Thresholds and ordinal data

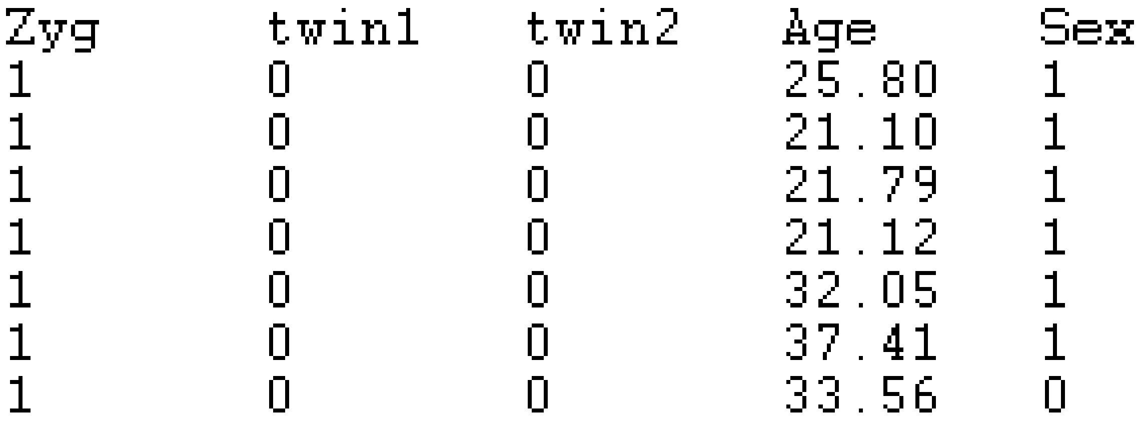
Sarah Medland – Boulder 2010

A plug for OpenMx?

- Very few packages can handle ordinal data adequately...
- OpenMx can also be used for more than just genetic analyses
  - Regression
  - Polycohoric correlations
  - Factor analysis…

Binary data

- File: two_cat.dat
  - Zyg twin1 twin2 Age Sex
- Trait – lifetime cannabis use
  - Never Smoked/Ever Smoked



Zyg	twin1	twin2	Age	Sex
1	0	0	25.80	1
1	0	0	21.10	1
1	0	0	21.79	1
1	0	0	21.12	1
1	0	0	32.05	1
1	0	0	37.41	1
1	0	0	33.56	0

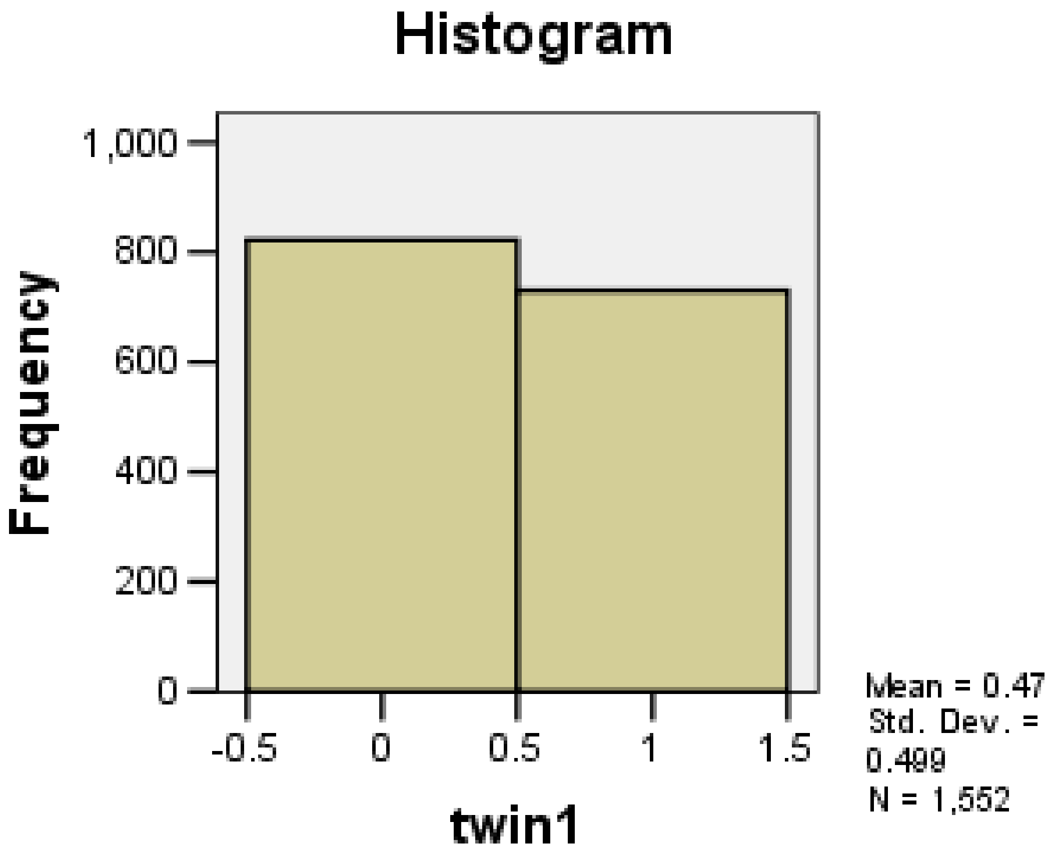
Twin 1 cannabis use

- 0 = never used

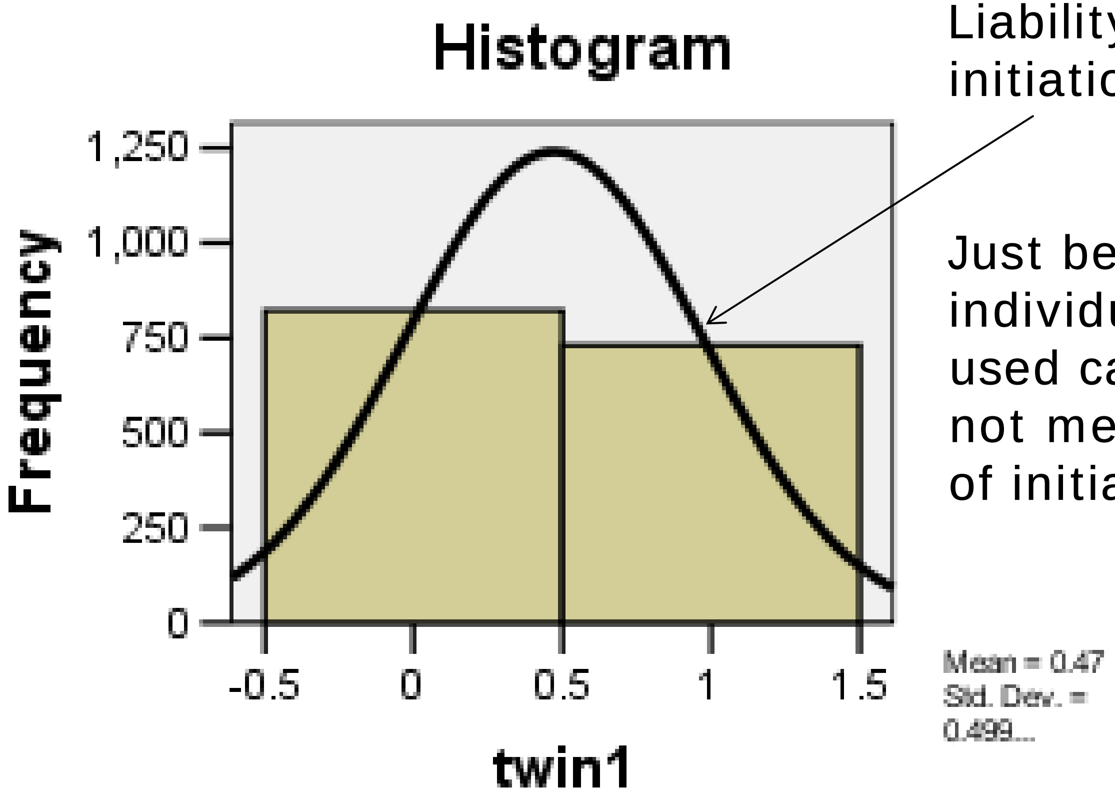
twin1

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	0	822	47.5	53.0	53.0
	1	730	42.2	47.0	100.0
	Total	1552	89.7	100.0	
Missing	System	179	10.3		
Total		1731	100.0		

Twin 1 cannabis use

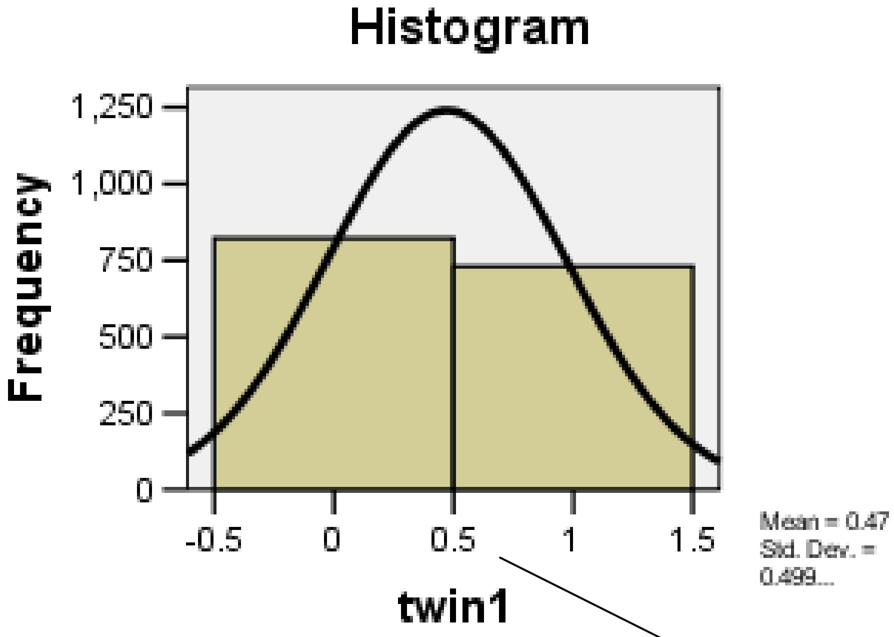


Twin 1 cannabis use



Liability or 'risk' of initiation distribution

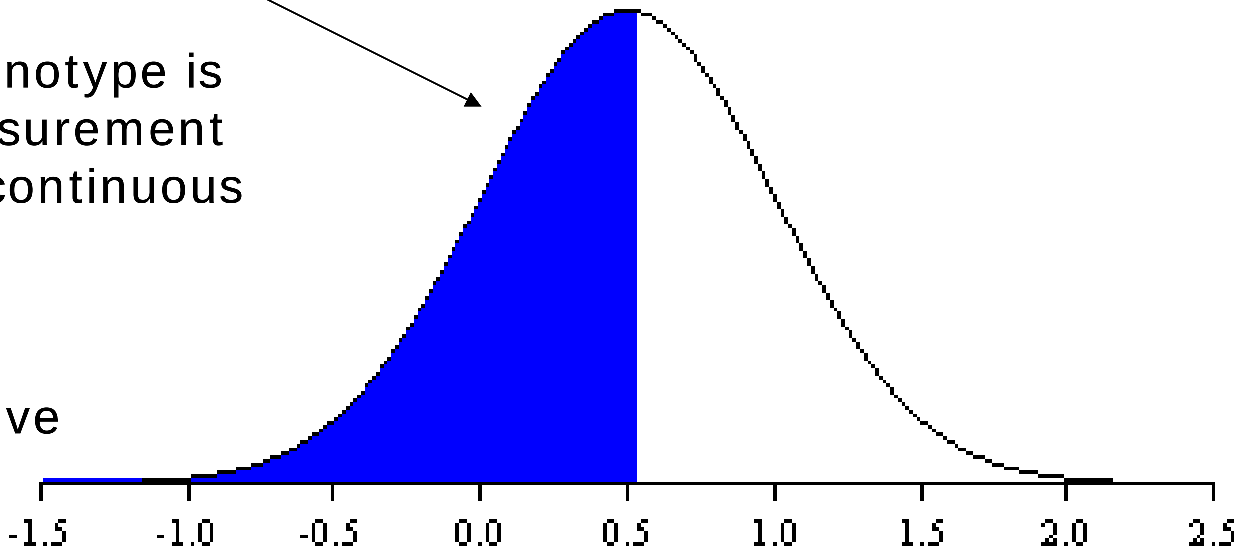
Just because an individual has never used cannabis does not mean their 'risk' of initiation is zero

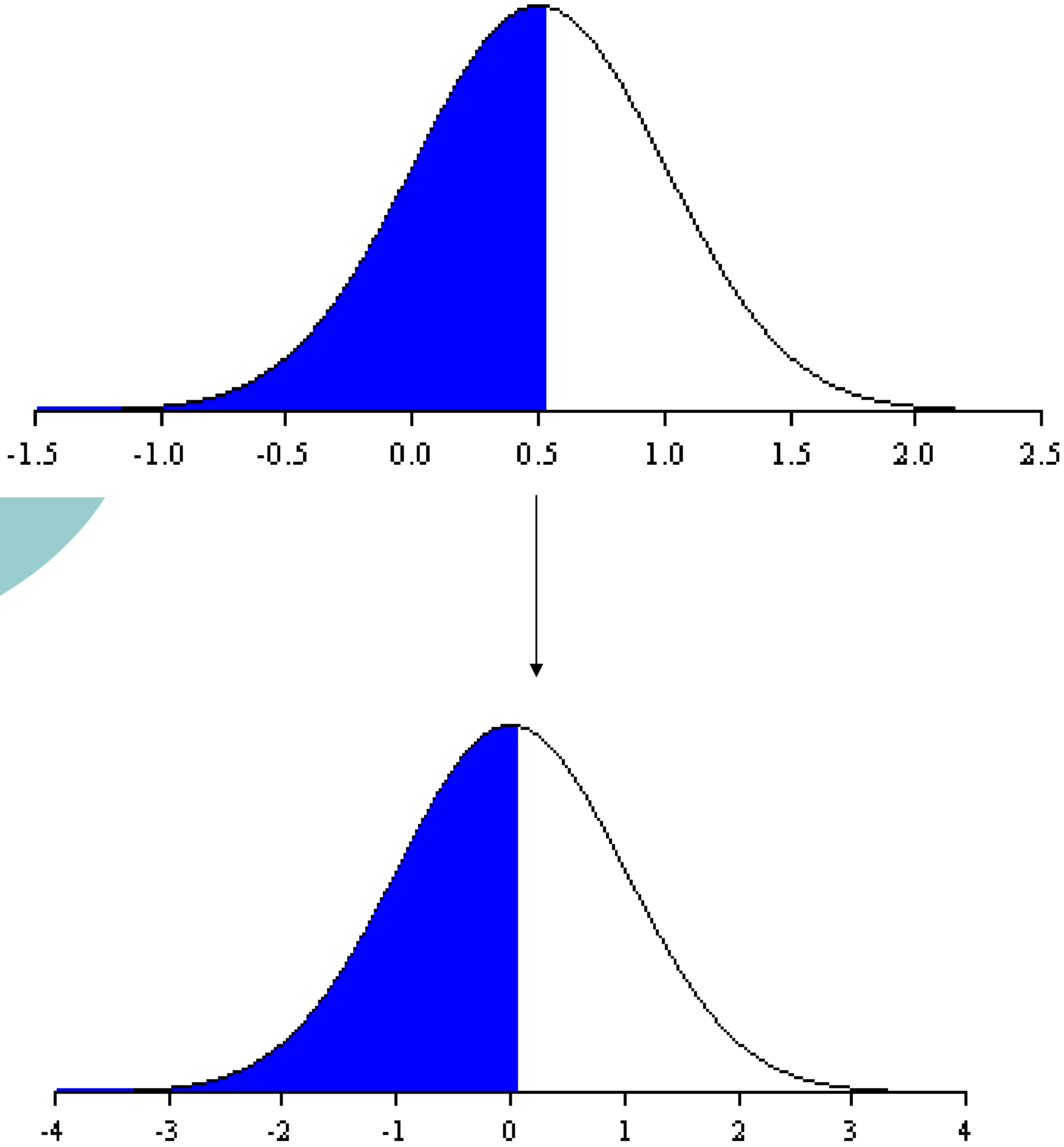


Mean	= .47
SD	=.499
Non Smokers	=53%

The observed phenotype is an *imperfect* measurement of an underlying continuous distribution

ie Obesity vs BMI
MDD vs quantitative depression scales



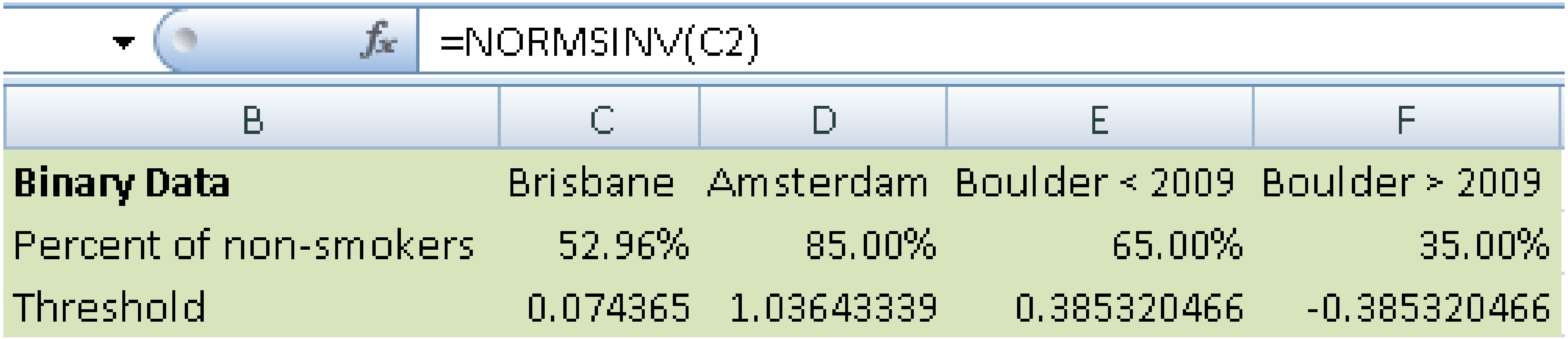


Raw data distribution	
Mean	= .47
SD	=.499
Non Smokers	=53%
Threshold	=.53

Standard normal distribution	
Mean	= 0
SD	=1
Non Smokers	=53%
Threshold	=.074

Threshold = .074 – Huh what?

- How can I work this out
  - Excell
    - =NORMSINV()
    - Thresholds.xls



=NORMSINV(C2)

B	C	D	E	F
Binary Data	Brisbane	Amsterdam	Boulder < 2009	Boulder > 2009
Percent of non-smokers	52.96%	85.00%	65.00%	35.00%
Threshold	0.074365	1.03643339	0.385320466	-0.385320466

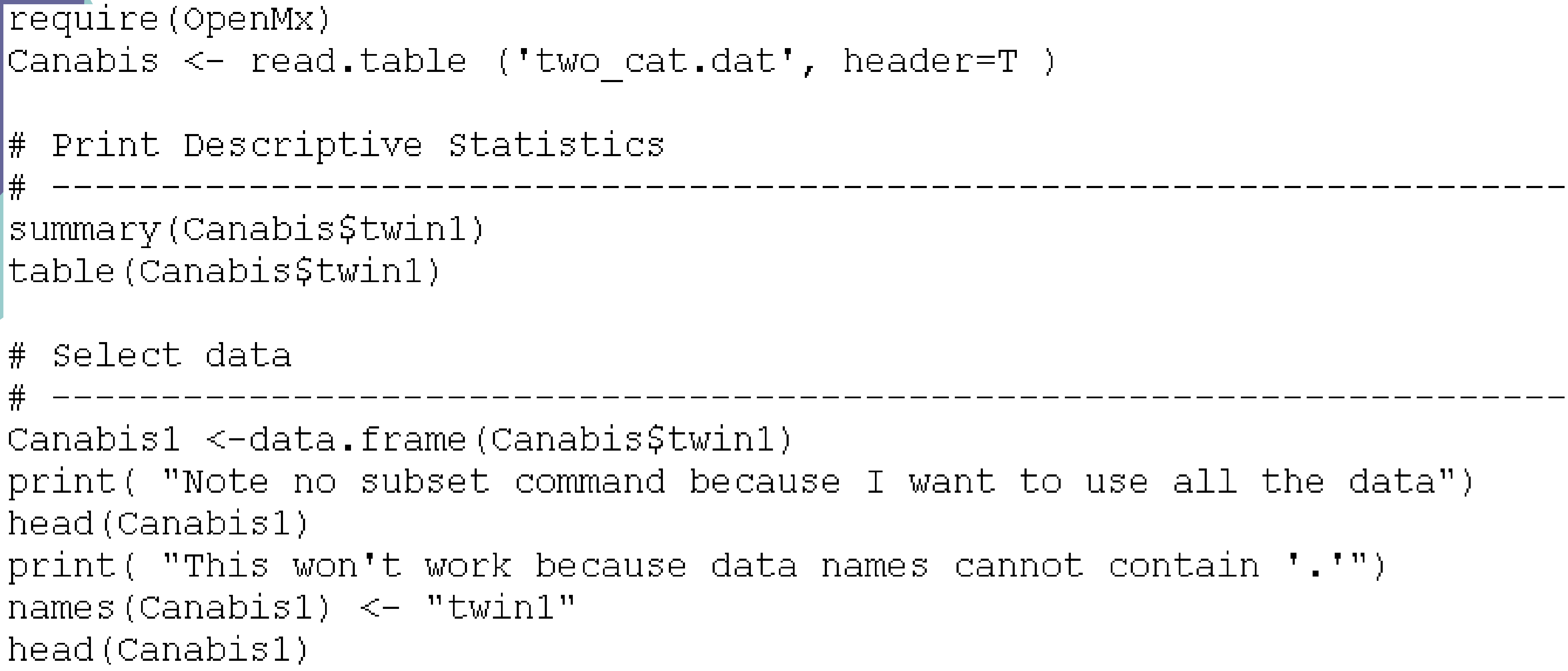
Why do we rescale the data this way?

- Convenience
  - Variance always 1
  - Mean is always 0
  - We can interpret the area under a curve between two z-values as a probability or percentage

Why do we rescale the data this way?

You could use other distributions but **you** would have to specify the fit function

Threshold.R



```
require(OpenMx)
Canabis <- read.table ('two_cat.dat', header=T )

# Print Descriptive Statistics
# ----------------------------------------------------------------------
summary(Canabis$twin1)
table(Canabis$twin1)

# Select data
# ----------------------------------------------------------------------
Canabis1 <-data.frame(Canabis$twin1)
print( "Note no subset command because I want to use all the data")
head(Canabis1)
print( "This won't work because data names cannot contain '.'")
names(Canabis1) <- "twin1"
head(Canabis1)
```

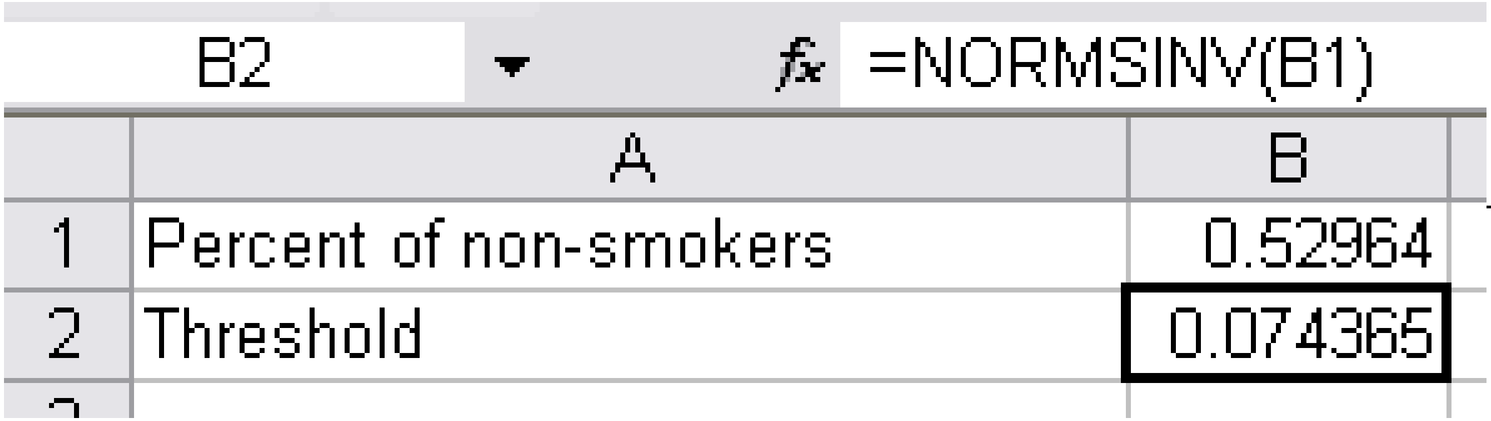
Threshold.R

```
# Specify and Run Saturated Model (Tetrachoric correlations) with RawData
# -----------------------------------------------------------------------
nvar <- 1
nthresh <- 1
Vars <-('twin1')
selVars <- ('twin1')

checkThresholdModel<- mxModel("checkThreshold",
mxModel("BrisbaneData",
  # Matrix & Algebra for expected means vector (SND), Thresholds and correlation
        mxMatrix(type="Zero", nrow=nvar, ncol=nvar, name="expMean" ),
        mxMatrix(type="Full", nrow=nvar, ncol=nthresh, free=TRUE, values=0,
                 name="expThresh", label="threshold1",dimnames=list('th1',selVars) ),
        mxMatrix(type="Stand", nrow=nvar, ncol=nvar, name="expCor"),
        mxData(Canabis1, type="raw"),

        mxFIMLObjective( covariance="expCor", means="expMean", dimnames=selVars,
                 thresholds="expThresh" )),
        mxAlgebra(BrisbaneData.objective, name="-2LL" ),
        mxAlgebraObjective("-2LL")

 )
checkThresholdFit <- mxRun(checkThresholdModel)
checkThresholdSumm <- summary(checkThresholdFit)
checkThresholdSumm
```



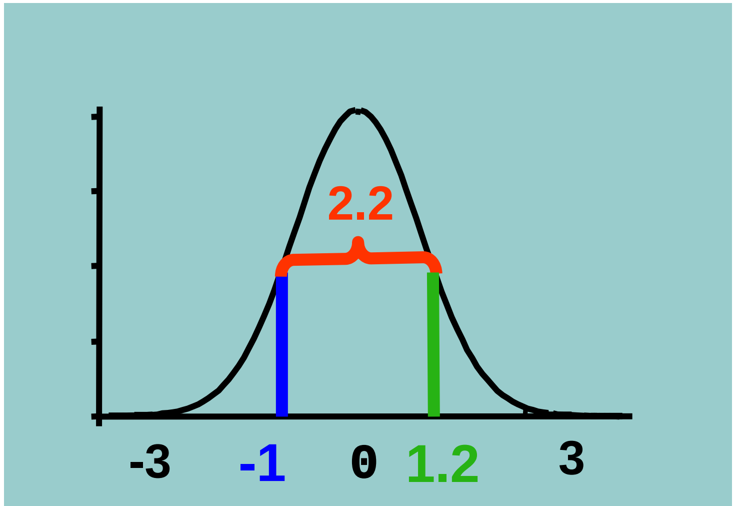
B2 =NORMSINV(B1)

	A	B
1	Percent of non-smokers	0.52964
2	Threshold	0.074365

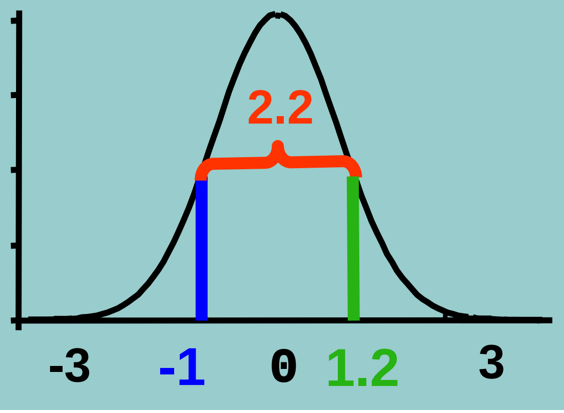
```
> # Generate Model Output
> # -----------------------------------------------------------------------
> ExpThresholds <- mxEval(BrisbaneData.expThresh, checkThresholdFit )
> ExpThresholds
         twin1
th1 0.07436543
> ExpTetraCor <-mxEval(BrisbaneData.expCor, checkThresholdFit )
> ExpTetraCor
      twin1
twin1     1
> minusTwiceLL <- mxEval(objective, checkThresholdFit )
> minusTwiceLL
         [,1]
[1,] 2146.072
```

What about more than 2 categories?

- Very similar
  - We create a matrix containing the 1st threshold and the displacements between subsequent matrices
  - We then add the 1st threshold and the displacement to obtain the subsequent thresholds



Mx Threshold Specification: 3+ Cat.

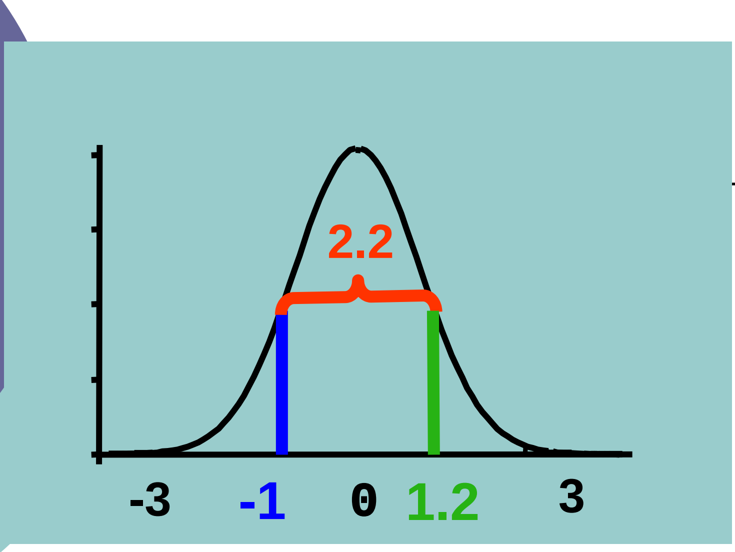


$$\mathbf{T} = \begin{array}{cc} \text{Twin 1} & \text{Twin 2} \\ \left[\begin{matrix} t_{11} \\ t_{21} \end{matrix}\right. & \left.\begin{matrix} t_{12} \\ t_{22} \end{matrix}\right] \end{array}$$

1st threshold

increment

Mx Threshold Specification: 3+ Cat.



$$\mathbf{T} = \begin{array}{cc} \text{Twin 1} & \text{Twin 2} \\ \left[\begin{array}{cc} t_{11} & t_{12} \\ t_{21} & t_{22} \end{array}\right] \end{array}$$

1st threshold

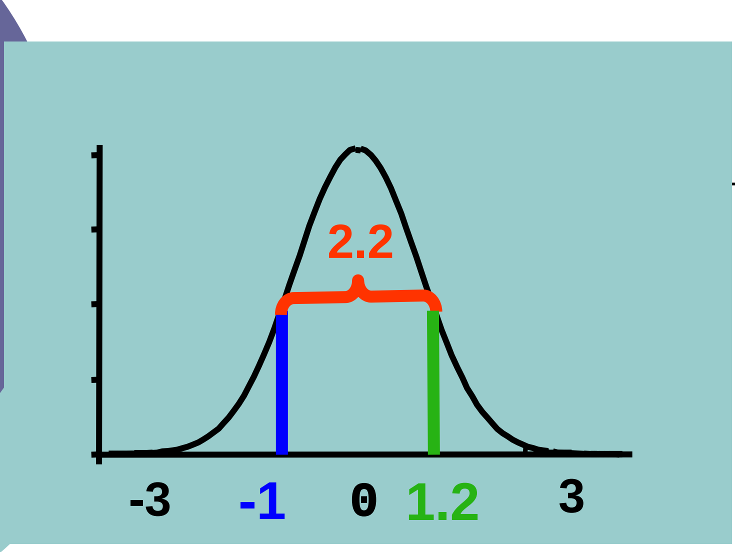
increment

Mx Threshold Model: `Thresholds L*T /`

$$\mathbf{L} * \mathbf{T} = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} * \begin{bmatrix} t_{11} & t_{12} \\ t_{21} & t_{22} \end{bmatrix}$$

$$= \begin{bmatrix} t_{11} & t_{12} \\ t_{11} + t_{21} & t_{12} + t_{22} \end{bmatrix}$$

Mx Threshold Specification: 3+ Cat.



$$\mathbf{T} = \begin{array}{cc} \text{Twin 1} & \text{Twin 2} \\ \hline \left[\begin{matrix} t_{11} \\ t_{21} \end{matrix}\right. & \left.\begin{matrix} t_{12} \\ t_{22} \end{matrix}\right] \end{array}$$

1st threshold

increment

Mx Threshold Model: `Thresholds L*T /`

$$\mathbf{L} * \mathbf{T} = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} * \begin{bmatrix} t_{11} & t_{12} \\ t_{21} & t_{22} \end{bmatrix}$$

$$= \begin{bmatrix} t_{11} & t_{12} \\ t_{11} + t_{21} & t_{12} + t_{22} \end{bmatrix}$$

2nd threshold

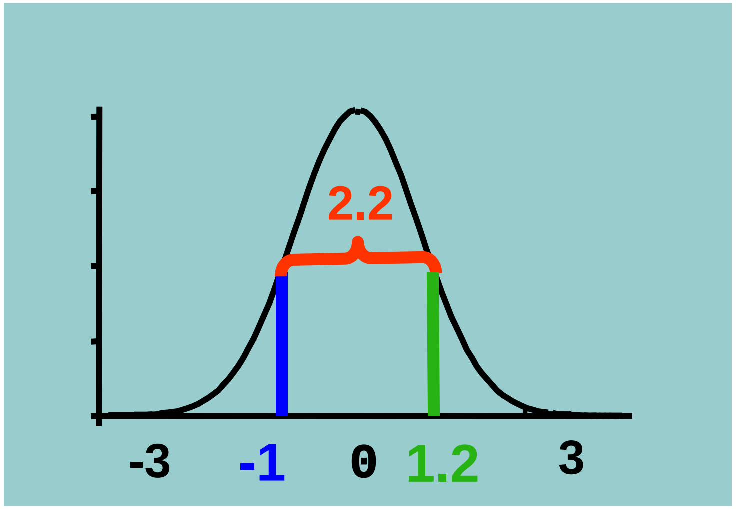
OpenMx code

$$\begin{bmatrix} t_{11} & t_{12} \\ t_{21} & t_{22} \end{bmatrix} = \begin{bmatrix} \text{zscore1 twin1} & \text{zscore1 twin2} \\ \text{zscore2 twin1 - zscore1 twin1} & \text{zscore2 twin2 - zscore1 twin2} \end{bmatrix}$$

```
mxMatrix( type="Full", nrow=2, ncol=2, free=TRUE,
values=.5, lbound= c(-3,0.0001, -3, 0.0001), name="T" ),
```

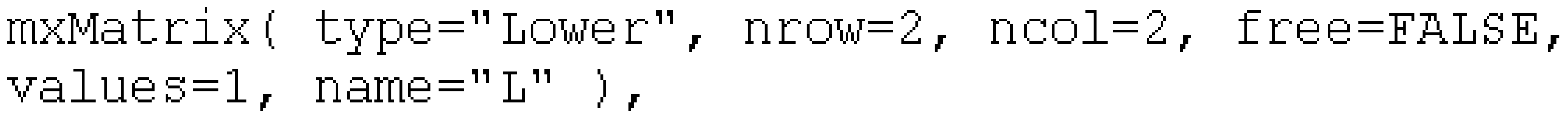
The bounds stop the thresholds going ‘backwards’

ie they preserve the ordering of the data



- Lower bounds
- -3 -3
- .0001 .0001

OpenMx code



```
mxMatrix( type="Lower", nrow=2, ncol=2, free=FALSE,
values=1, name="L" ),
```

$$\begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$$

OpenMx code

```
mxAlgebra( expression= L %*% T, dimnames=list(thRows,selVars)
, name="expThmz"),
```

$$\mathbf{L} * \mathbf{T} = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} * \begin{bmatrix} t_{11} & t_{12} \\ t_{21} & t_{22} \end{bmatrix}$$

$$= \begin{bmatrix} t_{11} & t_{12} \\ t_{11} + t_{21} & t_{12} + t_{22} \end{bmatrix}$$

2nd threshold

OpenMx code

```
mxMatrix( type="Full", nrow=2, ncol=2, free=TRUE,
values=.5, lbound= c(-3,0.0001, -3, 0.0001), name="T" ),

mxMatrix( type="Lower", nrow=2, ncol=2, free=FALSE,
values=1, name="L" ),

mxAlgebra( expression= L %*% T, dimnames=list(thRows,selVars)
, name="expThmz"),
```

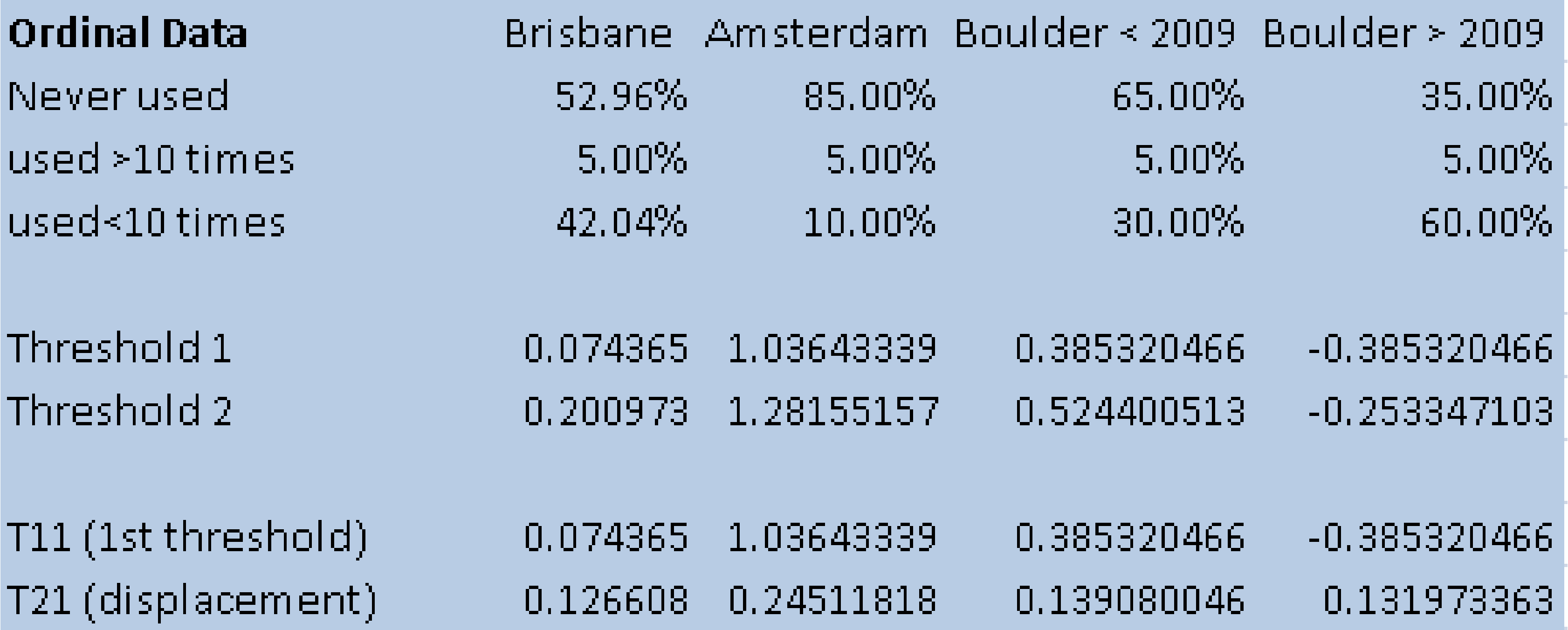
$$\mathbf{L} * \mathbf{T} = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} * \begin{bmatrix} t_{11} & t_{12} \\ t_{21} & t_{22} \end{bmatrix}$$

$$= \begin{bmatrix} t_{11} & t_{12} \\ t_{11} + t_{21} & t_{12} + t_{22} \end{bmatrix}$$

2nd threshold

Thresholds.xls

- Try changing the values



Ordinal Data	Brisbane	Amsterdam	Boulder < 2009	Boulder > 2009
Never used	52.96%	85.00%	65.00%	35.00%
used >10 times	5.00%	5.00%	5.00%	5.00%
used<10 times	42.04%	10.00%	30.00%	60.00%
Threshold 1	0.074365	1.03643339	0.385320466	-0.385320466
Threshold 2	0.200973	1.28155157	0.524400513	-0.253347103
T11 (1st threshold)	0.074365	1.03643339	0.385320466	-0.385320466
T21 (displacement)	0.126608	0.24511818	0.139080046	0.131973363