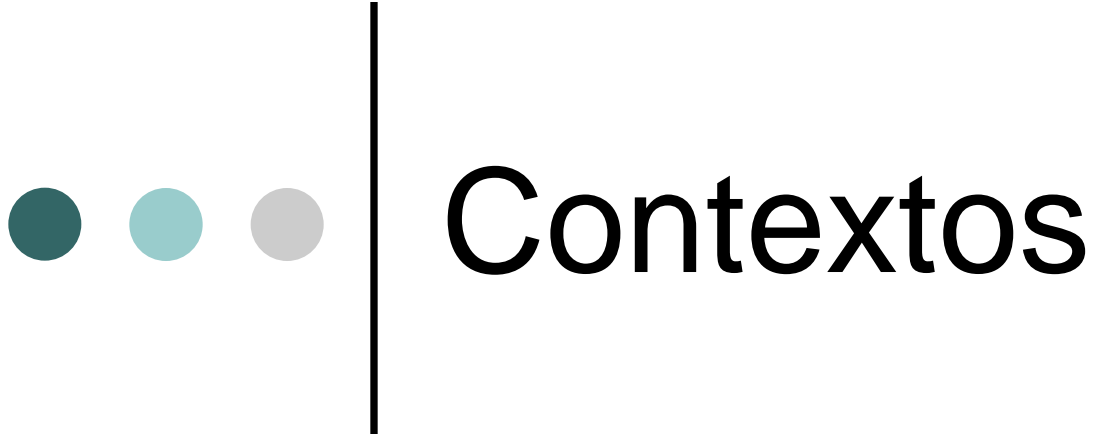


# Compresión de Datos

Contextos  
PPMC



Contextos



# Contextos

- Se parte de la base de que un texto tiene secuencias que se repiten
  - En un programa escrito en C, luego del carácter “;” es altamente probable un fin de línea.
- Se intenta **predecir** el símbolo siguiente. Para cada carácter, el o los caracteres precedentes son su contexto



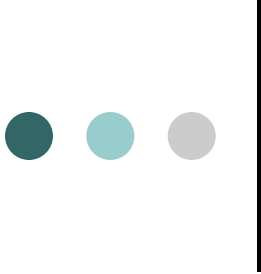
# Contextos

- El orden de un contexto es la cantidad de caracteres precedentes que se toman en cuenta para predecir.
- Orden 0 significa ningún carácter. Orden 1, un carácter.
- Se suele denotar de la siguiente forma:  $O(n)$ , para Orden  $n$ .
- Para cada carácter o conjunto de ellos, se almacena una tabla de probabilidades asociada.



# Contextos

- Ejemplo:
  - Fuente: DATATA
  - Orden: 1
- Utilizando solamente los caracteres de la fuente.
- Utilizando todos los caracteres ASCII.



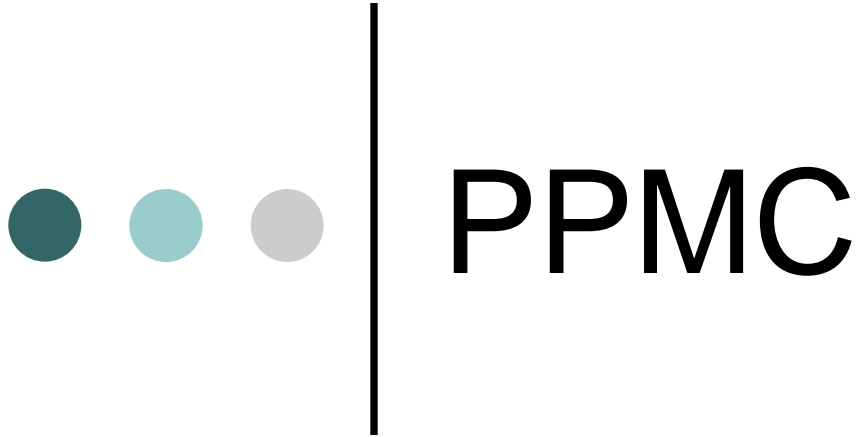
# Ventajas de utilizar contextos

- Utilizar distintas tablas de probabilidades según el contexto ayuda a predecir mejor
- Una mejor predicción equivale a mejores modelos probabilísticos
- Mejores modelos equivalen a una menor emisión en bits.



# Desventajas de utilizar contextos

- $O(0)$  necesita una tabla de 256 posiciones (una para cada carácter)
- $O(1)$  necesita una tabla de  $256 \times 256$  posiciones.  $O(2)$ , una tabla de  $256 \times 256 \times 256$ .
- El tamaño de las tablas crece exponencialmente.
- Si la fuente no tiene patrones repetidos (no es estructurada), utilizar contextos no mejora la compresión



Prediction by Partial Matching versión C





# PPMC

- No utiliza un solo contexto para comprimir, sino varios para una mejor predicción (hasta 6).
- Compresor híbrido
  - Estadístico: Por tener base en el compresor aritmético.
  - Predictor: Por la utilización de contextos.



# PPMC

- Problema de utilización de contextos al inicializar todos los caracteres en forma equiprobable.
- En el contexto del carácter “Q”, una “U” debería tener probabilidad casi 1.
- Luego de 20 veces de encontrado el patrón, la probabilidad es de, solamente,  $21/277$ .
- Es muy ineficiente, aprende lento.



# PPMC

- Si inicializamos los caracteres en frecuencia 0, no tenemos probabilidades para emitir.
- Se agrega un carácter especial (carácter de ESCAPE), que se inicializa con frecuencia 1.
- Cuando no se encuentra el carácter a emitir en el contexto actual, se emite un ESCAPE y se actualiza la tabla.



# PPMC

- Se usan varios contextos.
- Se comienza desde el contexto de mayor orden y si el carácter a emitir se encuentra con probabilidad 0, se emite un ESCAPE y se va al contexto inmediatamente menor.
- El último contexto es el -1. Contiene a los 256 caracteres ASCII y al EOF con frecuencia 1 (fija para todos).



# PPMC

## ○ Exclusión:

- Se excluye del contexto actual, los caracteres leídos en contextos anteriores.
- Si pasé por un contexto donde hubo caracteres con frecuencia mayor a 0 y no los utilicé, tampoco los necesito en la tabla del contexto actual.
- Aplicar exclusión mejora el nivel de compresión.



# PPMC

- Para la emisión se utiliza un compresor aritmético que aprovecha las distribuciones de probabilidad del contexto en el que estoy parado para emitir.
- Comienza con el intervalo inicial, y va emitiendo y normalizando siempre el mismo para cada contexto (no se utilizan intervalos diferentes para cada uno)



# PPMC

- Cuando se pasa de la emisión de un contexto a otro se hace “zoom” sobre el intervalo actual para la emisión correspondiente, y, para el próximo contexto, se utiliza el subintervalo obtenido (normalizado).
- La salida en bits es la del compresor aritmético. Se debe agregar información de control sobre el mayor orden de contexto utilizado.



# PPMC

- Ejemplo:
  - Fuente: DIVIDIDOS
  - Orden: 2





# PPMC

- Descompresión:

- Se toma la tira de bits emitida por el aritmético y se descomprime según éste.
- Con cada emisión que se obtiene, se van actualizando las tablas de los contextos, de la misma forma en que se comprimió.



# PPMC

- Mientras mas alto el orden, mejor nivel de compresión?
- La experiencia dicta que el óptimo está entre orden 4 y 5.
- Órdenes mas grandes agregan overhead por la emisión de ESCAPES.