most location measures try to capture **central** tendency.
  - the **median** $Q_y(1/2)$
  - the **mid-hinge** average of the first and third quartiles $\frac{1}{2}(Q_y(1/4) + Q_y(3/4))$
  - the **mid-range** $\frac{1}{2}(Q_y(1/N) + Q_y(1))$
  - the **trimean** $\frac{1}{4}(Q_y(1/4) + 2 \times Q_y(1/2) + Q_y(3/4))$

- Principally, reading off the vertical location of $Q_y(p)$ for any pre-determined p provides some location of measure.

Quantiles - Spread



- The quantile function can also be used to provide some straightforward and natural measures of of the **scale** or **spread** of the variate values:
  - the **range** $Q_y(1) - Q_y(1/N)$
  - the **inter-quartile range** $IQR_y = Q_y(3/4) - Q_y(1/4)$
  - the **central 100** $\times p$ **percent range**
- Alternatively, any of these measures might be divided by the difference in the corresponding p values.
  - That is, the **slope** of the line segment joining any two points $(p_1, Q_y(p_1))$ and $(p_2, Q_y(p_2))$ for $p_1 < p_2$ provides a measure of **scale**.
- **Exercise**: How do quantiles change after a power transformation? Explain.

Concentration in Quantile Plots

- Flatter regions in a quantile plot indicate areas where the variate values appear to be concentrated.
  - To quantify this we could draw a box with **fixed height** and see how many elmenets are within the box.

```
# Here's an R function that draws a single
# box between the pair of points (x[1],y[1]) and (x[2],y[2])
#
drawbox <- function(x,y, ...) {
```

```
  rect(xleft = x[1], ybottom = y[1], xright = x[2], ytop = y[2], ...)
}
```

```
### Quantiles:
qvals <- sort(y)
pvals <- ppoints(length(qvals))
plot(pvals, qvals, pch = 19, col=adjustcolor("grey", alpha = 0.5),
     xlim=c(0,1),
     xlab = "Proportion p",
     ylab = "Quantiles Q_y(p)",
     main = "1987 farming acreage for north east counties")


# Need some boundaries for the qvals range
qrange <- extendrange(qvals)
bins <- seq(qrange[1], qrange[2], length.out=15)
col <- adjustcolor("steelblue", 0.2)
border <- adjustcolor("black", 0.7)

# Draw one
i <- 1
drawbox(c(min(pvals),
          pvals[sum(qvals <= bins[i+1])]),
        bins[i:(i+1)],
        lty=1,
        lwd=2,
        col= col, border = border)

# Now the rest
for (i in c(3,7,12) ) {
  biny <- c(sum(qvals <= bins[i]),
            sum(qvals <= bins[i+1]))
  drawbox(pvals[biny],
          bins[c(i, i+1)],
          lty=1,
          lwd=2,
          col= col, border = border)
}
```

- The width of the box is proportional to the number of elements.
  - The greater the width, the greater the concentration.

Quantile Plot and Concentration

- We can produce all such boxes, with **fixed height**, to see how the concentration changes with p.

```
plot(pvals, qvals, pch = 19, col=adjustcolor("grey", alpha = 0.5),
     xlim=c(0,1),
     xlab = "Proportion p",
     ylab = "Quantiles Q_y(p)",
     main = "1987 farming acreage for north east counties")
```

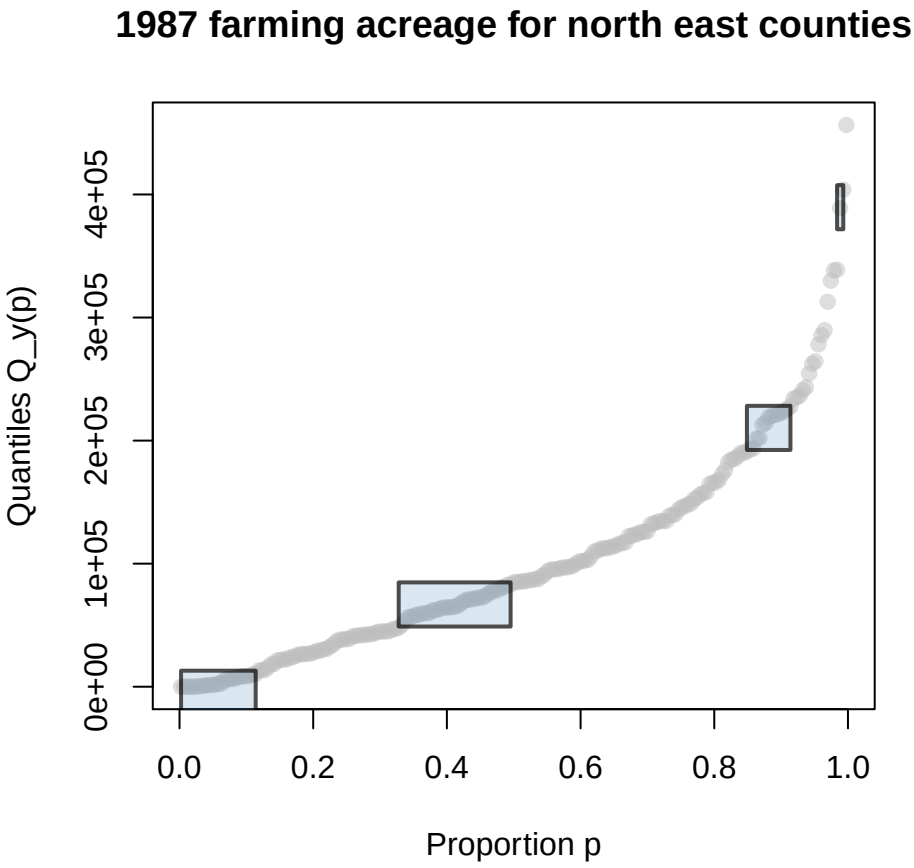


Figure 5: Concentration box on quantile plot

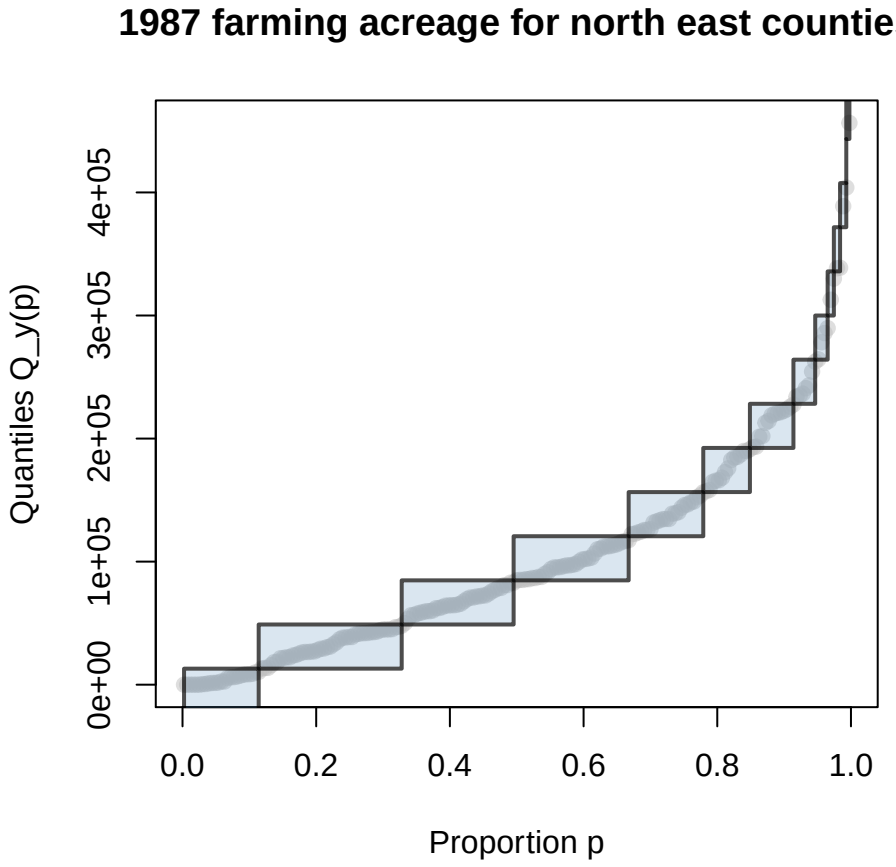


Figure 6: Contiguous concentration boxes on quantile plot

```
# Draw first one
i <- 1
drawbox(c(min(pvals),
          pvals[sum(qvals <= bins[i+1])]),
        bins[i:(i+1)],
        lty=1,
        lwd=2,
        col= col, border = border)

# Now the rest
for (i in 2:length(bins)) {
  biny <- c(sum(qvals <= bins[i]),
            sum(qvals <= bins[i+1]))
  drawbox(pvals[biny],
          bins[c(i, i+1)],
          lty=1,
          lwd=2,
          col= col, border = border)
}
```

- What do the width of the boxes indicate?
  - Now, if all of the boxes are moved to the left edge of the plot a familiar graphic appears.

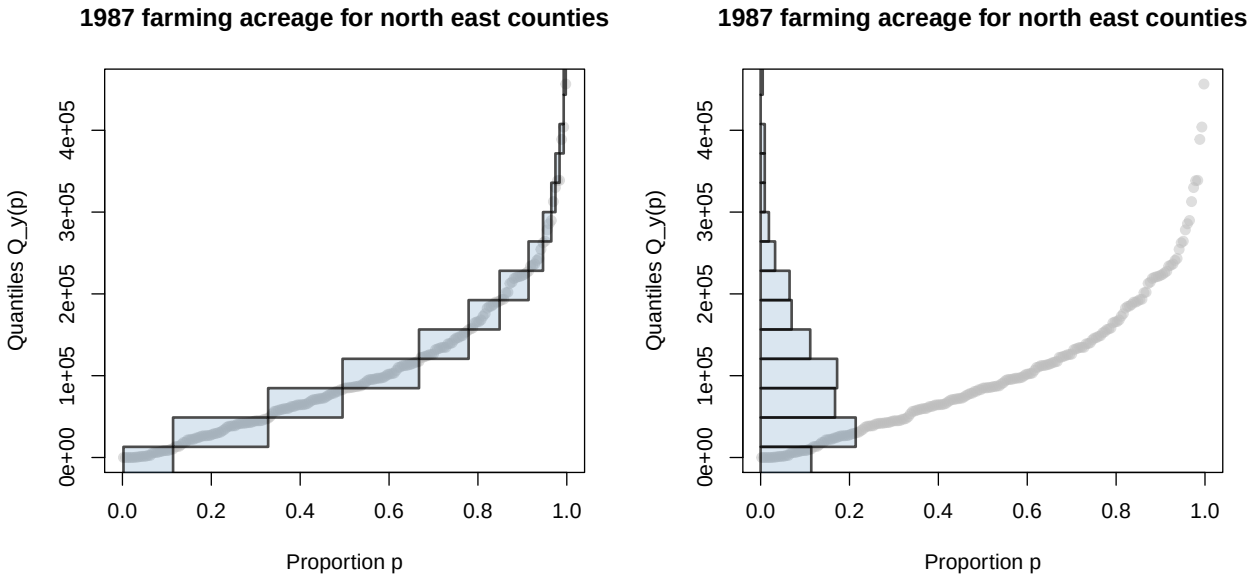


Figure 7: Contiguous concentration boxes on quantile plot

Quantile Plot and Histograms

- A histogram of the acreage (or any y variate) is formed from the boxes that identify concentrations on the quantile plot.