SOEE1475 Statistics and Data Analysis
Lecture 6: Temporal statistics

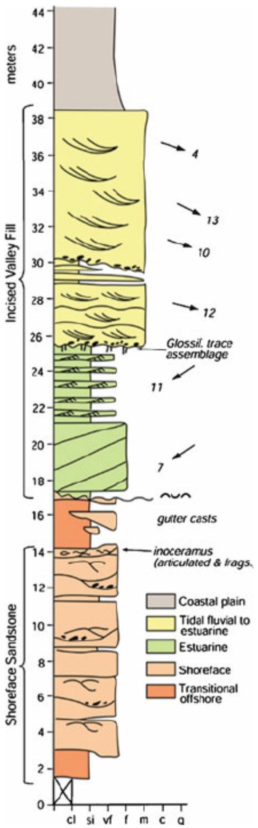
Graeme T. Lloyd

Today

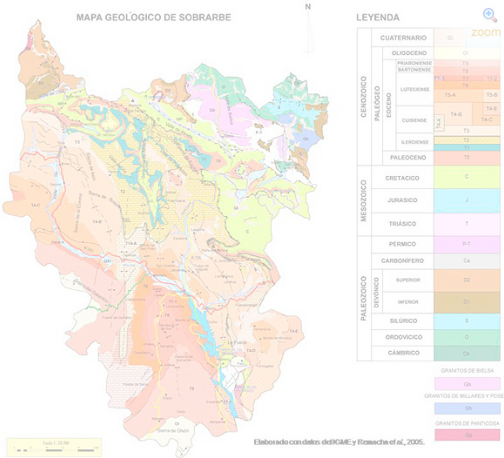
- Temporal patterns
- Kolmogorov-Smirnov test
- Markov chains

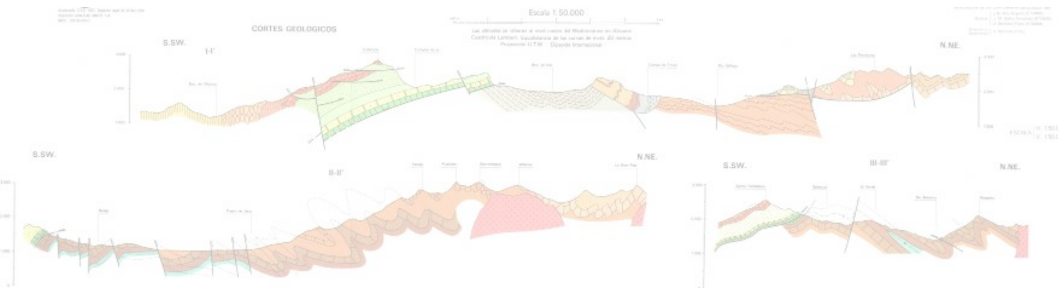
Spatial data in geology

1D

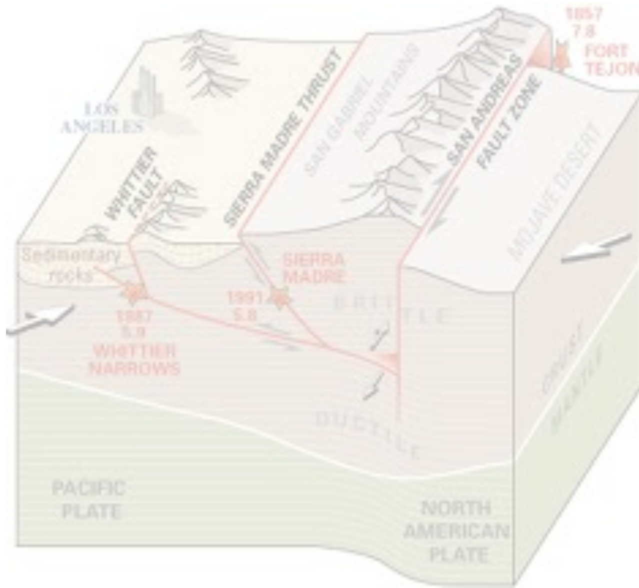


2D

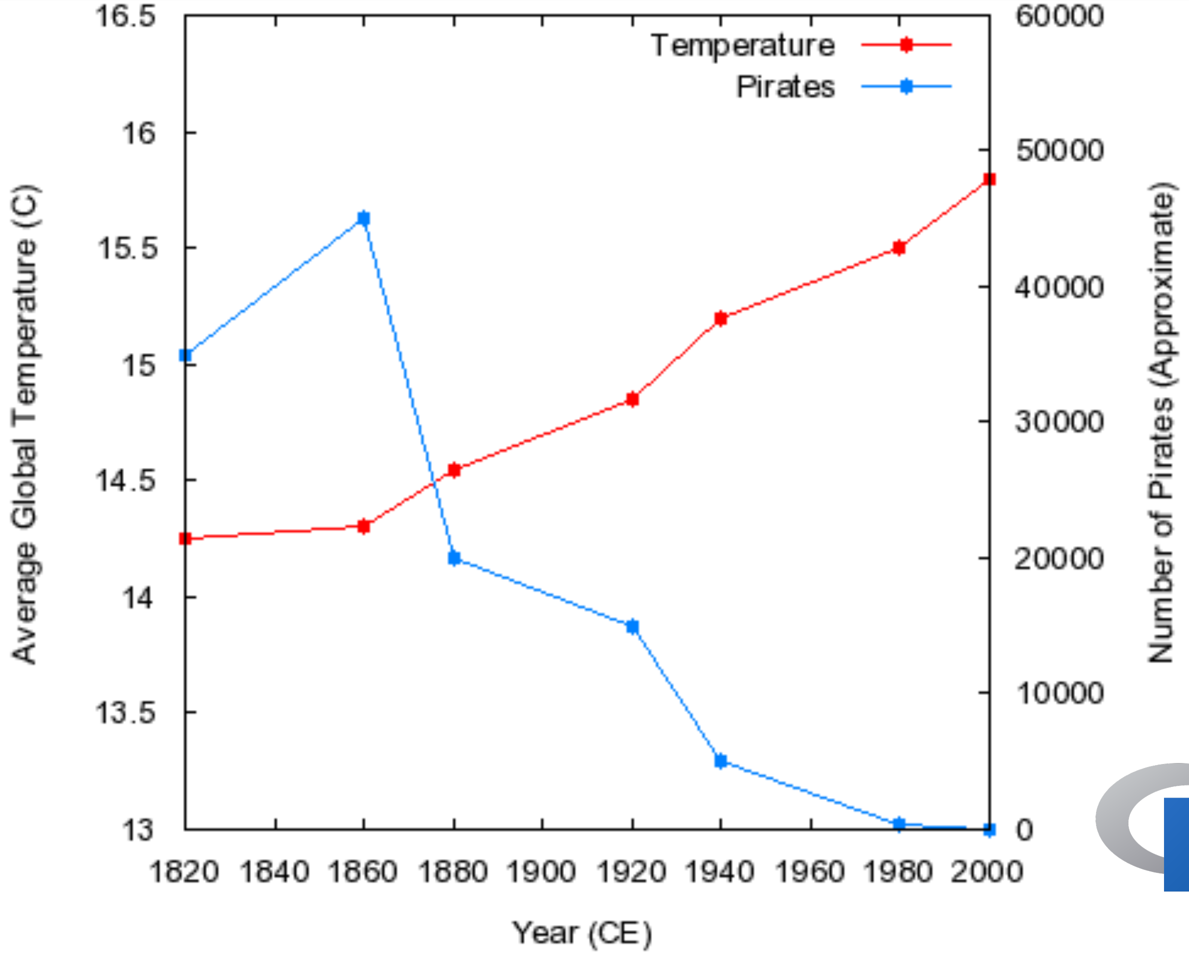




3D

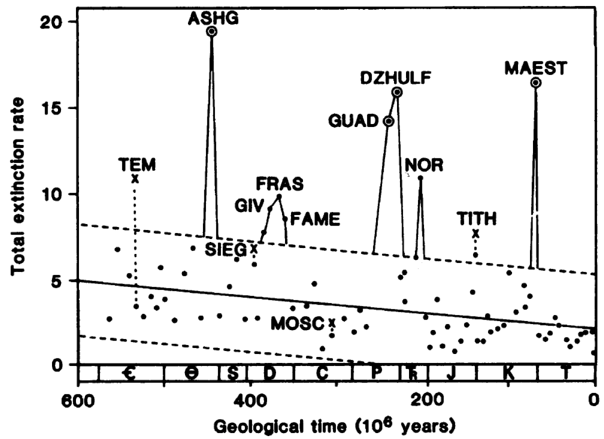


Temporal autocorrelation

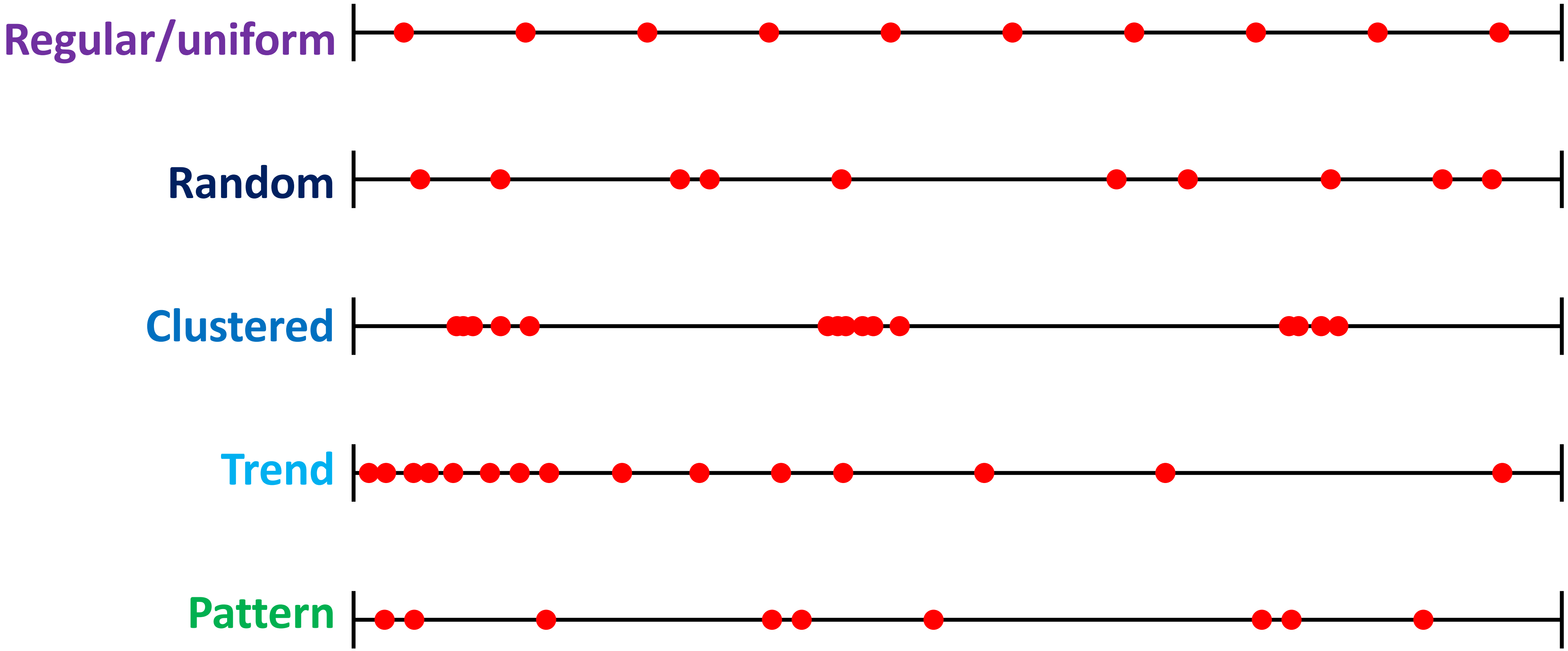



```
cor.test(x = pirates,
  y = temp)$estimate
     -0.9242241
cor.test(x = pirates,
  y = temp)$p.value
     0.00291353
```

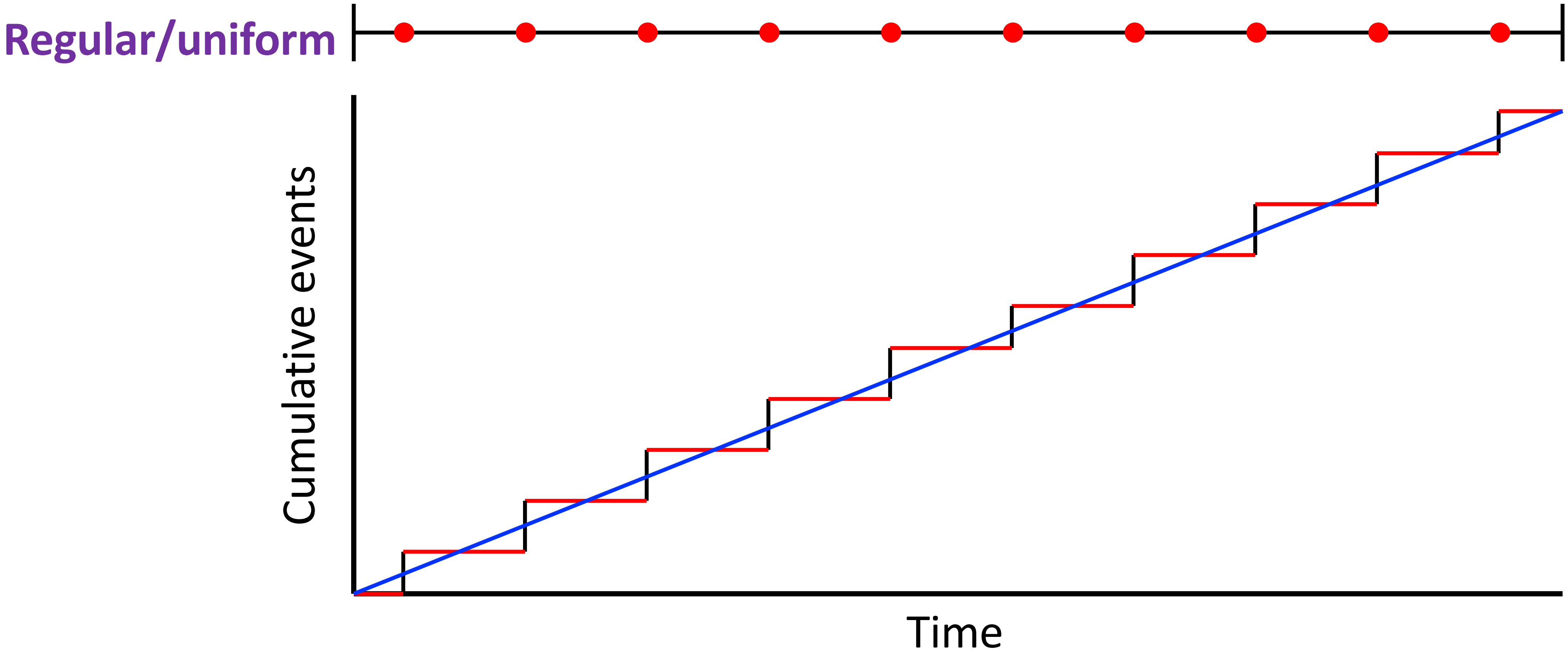
Series of events



Series of events



Testing for regular/uniform



Testing for regular/uniform

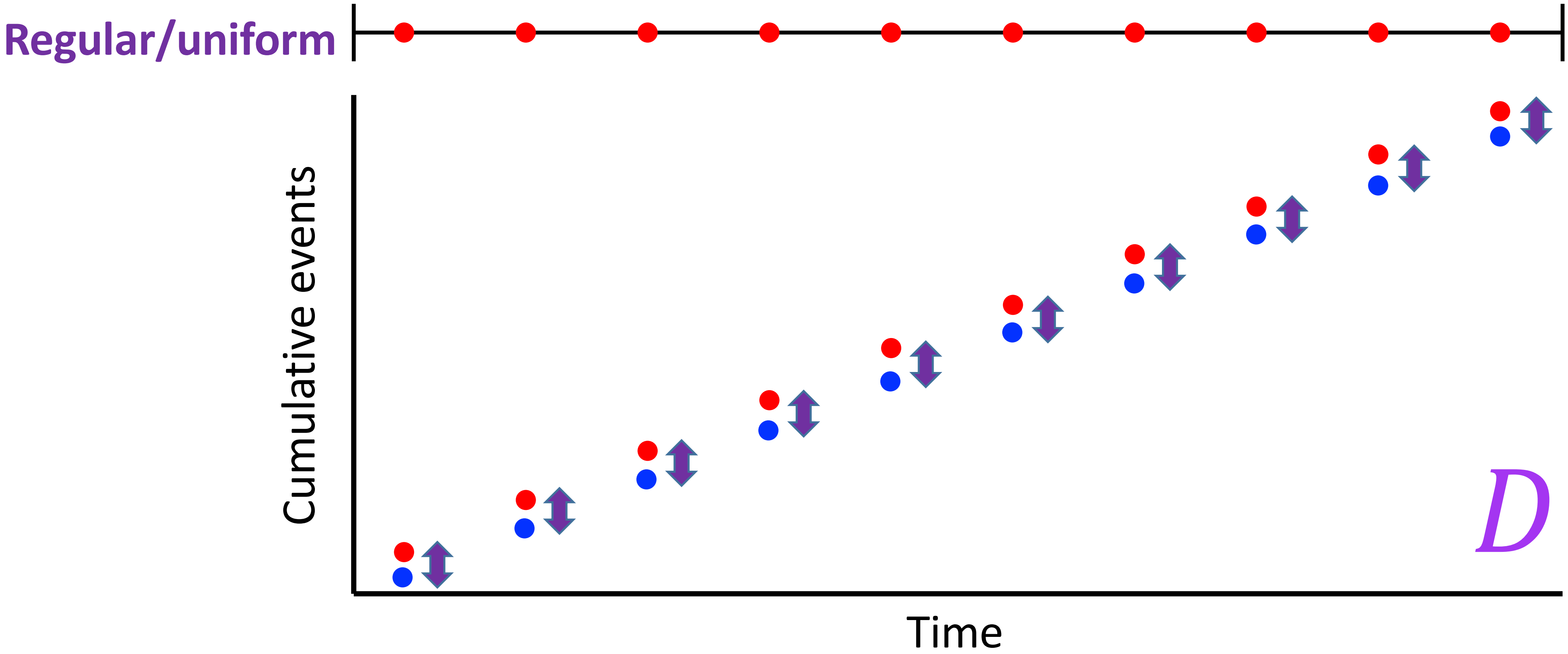
Continuous uniform

Probability density function

Continuous uniform

Cumulative distribution function

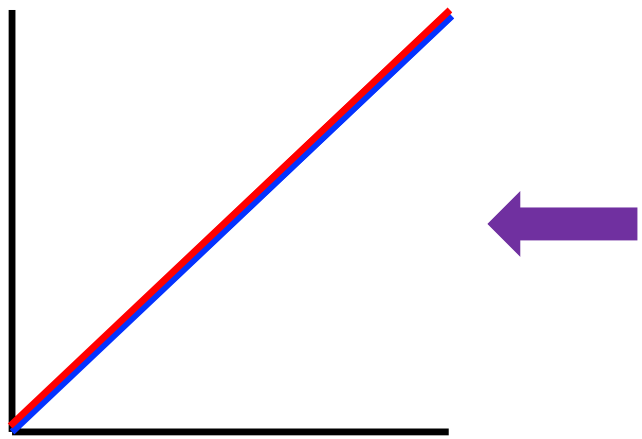
Kolmogorov-Smirnov test



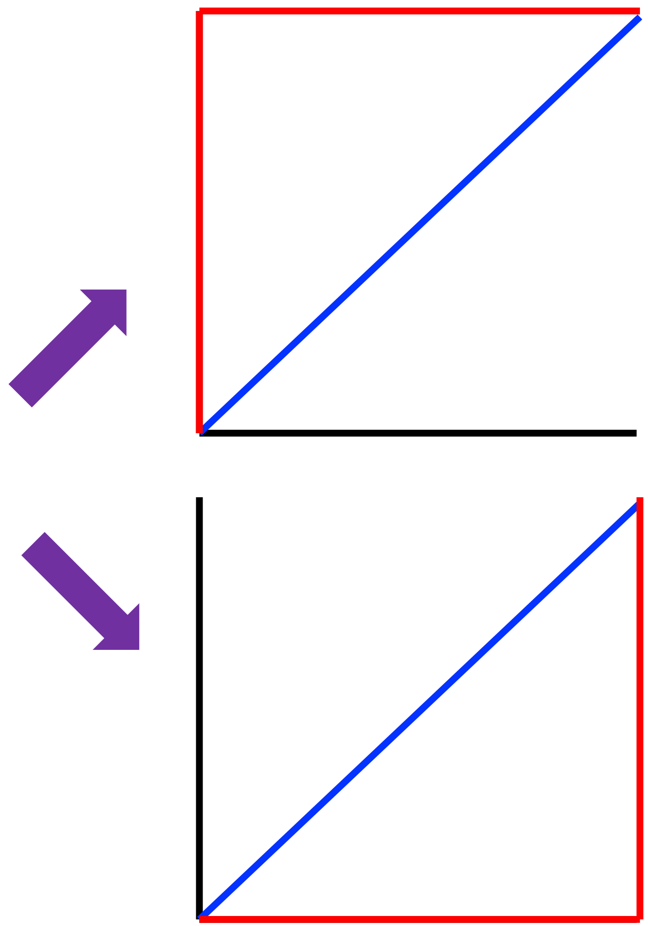
Kolmogorov-Smirnov test

$$0 \leq D \leq 1$$

Kolmogorov-Smirnov test



$$0 \leq D \leq 1$$



Kolmogorov-Smirnov test

```
x <- runif(20)
ks.test(x = x, y = "punif")$statistic
ks.test(x = x, y = "punif")$p.value
```

Kolmogorov-Smirnov test

Assesses "goodness of fit" between two distributions

Like Chi-squared, but for continuous rather than categorical data

Allows us to choose expected value (null model)

Also like Chi-squared; flexible

One way to test if data approximate a specific distribution type

E.g., are data uniform? Are data normal?

Kolmogorov-Smirnov test

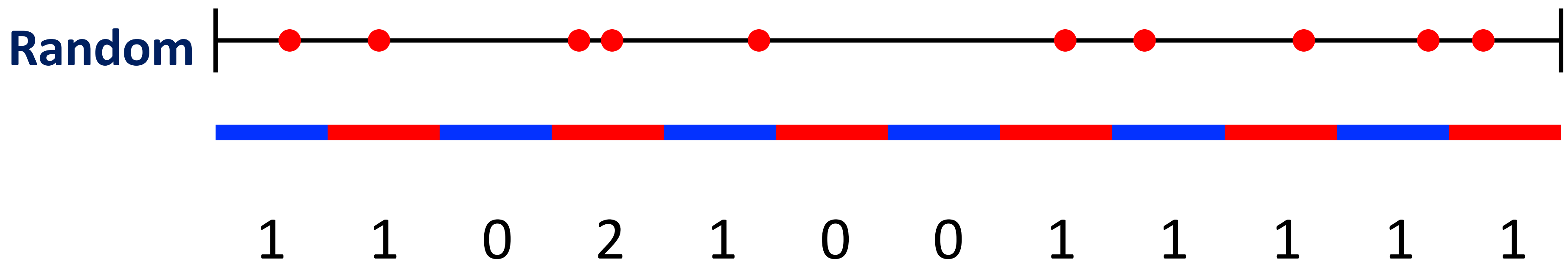
One-sample

```
ks.test(x = runif(20), y = "punif")
ks.test(x = runif(20), y = "pnorm")
```

Two sample

```
ks.test(x = runif(20), y = runif(20))
ks.test(x = runif(20), y = rnorm(20))
```

Testing for random



Expected?

Testing for random

Poisson distribution

Has single parameter (λ), the mean

$$P(X = x) = \frac{\lambda^k e^{-\lambda}}{k!}$$

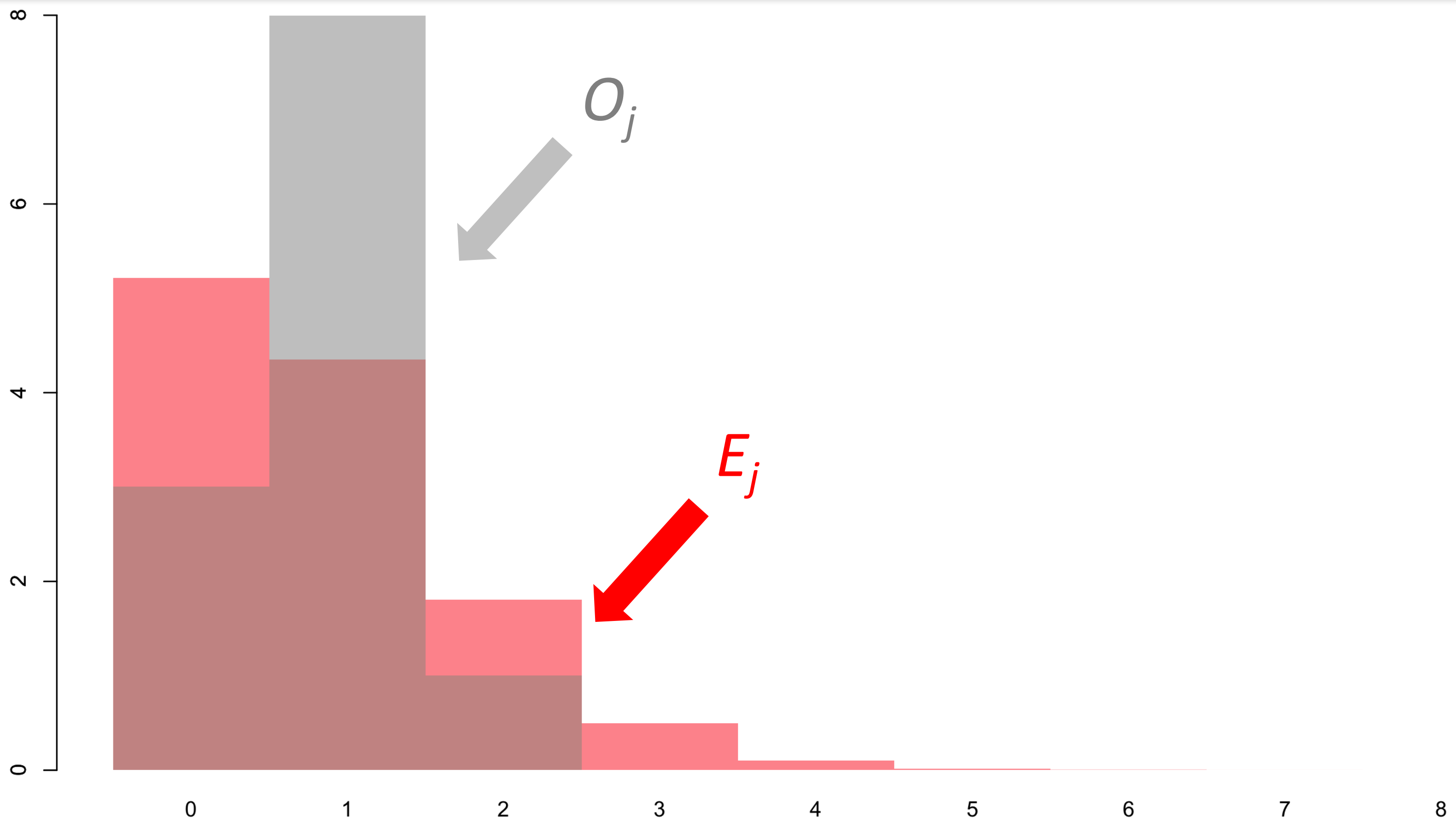
Expresses probability of given number of events

Assumes randomness; events can occur in space or time

Good null for random distribution of points in ~~space~~ time

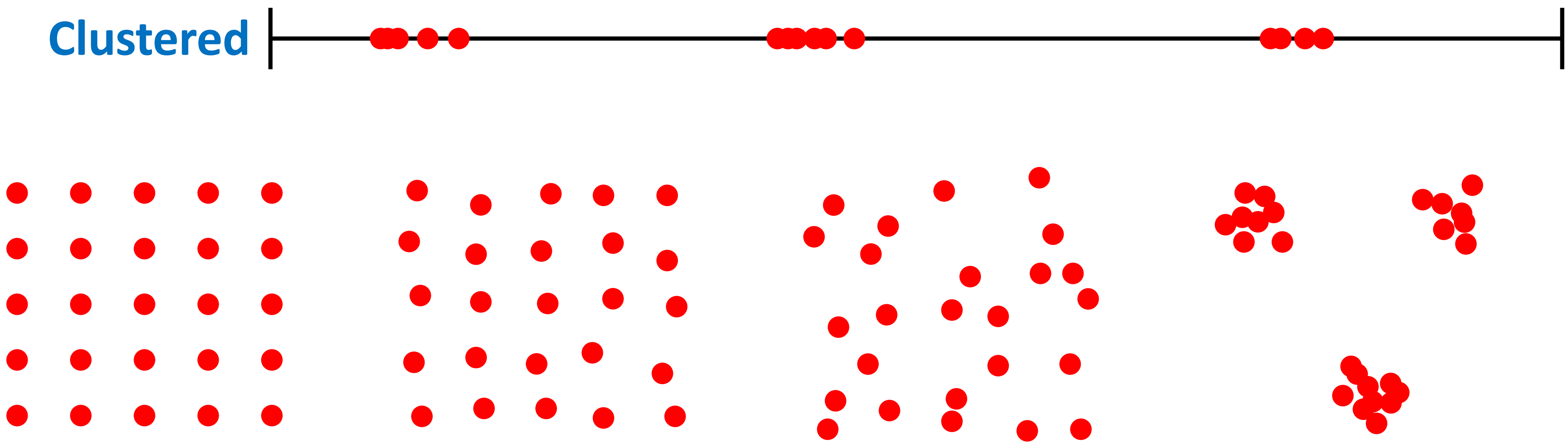
I.e., in example expected N time bins with k events

Testing for random



Testing for clustered

Clustered

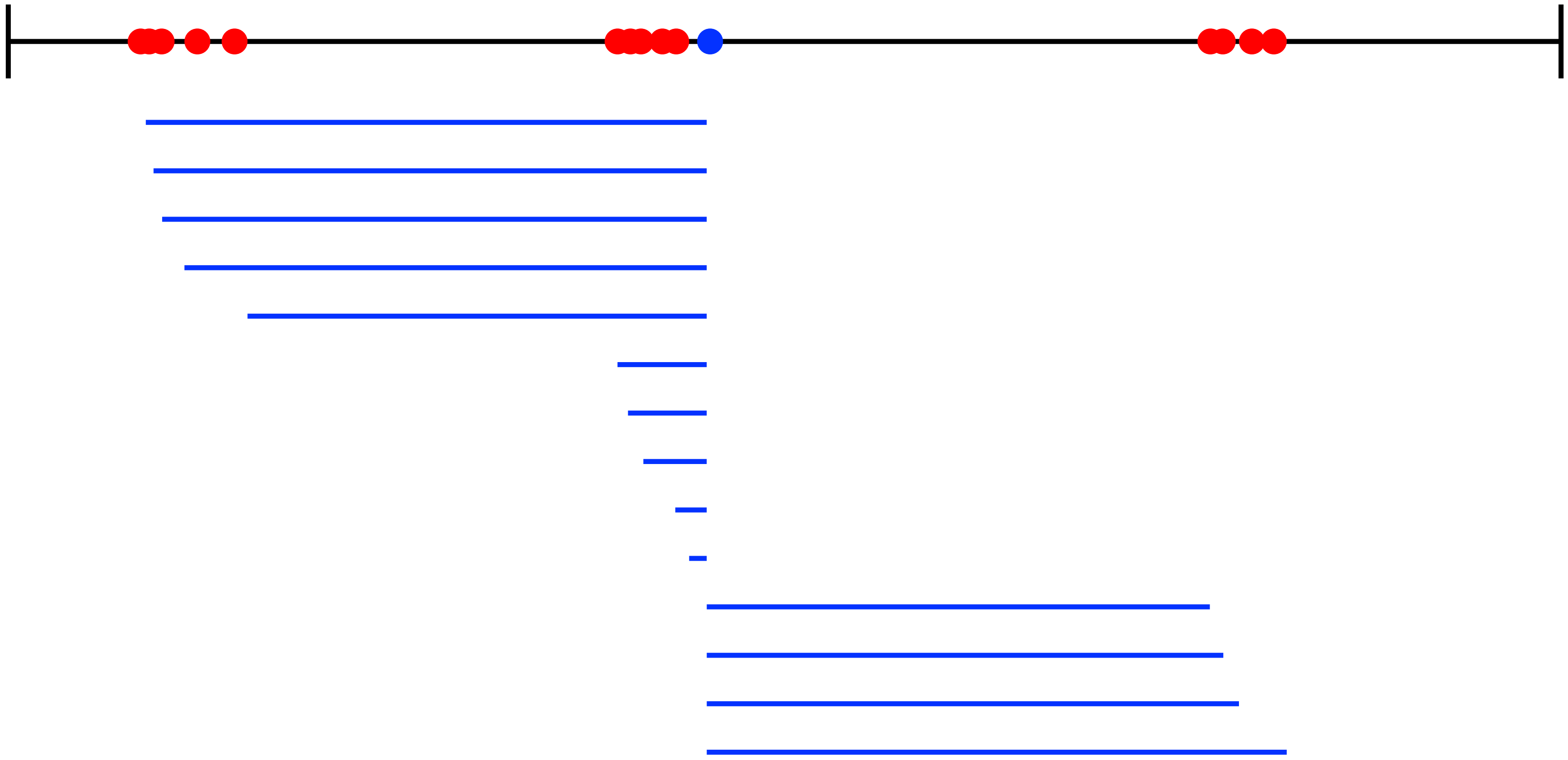


Regular | Uniform | Random | Clustered

$R = 2.15$ | $2.15 > R > 1$ | $R \approx 1$ | $R < 1$

Testing for clustered

Clustered



Testing for clustered

Mean nearest neighbour time

Captures average time to nearest event

$$\bar{t} = \frac{1}{n}\sum_{i=1}^{n} t_i$$

Expected mean nearest neighbour time

Uses total time (T) and number of events (n)

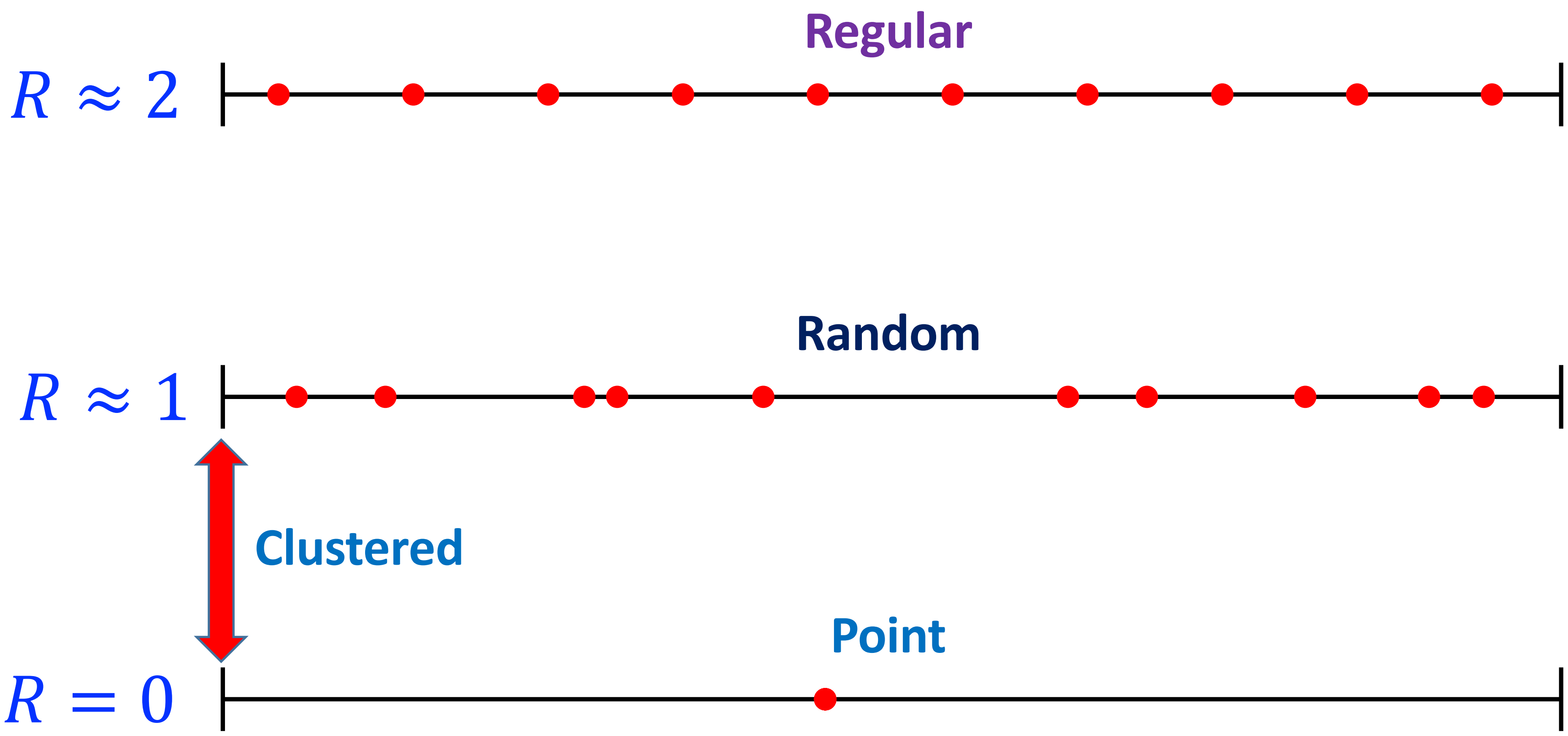
$$\bar{\delta} = \frac{1}{2}(T/n)$$

Nearest neighbour statistic

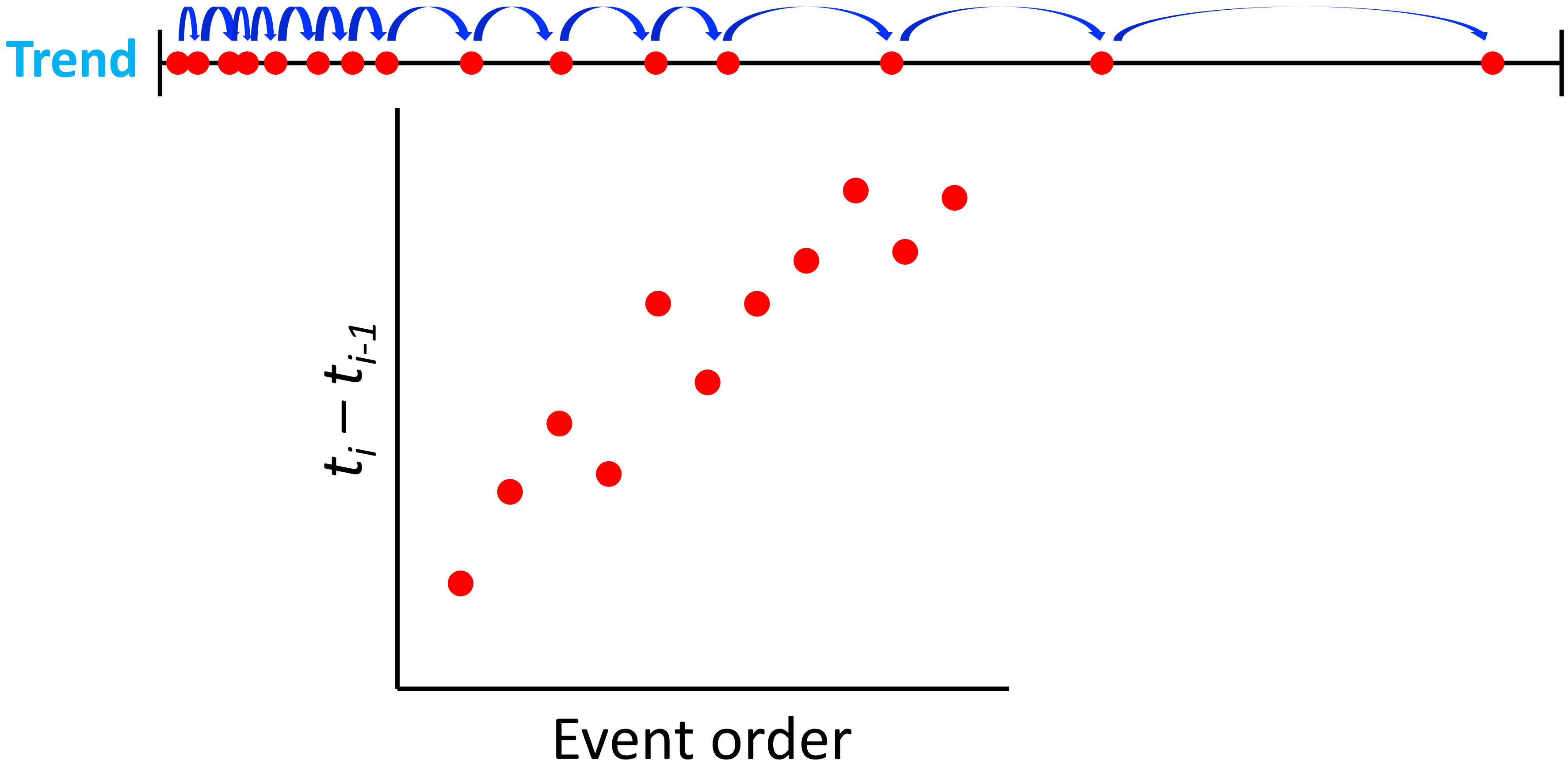
Varies from 0 to c. 2

$$R = \frac{\bar{t}}{\bar{\delta}}$$

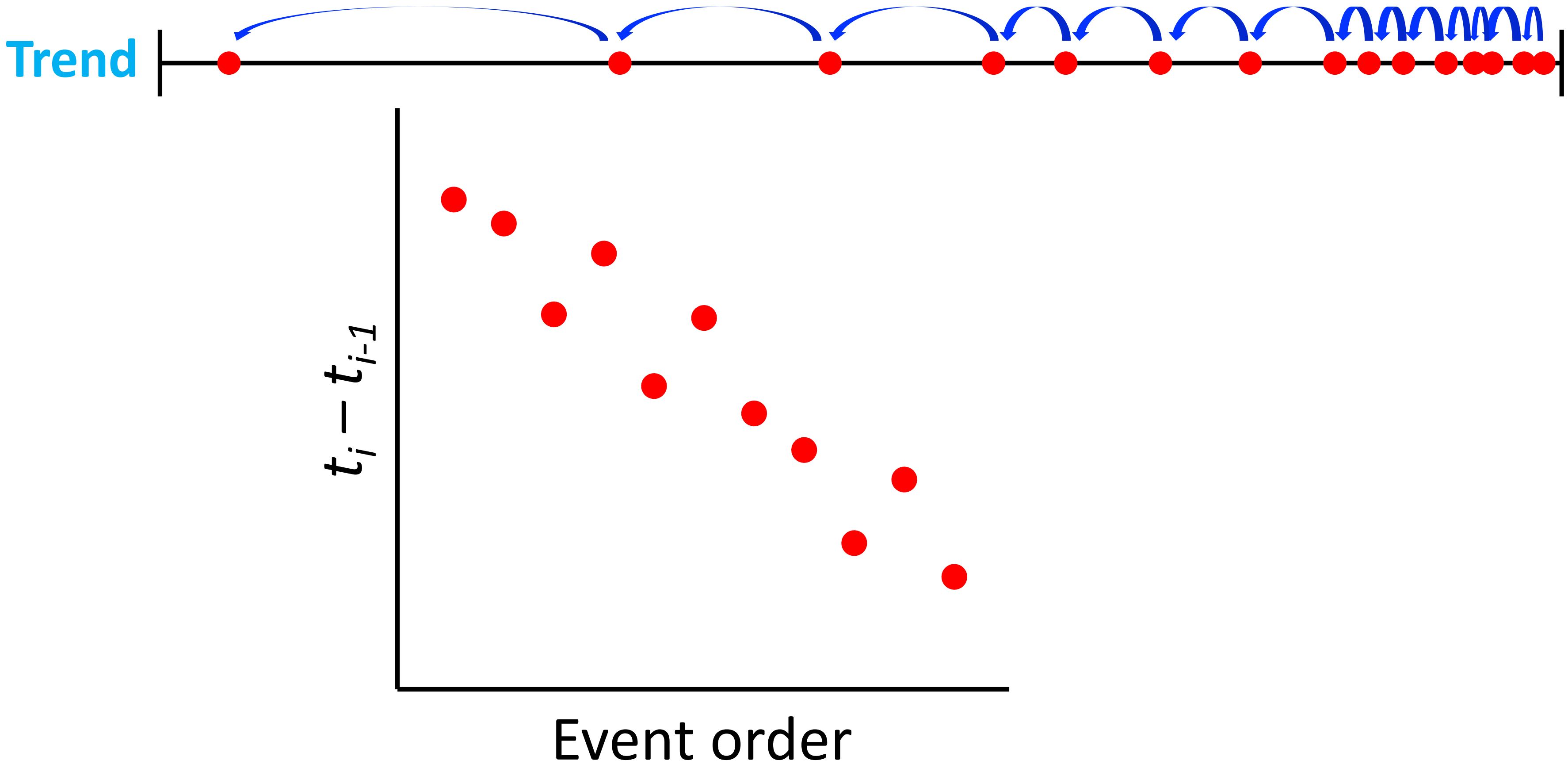
Testing for clustered



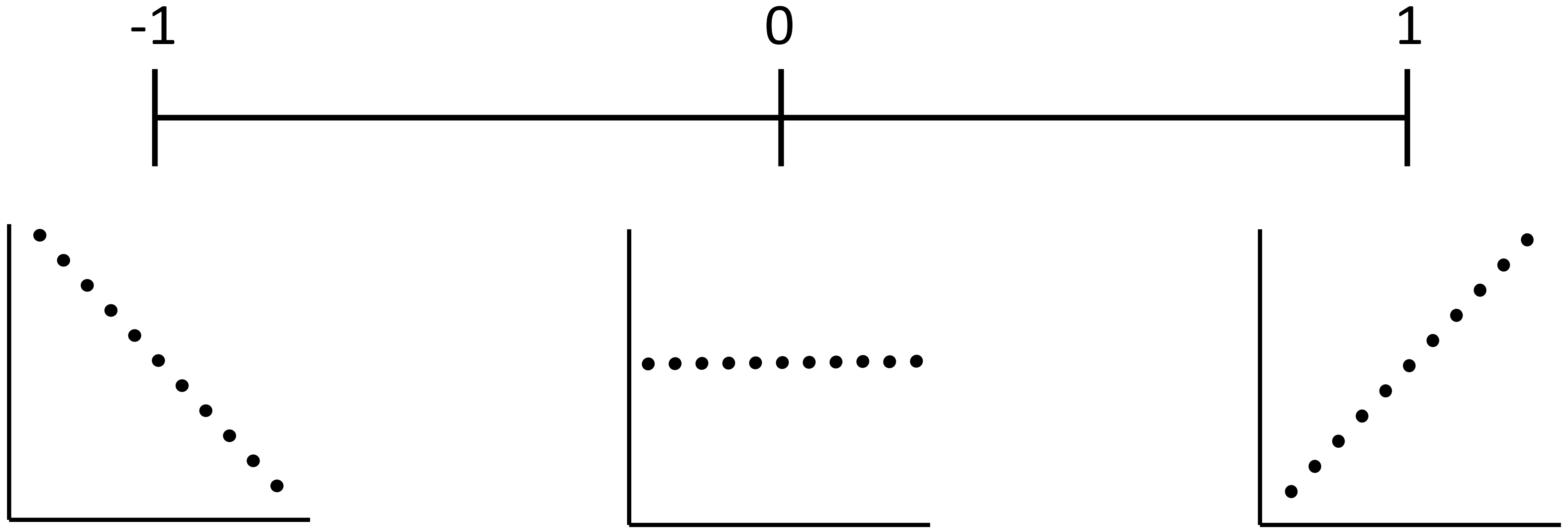
Testing for trend



Testing for trend



Testing for trend



Increasing event frequency

Decreasing event frequency

Testing for trend

Pearson correlation coefficient, r **PARAMETRIC**

$$r = \frac{\sum_{i=1}^{n}(Z_{xi} \times Z_{yi})}{n-1}$$

Spearman rank correlation coefficient, r' **NON-PARAMETRIC**

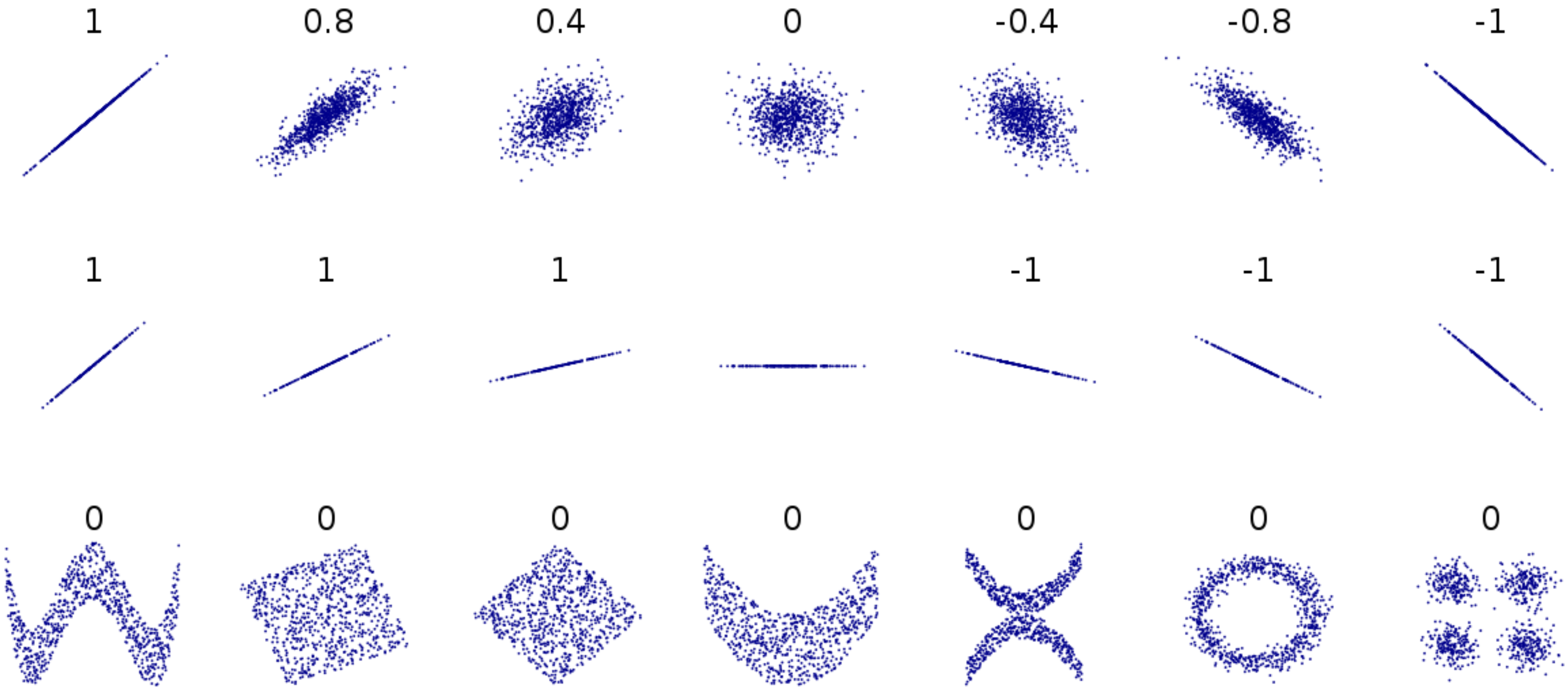
$$r' = 1 - \frac{6\sum_{i=1}^{n}[i - R(h_i)]^2}{n^3 - n}$$

Testing for trend

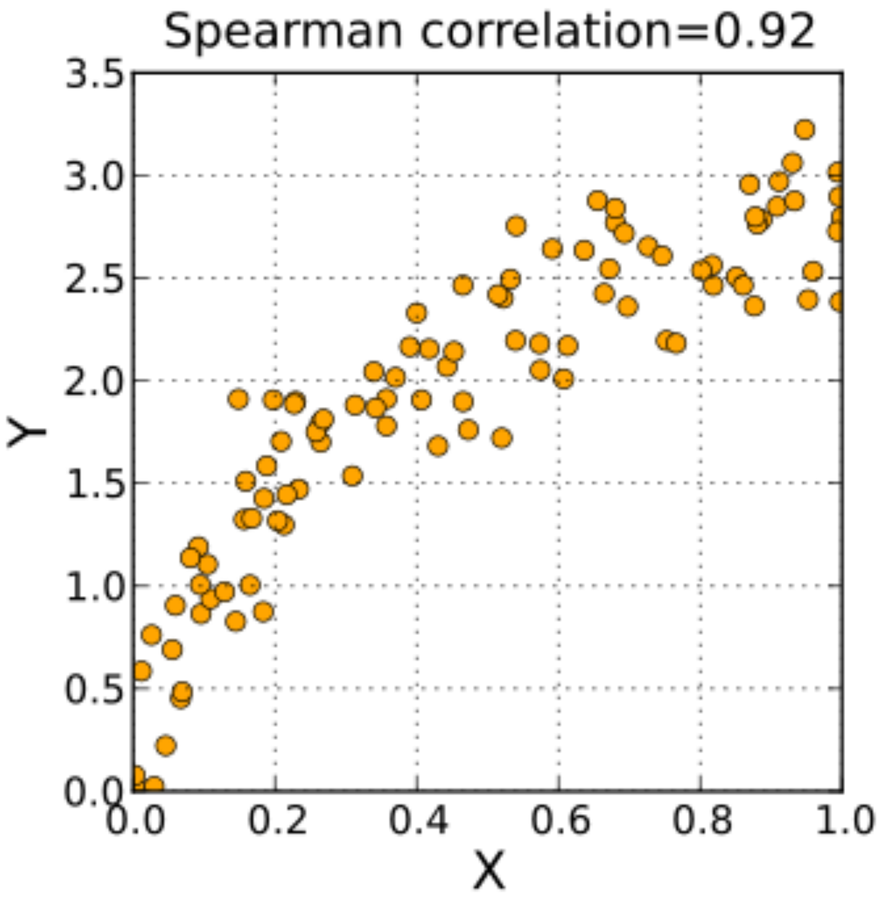
Spearman rank correlation coefficient, r’

```
cor.test(x = x, y = y, type = "spearman")$estimate
cor.test(x = x, y = y, type = "spearman")$p.value
```

Testing for trend

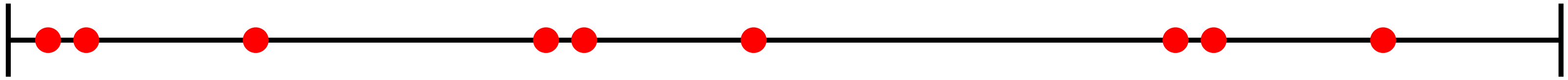


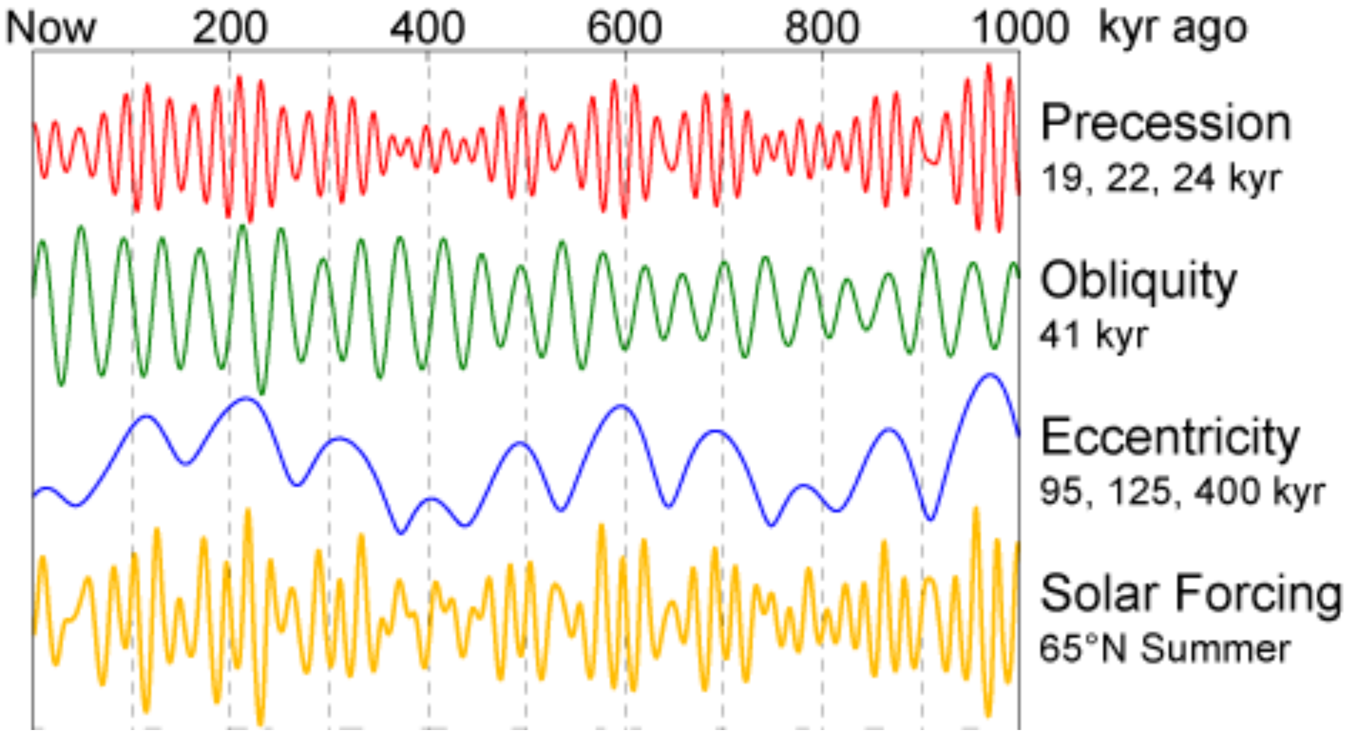
Testing for trend



Testing for pattern

Pattern



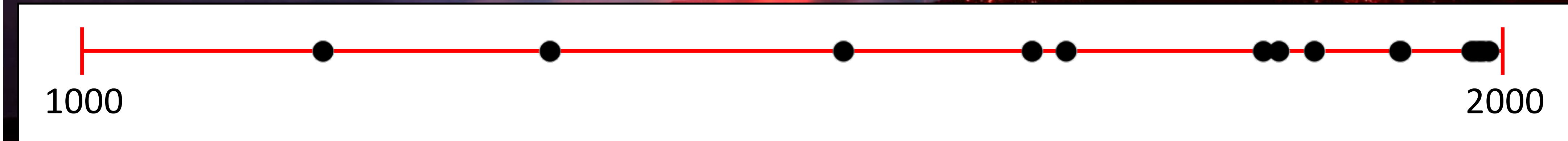


Exercise


Croatia
Bologna
Bosnia and
Herzegovina
Zadar
Sarajevo
Pisa
San Marino
Florence
Split
Italy
Adriatic Sea
Podgorica
Подгорица
Rome
Bari
Naples
Pompeii
Matera
Taranto
Lecce
Tyrrhenian Sea
Reggio
Calabria
Palermo
Ionian Sea
Trapani
Marsala
Sicilia
Catania
Syracuse

Exercise

```
EtnaEruptionYears <- c(1169, 1169, 1329, 1329, 1536, 1669, 1669,
  1693, 1832, 1843, 1868, 1928, 1929, 1979, 1981, 1984, 1985,
  1987, 1991)
```

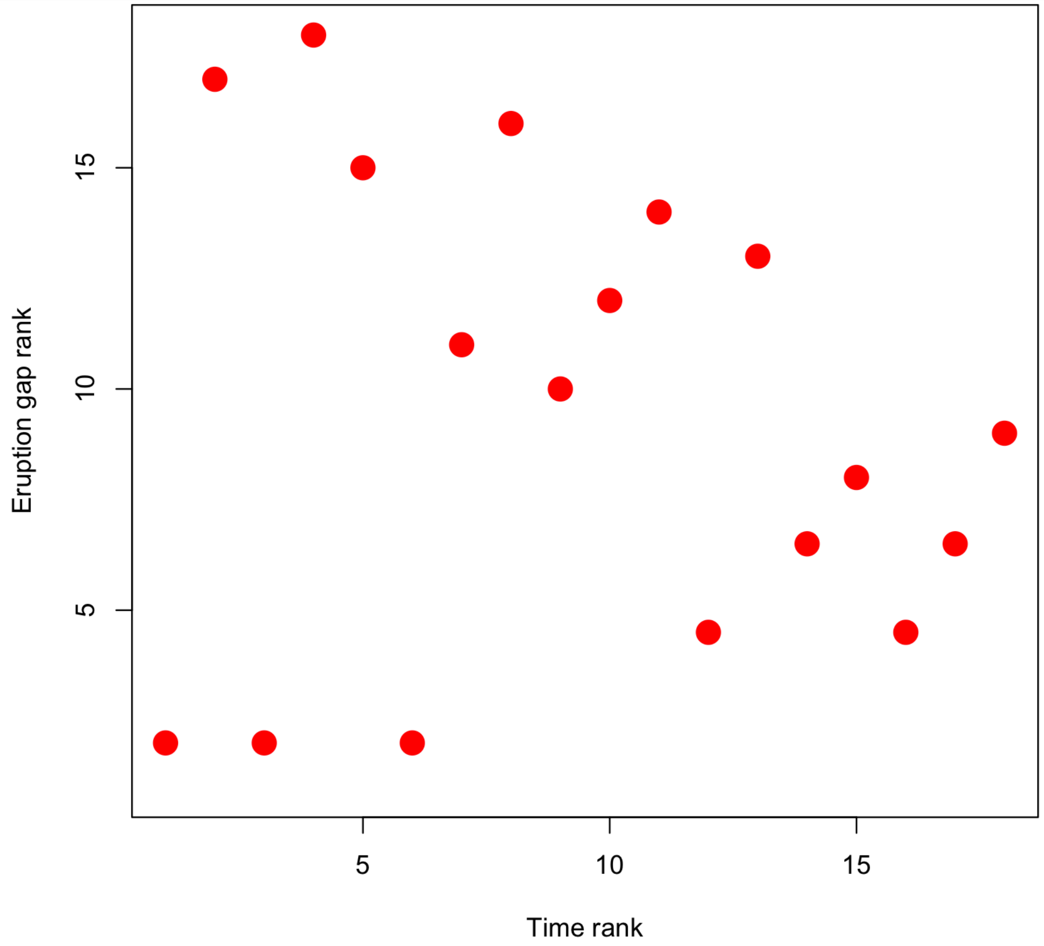


Exercise

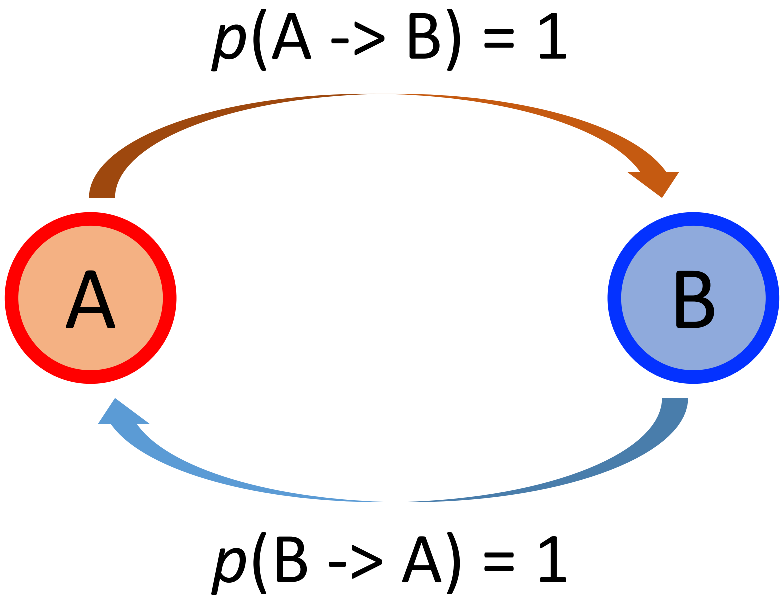
$$r' = 1 - \frac{6\sum_{i=1}^{n}[i - R(h_i)]^2}{n^3 - n}$$

```
diff(EtnaEruptionYears)
0 160 0 207 133 0 24 139 11 25 60 1 50 2 3 1 2 4

rank(diff(EtnaEruptionYears))
2 17 2 18 15 2 11 16 10 12 14 4.5 13 6.5 8 4.5 6.5 9

1:18 # temporal rank order of eruptions
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18
```

Exercise

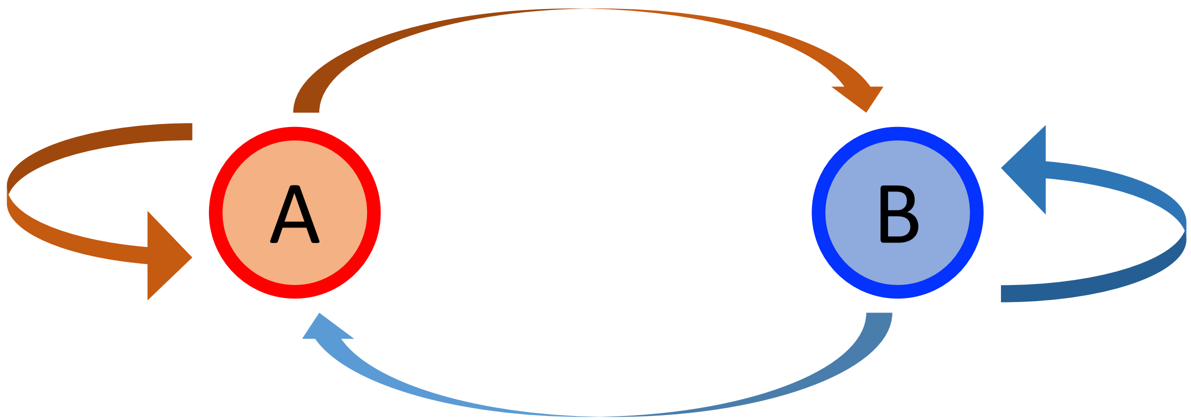


Markov chains

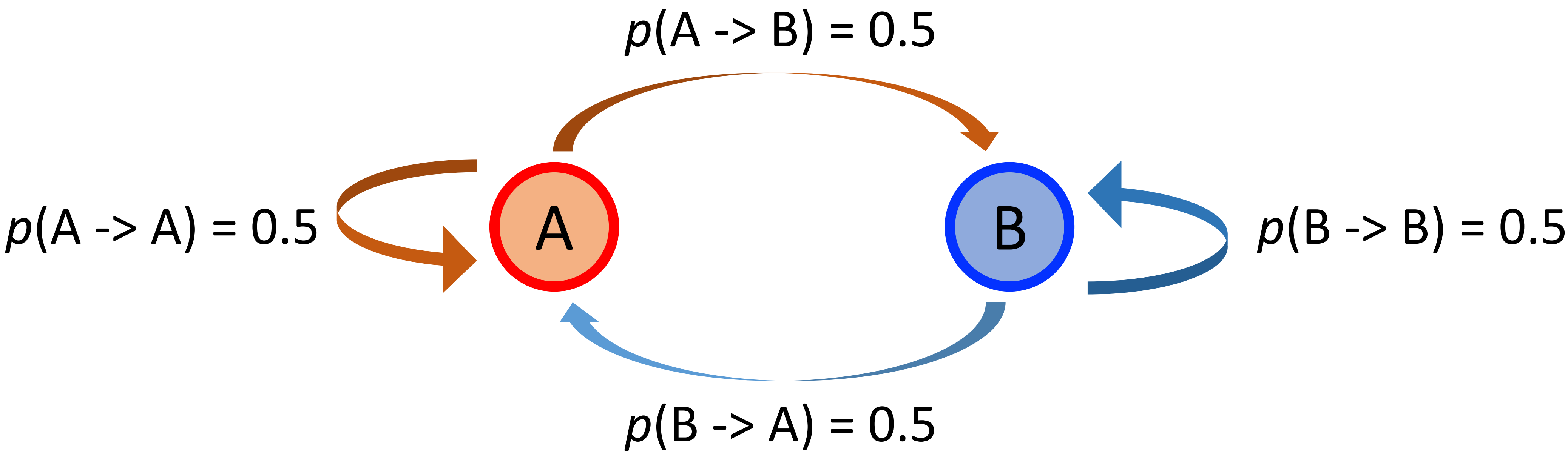


A B A B A B A B A B A B A B A B A B A B

Markov chains

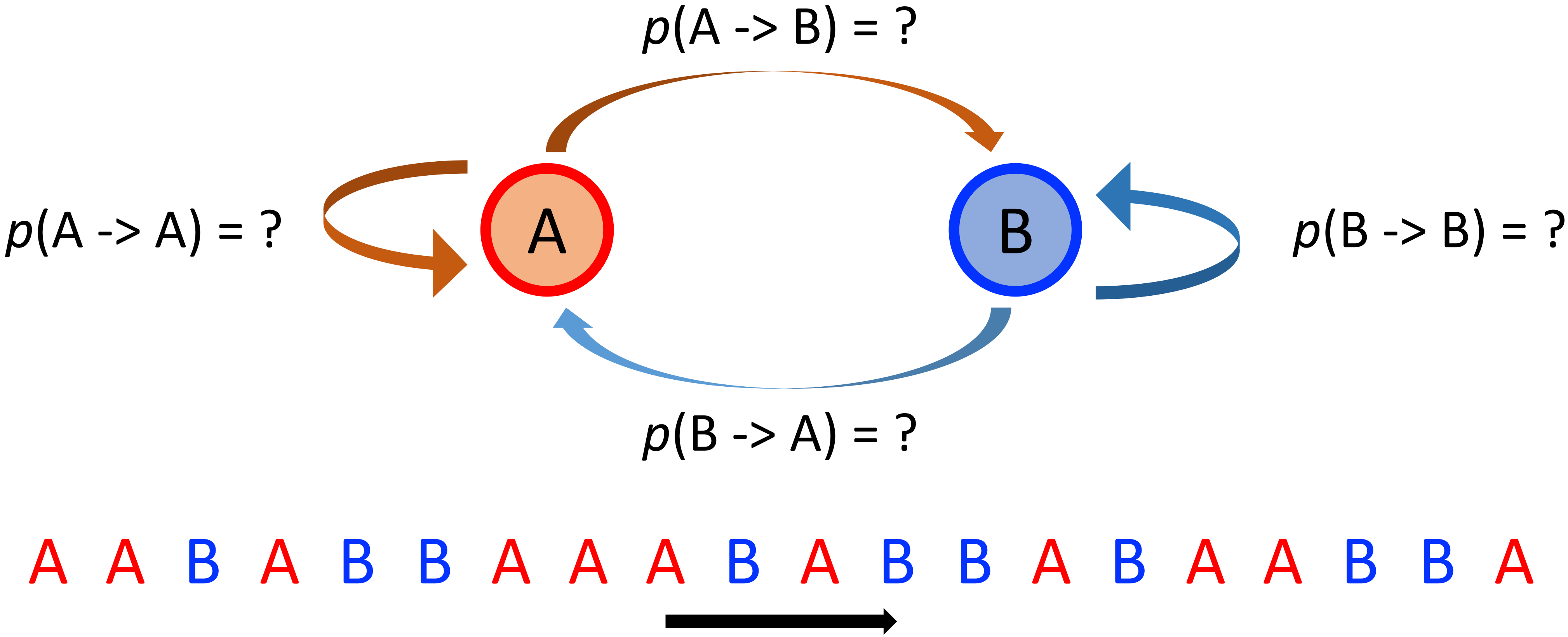


Markov chains



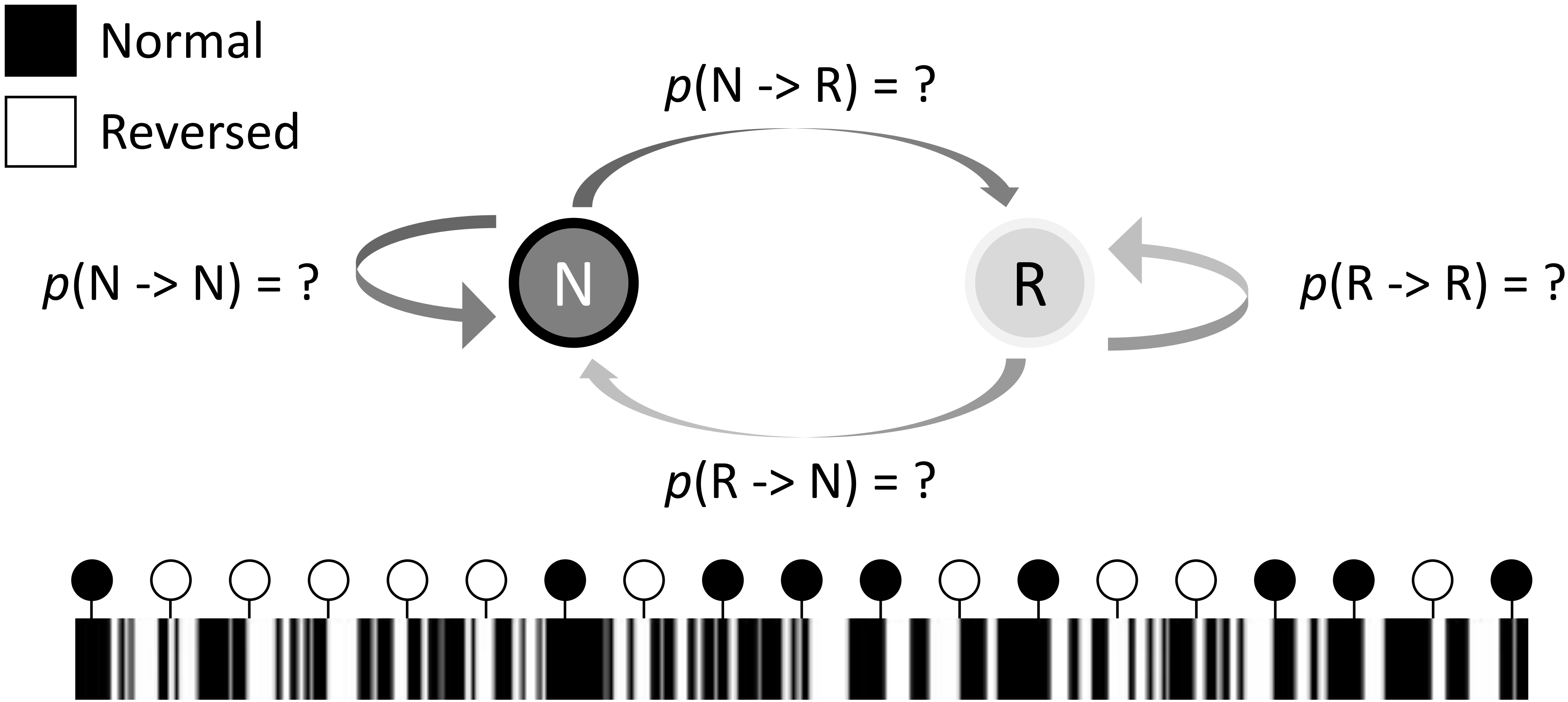
A A B A B B A A A B A B B A B A A B B A

Markov chains



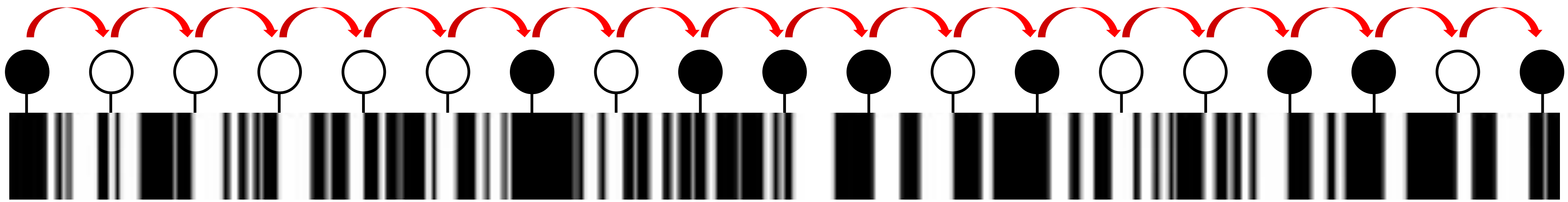
Markov chains

Markov chains



Markov chains

● → ●	3	*p*(N -> N)	= 3 / (3 + 5)	= 0.375
● → ○	5	*p*(N -> R)	= 5 / (3 + 5)	= 0.625
○ → ○	5	*p*(R -> R)	= 5 / (5 + 5)	= 0.500
○ → ●	5	*p*(R -> N)	= 5 / (5 + 5)	= 0.500



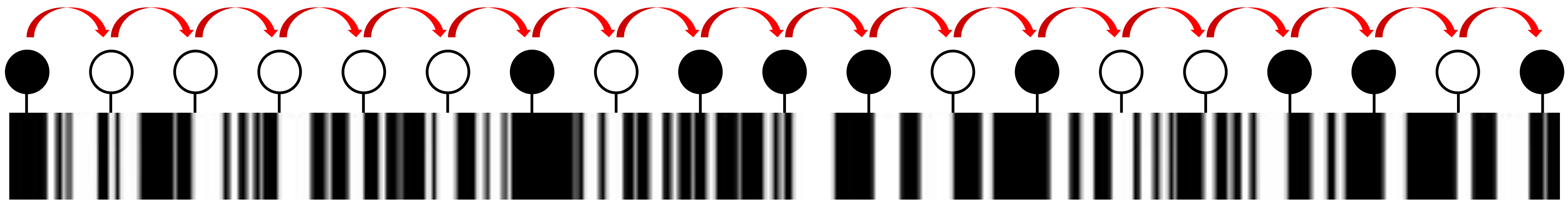
Markov chains

Transition frequency matrix

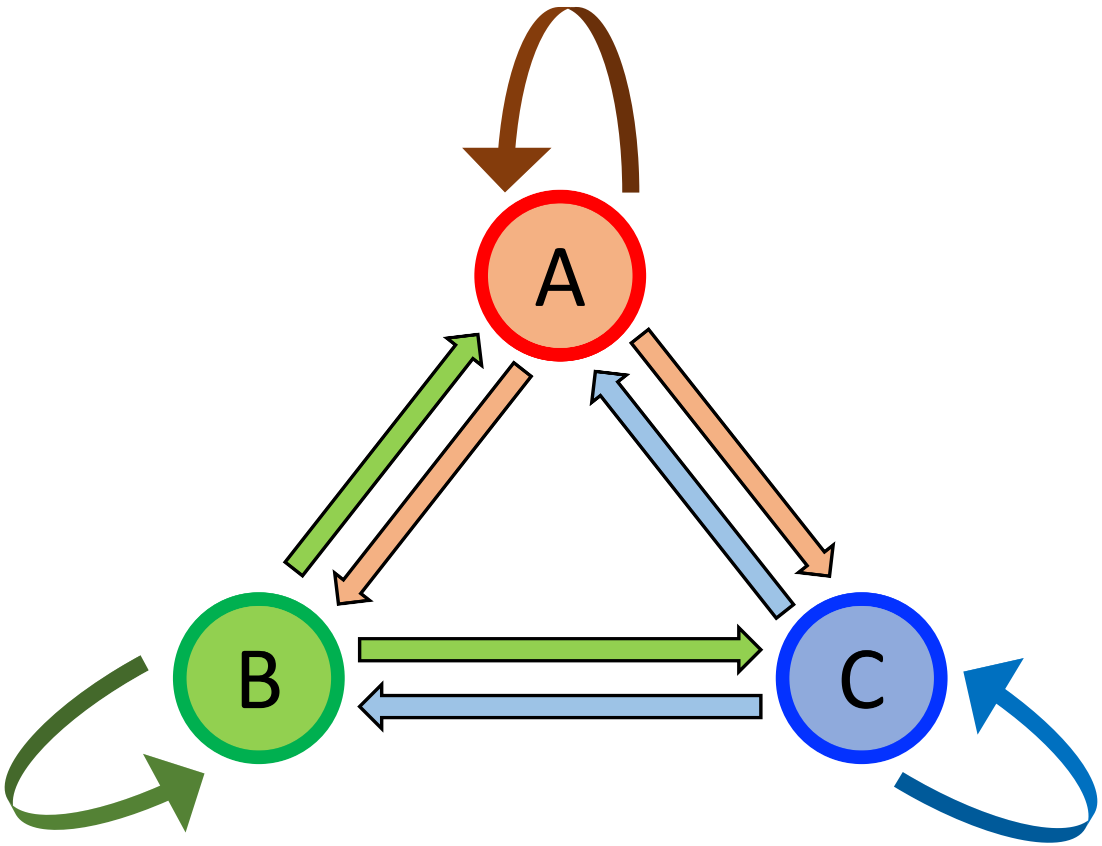
From \ To	●	○
●	3	5
○	5	5

Transition probability matrix

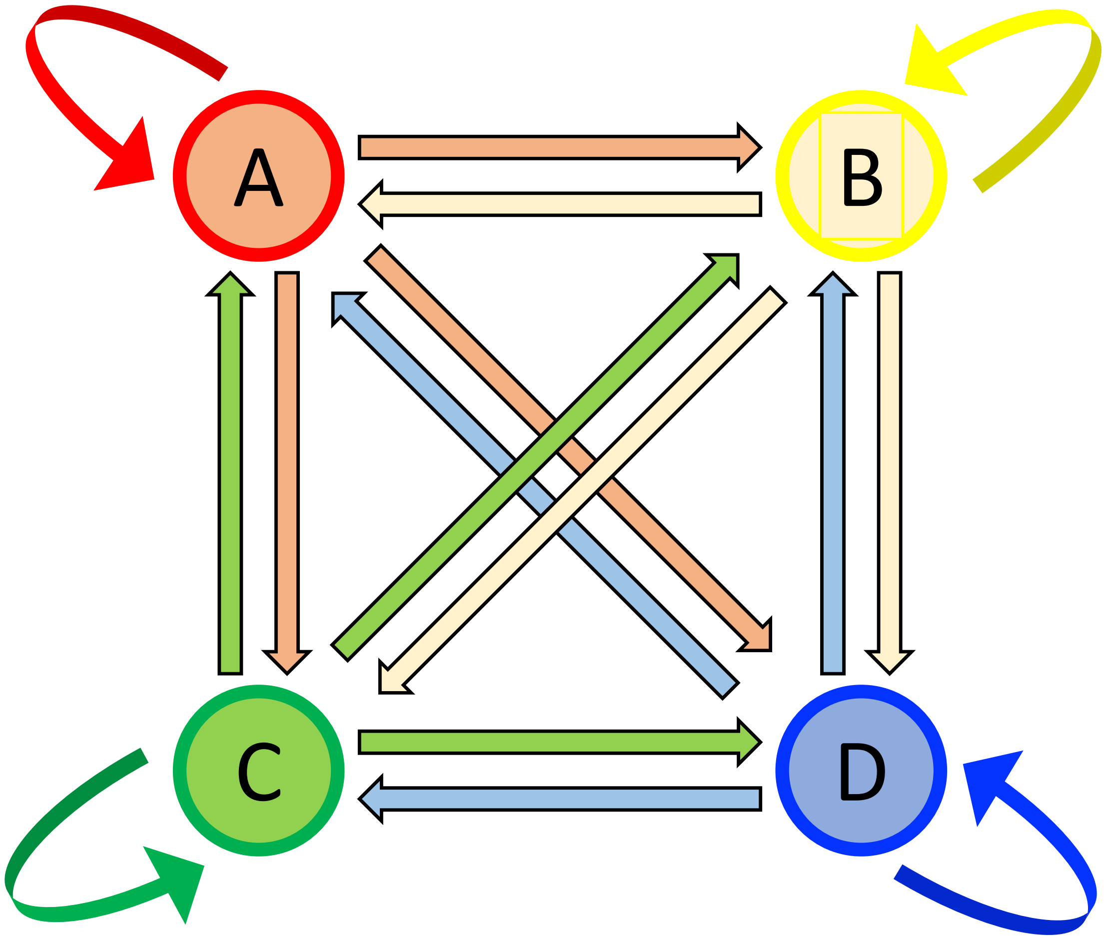
From \ To	●	○
●	0.375	0.625
○	0.500	0.500



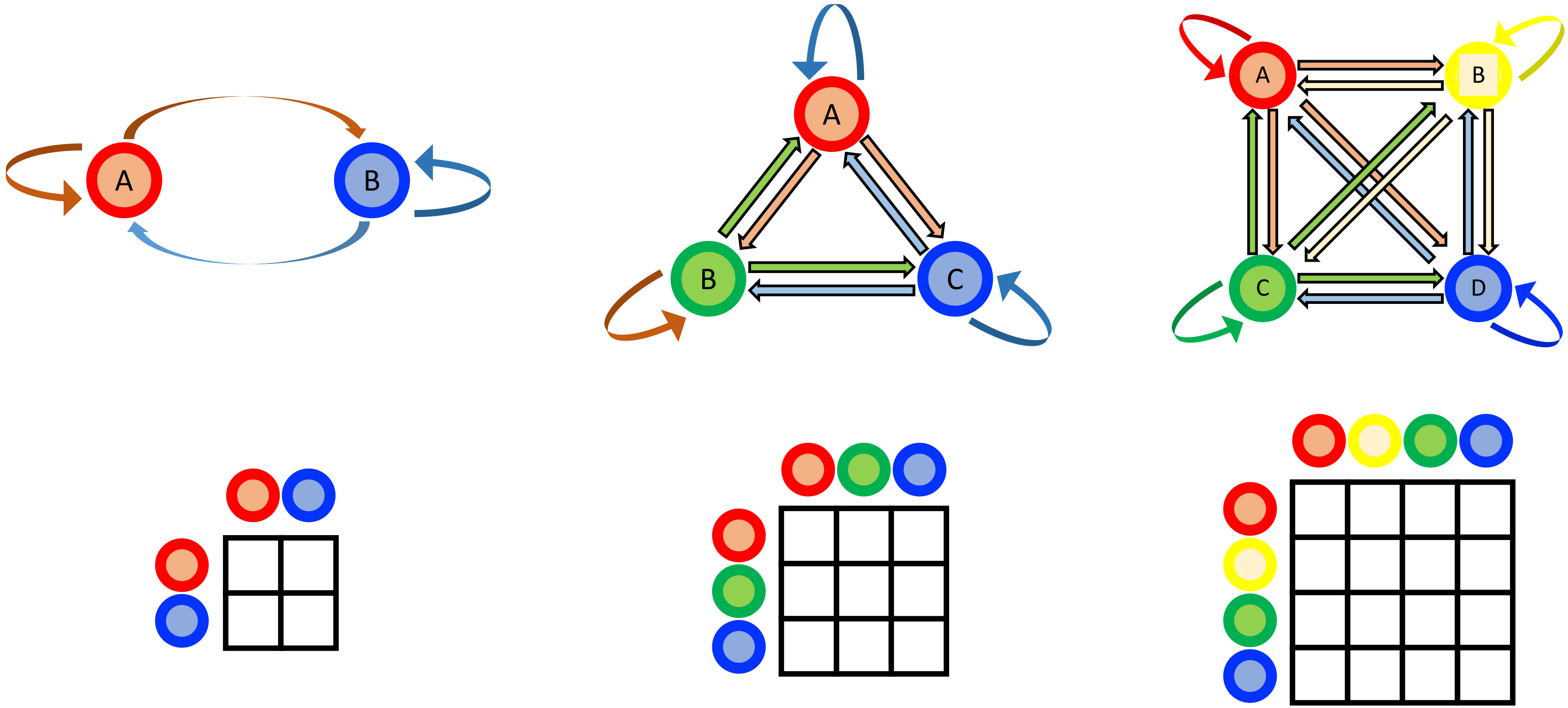
Markov chains



Markov chains

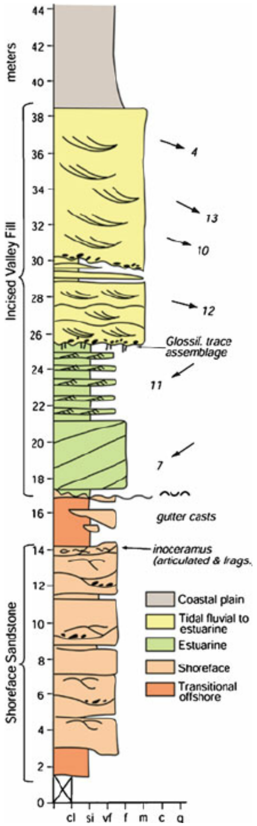


Markov chains

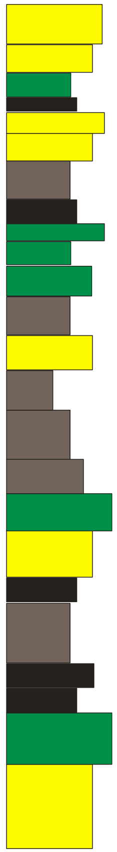


Markov chains

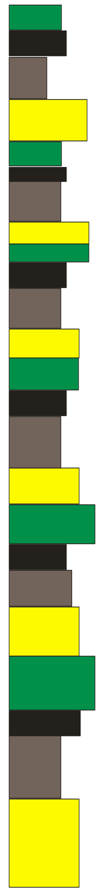
Sedimentary sequence



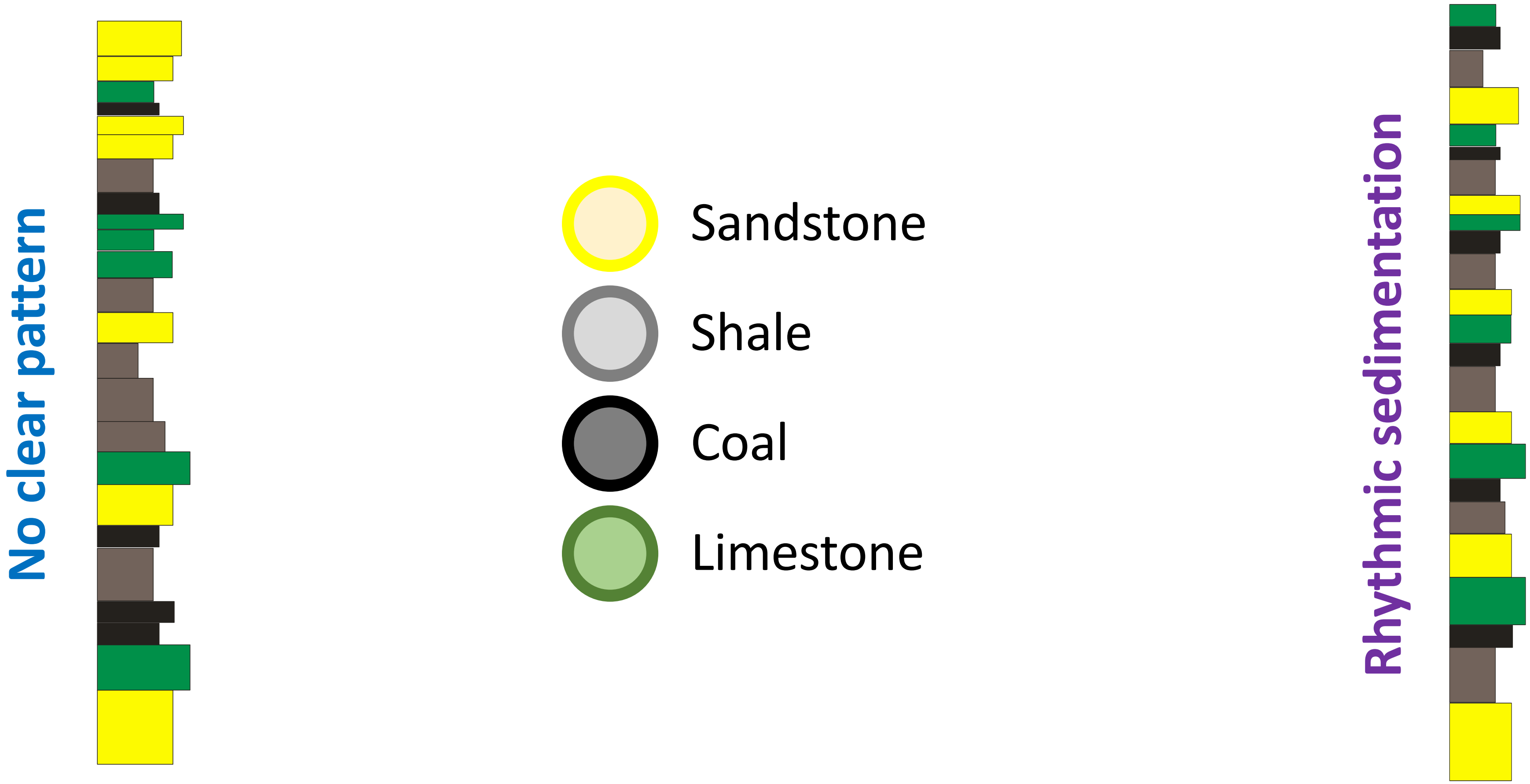
No clear pattern



Rhythmic sedimentation

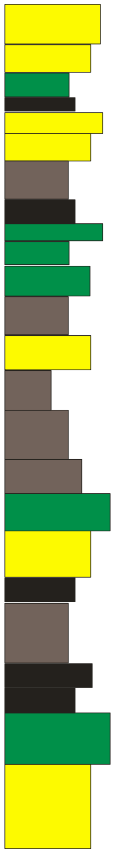


Markov chains



Markov chains

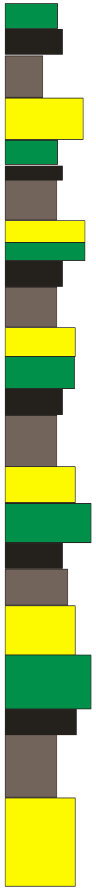
No clear pattern



Transition freq. matrix

2	1	1	2
2	2	1	1
1	2	1	1
1	1	2	2

Rhythmic sedimentation

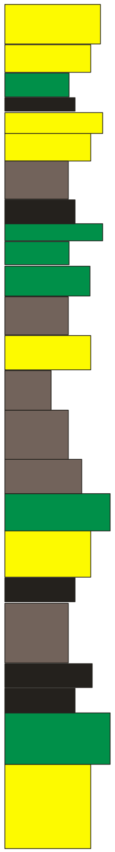


Transition freq. matrix

0	6	0	0
0	0	6	0
0	0	0	6
6	0	0	0

Markov chains

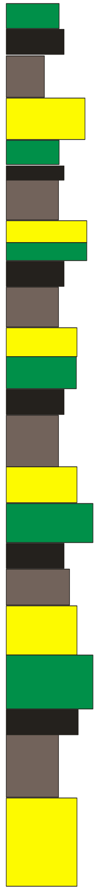
No clear pattern



Transition prob. matrix

0.3	0.2	0.2	0.3
0.3	0.3	0.2	0.2
0.2	0.3	0.2	0.2
0.2	0.2	0.3	0.3

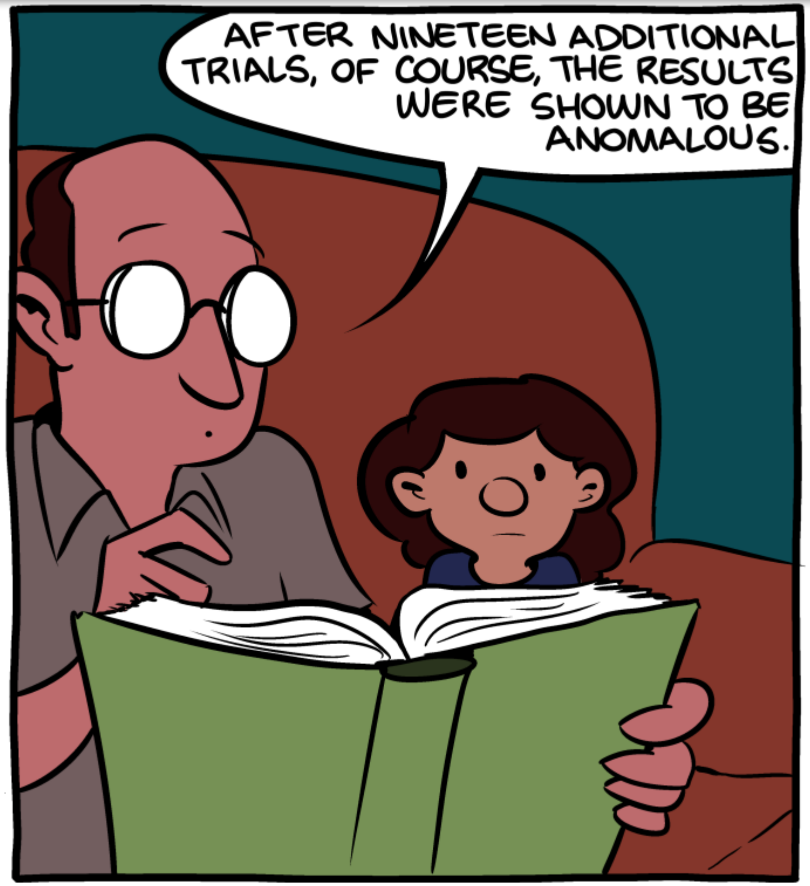
Rhythmic sedimentation



Transition prob. matrix

0	1	0	0
0	0	1	0
0	0	0	1
1	0	0	0

The importance of multiple replicates



"The Tortoise And The Hare" is actually a fable about small sample sizes.

Source: SMBC