

4 de enero de 2021

Una planta de reciclaje ha decidido introducir un sistema automático de clasificación de envases. Utilizando una cámara hiperespectral pretende distinguir latas de refresco y tetrabriks respecto de otros envases. Un análisis preliminar ha identificado que los dos atributos más relevantes para la clasificación son el área que ocupan los envases en la imagen y la intensidad que devuelve la cámara para una determinada longitud de onda (ambas medidas normalizadas entre 0 y 1). La Figura 4.1 (izquierda) muestra las medidas obtenidas para un conjunto de datos de entrenamiento.

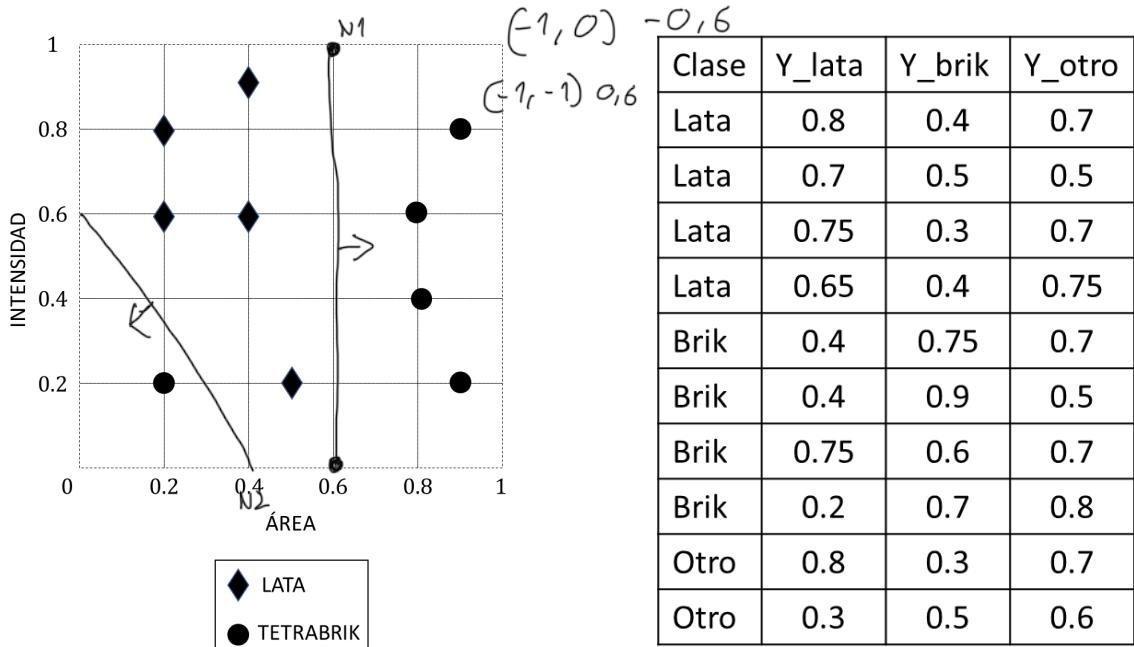


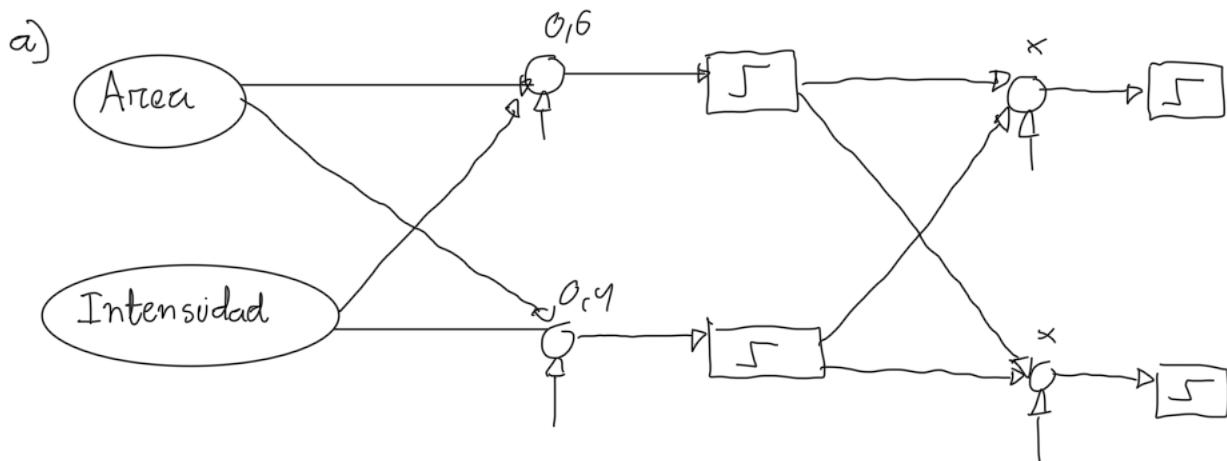
Figura 4.1 Datos de entrenamiento para la red neuronal (izquierda) y resultados obtenidos para el algoritmo de la competencia (derecha)

Se pide:

- Diseña la red neuronal más sencilla que, considerando en la capa de salida una neurona para cada clase, clasifique correctamente latas y tetrabriks. Nota: puedes elegir las funciones de activación sin considerar el algoritmo de optimización.
- Calcula la salida que devolverá la red para una muestra con medidas: (Área = 0.6, Intensidad = 0.4).

La empresa dispone de otro algoritmo de clasificación que incluye la clase “otros” en su diseño. Para un conjunto de datos de prueba se han obtenido los resultados mostrados en la parte derecha de la Figura 4.1. Se pide:

- Calcula la matriz de confusión suponiendo que la clase ganadora es la de mayor salida.
- Calcula la precisión, el *recall* y el *f1-score* para la clase “LATA”.



La primera neurona N1 tiene parámetros

$$\vec{v_1} = p_2 - p_1 = \begin{pmatrix} 0, 6, 1 \end{pmatrix} - \begin{pmatrix} 0, 6, 0 \end{pmatrix} = \begin{pmatrix} 0, 1, 1 \end{pmatrix}$$

$$w_1^T p_1 + b_1 = 0 \Rightarrow b_2 = -w_{1x} p_1 x - w_{1y} p_1 y = -1 \cdot 0,6 - 0 = -0,6$$

La segunda neurona N2 tiene parámetros:

$$\vec{V}_2 = r^4 - r^3 = (0, 0.6) - (0.6, 0) = (-0.6, 0.6)$$

$$w_2 = (-1, -1) \quad b_2 = 0.6$$

Salida de la primera corriente de 2º corriente

| | | |
|-----------|------------|------------|
| \square | y_1 0 | y_2 0 |
| • | 0 | 1 |
| | 1 | 0 |

Diseñar 2 neuronas (N_3, N_4)

$$w_3 x + b_3 \quad w_4 x + b_4$$

$w_3 x + y_1 + w_4 y_2 + b_3 \geq 0 \quad \text{Si}$

$w_4^T y + b_4 \quad \text{Si}$

Nº clasificar tetrabrick

$$w_4x + y_1 + w_4y \cdot y_2 + b_4 \geq 0 \text{ si } \Delta \\ \leq 0 \text{ si } \varnothing$$

$$\left\{ \begin{array}{l} w_4y + b_4 > 0 \\ w_4x + b_4 > 0 \\ b_4 = -0,5 \end{array} \right.$$

b) $w_1 = (1, 0)$ $b_1 = -0,6$
 $w_2 = (-1, -1)$ $b_2 = 0,6$
 $w_3 = (-1, 1)$ $b_3 = 0,5$
 $w_4 = (1, 1)$ $b_4 = -0,5$

$$\text{Escalón}(z_1) = \begin{cases} 1 & \text{si } z_1 \geq 0 \\ 0 & \text{si } z_1 < 0 \end{cases}$$

Entrada $(0,6, 0,4)$

$$w_1 = 1 \cdot 0,6 + 0 \cdot 0,4 - 0,6 = 0 \rightarrow 1$$

$$w_2 = -1 \cdot 0,6 - 1 \cdot 0,4 + 0,6 = -0,4 \rightarrow 0$$

Salida capa 2
Entrada $\boxed{(1, 0)}$

$$w_3 = (-1 \cdot 1 - 1 \cdot 0 + 0,5) = -0,5 \rightarrow 0$$

$$w_4 = (1 \cdot 1 + 1 \cdot 0 - 0,5) = 0,5 \rightarrow 1$$

Salida capa 2 $\rightarrow (0,1)$ Tetrabrick

c) clase predicha \rightarrow Lata Tetrabrick

| clase real | Lata | Brick | Otro |
|------------|------|-------|------|
| Lata | 3 | 0 | 1 |
| Brick | 1 | 2 | 1 |
| OTRO | 1 | 0 | 1 |

Matriz de confusión para lata

clase predicha \rightarrow LATA

| clase real | LATA | NO LATA |
|------------|------|---------|
| LATA | 3 | 1 |
| NO LATA | 2 | 4 |

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} = \frac{3}{3+2} = \frac{3}{5} = 0,6$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} = \frac{3}{3+1} = \frac{3}{4} = 0,75$$

$$\text{F1-Score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} = 2 \cdot \frac{\frac{3}{5} \cdot \frac{3}{4}}{\frac{3}{5} + \frac{3}{4}} = 0,66$$

9 de septiembre de 2021

Zara-Uber está desarrollando un módulo de percepción para conducción autónoma en su flota de vehículos. Para ello, ha recogido muestras de coches, ciclistas y peatones utilizando una cámara convencional. En una primera fase, se plantea realizar un reconocimiento de estos tres tipos de elementos a través de su color y su relación de aspecto en la imagen. La Figura 4.1 muestra las medidas obtenidas.

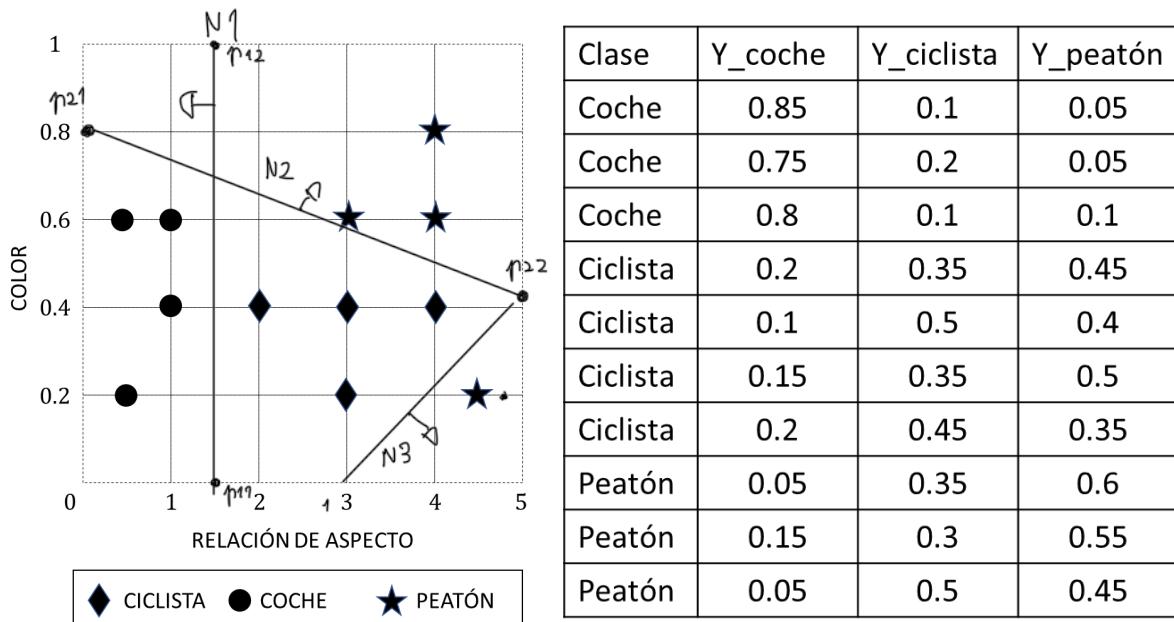


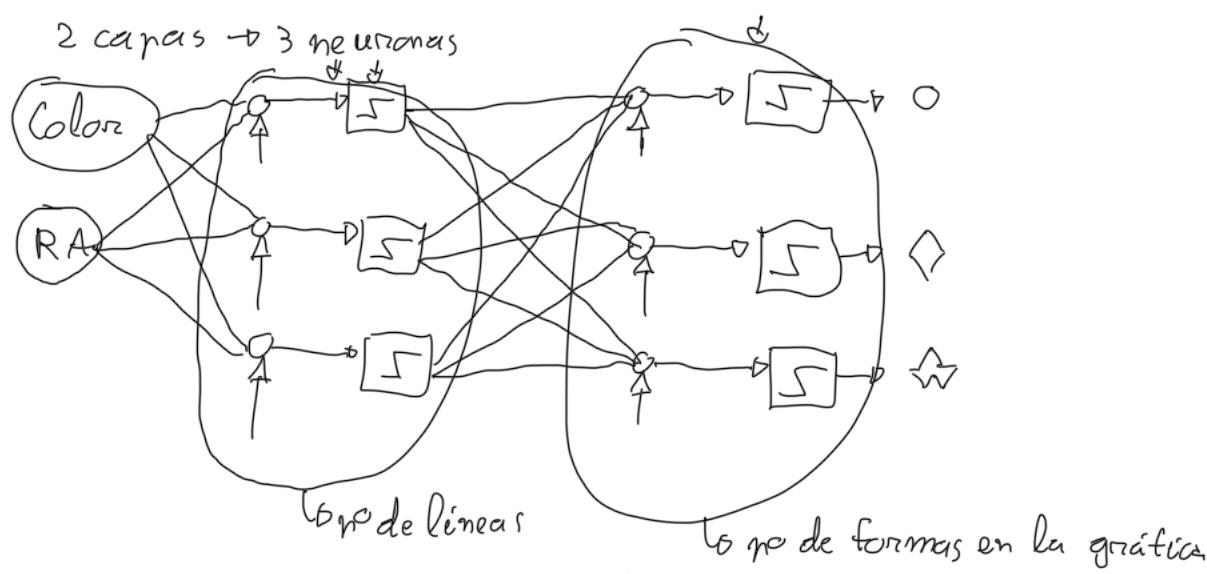
Figura 4.1 Datos de entrenamiento para la red neuronal (izquierda) y resultados obtenidos para el segundo algoritmo (derecha)

Se pide:

- Diseña la red neuronal más sencilla que, considerando en la capa de salida una neurona para cada clase, clasifique correctamente coches, ciclistas y peatones. Nota: puedes elegir las funciones de activación sin considerar el algoritmo de optimización.

La empresa dispone de otro algoritmo de clasificación para un conjunto de datos de prueba ha obtenido los resultados mostrados en la parte derecha de la Figura 4.1. Se pide:

- Dibuja la curva *precision-recall* para la clase "Ciclista" utilizando los umbrales de clasificación $\tau = (0.3, 0.4, 0.9)$.



La primera neurona N1 tiene parámetros:

$$\vec{V}_1 = p_{12} - p_{11} = (1.5, 1) - (1.5, 0) = (0, 1)$$

$w_1^x + p_{12x} + b_1 = \boxed{b_1 = -w_1x p_{12x} - w_1y p_{12y} = 1 \cdot 1.5 - 0 \cdot 1 = 1.5}$

$w_1 = (1, 0)$

$w_1 = (-1, 0)$

$$w_1^x + p_{12x} + b_1 = \boxed{b_1 = -w_1x p_{12x} - w_1y p_{12y} = 1 \cdot 1.5 - 0 \cdot 1 = 1.5}$$

La segunda neurona N2 tiene parámetros:

$$\vec{V}_2 = p_{22} - p_{21} = (5, 0.4) - (0, 0.8) = (5, -0.4)$$

$w_2^x + p_{22x} + b_2 = \boxed{b_2 = -w_2x p_{22x} - w_2y p_{22y} = -0.4 \cdot 5 - 5 \cdot 0.4 = -4}$

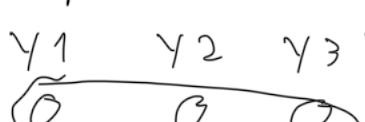
$w_2 = (-0.4, -5)$

$w_2 = (+0.4, +5)$

$$w_2^x + p_{22x} + b_2 = \boxed{b_2 = -w_2x p_{22x} - w_2y p_{22y} = -0.4 \cdot 5 - 5 \cdot 0.4 = -4}$$

N3

Salida de la primera capa = Entrada 2^a capa



\rightarrow Según el número de neuronas

Diseñan 3 \rightarrow el número de formas
neuronas (N_4, N_5, N_6)

N_4 clasificar ciclistas ◊

$$w_4x \cdot y_1 + w_4y \cdot y_2 + w_4z \cdot y_3 + b_4 \geq 0 \text{ si } \diamond$$

$w_4x + b_4$

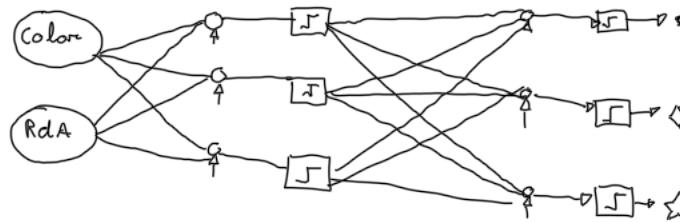
$\leq 0 \text{ si } 0 \star$

$$\left. \begin{array}{l} \diamond \quad y = (0, 0, 0) \\ \circ \quad y = (1, 0, 0) \\ \star \quad y = (0, 1, 0) \\ \quad \quad y = (0, 0, 1) \end{array} \right\} \begin{array}{l} b_4 \geq 0 \\ w_4x + b_4 \leq 0 \\ w_4y + b_4 \leq 0 \\ w_4z + b_4 \leq 0 \end{array} \quad \begin{array}{l} b_4 = 1 \\ w_4x = -2 \\ w_4y = -2 \\ w_4z = -2 \end{array}$$

N_5 clasifican

Puedo resolver el problema con una capa? No
¡son los datos...

Necesito 2 capas
2^a capa → 3 neuronas



La primera neurona N1 tiene parámetros:

$$\vec{v}_1 = p_2 - p_1 = (1.5, 1) - (1.5, 0) = (0, 1) \quad \begin{array}{l} \rightarrow w_1 = (1, 0) \\ \boxed{w_1 = (-1, 0)} \end{array}$$

$$w_1^T + p_2 + b_1 \Rightarrow [b_1 = -w_{1x}p_{2x} - w_{1y}p_{2y} = 1 \cdot 1.5 - 0 \cdot 1 = 1.5]$$

La segunda neurona N2 tiene parámetros:

$$\vec{v}_2 = p_2 - p_1 = (5, 0.4) - (0, 0.8) = (5, -0.4) \quad \begin{array}{l} \rightarrow w_2 = (-0.4, -5) \\ \boxed{w_2 = (0.4, 5)} \end{array}$$

$$w_2^T + p_2 + b_2 \Rightarrow [b_2 = -w_{2x}p_{2x} - w_{2y}p_{2y} = -0.4 \cdot 5 - 5 \cdot 0.4 = -4]$$

La tercera neurona N3 tiene parámetros:

$$\vec{v}_3 = p_2 - p_1 = (5, 0.4) - (3, 0) = (2, 0.4) \quad \begin{array}{l} \rightarrow w_3 = (0.4, 2) \\ \boxed{w_3 = (0.4, -2)} \end{array}$$

$$w_3^T + p_2 + b_3 \Rightarrow [b_3 = -w_{3x}p_{2x} - w_{3y}p_{2y} = -5 \cdot 0.4 + 2 \cdot 0.4 = -1.2]$$

Salida de la primera capa = Entrada 2^a capa

| | y_1 | y_2 | y_3 |
|---|-------|-------|-------|
| ◇ | 0 | 0 | 0 |
| ○ | 1 | 0 | 0 |
| ★ | 0 | 1 | 0 |
| | 0 | 0 | 1 |

Diseñar 3 neuronas (N^4, N^5, N^6)

$$\begin{array}{lll} w^4 b^4 & w^5 b^5 & w^6 b^6 \\ w^4_x w^4_y & w^5_x w^5_y & w^6_x w^6_y \\ w^4_z & w^5_z & w^6_z \end{array}$$

N^4 clasificar ciclistas ◇

$$w^4_x \cdot y_1 + w^4_y \cdot y_2 + w^4_z \cdot y_4 + b^4 \geq 0 \quad \text{si } \diamondsuit$$

$$< 0 \quad \text{si } ○$$

$$w^4_x + b^4$$

$$\begin{array}{ll} \diamondsuit \quad y = (0, 0, 0) & \left\{ \begin{array}{ll} b^4 \geq 0 & b^4 = 1 \\ w^4_x + b^4 \leq 0 & w^4_x = -2 \\ w^4_y + b^4 \leq 0 & w^4_y = -2 \\ w^4_z + b^4 \leq 0 & w^4_z = -2 \end{array} \right. \\ ○ \quad y = (1, 0, 0) & \\ ★ \quad y = (0, 1, 0) & \\ \quad y = (0, 0, 1) & \end{array}$$

N5 clasificar coche O

$$w_{5x} \cdot y_1 + w_{5y} \cdot y_2 + w_{5z} \cdot y_3 \geq 0 \text{ si } O \\ \leq 0 \text{ si } \square \star$$

$$\begin{array}{l} O \quad y = (1, 0, 0) \\ \square \quad y = (0, 0, 0) \\ \star \quad y = (0, 0, 1) \\ \diamond \quad y = (0, 1, 0) \end{array} \left\{ \begin{array}{l} w_{5x} + b_5 \geq 0 \\ b_5 < 0 \\ w_{5y} + b_5 \geq 0 \\ w_{5z} + b_5 \geq 0 \end{array} \right. \boxed{\begin{array}{l} b_5 = -1 \\ w_{5x} = 2 \\ w_{5y} = 0 \\ w_{5z} = 0 \end{array}}$$

N6 clasificar peatón

$$w_{6x} \cdot y_1 + w_{6y} \cdot y_2 + w_{6z} \cdot y_3 \geq 0 \text{ si } \star \\ \leq 0 \text{ si } O \quad \square$$

$$\begin{array}{l} \star \quad y = (0, 0, 1) \\ \diamond \quad y = (0, 1, 0) \\ \square \quad y = (0, 0, 0) \\ O \quad y = (1, 0, 0) \end{array} \Rightarrow \left\{ \begin{array}{l} w_{6y} + b_6 \geq 0 \\ w_{6z} + b_6 \geq 0 \\ b_6 < 0 \\ w_{6x} + b_6 \geq 0 \end{array} \right. \boxed{\begin{array}{l} b_6 = -1 \\ w_{6y} = 2 \\ w_{6z} = 2 \\ w_{6x} = 0 \end{array}}$$

b) curva precisión recall $\tau = (0.3, 0.4, 0.9)$

O, 3

| | | predicha | |
|------|----|----------|----|
| | | +C | -C |
| real | +C | 4 | 0 |
| | -C | 3 | 3 |

$$P = \frac{TP}{TP + FP} \quad R = \frac{TP}{TP + FN}$$

$$P = \frac{4}{4+3} = \frac{4}{7} = 0,57 \quad R = 1$$

O, 4

Predicha

| | | +C | -C |
|------|----|----|----|
| real | +C | 2 | 2 |
| | -C | 1 | 5 |

$$P = \frac{2}{2+3} = 0,67 \quad R = 0,5$$

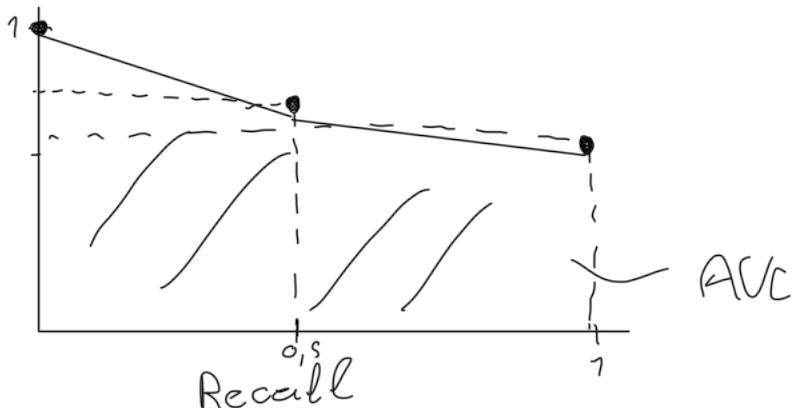
O, 9

Predicha

| | | +C | -C |
|------|----|----|----|
| real | +C | 0 | 4 |
| | -C | 0 | 6 |

$$P = \frac{0}{0+6} = 0 \quad R = \frac{0}{0+4} = 0$$

Precision



20 de enero de 2022

La empresa Farma-tests quiere utilizar una red neuronal para detectar de forma rápida y económica posibles casos positivos de COVID. Para ello, pretende utilizar dos medidas de temperatura, una en la frente (T1) y otra en la garganta (T2) con las que clasificar a las personas en tres niveles de riesgo de tener COVID (BAJO, MEDIO y ALTO). Las medidas obtenidas para varios casos clasificados manualmente se encuentran en la Figura 4.:

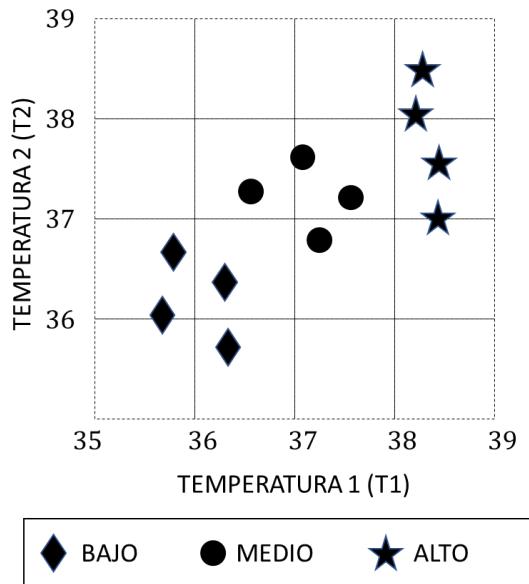


Fig. 4.1. Datos de entrenamiento para la red neuronal.

| Clase | Y_ALTO | Y_BAJO |
|-------|--------|--------|
| ALTO | 0.65 | 0.35 |
| ALTO | 0.75 | 0.25 |
| ALTO | 0.80 | 0.20 |
| ALTO | 0.45 | 0.55 |
| ALTO | 0.65 | 0.35 |
| BAJO | 0.55 | 0.45 |
| BAJO | 0.65 | 0.35 |
| BAJO | 0.45 | 0.55 |

Tabla 4.1 Resultados del tercer algoritmo.

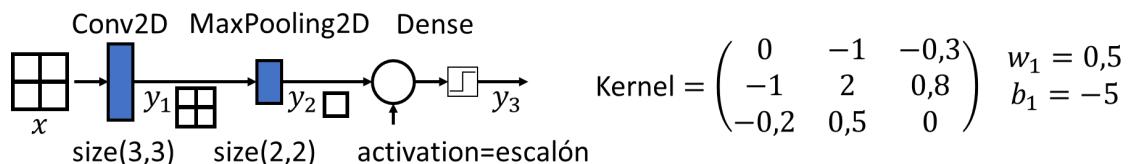


Fig. 4.2. Red convolucional (con pesos obtenidos a la derecha)

Se pide:

- a) Diseña la red neuronal más sencilla que, considerando en la capa de salida una neurona para cada clase, clasifique correctamente todos los datos de entrenamiento.

La empresa dispone de otro algoritmo de clasificación basado en una red neuronal convolucional, mostrada en la Figura 4.2, y de un tercer método cuyos resultados para un conjunto de datos de prueba se muestran en la Tabla 4.1. Se pide:

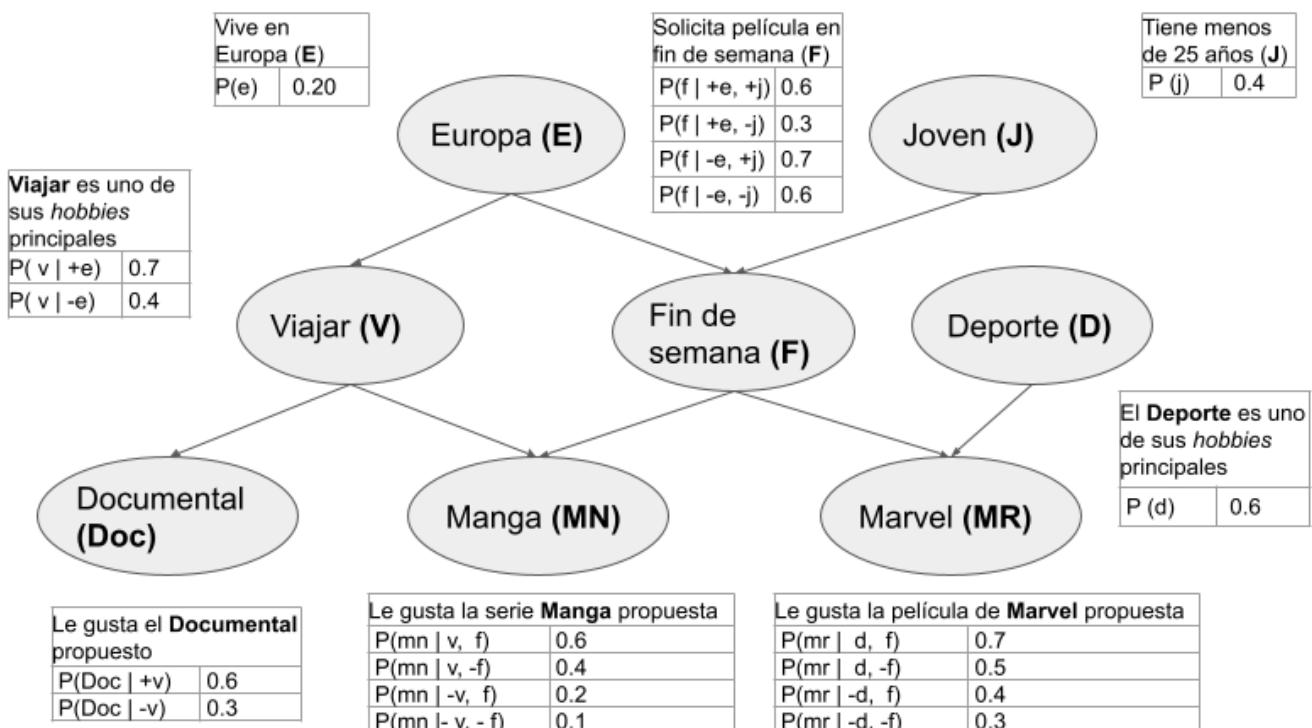
- b) Calcula el valor de la salida que tendrá la red para la entrada $x = \begin{pmatrix} 3 & 5 \\ 1 & 4 \end{pmatrix}$.
- c) Calcula la matriz de confusión, la precisión y el *recall* para la clase ALTO suponiendo que la clase ganadora es la de mayor salida.

20 de enero de 2022

Una plataforma de *streaming* utiliza estadísticas sobre las costumbres y hobbies de sus usuarios en distintas zonas geográficas para recomendar “*¿qué ver?*” cuando el usuario solicita una recomendación.

Teniendo en cuenta la red Bayesiana que modela la información disponible, *¿qué debería recomendar ver el sistema a un usuario que se conecta desde Europa, en fin de semana, y en su perfil de usuario pone que NO le gustan las series Manga?* Es decir, estima la probabilidad de que le guste ver una película de Marvel y la de que le guste ver un documental.

Estima estas dos probabilidades aplicando el algoritmo de eliminación de variables. Detalla claramente el proceso seguido y los motivos claros si decides ignorar alguna variable.



9 de septiembre de 2022

Zara-Uber vuelve a las andadas con su sistema de percepción para conducción autónoma. Quiere reentrenar su sistema para que también sea capaz de reconocer a las mascotas en las imágenes. Ha decidido utilizar el área que ocupa cada elemento en la imagen y su relación de aspecto como atributos. La Figura 4.1 muestra las medidas obtenidas para las tres clases que quiere reconocer.

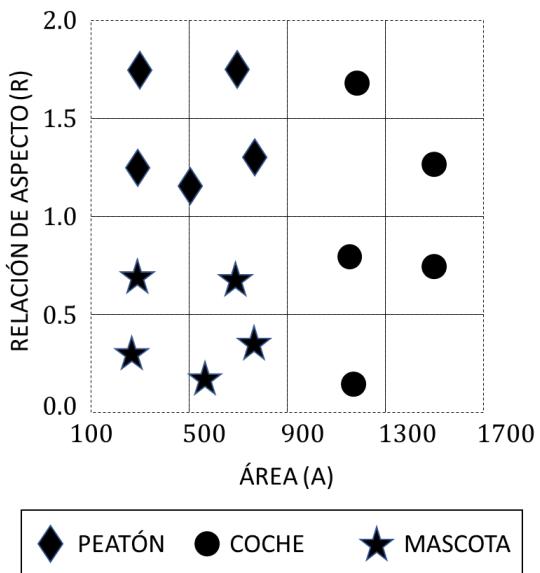


Fig. 4.1. Datos de entrenamiento para la red neuronal.

| Clase | Y_coche | Y_mascota | Y_peatón |
|---------|---------|-----------|----------|
| Coche | 0.3 | 0.5 | 0.2 |
| Coche | 0.7 | 0.2 | 0.1 |
| Coche | 0.8 | 0.1 | 0.1 |
| Mascota | 0.2 | 0.3 | 0.5 |
| Mascota | 0.1 | 0.6 | 0.3 |
| Mascota | 0.1 | 0.8 | 0.1 |
| Mascota | 0.5 | 0.2 | 0.3 |
| Peatón | 0.1 | 0.1 | 0.8 |
| Peatón | 0.2 | 0.3 | 0.5 |
| Peatón | 0.1 | 0.5 | 0.4 |

Tabla 4.1 Resultados de YOLO.

Se pide:

- Diseña la red neuronal más sencilla que, considerando en la capa de salida una neurona para cada clase, clasifique correctamente todos los datos de entrenamiento.
- Calcula la salida de la red neuronal que has diseñado para una persona con medidas $(A,R) = (600, 0.9)$.

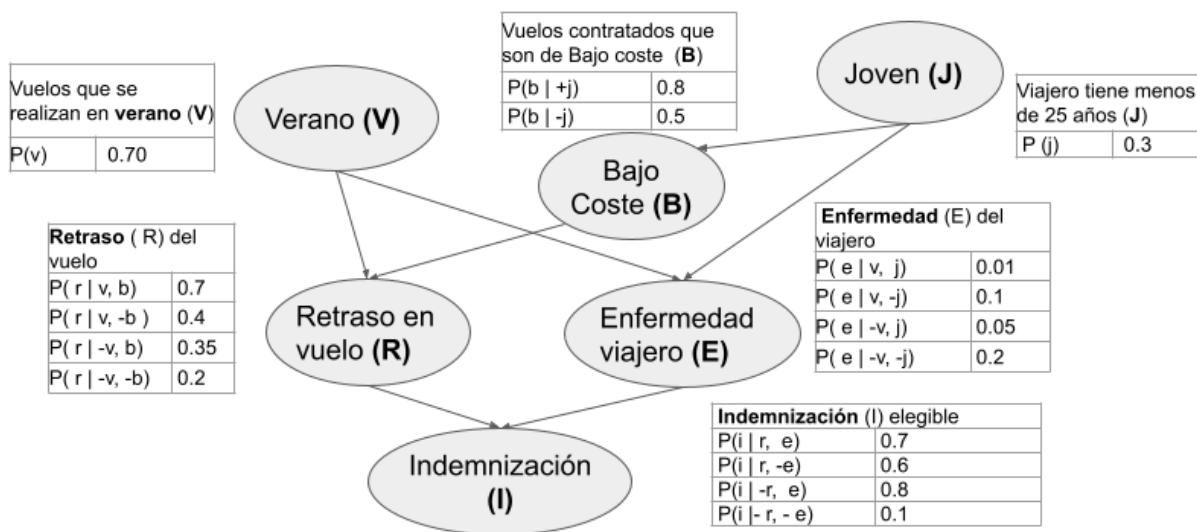
La empresa también quiere estudiar la calidad del sistema de percepción YOLO, como posible alternativa en su flota de vehículos. Los resultados de dicho sistema se muestran en la Tabla 4.1. Se pide:

- Calcula la matriz de confusión suponiendo que la clase ganadora es la de mayor salida.
- Calcula el F1-score de la clase Mascota.

9 de septiembre de 2022

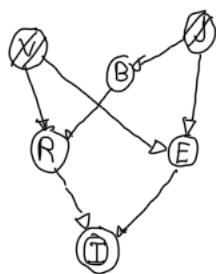
Una agencia de viajes quiere estudiar si debiera cobrar el seguro de viaje más alto según la franja de edad de los viajeros (≤ 25 años, o no). Para ello estudia las probabilidades de tener que pagar indemnización a los viajeros en distintas circunstancias. Responde a estas dos preguntas:

- Este verano, ¿sería lógico cobrar más a los viajeros menores de 25 años o no? (es decir, estima la probabilidad de indemnización en verano, ¿es mayor si el pasajero es menor de 25, o si no lo es?).
Estima estas probabilidades aplicando el algoritmo de eliminación de variables. Detalla claramente el proceso seguido y los motivos claros si decides ignorar alguna variable.
- ~~Si sabemos que el viajero no está enfermo, ¿sigue influyendo la edad en la probabilidad de indemnización? (es decir, ¿son I y J dependientes en este caso?)~~



$$d P(\pm I+J, +V) ?$$

$$P(\mp I-J, +V) ?$$



Paso 1: Ignorar nodos no ancestros de Q y E

Paso 2: Eliminar R, B, y E

Empezamos por B (lo unimos con R)

| V | J | R | B | $P(RI+V, B)$ | $P(BI+J)$ | |
|---|---|---|---|--------------|-----------|--------|
| + | + | + | + | 0.7 | 0.8 | |
| + | + | - | - | 0.4 | 0.2 | |
| + | + | - | + | 0.3 | 0.8 | |
| + | + | - | - | 0.6 | 0.2 | |
| | | | | | 0.56 | 0.64 |
| | | | | | 0.08 | |
| | | | | | 0.24 | |
| | | | | | 0.12 | 0.36 |

| R | $P(RI+V, +J)$ |
|---|---------------|
| + | 0.64 |
| - | 0.36 |

Eliminamos R (lo unimos con I)

$P(R, I | +V, +J, E)$

| +V | +J | E | I | R | $P(RI+V, +J)$ | $P(I R, E)$ | |
|----|----|---|---|---|---------------|---------------|--------|
| + | + | + | + | + | 0.64 | 0.7 | |
| + | + | - | - | - | 0.36 | 0.8 | |
| + | - | + | - | - | 0.64 | 0.2 | |
| - | + | + | + | - | 0.36 | 0.2 | |
| - | + | - | - | - | 0.64 | 0.6 | |
| - | - | + | + | - | 0.36 | 0.1 | |
| - | - | - | - | - | 0.64 | 0.4 | |
| | | | | | 0.36 | 0.036 | 0.42 |
| | | | | | | 0.384 | |
| | | | | | | 0.036 | |
| | | | | | | 0.256 | 0.58 |
| | | | | | | 0.324 | |

| E | $P(I +V, +J, E)$ |
|---|--------------------|
| + | 0.736 |
| - | 0.264 |
| - | 0.42 |
| - | 0.58 |

Eliminamos E (lo unimos con I)

$P(I, EI | +V, +J)$

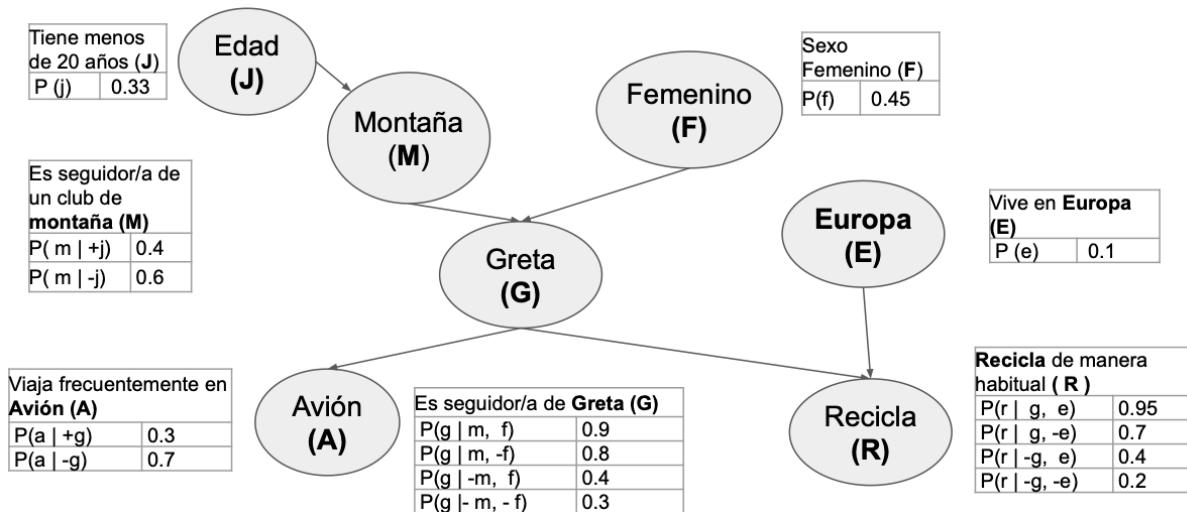
| +V | +J | I | E | $P(EI+V, +J)$ | $P(I EI, +J)$ | |
|----|----|---|---|---------------|-----------------|-----------|
| + | + | + | + | 0.736 | 0.01 | |
| + | + | - | - | 0.42 | 0.99 | |
| - | + | + | - | 0.264 | 0.01 | |
| - | - | - | - | 0.58 | 0.99 | |
| | | | | 0.00736 | 0.00736 | 0.42316 |
| | | | | 0.4158 | | |
| | | | | 0.00264 | 0.00264 | 0.52684 |
| | | | | 0.5742 | 0.5742 | |

| I | $P(I +V, +J)$ |
|---|-----------------|
| + | 0.42316 |
| - | 0.57684 |

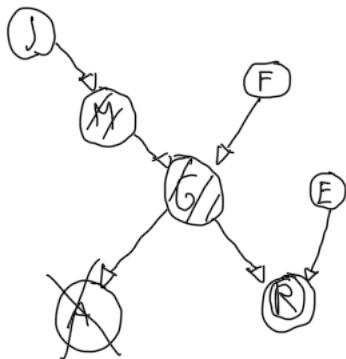
16 de enero de 2023

Se está haciendo un estudio sobre reciclaje analizando información obtenida a través de Twitter. La información recopilada está reflejada en la siguiente red Bayesiana ¿Podrías responder a las siguientes preguntas a partir de ella?

- a) ~~¿Según este modelo, influye el hecho de viajar en avión de manera habitual (A) con el hecho de reciclar de manera habitual (R)?~~
- b) ¿Cuál es la probabilidad de que alguien que es seguidor de Greta y de un club de montaña recicle de manera habitual?
- Completa las probabilidades que faltan en las tablas de la red para tener las distribuciones de probabilidad completas.
 - Escribe (de manera formal) la probabilidad que vas a calcular para responder esta pregunta, indicando que variable o variables son “query” o “evidencia”
 - Calcula dicha probabilidad con el algoritmo de eliminación de variables. Explica en detalle los pasos que vas realizando.



$$b) P(R|G, M)$$



Paso 1 Ignorar nodos no ancestros de G y E \rightarrow A

Paso 2 Eliminar J, F, E

Empezamos por E:

| $P(R, E G)$ | | | $P(R G)$ |
|---------------|---|---|------------|
| G | R | E | |
| + | + | + | 0.725 |
| + | + | - | 0.275 |
| + | - | + | |
| + | - | - | |

| $P(R G, E)$ | $P(E)$ | |
|---------------|--------|--|
| 0.45 | 0.1 | |
| 0.7 | 0.4 | |
| 0.05 | 0.1 | |
| 0.3 | 0.9 | |

$$P(R | G)$$

$$0.725 \\ 0.275$$

$$P(G | R) 0.809$$

$$P(M) 0.534$$

Paso 3 Unir todo y marginalizar

| G | $P(R G)$ | $P(G M)$ | $P(M)$ |
|---|------------|------------|--------|
| + | 0.725 | 0.809 | 0.534 |
| - | 0.275 | 0.804 | 0.534 |

$$+ = 0.725 \\ - = 0.275$$

16 de enero de 2023

La marca de relojes Rolex, para diferenciarse de su competidora Casio te ha encargado que diseñas y evalúes diferentes algoritmos de clasificación. En el primer caso quiere utilizar un perceptrón que, en base al tamaño de la esfera del reloj y su precio distinga los dos tipos de relojes. La Figura 4.1 muestra los datos que se han recopilado para entrenar dicha red (**iOJO! el escalado de la figura no es lineal**):

- a) Diseña la red neuronal más sencilla que, considerando en la capa de salida una neurona para cada clase, clasifique correctamente todos los relojes de las dos marcas

En segundo lugar, ha utilizado una CNN ya existente que ha testeado con diferentes imágenes. Los resultados de dicho clasificador se muestran en la Tabla 4.1.

- b) Dibuja la curva precisión-recall utilizando como umbrales $\tau = (0, 0.2, 0.6, 0.8, 1)$

Por último, ha diseñado una CNN con la arquitectura y pesos que se muestran en la Figura 4.2. La capa “flatten” genera el vector a partir de la matriz siguiendo un orden de filas.

- c) Calcula la salida de dicha red para una entrada $x = \begin{pmatrix} 10 & 20 \\ 5 & 2 \end{pmatrix}$

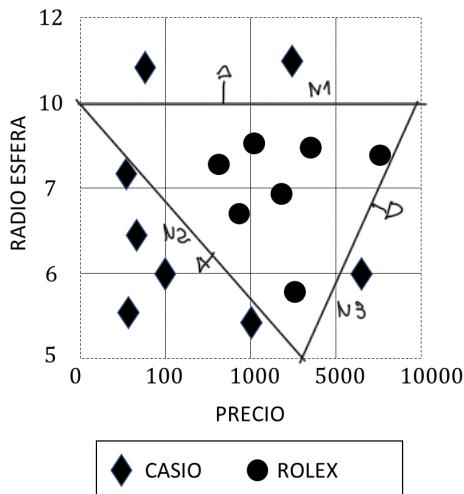


Fig. 4.1. Datos de entrenamiento para la primera red neuronal.

| Clase | Y_rolex | Clase | Y_rolex |
|-------|---------|-------|---------|
| Rolex | 0.85 | Casio | 0.75 |
| Rolex | 0.75 | Casio | 0.05 |
| Rolex | 0.95 | Casio | 0.55 |
| Rolex | 0.4 | Casio | 0.65 |
| Rolex | 0.4 | Casio | 0.1 |
| Rolex | 0.9 | Casio | 0.25 |

Tabla 4.1 Resultados de la segunda red neuronal.

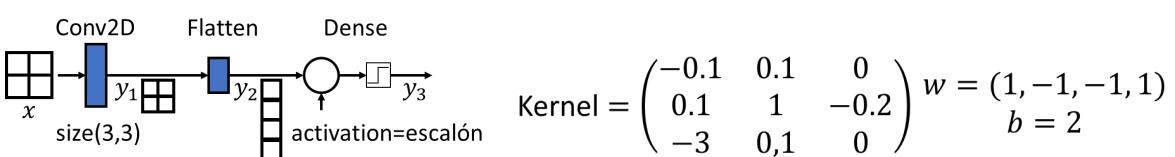
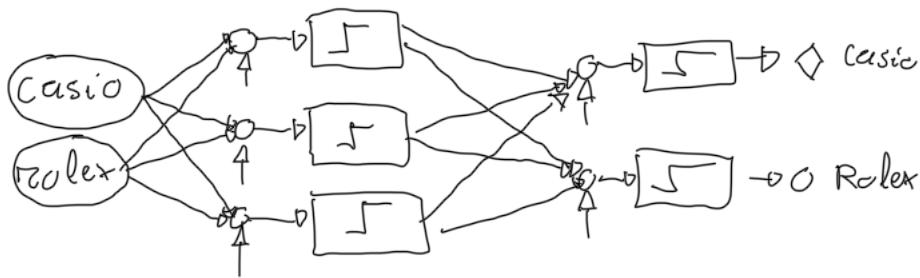


Fig. 4.2. Arquitectura y pesos de la tercera red neuronal

Necesito 2 capas \rightarrow 3 neuronas



Calculo pesos N1

$$\vec{v}_1 = p_2 - p_1 = (10.000, 10) - (0, 10) = (10.000, 0) \xrightarrow[w_1^T(0, -10.000)]{w_1^T(0, 10.000)} \\ w_1^T p_2 + b_1 = 0 \Rightarrow [b_1 = -w_1 x p_2 x - w_1 y p_2 y = -10.000 \cdot 10 = -100.000]$$

Calculo pesos N2

$$\vec{v}_2 = p_2 - p_1 = (3.500, 0) - (0, 10) = (3.500, -10) \xrightarrow[w_2^T(10, 3500)]{w_2^T(-10, -3500)} \\ w_2^T p_2 + b_2 = 0 \Rightarrow [b_2 = -w_2 x p_2 x - w_2 y p_2 y = -3500 \cdot (-10) = 35.000]$$

Calculo pesos N3

$$\vec{v}_3 = p_2 - p_1 = (10.000, 10) - (3.500, 0) = (6.500, 10) \xrightarrow[w_3^T(-10, 6500)]{w_3^T(10, 6500)} = w_3$$

$$w_3^T p_2 + b_3 = 0 \Rightarrow [b_3 = -w_3 x p_2 x - w_3 y p_2 y = -10 \cdot 10.000 - 10 \cdot 6.500 = -35.000]$$

Salida de la primera capa = Entrada 2^a capa

| | y_1 | y_2 | y_3 |
|------------|-------|-------|-------|
| \diamond | 1 | 0 | 0 |
| | 0 | 0 | 1 |
| | 0 | 1 | 0 |
| \circ | 0 | 0 | 0 |

Diseñar 3 neuronas (N_4, N_5)

$$w_{4x} w_{4y} \quad w_{5x} w_{5y} \\ w_{4z} \quad w_{5z}$$

N_4 clasificar Casio

$$w_{4x} \cdot y_1 + w_{4y} \cdot y_2 + w_{4z} \cdot y_3 + b_4 \geq 0 \quad \text{Si } y \in \mathbb{Q} \\ \leq 0 \quad \text{Si } y \notin \mathbb{Q}$$

$$\left. \begin{array}{l} \text{Y } Y = (1, 0, 0) \\ \text{Y } Y = (0, 1, 0) \\ \text{Y } Y = (0, 0, 1) \\ \text{O } Y = (0, 0, 0) \end{array} \right\} \quad \left. \begin{array}{l} w_{4x} + b_4 \geq 0 \\ w_{4y} + b_4 \geq 0 \\ w_{4z} + b_4 \geq 0 \\ b_4 \leq 0 \end{array} \right\} \quad \begin{array}{l} b_4 = -1 \\ w_{4x} = 1 \\ w_{4y} = 1 \\ w_{4z} = 1 \end{array}$$

N_5 clasificar Cache

$$\begin{array}{ll} b_{5x} \geq 0 & b_{5z} = 0 \\ w_{5x} + b_{5y} \leq 0 & w_{5x} = -1 \\ \vdots & w_{5y} = -1 \\ w_{5z} \leq 0 & w_{5z} = -1 \end{array}$$

| Clase | YRolex | 0 | 0.2 | 0.6 | 0.8 | 1 |
|-------|--------|---|-----|-----|-----|---|
| Rolex | 0.85 | * | * | * | * | |
| Rolex | 0.75 | * | * | * | | |
| Rolex | 0.45 | * | * | * | * | |
| Rolex | 0.4 | * | * | * | * | |
| Rolex | 0.4 | * | * | * | * | |
| Casio | 0.75 | * | * | * | | |
| Casio | 0.05 | * | | | | |
| Casio | 0.55 | * | * | | | |
| Casio | 0.65 | * | * | * | | |
| Casio | 0.1 | * | | | | |
| Casio | 0.25 | * | * | | | |

| Recall | | Prize dichiar | |
|--------|---|---------------|----|
| | | 0 | 1 |
| Recall | 0 | +R | -R |
| | 1 | 6 | 0 |

$$P = \frac{6}{6+6} = \frac{1}{2} \quad R = \frac{6}{6+0} = 1$$

| Recall | | Prize dichiar | |
|--------|-----|---------------|----|
| | | 0.2 | 1 |
| Recall | 0.2 | +R | -R |
| | 1 | 6 | 0 |

$$P = \frac{6}{6+4} = \frac{6}{10} = \frac{3}{5} \quad R = \frac{6}{6+0} = 1$$

| Recall | | Prize dichiar | |
|--------|-----|---------------|----|
| | | 0.6 | 1 |
| Recall | 0.6 | +R | -R |
| | 1 | 4 | 2 |

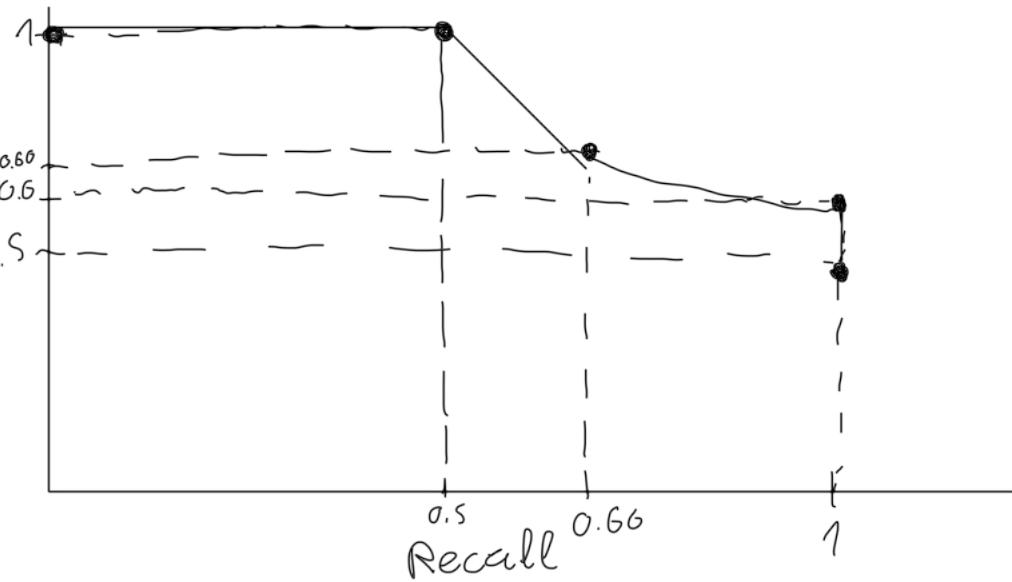
$$P = \frac{4}{4+2} = \frac{2}{3} \quad R = \frac{4}{4+0} = \frac{2}{3}$$

| Recall | | Prize dichiar | |
|--------|-----|---------------|----|
| | | 0.8 | 1 |
| Recall | 0.8 | +R | -R |
| | 1 | 3 | 3 |

$$P = \frac{3}{3+0} = 1 \quad R = \frac{3}{3+3} = \frac{1}{2}$$

| Recall | | Prize dichiar | |
|--------|---|---------------|----|
| | | 1 | 1 |
| Recall | 1 | +R | -R |
| | 1 | 0 | 6 |

$$P = \frac{0}{0+6} = 0 \quad R = \frac{0}{6} = 0$$



$$y_1 = \begin{pmatrix} 6.5 & 5.6 \\ 6.2 & 3.5 \end{pmatrix} \quad y_2 = 6.5 - 6.2 - 5.6 + 3.5 + 2 = 0.2 > 0 \rightarrow 1$$

23 de junio de 2023

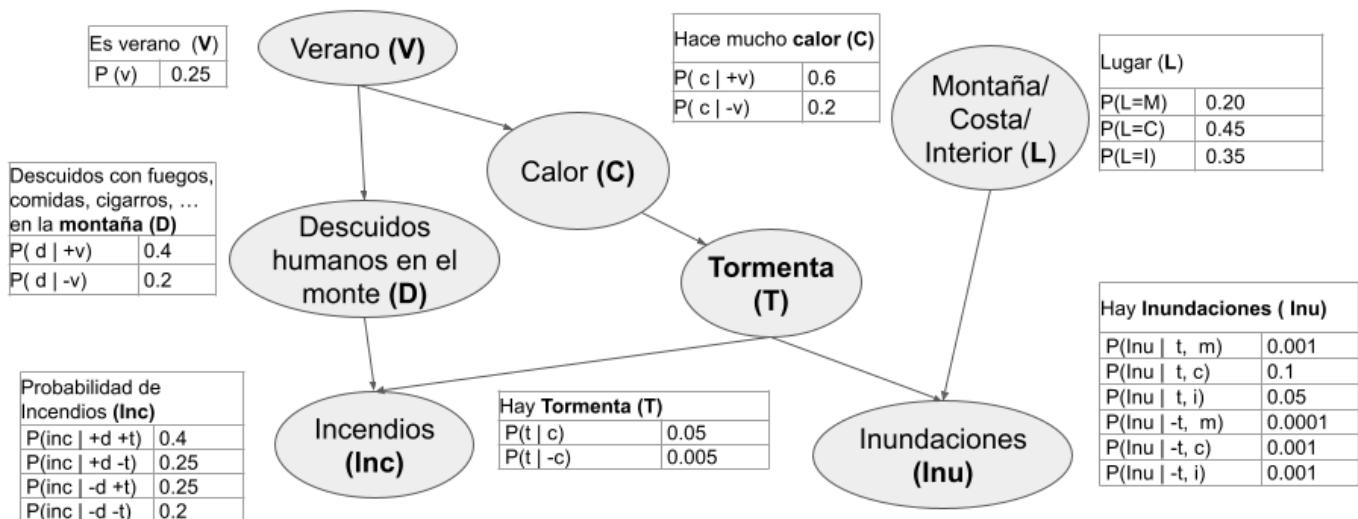
Se está haciendo un estudio sobre incidentes naturales (incendios e inundaciones) para mejorar la distribución de recursos y prevención. ¿Podrías responder a las siguientes preguntas a partir de los datos recopilados en el modelo de esta red?

Completa las probabilidades que faltan en las tablas de la red para tener las distribuciones de probabilidad completas

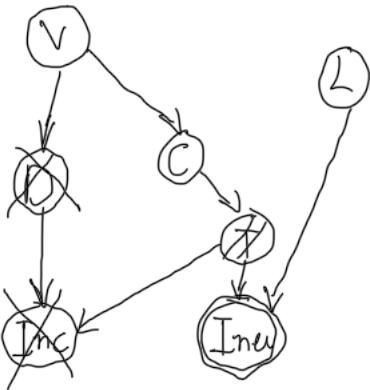
Escribe (de manera formal) la probabilidad que vas a calcular para responder cada una de las siguientes preguntas:

a) ¿Cuál es la probabilidad de inundaciones los días que hay tormenta (T)? Calcula dicha probabilidad con el algoritmo de eliminación de variables. Explica en detalle los pasos que vas siguiendo para realizarlo.

b) Si ocurre un Incendio (Inc) un día con temperaturas muy altas (C), ¿cuál es la probabilidad de que haya habido una tormenta? Calcula dicha probabilidad con el algoritmo de eliminación de variables. Explica en detalle los pasos que vas siguiendo para realizarlo.



a) $P(I_{Inu} | T)$



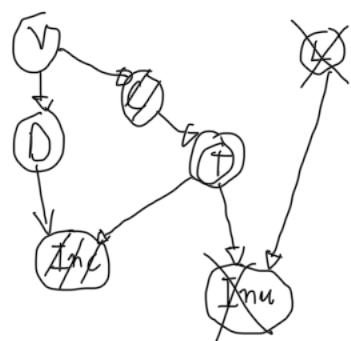
Paso 1: Ignorar nodos no ancestros de Q y E:
D, Inc

Paso 2: Eliminar V, C, L

Empezamos por L (lo unimos con I_{Inu})

| +T | I_{Inu} | L | $P(I_{Inu} L, T)$ | $P(L)$ | | I_{Inu} | $P(I_{Inu} T)$ |
|----|---------|---|---------------------|--------|--------|-----------|------------------|
| + | + | M | 0.001 | 0.20 | 0.0002 | | |
| + | + | C | 0.1 | 0.45 | 0.045 | | |
| + | + | I | 0.05 | 0.35 | 0.0175 | | |
| + | - | M | 0.999 | 0.20 | 0.1948 | | |
| + | - | C | 0.9 | 0.45 | 0.405 | | |
| + | - | I | 0.95 | 0.35 | 0.325 | | |

b) $P(T | +Inc, +C)$



Paso 1: Ignorar nodos no ancestros de Q y E:

L y I_{Inu}

Paso 2: Eliminar V y D

Empezamos por D (lo unimos con Inc)

| +Inc | V | T | D | $P(+Inc D, V, T)$ | $P(+Inc D, T)$ | $P(D V)$ | | Inc | V | T | $P(+Inc V, T)$ |
|------|---|---|---|---------------------|------------------|------------|------|-------|-----|-----|------------------|
| + | + | + | + | 0.9 | 0.9 | 0.16 | 0.31 | + | + | + | 0.31 |
| + | + | - | - | 0.25 | 0.25 | 0.15 | | + | + | - | 0.22 |
| + | - | + | - | 0.25 | 0.25 | 0.1 | 0.22 | + | + | - | 0.28 |
| + | - | - | - | 0.2 | 0.2 | 0.12 | | + | - | + | 0.21 |
| - | + | + | + | 0.4 | 0.4 | 0.2 | 0.28 | + | - | + | 0.21 |
| - | + | - | - | 0.25 | 0.25 | 0.2 | | + | - | - | 0.21 |
| - | - | + | + | 0.25 | 0.25 | 0.2 | 0.28 | + | - | + | 0.21 |
| - | - | - | - | 0.2 | 0.2 | 0.16 | 0.21 | | | | |

• Eliminamos V (lo unimos con Inc, C)

| +INC | +C | T | V | $P(V)$ | $P(+INC T, V)$ | $P(+C V)$ | | T | $P(+INC, +C T)$ |
|------|----|---|---|--------|------------------|-------------|--------|---|-------------------|
| + | + | + | + | 0.25 | 0.31 | 0.6 | 0.0465 | + | 0.0885 |
| + | + | + | - | 0.75 | 0.28 | 0.2 | 0.042 | | |
| + | + | - | + | 0.25 | 0.22 | 0.6 | 0.033 | | |
| + | + | - | - | 0.75 | 0.21 | 0.2 | 0.0315 | - | 0.0645 |

Paso 3: Unir todo y marginalizar

$$0.0885 \cdot 0.05 = 0.004425 \quad \text{Marginalizar}$$

$$0.0645 \cdot 0.95 = 0.061275 \quad \text{Marginalizar}$$

$$\begin{aligned} &0 \\ &0.8832 \end{aligned}$$

23 de junio de 2023

En la parte izquierda de la Figura 1 se muestra una arquitectura de red neuronal con los pesos obtenidos durante el entrenamiento. En la parte derecha de la Figura 1 se muestran los resultados obtenidos evaluando dicha red con cierto conjunto de datos de prueba.

Calcula el valor de la salida de la red neuronal para la siguiente entrada

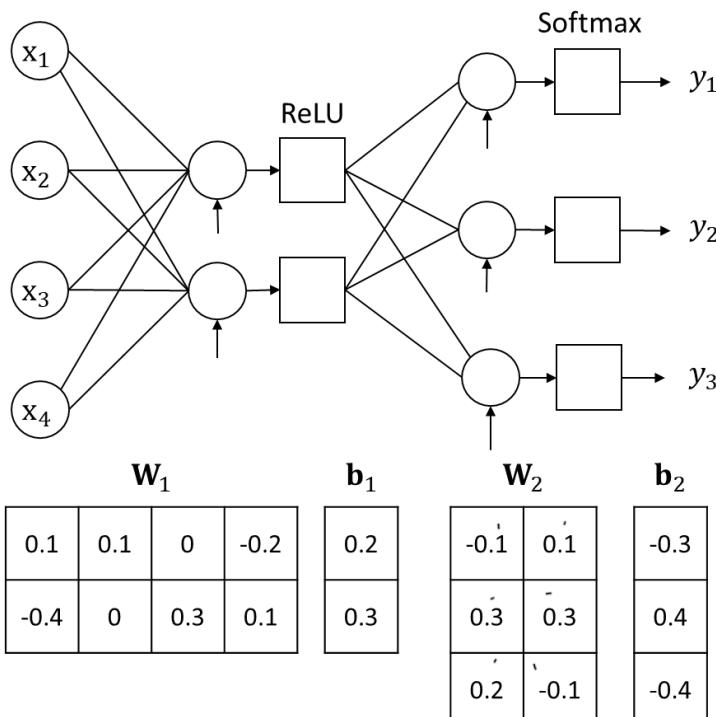
$$\mathbf{x}^T = [x_1, x_2, x_3, x_4]^T = [1, 2, 3, 4]^T$$

Obtén la matriz de confusión de los datos de prueba.

Calcula el *accuracy* de la red para la clase 1.

Dibuja la curva *precision-recall* para la clase 3 con vector de umbrales

$$\tau = [0, 0.2, 0.5, 0.8, 1].$$



| Clase | y_1 | y_2 | y_3 | |
|-------|-------|-------|-------|-----|
| 1 | 0.6 | 0.1 | 0.3 | 0 |
| 1 | 0.8 | 0.1 | 0.1 | 0.1 |
| 1 | 0.4 | 0.1 | 0.5 | 0.5 |
| 1 | 0.9 | 0.0 | 0.1 | 0.9 |
| 1 | 0.5 | 0.1 | 0.4 | 0.5 |
| 2 | 0.3 | 0.3 | 0.4 | 0.3 |
| 2 | 0.1 | 0.4 | 0.5 | 0.9 |
| 2 | 0.0 | 0.9 | 0.1 | 0.9 |
| 2 | 0.2 | 0.6 | 0.2 | 0.9 |
| 2 | 0.1 | 0.2 | 0.7 | 0.9 |
| 3 | 0.0 | 0.2 | 0.8 | 0.9 |
| 3 | 0.1 | 0.2 | 0.7 | 0.9 |
| 3 | 0.1 | 0.1 | 0.8 | 0.9 |
| 3 | 0.0 | 0.1 | 0.9 | 0.9 |
| 3 | 0.3 | 0.1 | 0.6 | 0.9 |

Figura 1. (Izquierdo) Arquitectura y pesos de la red neuronal. (Derecha) Resultados de la red para un conjunto de datos de prueba.

Salida capa 1 y Entrada capa 2

$$\text{ReLU} \begin{pmatrix} 6 \cdot 1 \cdot 1 + 0 \cdot 1 \cdot 2 - 0 \cdot 2 \cdot 4 + 0 \cdot 2 \\ -0 \cdot 4 + 0 \cdot 3 \cdot 3 + 0 \cdot 1 \cdot 4 + 0 \cdot 3 \end{pmatrix} = \text{ReLU} \begin{pmatrix} -0.3 \\ 1.2 \end{pmatrix} = \begin{pmatrix} 0 \\ 1.2 \end{pmatrix}$$

Entrada capa 2 y salida capa 2

$$\text{Softmax} \begin{pmatrix} 0.1 \cdot 1 \cdot 1.2 - 0.3 \\ 0.3 \cdot 1 \cdot 1.2 + 0.4 \\ -0.1 \cdot 1 \cdot 1.2 - 0.4 \end{pmatrix} = \text{Softmax} \begin{pmatrix} -0.18 \\ 0.76 \\ -0.52 \end{pmatrix} = \begin{pmatrix} 0.234 \\ 0.594 \\ 0.167 \end{pmatrix} \quad \begin{aligned} y_1 &= 0.234 \\ y_2 &= 0.594 \\ y_3 &= 0.167 \end{aligned}$$

predicha

| | | 1 | 2 | 3 |
|------|---|---|---|---|
| real | 1 | 4 | 0 | 1 |
| | 2 | 0 | 2 | 3 |
| | 3 | 0 | 0 | 5 |

$$Acc = \frac{TP + TN}{TP + TN + FP + FN} = \frac{4 + 10}{4 + 10 + 0 + 1} = \frac{14}{15} = 0.93$$

predicha

| | | 0 | -3 |
|------|----|----|----|
| real | 0 | 3 | -3 |
| | 3 | 5 | 0 |
| | -3 | 10 | 0 |

0.2

| | | 3 | -3 |
|------|----|---|----|
| real | 3 | 5 | 0 |
| | -3 | 7 | 3 |
| | | | |

$$R = \frac{5}{5+0} = 1 \quad P = \frac{5}{5+7} = 0.42$$

0.5

| | | 3 | -3 |
|------|----|---|----|
| real | 3 | 5 | 0 |
| | -3 | 3 | 7 |
| | | | |

$$R = \frac{5}{5+0} = 1 \quad P = \frac{5}{5+3} = 0.62$$

0.8

| | | 3 | -3 |
|------|----|---|----|
| real | 3 | 3 | 2 |
| | -3 | 0 | 10 |
| | | | |

$$R = \frac{3}{3+2} = 0.67 \quad P = \frac{3}{3+0} = 1$$

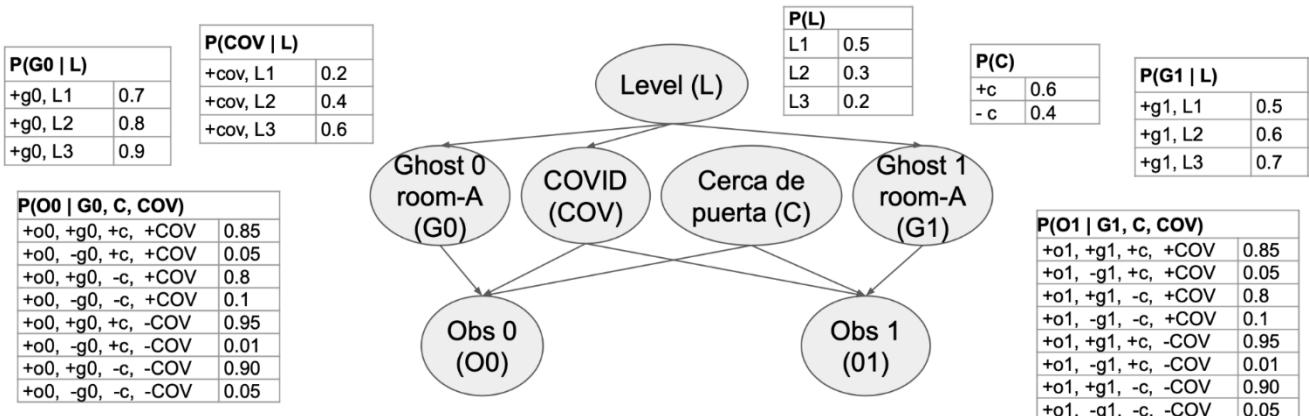
16 de enero de 2024

En un nuevo juego de PACMAN, se puede cambiar entre habitaciones, pero una vez que PACMAN entra a una habitación, tiene que pasar allí un buen rato antes de salir, así que mejor estar seguro de si merece la pena entrar. Los fantasmas pueden aparecer o desaparecer del juego de manera aleatoria, así que, aunque no lo veamos en nuestra habitación, no sabemos a priori si va a estar al otro lado de las puertas.

Para ayudar al jugador, se puede **estimar la probabilidad de que haya alguno de los fantasmas en otra habitación**, y así decidir si entra a ella o no. Por simplificar, asumimos 2 fantasmas, 2 habitaciones (la actual, room-B, y otra, room-A) y las variables:

- **GHOST_0_A (G0)** - variable binaria, GHOST_0 está o no está en la habitación room-A.
- **OBS_0 (O0)** - medida ruidosa de un sensor de olor, específico del ghost-0. Variable binaria que dice si huele o no al GHOST_0.
- **CERCA (C)** - variable binaria que representa si PACMAN está CERCA o no de la puerta. (Si Pacman está lejos, la medida del sensor tendrá más ruido).
- **COVID (COV)** - variable binaria que dice si PACMAN ha cogido el COVID (entonces le falla más el olfato).
- **LEVEL (L)** - nivel del juego en el que estamos. Hay 3 niveles de dificultad: L1, L2 y L3.
- **GHOST_1_A** y **OBS_1**: equivalentes a GHOST_0_A y OBS_0 pero para otro fantasma.

El modelado de UNA HABITACIÓN (room-A) corresponde a esta red Bayesiana:

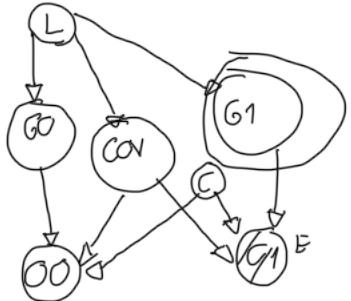


Empezamos a jugar, así que **estamos en el Nivel 1 (L1)**, y sabemos que PACMAN está cerca de la puerta.

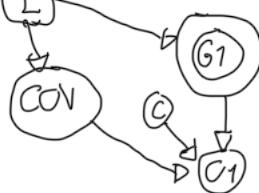
- ¿Cuál es la probabilidad de que el **GHOST_1** esté en “room A” si el sensor **Obs-1** NO detecta el olor de **GHOST_1**? Aplica el algoritmo de Eliminación de Variables. Acuérdate de completar las tablas de los factores que no estén completos y de escribir de manera formal las probabilidades que vas a calcular. Describe brevemente el proceso que sigues para obtener la probabilidad pedida.
- Si además sabemos que el **PACMAN ha cogido el COVID**, ¿cuál es la probabilidad de que **GHOST_0** esté en “room A” si el sensor **Obs-0** NO detecta el olor de **GHOST_0**? Describe brevemente el proceso que sigues para obtener la probabilidad pedida.

$$a) P(G_1 | -O_1, L_1 + C)$$

Paso 1: Ignorar nodos no ancestros de G_1 y E



Paso 2: Eliminar COV



| $P(-O_1, C, G_1, L)$ | | | | $P(O_1 Cov, C, G_1)$ | $P(Cov L)$ | |
|----------------------|-------|-----|-------|----------------------|------------|-------|
| L | O_1 | C | G_1 | $P(O_1 Cov, C, G_1)$ | $P(Cov L)$ | |
| - | - | - | - | 0.15 | 0.2 | 0.03 |
| - | - | - | + | 0.05 | 0.8 | 0.04 |
| - | - | + | = | 0.95 | 0.2 | 0.19 |
| - | - | + | + | 0.99 | 0.6 | 0.742 |

0.0665

0.4334

$$b) P(G_0 | -O_0, +Cov, +C, L_1)$$

| G_0 | $P(G_0 L_1)$ | $P(-O_0 G_0, +Cov, +C)$ | $P(+Cov L_1)$ | $P(+C)$ | $P(L_1)$ | |
|-------|----------------|---------------------------|-----------------|---------|----------|---------|
| + | 0.7 | 0.15 | 0.2 | 0.6 | 0.5 | 0.0063 |
| - | 0.3 | 0.95 | 0.2 | 0.6 | 0.5 | 0.17307 |

16 de enero de 2024

Para mejorar el sistema de puntuación de los trucos de skate en los Juegos Olímpicos de París se va a instalar un sistema de visión por computador que utilice inteligencia artificial para reconocer diferentes trucos. Dado un clip de video, se quiere distinguir cuando el deportista hace tres trucos diferentes: *Ollie*, *Kickflip* y *Hippie-Jump*. Se ha filmado a varios expertos haciendo trucos y se ha medido la altura media del skate (*H*) y la variación de la intensidad de los píxeles en la zona del skate (*G*) durante el truco. Por simplicidad, ambos valores han sido normalizados para que estén entre 0 y 1. La Figura 6.1 muestra los datos recopilados.

- a) Diseña la red neuronal más sencilla que, considerando en la capa de salida una neurona para cada clase, clasifique correctamente todos los trucos del dataset.

También se ha evaluado un modelo de red avanzado basado en Transformers para reconocer los mismos trucos, donde además se ha incorporado la clase *other*, para indicar que el truco realizado no es ninguno de los tres anteriores. La Tabla 6.1 muestra la salida de dicho modelo para un conjunto de datos de prueba.

- b) Calcula la matriz de confusión de dicho modelo suponiendo como clase ganadora la de mayor salida.
 c) Calcula la precisión media del modelo (obtenida como el promedio de la precisión de cada clase).

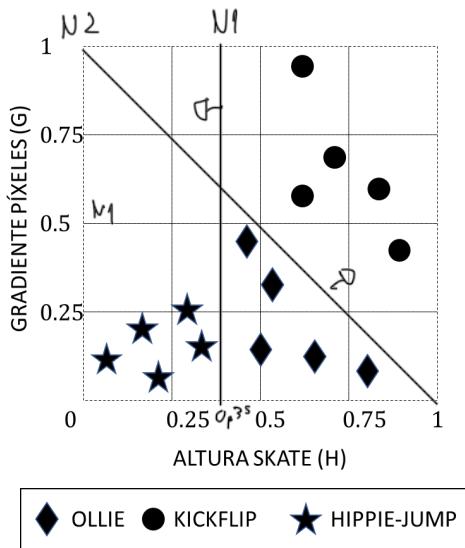
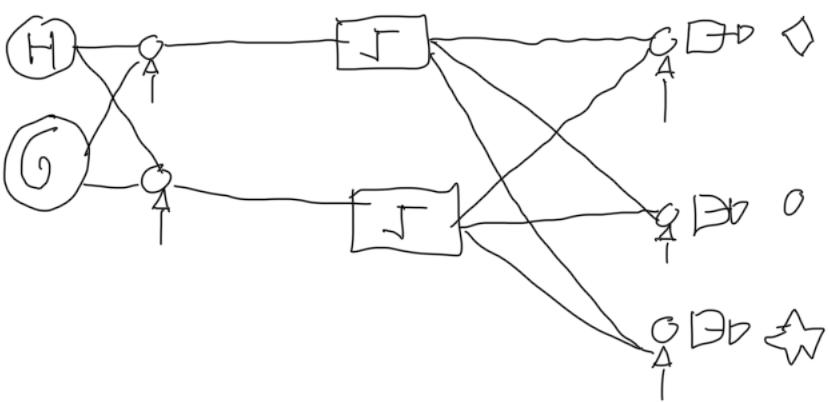


Fig. 6.1. Datos de entrenamiento para la primera red neuronal.

Tabla 6.1 Resultados de la segunda red neuronal.

$$\begin{aligned}
 & \text{predicha} \\
 & \begin{array}{c|c|c|c|c}
 & O & K & H & Ot \\
 \hline
 O & 3 & 2 & 0 & 0 \\
 \hline
 K & 1 & 2 & 0 & 2 \\
 \hline
 H & 0 & 0 & 3 & 2 \\
 \hline
 Ot & 1 & 0 & 1 & 3
 \end{array} \\
 & P_O = \frac{TP}{TP+FP} = \frac{3}{3+2} = \frac{3}{5} \\
 & P_K = \frac{2}{2+2} = \frac{1}{2} \\
 & P_H = \frac{3}{3+1} = \frac{3}{4} \\
 & P_{Ot} = \frac{3}{3+4} = \frac{3}{7} \\
 & P_{\text{media}} = 0,569
 \end{aligned}$$



Calculo pesos N1

$$\vec{v}_1 = p_2 - p_1 = (0, 35, 1) - (0, 35, 0) = (0, 1) \quad \begin{cases} w_1 = (-1, 0) \\ w_2 = (1, 1) \end{cases}$$

$$w_1^T p_2 + b_1 = 0 \Rightarrow [b_1 = -w_1 x \cdot p_2 - w_1 y \cdot p_2 z = 1 \cdot 0, 35 - 0 \cdot 1 = 0, 35]$$

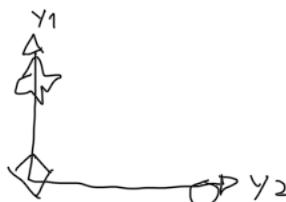
Calculo pesos N2

$$\vec{v}_2 = p_2 - p_1 = (1, 0) - (0, 1) = (-1, 1) \quad \begin{cases} w_2 = (1, 1) \\ w_3 = (-1, -1) \end{cases}$$

$$w_2^T p_2 + b_2 = 0 \Rightarrow [b_2 = -w_2 x \cdot p_2 - w_2 y \cdot p_2 z = -1 \cdot 1 - 1 \cdot 0 = -1]$$

Salida de la primera capa = Entrada 2^a capa

| | y_1 | y_2 |
|---|-------|-------|
| ◇ | 0 | 0 |
| ○ | 0 | 1 |
| ★ | 1 | 0 |



Definir 3 neuronas (N_H, N_S, N_G)

$$\begin{array}{lll} w_4 b_4 & w_5 b_5 & w_6 b_6 \\ w_4 + w_4 y & w_5 + w_5 y & w_6 + w_6 y \end{array}$$

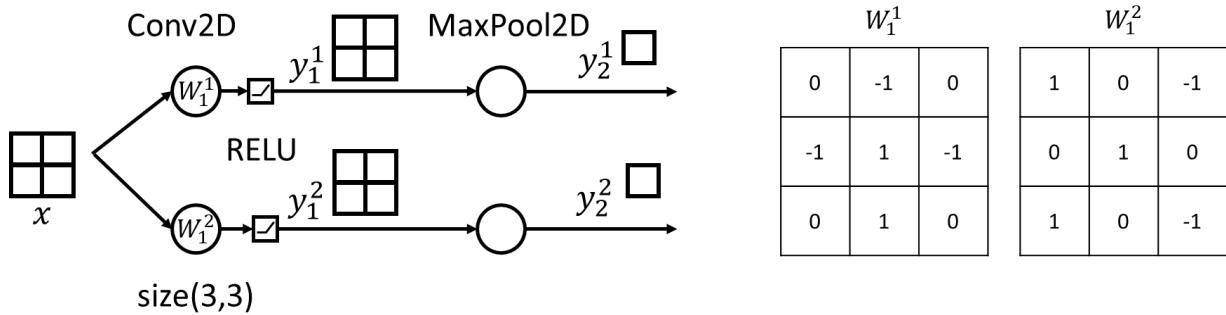
N_H clasificar OLLIE

$$w_4 x \cdot y_1 + w_4 y \cdot y_2 + b_4 \geq 0 \quad \text{si } y \in \diamondsuit \\ < 0 \quad \text{si } y \in \circlearrowleft$$

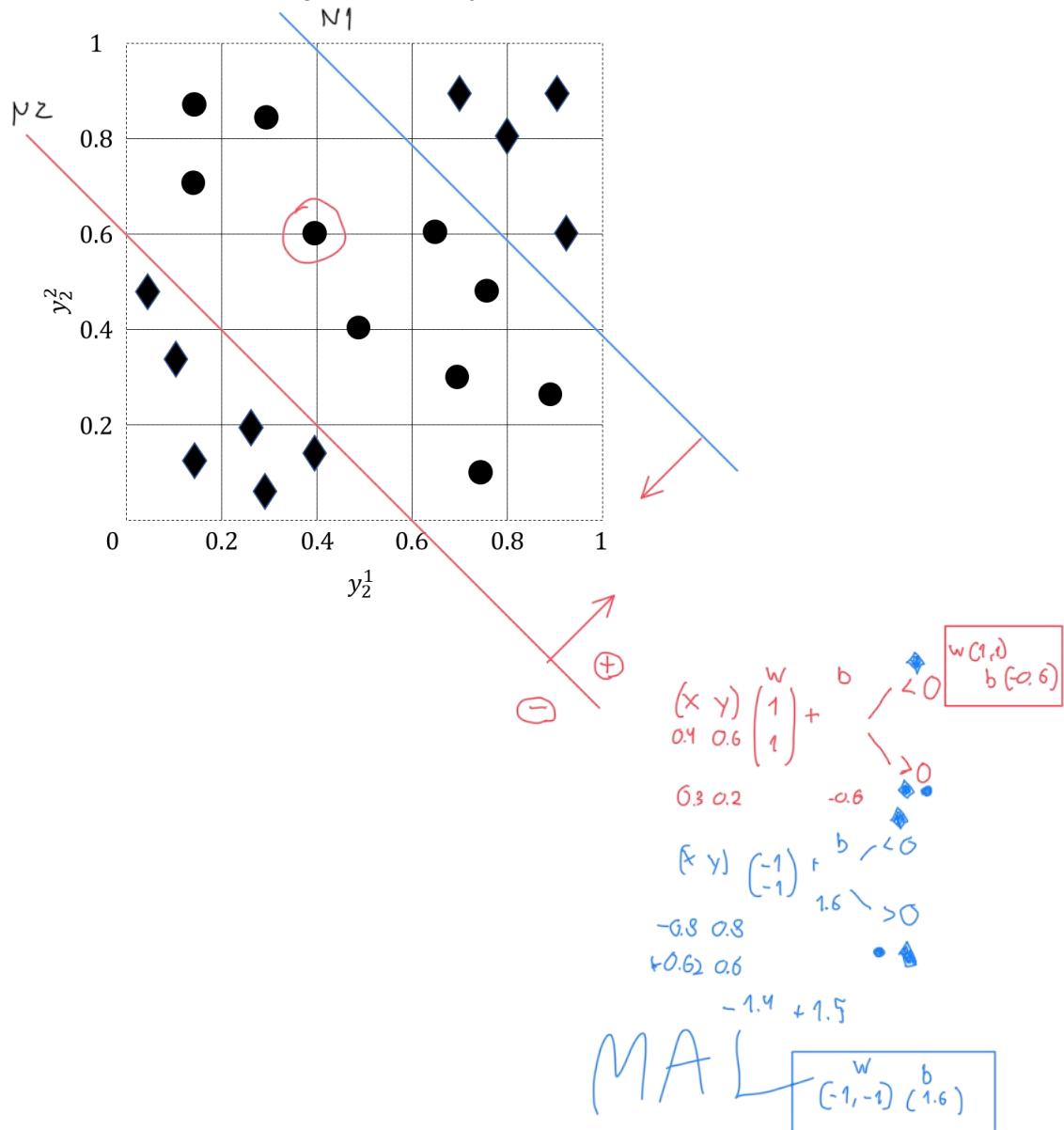
$$\begin{array}{ll} \diamondsuit \quad y = (0, 0) & \left\{ \begin{array}{ll} b_4 \geq 0 & b_4 = 0 \\ w_4 x + b_4 < 0 & w_4 x = -1 \\ w_4 y + b_4 < 0 & w_4 y = -1 \end{array} \right. \\ \circlearrowleft \quad y = (0, 1) & \\ \leftarrow \quad y = (1, 0) & \end{array}$$

20 de junio de 2024

En la siguiente Figura se muestra una arquitectura de red neuronal convolucional con los pesos obtenidos durante el entrenamiento mostrados en la parte derecha. Calcula el valor de la salida de la red neuronal para la siguiente entrada $x = \begin{pmatrix} 1 & 4 \\ 2 & 3 \end{pmatrix}$



En la siguiente Figura se muestra el valor de la salida para el conjunto de datos de entrenamiento de la red neuronal anterior. Dicho conjunto se compone de imágenes de baja resolución en las que se pretende detectar cuando ocurre algún tipo de anomalía. Diseña la red neuronal más sencilla que, considerando **una sola neurona en la capa de salida** (positiva para la clase “Anomalía”, **rombos en la figura**) clasifique correctamente todas las imágenes del conjunto de entrenamiento.



$$w_1$$

$$\begin{pmatrix} 1-4+2 & 4-1+3 \\ 2-1-3 & -4-2+3 \end{pmatrix} = \begin{pmatrix} -1 & 6 \\ -2 & -3 \end{pmatrix}$$

$\downarrow \begin{matrix} \text{RELU} \\ (\text{Si } i < 0) \Rightarrow 0 \end{matrix}$

$$\begin{pmatrix} 0 & 6 \\ 0 & 0 \end{pmatrix} \xrightarrow{\text{MAX}} 6$$

w₂

$$\begin{pmatrix} -2 & 6 \\ -2 & 4 \end{pmatrix} \xrightarrow{\text{RELU}} \begin{pmatrix} 0 & 6 \\ 0 & 4 \end{pmatrix} \xrightarrow{\text{MAX}} 6$$

20 junio 29

$$y_1^1 = \text{ReLU} \begin{pmatrix} 1-4+2 & -1+4+3 \\ -1+2-3 & -4-2+3 \end{pmatrix} = \text{ReLU} \begin{pmatrix} -1 & 6 \\ -2 & -3 \end{pmatrix} = \begin{pmatrix} 0 & 6 \\ 0 & 0 \end{pmatrix} \Rightarrow y_2^1 = 6$$

$$y_2^2 = \text{ReLU} \begin{pmatrix} 1-3 & 4+2 \\ -4+2 & 1+3 \end{pmatrix} = \text{ReLU} \begin{pmatrix} -2 & 6 \\ -2 & 1 \end{pmatrix} = \begin{pmatrix} 0 & 6 \\ 0 & 1 \end{pmatrix} \Rightarrow y_2^2 = 6$$

Calculo pesos N1

$$\vec{v}_2 = p_2 - p_1 = (1, 0.4) - (0.4, 1) = (0.6, -0.6)$$

(0.6, 0.6) = w₁
(0.6, -0.6)

$$w_1^T p_2 + b_1 = 0 \Rightarrow [b_1 = -w_1 x \cdot p_2 x - w_1 y \cdot p_2 y = -0.6 \cdot 1 - 0.6 \cdot 0.4 = -0.36]$$

Calculo pesos N2

$$\vec{v}_2 = p_2 - p_1 = (0.6, 0) - (0, 0.6) = (0.6, -0.6) \Rightarrow w_2 = (-0.6, -0.6)$$

$$b_2 = 0.36$$

Selvada de la primera capa = entrada 29 con

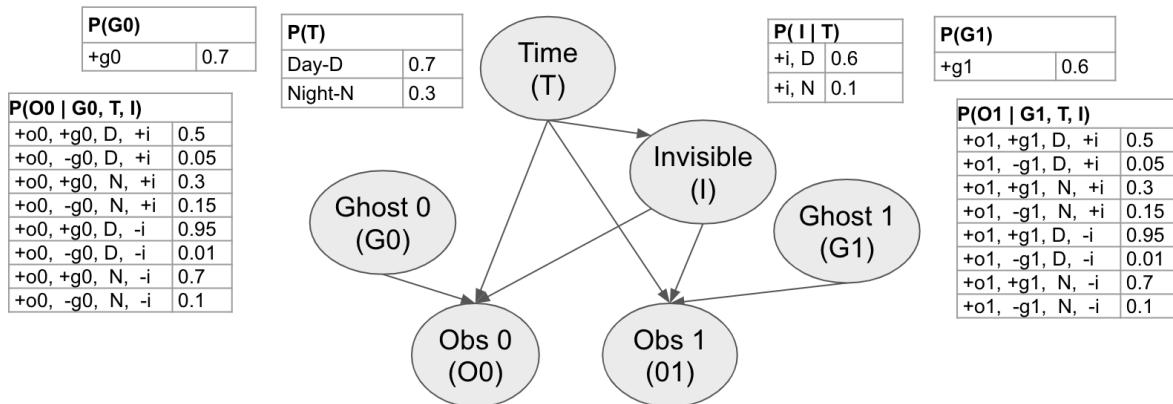
| | | | | |
|---|----------------|----------------|-------------------------|------------------------------|
| | y ₁ | y ₂ | | |
| ◇ | 1 | 0 | \diamond $y = (1, 0)$ | $w_3 x + b_3 \geq 0$ |
| | 0 | 1 | $y = (0, 1)$ | $w_3 y + b_3 \geq 0$ |
| ○ | 0 | 0 | $y = (1, 1)$ | $w_3 x + w_3 y + b_3 \leq 0$ |

$b_3 = 1.5$
 $w_3 x = 1$
 $w_3 y = 1$

20 de junio de 2024

En un nuevo juego de PACMAN, la visibilidad de los fantasmas varía mucho a lo largo de una partida. Según si el juego está en un periodo de día o noche (variable T) o si los fantasmas han activado su Invisibilidad (variable I), la fiabilidad de las medidas del sensor de visión de PACMAN cambia. Asumimos que puede haber hasta 2 fantasmas, y las siguientes variables:

- **GHOST_0 (G0)** - variable binaria, representa si el GHOST_0 está o no está
- **OBS_0 (O0)** - medida ruidosa de presencia, específico del ghost-0. Variable binaria que dice si ve o no al GHOST_0.
- **GHOST_1 y OBS_1** son equivalentes a GHOST_0 y OBS_0 pero para otro fantasma
- **Time (T)** - representa la fase del juego. Hay 2 posibles valores para esta variable *Day D*, y *Night N*
- **Invisible (I)** - variable binaria que representa si la invisibilidad de los fantasmas está activada o no. Cuando está activada, el fantasma solo se ve de manera intermitente, por lo cual baja mucho la precisión de la observación.



EN AMBOS CASOS: aplica el algoritmo de Eliminación de Variables. Acuérdate de completar las tablas de los factores que no estén completas y de escribir de manera formal la probabilidad que vas a calcular. Describe brevemente el proceso que sigues para obtener la probabilidad pedida.

- ¿Cuándo es más probable que haya un fantasma, aunque PACMAN diga que no lo ve, por la mañana o por la noche? Es decir, estima las siguientes probabilidades: ¿Cuál es la probabilidad de que el GHOST_0 esté presente aunque PACMAN no lo vea (es decir, O0 = FALSE) cuando el juego está en periodo de *Day*? ¿y cuando está en periodo de *Night*?
- Si el Ghost 1 está presente, pero PACMAN no lo ve, y el juego indica que es periodo de *Day* ¿cuál es la probabilidad de que la Invisibilidad esté activada?