AUTOMAT[R]IX:

Learning Simple Matrix Pipelines

Lidia Contreras-Ochando, César Ferri, José Hernández-Orallo

{liconoc, jorallo}@upv.es, cferri@dsic.upv.es

VRAIN – Valencian Research Institute for Artificial Intelligence (Universitat Politècnica de València)

Introduction

NA	0.30	0.50	NA	NA	NA	NA
NA	NA	NA	0.90	NA	NA	0.40
NA	NA	NA	NA	NA	NA	NA
NA	NA	NA	NA	0.60	NA	NA
NA	NA	NA	NA	NA	NA	NA

1	2
1	3
2	4
4	5
2	7

I need to transform the matrix on the left into the matrix on the right (position of non-empty values).
Can you code it?

Matrices are a very common way of representing and working with data
There are a Lot of functions in different programming languages
What if I don't have programming knowledge?

Problem: Induce R programs to transform an input matrix to an output matrix using only few examples from the result

Problem Definition

Considering:

1	3	5
4	NA	6
NA	NA	7

An input matrix A $(\mathrm{m}\ x\ n)$

·	2	·

A partially filled matrix B $(m' x\ n')$

"How to know the number of NA in each column"

Optionally, some textual hint T in natural language

We look for a combination of functions (in R[1]) f such that $f(A) = S$

`colSums(is.na(A))` = | 1 | 2 | 0 |

where S is a matrix $(m' x\ n')$, such that for every non-empty $b_{ij} \in B$ there is a $s_{ij} \in S$ such that $b_{ij} = s_{ij}$

Method

Dimensional Constraints:
Each primitive g in the background knowledge G includes a tuple (m_{min}, n_{min}, τ)
- m_{min} n_{min} : minimum size for the input .
- τ: $\mathbb{R}^{2'} \rightarrow \mathbb{R}^2$: type function which maps the dimension of input to output.

Probabilistic Model:
We guide the search using the probability:

$$p(g|T, A, B) = \gamma \cdot p(g) + (1-\gamma) \cdot p_0(g|T)$$

Prior Probability

Frequency of use of the 2000 most used R functions on GitHub[2]

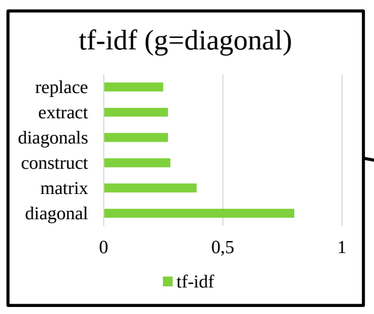
	Function	Freq.
1	length	0,33
2	nrow	0,09
3	is.na	0,08
...	...	...

Being n_g the absolute frequency of use:

$$p(g) = \frac{n_g}{\sum_{g \in G} n_g}$$

Frequency Model

tf-idf values from R help documentation



Text hint provided by the user

T= "How to extract the diagonal of a matrix in R"

$s(H_g, T)$ = cosine similarity

The text hint probability is defined as:

$$p_0(g|T) = \frac{s(H_g|T)}{\sum_{g \in G} s(H_g, T)}$$

Weights:

- Final Dimension: α gives more weight to those transformations where the output size matches the size of B

$$p^*(g_1 g_2 \dots g_d) = (1 + \alpha m) \prod_{i=1}^{d} p(g_i | g_{i-1} g_{i-2} \dots g_1, T, A, B)$$

- Sequential Dependencies: β reduces the weight to those transformations repeating functions

$$p(g_i | g_{i-1} \dots g_1, T, A, B) = \beta p(g|T, A, B) + (1-\beta) p(g_i | g_{i-1} g_{i-2} \dots g_1, T, A, B)$$

$$p(g_i | g_{i-1} g_{i-2} \dots g_1, T, A, B) = 0 \quad \text{if } g_i \in \{g_{i-1}, g_{i-2}, g_{i-1}\}$$

$$= p(g|T, A, B) \quad \text{otherwise}$$

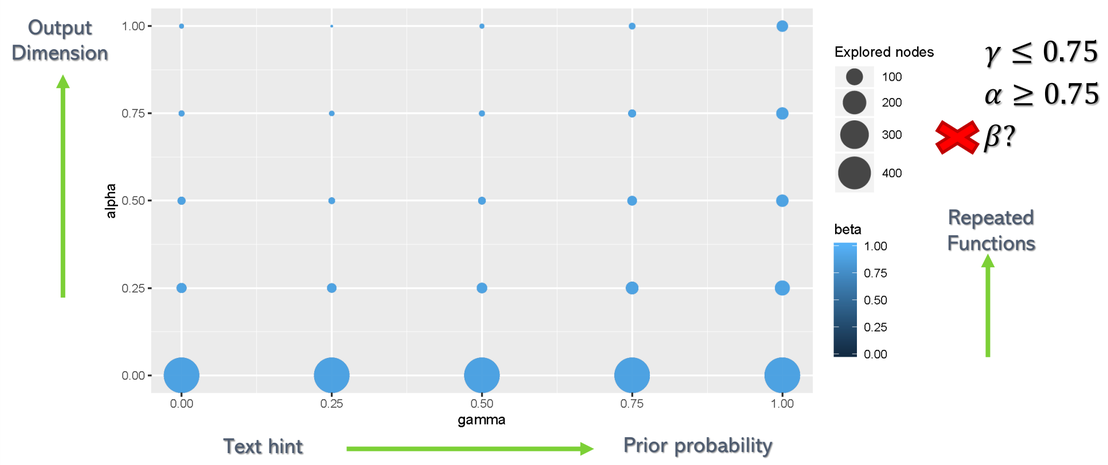
β=[0,1]

Experiments & Results

Background Knowledge (functions)
34 R functions related to matrices (base, stats and Matrix packages)

Data
- Artificial data: 400 pairs of matrices A,B (with 80% empty cells).
- 30 questions/answers from stackoverflow

Parameter Setting

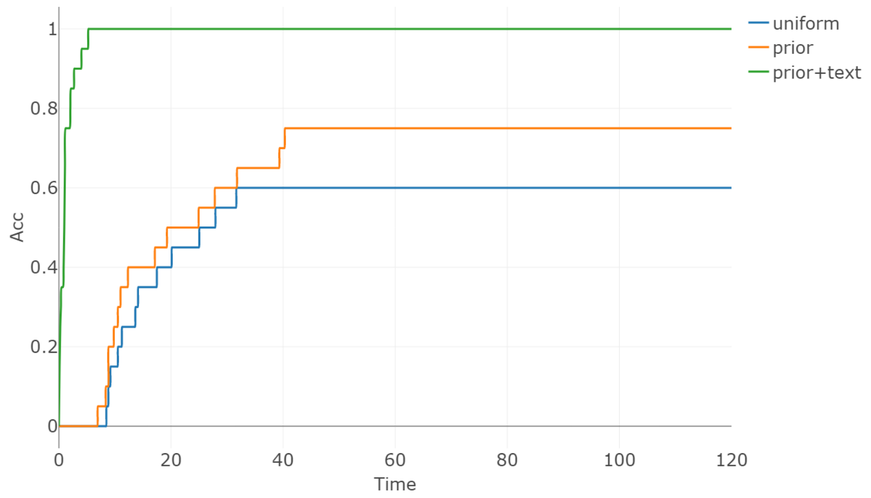


$\gamma \leq 0.75$
$\alpha \geq 0.75$
β?

Average of the time needed to find the solution, depending on the value of α, β and γ parameters

Strategies
- **Uniform:** Considering $p(g)$ uniform.
- **Prior**: Building a Dynamic Background Knowledge using the prior probability
- **Prior + Text:** Building a Dynamic Background Knowledge using the prior probability and the frequency model generated with T

Results



Percentage of cases that are solved and the time needed to find the solution

Conclusions

We have created a new system able to solve matrix transformations:
- Based on a breadth-search approach; Pruned by the dimensions of the matrices; Guided by a strategy based on dynamic probabilities from:
  - A prior value depending on the primitive frequency on Github.
  - tf-idf values of text hints provided by the user and the R help documentation of the functions.

Future work:
- Add new characteristics (constraints)
- Include more primitives and new data structures
- Create a visual interface or R package
- Replicate for other languages such as Python.

[1]R: https://www.r-project.org/
[2]The 2000 most used R functions on GitHub : shorturl.at/pDFRZ