

# Assessing Data Mining Results on Matrices with Randomization

Markus Ojala

Department of Information and Computer Science



**Aalto University**  
School of Science  
and Technology



# Introduction

## Setting

- Numerical  $n \times d$  matrix  $D$
- Data mining algorithm  $\mathcal{S}(D)$
- E.g., PCA,  $k$ -means clustering, or maximum correlation

	Prices (€/ kg)		
	Milk	Bread	...
Store 1	0.69	2.49	...
Store 2	0.79	2.79	...
Store 3	0.79	2.49	...
Store 4	0.89	2.89	...
	⋮	⋮	⋮

## Problem

- Is the result **significant** or just a **random effect** in the data?
- Is the result explained by some basic properties of the data?

# Example

Prices (€/ kg)

	Milk	Bread	Banana	Cheese	Ham	Salmon
Store 1	0.69	2.49	0.99	5.49	6.49	5.99
Store 2	0.79	2.79	1.19	6.69	7.13	6.99
Store 3	0.79	2.49	1.29	6.39	7.59	6.49
Store 4	0.89	2.89	0.99	6.59	6.99	7.49
Store 5	0.89	3.19	1.49	7.09	7.39	11.69
Store 6	0.99	3.59	1.79	8.09	8.69	9.59
Store 7	0.99	3.29	1.69	6.89	9.19	12.99
Store 8	1.19	4.59	1.99	8.49	8.59	16.99
Store 9	1.19	4.29	2.49	8.99	9.39	18.99
Store 10	1.29	3.99	2.19	7.79	9.99	14.49

High correlation: 0.9323 — is this interesting?

# Example

Prices (€/ kg)

	Milk	Bread	Banana	Cheese	Ham	Salmon	
Store 1	0.69	2.49	0.99	5.49	6.49	5.99	Cheap
Store 2	0.79	2.79	1.19	6.69	7.13	6.99	
Store 3	0.79	2.49	1.29	6.39	7.59	6.49	⋮
Store 4	0.89	2.89	0.99	6.59	6.99	7.49	⋮
Store 5	0.89	3.19	1.49	7.09	7.39	11.69	Average
Store 6	0.99	3.59	1.79	8.09	8.69	9.59	
Store 7	0.99	3.29	1.69	6.89	9.19	12.99	⋮
Store 8	1.19	4.59	1.99	8.49	8.59	16.99	⋮
Store 9	1.19	4.29	2.49	8.99	9.39	18.99	Expensive
Store 10	1.29	3.99	2.19	7.79	9.99	14.49	

High correlation: 0.9323 — is this interesting?

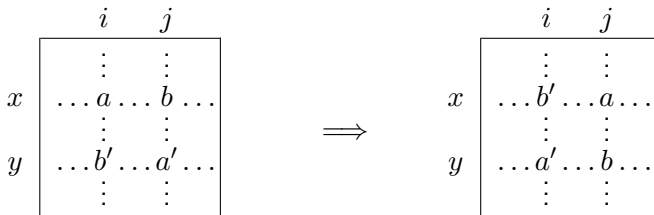
NOT! — general price levels in stores explain the correlation

# Previous Approaches

Randomization of binary matrices (Gionis et al., 2007)

Randomization of real-valued matrices (Ojala et al., 2009)

- Is the result explained by **row and column value distributions**?
- Preserve row and column value distributions in randomization
- Is the original result better than results on randomized data?
- Randomization methods: *GeneralMetropolis*, *SwapDiscretized*



# New Improved Approach (1/2)

## Contributions

1. A new method generalizing *SwapDiscretized*
  - Preserves the row and column distributions more accurately
2. Support for features measured in different scales
  - Preserve feature-wise **rank distributions** of observations

Original matrix						Feature-wise ranks					
0.69	2.49	0.99	5.49	6.49	5.99	1	1	1	1	1	1
0.79	2.79	1.19	6.69	7.13	6.99	2	3	4	3	4	3
0.79	2.49	1.29	6.39	7.59	6.49	3	2	2	4	2	2
0.89	2.89	0.99	6.59	6.99	7.49	4	4	3	2	5	4
0.89	3.19	1.49	7.09	7.39	11.69	5	5	5	7	3	6
			...						...		

# New Improved Approach (2/2)

## Contributions

3. Support for non-Gaussian value distributions
  - Scale does not matter, only the ordering
4. Support for missing values and sparse structure
  - Sparsity and missing values of rows and columns are preserved
5. No need for manual tuning: theoretically justified parameters
  - Kolmogorov-Smirnov test
  - Approximation of mixing time

# Experiments (1/2)

## Datasets

- Various real-life datasets
- Sparse structure, missing values, dissimilar features etc.

**Artificial:** Random

**UCI:** Iris, Wine, Water, Breast, Wdbc, WineQuality, Letter

**Others:** Gene, Jester, MovieLens

## Assessing data mining results

- K-means clustering
- Principal component analysis
- Correlations between features



# Experiments (2/2)

## K-means and PCA

- RANDOM: results are **nonsignificant**
- Real-life datasets: all results are **significant**

## Correlations between features

- Benjamini-Hochberg for multiple correction

	# Significant / # Pairs	Threshold
RANDOM	0 / 4950	–
BREAST	1 / 36	$\geq 0.91$
LETTER	17 / 120	$\geq 0.49$
GENE	428 / 1770	$\geq 0.42$

# Conclusions

- Interesting result on a matrix?  
= not explained by row and column value distributions
- Introduced a new practical randomization method
- Supports various real-life matrices
- Implementation freely available:  
<http://www.cis.hut.fi/mrojala/randomization/>