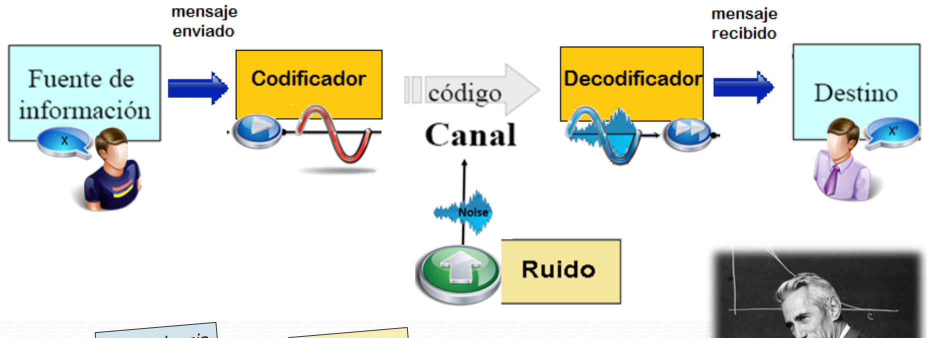
TEORÍA DE LA INFORMACIÓN

Tratamiento probabilístico de la Información

Fuentes de información

INGENIERÍA DE SISTEMAS 2017

TRANSMISIÓN DE INFORMACIÓN



redundancia en los datos

Medida de información

técnicas de compresión

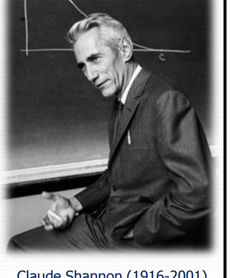
ruido en la transmisión

pérdida de información

capacidad del canal

decodificación automática

errores en los datos recibidos



Claude Shannon (1916-2001)

FUENTES DE INFORMACIÓN

Fuente de información → $s_1\ s_1\ s_2\ s_1\ s_3\ s_2\ \ldots$

en cada instante *t* genera un símbolo s_i elegido dentro del conjunto de símbolos posibles, según su probabilidades de emisión (*modelo*)

Clasificación:

Según el rango de valores que pueden generar
- F continua
- F. discreta

Según la relación entre sus símbolos
- F. sin memoria (o de memoria nula)
- F. con memoria (de orden *k)*

FUENTES SIN MEMORIA

- Los símbolos son estadísticamente independientes
- La fuente se describe mediante el conjunto de símbolos posibles s_i y la probabilidad de ocurrencia $p(s_i)$ asociada a cada uno
- Las probabilidades de emisión son constantes en el tiempo

Ejemplo

Una fuente S emite símbolos de un conjunto $\{s_0, s_1, s_2\}$ con probabilidades: 0.3, 0.5, 0.2, respectivamente

$P(S=s_0) = 0.3$
$P(S=s_1) = 0.5$
$P(S=s_2) = 0.2$

$X = \{s_0, s_1, s_2\}$ $\quad p(x) = \{0.3, 0.5, 0.2\}$

FUENTES CON MEMORIA

- Los símbolos son estadísticamente dependientes
- La probabilidad de emitir un símbolo en *t+1* está condicionada por la ocurrencia de los símbolos anteriores (en *t*, *t-1*, …)

					i	j
$S_{(0)}$	$S_{(1)}$	$S_{(2)}$	…	$S_{(t-1)}$	$S_{(t)}$	$\mathbf{S_{(t+1)}}$

$p_{j/i}$

cadena de Markov

- **Hipótesis de Markov:** la probabilidad de emitir un símbolo depende SÓLO del símbolo emitido en el instante anterior

$$P\left[S(t+1)=j \middle/ S(0)=s_0, S(1)=s_1, \ldots, S(t)=i\right] = P\left[S(t+1)=j \middle/ S(t)=i\right]$$

PROCESOS ESTOCÁSTICOS

- Una fuente con memoria se puede modelar como un proceso estocástico markoviano de *random walk*

- Proceso estocástico → fenómeno que evoluciona en el tiempo de manera impredecible, desde el punto de vista del observador

Ejemplos

Señales (sonido, imágenes, video, …)

clima (viento, humedad, temperatura,…)

Procesos económicos (acciones, stocks, PBI,…)

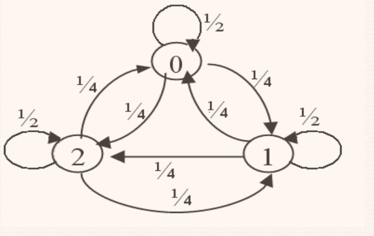
evolución de poblaciones, superficies, …

FUENTES CON MEMORIA (ORDEN 1)

- En una fuente markoviana (memoria de orden 1) la probabilidad de emitir un símbolo en *t+1* depende sólo del símbolo emitido en *t*
- Si las probabilidades de transición son constantes en el tiempo → proceso markoviano *homogéneo*

Ejemplo

una fuente markoviana emite 3 símbolos 0, 1, 2 con las siguientes probabilidades de transición:



Grafo de transición de estados:
- Nodos: estados (símbolos)
- Arcos: transiciones posibles

FUENTES CON MEMORIA (ORDEN 1)

Las probabilidades de transición de estados (condicionales) se pueden disponer en una matriz de pasaje o de transición:

$$M_{j/i} = M = \begin{array}{c} \\ s_1 \\ s_2 \\ \vdots \\ s_j \\ \vdots \end{array} \begin{array}{c} \begin{matrix} s_1 & s_2 & \quad & s_i & \dots \end{matrix} \\ \begin{bmatrix} p_{1/1} & p_{1/2} & \cdots & p_{1/i} & \cdots \\ p_{2/1} & p_{2/2} & \cdots & p_{2/i} & \cdots \\ \vdots & \vdots & \ddots & \vdots & \ddots \\ p_{j/1} & p_{j/2} & \cdots & p_{j/i} & \ddots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \end{array}$$

probabilidades condicionales

$$\sum_j P(S(t+1)=s_j \,/\, S(t)=s_i) = \sum_j p_{j/i} = 1$$

VECTORES DE ESTADO

Para una fuente markoviana:

- $V_t \rightarrow$ Vector de estado en *t* o distribución de prob. de emisión en *t*
- Conocido V_t, se puede obtener $\mathbf{V_{t+1} = M_{j/i} . V_t}$ ($M_{j/i}$ constante)
- Dada la condición inicial $\mathbf{V}_0$: $\mathbf{V_1 = M . V_0}$

$$\mathbf{V_2 = M. V_1 = M. (M.V_0) = M^2. V_0}$$

....

$$\mathbf{V_{t+1} = M. V_t = M^{t+1} . V_0}$$

Ejemplo

$$M = \begin{array}{c|ccc|} & 0 & 1 & 2 \\ 0 & \frac{1}{2} & \frac{1}{4} & \frac{1}{4} \\ 1 & \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \\ 2 & \frac{1}{4} & \frac{1}{4} & \frac{1}{2} \end{array}$$

	V0	V1	V2	V3	
$p_0(t)$	1	1/2	3/8	11/32	...
$P_1(t)$	0	1/4	5/16	21/64	...
$p_2(t)$	0	1/4	5/16	21/64	...

VECTORES DE ESTADO
por muestreo computacional

Generar la emisión de un símbolo por muestreo o simulación computacional:

Para el ejemplo anterior:

condiciones iniciales (*según el problema*):

$$V_0 = \begin{bmatrix} 1/3 \\ 1/3 \\ 1/3 \end{bmatrix} \quad V0_{acum} = \begin{bmatrix} 1/3 \\ 2/3 \\ 1 \end{bmatrix}$$

$$M = \begin{array}{c|ccc|} & 0 & 1 & 2 \\ 0 & \frac{1}{2} & \frac{1}{4} & \frac{1}{4} \\ 1 & \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \\ 2 & \frac{1}{4} & \frac{1}{4} & \frac{1}{2} \end{array}$$

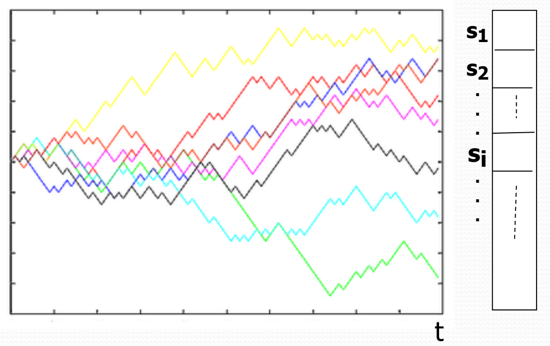
$$M_{acum} = \begin{array}{c} \\ 0 \\ 1 \\ 2 \end{array} \begin{array}{c} \begin{array}{ccc} 0 & 1 & 2 \end{array} \\ \begin{bmatrix} ½ & ¼ & ¼ \\ ¾ & ¾ & ½ \\ 1 & 1 & 1 \end{bmatrix} \end{array}$$

```
PrimerSimb ()
{ r=rand()
  for(i=0 to 2)
    if (r < V0acum[i])
      return i; }
```

```
Sig_dado_Ant (col)
{ r=rand()
  for(i=0 to 2)
    if ( r < Macum[i, col] )
      return i }
```

VECTORES DE ESTADO
por muestreo computacional

Una secuencia de símbolos (trayectoria) representa <u>un</u> posible mensaje emitido por la fuente estocástica



$$P(S_t) = s_i \approx \frac{\#si\ en\ t}{\#trayectorias}$$

VECTORES DE ESTADO
por muestreo computacional

Para el ejemplo anterior:

```
PrimerSimb ()
{ r=rand()
   for(i=0 to 2)
     if (r<V0acum[i])
        return i; }
```

```
Sig_dado_Ant (col)
{r=rand()
  for(i=0 to 2)
    if ( r < Macum[i, col] )
       return i }
```

```
converge (A[ ], B[] )
{ for (i=0 to 2)
   { if (abs(A[i]-B[i]) > ξ )
        return FALSE }
     return TRUE }
```

```
Calcular_Vector_Estado (int t)
{
  emisiones= [0,0,0] // cantidad de cada si
  Vt= [0,0,0]   // Vector de estado actual
  Vt_ant= [-1,0,0] // Vector de estado anterior
  tray=0  //cant. trayectorias
  pasos  //cant. transiciones o pasos
  while( not converge (Vt, Vt_ant) and (tray > T_MIN) )
  {  s=PrimerSimb ();
     for (pasos= 0 to t)
        s=Sig_dado_Ant (s)
     emisiones[s]++
     tray++
     Vt_ant ← Vt
     Vt ← emisiones/tray
  }
   return Vt
}
```

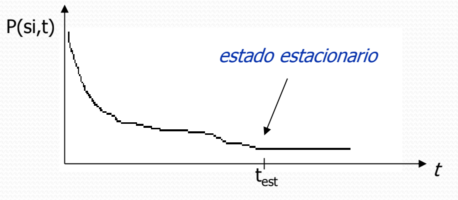
ESTADO ESTACIONARIO

- Los vectores de estado van variando con la evolución del proceso de emisión de símbolos, hasta estabilizarse o estacionarse
  → **estado estacionario (V*)**

Ejemplo

	V0	V1	V2	V3		V*	V*
$p_0(t)$	1	1/2	3/8	11/32	...	0,333336	0,333334
$P_1(t)$	0	1/4	5/16	21/64	...	0,333332	0,333333
$p_2(t)$	0	1/4	5/16	21/64	...	0,333332	0,333333

Comprobación: V* = M.V*



El estado estacionario es independiente de las condiciones iniciales (V_0)

ESTADO ESTACIONARIO

Analíticamente:

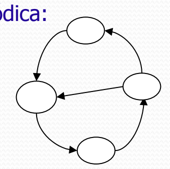
$$V^* = M \cdot V^* \rightarrow \boxed{\begin{array}{l}(M - I)\,V^* = 0 \ ^{(1)} \\ \sum v_i^* = 1 \ ^{(2)}\end{array}}$$

Sistema de ecuaciones
(1) eliminar una de las ecuaciones
(2) Incorporar necesariamente

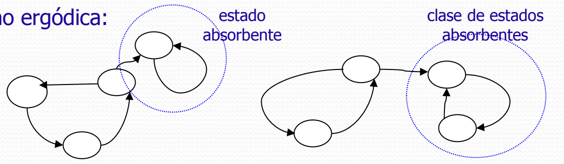
Condiciones de existencia de V*:

- Conjunto finito de estados (F. discreta)
- Fuente ergódica (todos los estados son alcanzables desde otro estado - no hay estados o clases absorbentes)

F. ergódica:



F. no ergódica:



Ejemplo

$$M = \begin{array}{c|ccc|} & 0 & 1 & 2 \\ 0 & \frac{1}{2} & \frac{1}{4} & \frac{1}{4} \\ 1 & \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \\ 2 & \frac{1}{4} & \frac{1}{4} & \frac{1}{2} \end{array}$$

$$(M - I)\, V^* = 0$$
$$\sum v_i^* = 1 \quad (3')$$

$$M - I = \begin{array}{c|ccc|l} & 0 & 1 & 2 & \\ 0 & -\frac{1}{2} & \frac{1}{4} & \frac{1}{4} & (1) \\ 1 & \frac{1}{4} & -\frac{1}{2} & \frac{1}{4} & (2) \\ 2 & \frac{1}{4} & \frac{1}{4} & -\frac{1}{2} & (3) \end{array}$$

(1) $-\frac{1}{2}\, v0^* + \frac{1}{4}\, v1^* + \frac{1}{4}\, v2^* = 0$

(2) $\frac{1}{4}\, v0^* - \frac{1}{2}\, v1^* + \frac{1}{4}\, v2^* = 0$

(3) Se puede descartar

(3´) $v0^* + v1^* + v2^* = 1$

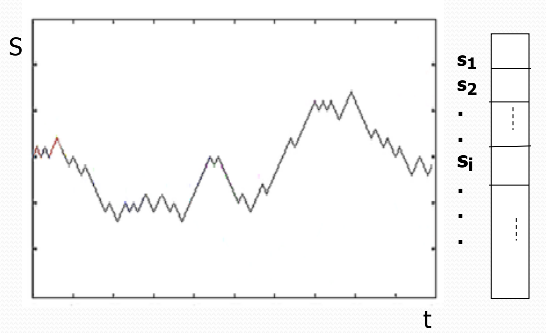
Resolviendo el sistema de ecuaciones en este caso se obtiene :

$$V^* = \begin{bmatrix} 1/3 \\ 1/3 \\ 1/3 \end{bmatrix}$$

Desarrollo en clase

ESTADO ESTACIONARIO
por muestreo computacional

Cuando un proceso se encuentra en estado estacionario es suficiente con simular una única trayectoria:



$$P(S^* = s_i) \approx \frac{\#s_i}{\#transiciones}$$

VECTORES DE ESTADO
por muestreo computacional

Para el ejemplo anterior:

```
PrimerSimb ()
{ r=rand()
   for(i=0 to 2)
      if (r<V0_acum[i])
         return i; }
```

```
Sig_dado_Ant (col)
{r=rand()
  for(i=0 to 2)
    if ( r < Macum[i, col] )
       return i }
```

```
converge (A[ ], B[] )
{ for (i=0 to 2)
   { if (abs(A[i]-B[i]) > ξ )
        return FALSE }
   return TRUE }
```

```
Calcular_Vector_Estacionario
{
  emisiones= [0,0,0] // cantidad de cada s_i
  V*= [0,0,0]   // Vector estac. actual
  V_ant= [-1,0,0] // Vector estac. anterior
  pasos=0  //cant. transiciones o pasos

  s=PrimerSimb ();
  while( not converge (Vt, Vt_ant) and (pasos >P_MIN) )
  {
     s=Sig_dado_Ant (s)
     emisiones[s]++
     pasos++
     V*_ant ← V*
     V* ← emisiones/pasos
  }
   return V*
}
```

BIBLIOGRAFÍA

Abramson N., **Teoría de la Información y Codificación**, Ed. Paraninfo, 1981

Papoulis A., Pillai U., **Probability Random Variables and Stochastic Processes**, 4th.ed., McGraw-Hill, 2002

Shannon C., Weaver W., **Teoría Matemática de la Comunicación**, Ed.Forja, 1981