

EST-XXX

NOMBRE DEL CURSO

Juan Zamora O.

FECHA.



PONTIFICIA  
UNIVERSIDAD  
CATÓLICA DE  
VALPARAÍSO

# Estructura de la Presentación

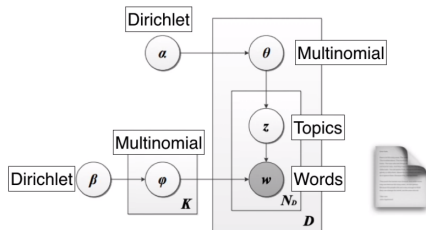
- 1 Aproximaciones Low-Rank para Clustering
- 2 Ejemplo numérico: datos iniciales
- 3 Ejemplo con 2 columnas de igual ancho
- 4 Pseudocódigo
- 5 Código en R
- 6 Código en Python

# Aproximaciones Low-Rank para Clustering

Una matriz  $X$  de rango  $r$  admite una factorización de la forma

$$X = BC^T, B \in \mathbf{R}^{m \times r}, C \in \mathbf{R}^{n \times r}$$

$X$  es aproximada con bajo rango (low-rank) cuando  $\text{rango}(X) \ll \min(m, n)$



## Ejemplo numérico: datos iniciales

$$V = \begin{bmatrix} 5 & 3 \\ 3 & 2 \\ 4 & 1 \end{bmatrix}$$

## Ejemplo con 2 columnas de igual ancho

Left column text.  
Another text line.

- Item 1.
- Item 2.
- Item 3.

# Pseudocódigo

```
1 for i:=maxint to 0 do
2 begin
3     j:=square(root(i));
4 end;
```

# Código en R

```
1 library(topicmodels)
2
3 # Entrenamiento del modelo
4 lda_model <- LDA(dtm_train, k = 2, control = list(seed =
    1234))
5
6 # Cálculo de perplexidad en datos de prueba
7 log_likelihood <- mean(sapply(seq_len(nrow(dtm_test)),
    function(i) {
8     logLikelihood(lda_model, dtm_test[i, ]) # Puede requerir
        función auxiliar
9     }))
10
11 num_words <- sum(colSums(dtm_test))
12 perplexity <- exp(-log_likelihood / num_words)
13
14 cat("Perplexity:", perplexity, "\n")
```

# Codigo en Python

```
1
2 for i in range(2, num):
3     if (num % i) == 0:
4         # if factor is found, set flag to True
5         flag = True
6         # break out of loop
7         break
```



## Codigo en Python

```
1
2 for i in range(2, num):
3     if (num % i) == 0:
4         # if factor is found, set flag to True
5         flag = True
6         # break out of loop
7         break
```