17 de diciembre, 2019.

Examen

Nombre:____

a. Tiempo: 9:00 a 11:20.

b. Está permitido usar apuntes personales, escritos a mano.

c. No está permitido usar impresiones o fotocopias.

d. A excepción de cuando el profesor o los ayudantes indiquen, cualquier tipo de comunicación con otra persona que habite este planeta será considerada como caso de copia. Comunicaciones espirituales o con otros mundos son bienvenidas.

e. No olvide silenciar su teléfono celular.

Pregunta	Puntaje Máximo	Puntaje Obtenido
P1	14	
P2	12	
P3	12	
P4	12	
P5	12	
P6	12	
P7	12	
P8	12	
	TOTAL	
	NOTA FINAL	

1. (*14 puntos*) **Responda indicando si la expresión es verdadera o falsa, o respondiendo directamente la consulta respectiva. En todo los casos fundamente brevemente su respuesta.**

1.1. Un problema que los investigadores de IA consideran tiene el Test de Turing es la necesidad de implementar el engaño.

Solución: ***Verdadero***. *Es necesario que la máquina simule ser un humano y eso implica dar información falsa (por ejemplo, tener un padre y una madre).*

1.2. Un programa en lógica Π finito puede tener un modelo infinito y un conjunto de términos instanciados finito.

Solución: ***Falso***. *Si el programa es finito, la única manera que tengamos un modelo infinito es que la versión instanciada (grounded) del problema, sea infinito. Para ello es necesario que el conjunto de términos instanciados, también sea infinito.*

1.3. El siguiente programa tiene 12 modelos.

```
p(1..3).
q(1..4).
1 { r(X,Y) : p(X)} 1:- q(Y).
```

Solución: ***Falso***. *El programa tiene* 3^4

1.4. Cuando no es posible encontrar una heurística para un problema de búsqueda con costos uniformes, entonces siempre se debiera preferir usar BFS en vez de A* con $h = 0$ para encontrar una solución óptima, porque usualmente BFS expandirá menos estados.

Solución: ***Verdadero***. *La condición de término de BFS permite que la búsqueda termine antes porque se revisa el objetivo antes de agregar a Open y no al sacar de la Open.*

1.5. El algoritmo minimax requiere memoria lineal en la profundidad de la rama más larga del especio de búsqueda.

Solución: ***Verdadero***. *El algoritmo es recursivo y por lo tanto solo necesita memoria lineal en la rama.*

1.6. El número máximo de veces que A* puede extraer un nodo de la lista Open corresponde al número de nodos en el espacio de búsqueda.

Solución: ***Falso***. *En general puede ocurrir que los nodos pueden entrar y salir varias veces de la lista Open.*

1.7. Se tiene una red neuronal tipo feedforward con 1 capa oculta. El número de neuronas en la capa de entrada es 10, en la capa oculta 100 y en la de salida 10. La función de activación de cada neurona es binaria tipo switch (on/off). ¿Cuál es el número máximo de categorías excluyentes que esta red puede clasificar?.

Solución: *Codificación binaria permite:* 2^{10} *posibles clases.*

1.8. Para el mismo caso anterior, si la codificación de las neuronas en la capa de salida de la red se restringe al tipo one-hot, ¿Cuál es el número máximo de categorías excluyentes que esta red puede clasificar?.

Solución: *Codificación one-hot permite:* 10 *posibles clases.*

1.9. Para un clasificador de SVM binario como el visto en clases, al momento de test (uso del modelo) siempre es posible calcular el margen de clasificación para cada posible punto clasificado por el modelo.

Solución: ***Verdadero***. *El margen está dado por la distancia del punto al plano, lo cual se puede calcular pues tanto el punto como el plano son conocidos.*

1.10. Como regla general, para el modelamiento de un problema usando una red de Bayes para cada nodo siempre existirá una conexión, ya sea con un nodo padre, con un nodo hijo, o con ambos.

Solución: ***Falso***. *El nodo puede representar una variable que es independiente de todas las otras, por tanto, no tiene conexión con otro nodo.*

1.11. Durante el entrenamiento de una red neuronal, la tasa de aprendizaje cumple el rol de suavizar cambios bruscos en la dirección en la cual se mueve el gradiente entre actualizaciones del valor de los pesos. Mientras más alto es el valor de esta tasa, el proceso de entrenamiento en menos suscestible a cambios bruscos en la dirección del gradiente.

Solución: ***Falso***. *El rol de la tasa de aprendizaje es controlar la magnitud del paso de actualización de pesos en la dirección contraria al gradiente que corresponda. La tasa indicada corresponde a la tasa que controla un factor de momemtum.*

1.12. Si el modelo de aprendizaje de máquina tiene un menor número de parámetros, mayor será la posibilidad de sufrir problemas de sobreajuste.

Solución: ***Verdadero***. *Mayor número de parámetros implica un mayor número de grados de libertad, por ende, mayor probabilidad de sobreajustar los datos.*

1.13. Si el tamaño del set de entrenamiento es mayor, mayor será la posibilidad de sufrir problemas de sobreajuste.

Solución: ***Falso***. *En general, una mayor cantidad de datos, en el límite todos los datos posibles, permiten entrenar un modelo con mayor capacidad de generalización.*

1.14. Un árbol de decisión en que cada nodo tiene como máximo 2 nodos hijo puede solamente modelar problemas de clasificación binaria.

Solución: ***Falso***. *El número de descendientes de cada nodo no esta determinado por el número de clases, sino que por el número de valores que toma cada variable o test que se ejecuta en el nodo.*

2. (*12 puntos*) (*Answer Set Programming*)

En esta pregunta buscamos comprender cómo desarrollar un agente inteligente capaz de jugar al juego *Buscaminas* (*Minesweeper*, en inglés), usando la herramienta de programación en lógica. El Buscaminas se juega en un tablero rectangular, en donde hay una cantidad desconocida de celdas que contienen una *mina*. Cada celda que contiene una mina se encuentra además *oculta*. Existe un número de celdas que están ocultas pero que no contienen una mina. Cada celda que no está oculta puede desplegar un número, que indica cuántas de sus celdas *vecinas* contienen minas. Si una celda no oculta no despliega un número, significa que ninguna de sus celdas vecinas contiene una mina. Cada celda tiene como vecinas a las ocho celdas adyacentes. La siguiente figura muestra un posible tablero.

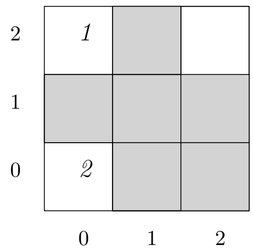


Figure 1: Un tablero de buscaminas de 3×3. Las celdas no ocultas son $(0,0)$, $(0,2)$ y $(2,2)$.

Para representar el problema usando programación en lógica, se utilizan los siguientes predicados.

- $celda(x, y)$ expresa que (x, y) es una celda del tablero.
- $vecina(x, y, x', y')$ expresa que (x, y) es vecina de (x', y').
- $oculta(x, y)$ expresa que la celda (x, y) está oculta.
- $libre(x, y)$ expresa que la celda (x, y) está libre, es decir, no contiene una mina.
- $mina(x, y)$ expresa que la celda (x, y) contiene una mina.

Conteste las siguientes preguntas (5/12 puntos por pregunta):

a. Escriba reglas que definan el predicado $vecina$.

Solución:

```
nulas(0,0).
inc(-1,0,1).
vecina(X,Y,X+Dx,Y+Dy) :- celda(X+Dx,Y+Dy), inc(Dx), inc(Dy), not nulas(Dx,Dy).
```

b. Escriba una regla que exprese el hecho que cada celda del tablero está libre o contiene una mina.

Solución:

```
1 {libre(X,Y);mina(X,Y)} 1 :- celda(X,Y).
```

c. Escriba reglas que expresen que:

(a) Hay exactamente dos minas en las celdas vecinas a $(0,0)$.

Solución:

```
2 {mina(X,Y) : vecina(0,0,X,Y)} 2
```

(b) Hay exactamente una mina en las celdas vecinas a $(0,2)$.

Solución:

```
1 {mina(X,Y) : vecina(0,2,X,Y)} 1
```

d. Explique detalladamente cómo haría usted para, usando un solver como clingo, obtener dónde jugar sin riesgo de que la celda explote (por la presencia de una mina).

Solución: *Computo todos los modelos usando el solver. Si todos los modelos coinciden en que* $libre(x, y)$ *para cierta celda* (x, y) *que está oculta, entonces podemos jugar ahí.*

e. Después de hacer una jugada, se revelan nuevos números en el tablero. Explique qué debe *agregar* y *eliminar* al programa para poder tomar la siguiente decisión.

Solución: *Al jugar el tablero despliega más números. Basta con agregar reglas como las de la pregunta c) para representar la información de esas reglas. Es necesario eliminar la regla* $oculta(x, y)$ *donde* (x, y) *es la celda donde recién se jugó.*

3. (*12 puntos*) (*Búsqueda*)

Una heurística si dice *consistente* si y solo si:

i. $h(s) \leq c(s,t)+h(t)$ para todo estado s y todo estado t que es sucesor de s, donde $c(s,t)$ representa el costo de la acción que lleva desde s a t, y

ii. $h(s_g) = 0$ para todo estado objetivo s_g

Conteste las siguientes preguntas:

a. (*8 pts*) Demuestre que si h es consistente, cada vez que A* expade un nodo s, todo t que es sucesor de s y que es agregado a Open en esa misma iteración es tal que $f(n') \geq f(n)$.

Solución: *Sumando $g(s)$ a ambos lados obtenemos:*

$$g(s) + h(s) \leq g(s) + c(s,t) + h(t)$$

Si t es agregado a Open entonces $g(t) = g(s) + c(s,t)$ por lo que podemos reemplazar arriba y obtener $g(s) + g(s) \leq g(t) + h(t)$ que es equivalente a lo que se quiere mostrar.

b. (*4 pts*) Diga por qué esta relación podría no cumplirse con algunos sucesores de s que ya estaban en Open.

Solución: *Cuando un sucesor de s, digamos t, no se agrega a Open, entonces se da que $g(s) + c(s,t) \geq g(t)$, por esta razón la desigualdad de arriba podría perfectamente invalidarse de hacer la sustitución que hicimos arriba.*

4. (*12 puntos*) (*Búsqueda, planeamiento*)

Los problemas de búsqueda determinísticos pueden ser descritos por un estado inicial, un conjunto de acciones con precondiciones y efectos, y una condición objetivo. Una forma de lograr esto es definir los estados como un conjunto de átomos. Por ejemplo un estado inicial podría ser $S_{init} = \{en(sala), soleado(hoy)\}$. Las acciones, a su vez, están definidas por una precondición y un efecto. Formalmente cada acción a tiene una precondición $pre(a)$, un efecto positivo $add(a)$ y un efecto negativo $del(a)$. Esto permite definir el estado sucesor usando simple aritmética de conjuntos. Si s es un estado, a es ejecutable en s si y solo si $pre(a) \subseteq s$. Si a es ejecutable en s, el estado resultante queda dado por $(s \cup add(a)) \setminus del(a)$.
Por ejemplo, la acción $caminar(sala, metro)$ podría tener la precondición $\{en(sala)\}$, el efecto positivo $\{en(metro)\}$ y el efecto negativo $\{en(sala)\}$. Así, $caminar(sala, metro)$ es aplicable en S_{init} (porque $\{en(sala)\} \subseteq S_{init}$) al aplicar esta acción sobre S_{init}, obtendríamos el estado $\{en(metro), soleado(hoy)\}$.
Finalmente, la condición objetivo G se define como un conjunto de átomos que queremos que se cumplan. De esta forma, si llegamos a un estado que sea superconjunto de G, habremos logrado nuestro objetivo. Por ejemplo para decir que nuestro objetivo es estar en la piscina, podríamos $G = \{en(piscina)\}$. En otras palabras, cualquier estado que contenta el átomo $en(piscina)$ es un estado objetivo.
Sea P un problema de búsqueda definido de la manera descrita arriba.

a. (*4 pts*) Sea P^{pre} el problema que resulta de cambiar las precondiciones de toda acción de P por el conjunto vacío y dejar los otros elementos intactos. Justifique que P^{pre} es una relajación del problema P.

Solución: *Al elimnar las precondiciones toda acción es aplicable en todo estado. El grafo del espacio de estados queda con más estados y más arcos. Esto garantiza que todo camino que existía en el grafo original ya también existe en el nuevo grafo. Por lo tanto, esta es una relajación al problema.*

b. (*4 pts*) Sea P^{-} el problema que resulta cambiar los efectos negativos de todas las acciones de P por el conjunto vacío y dejar los otros elementos intactos. Justifique que P^{-} es una relajación del problema P.

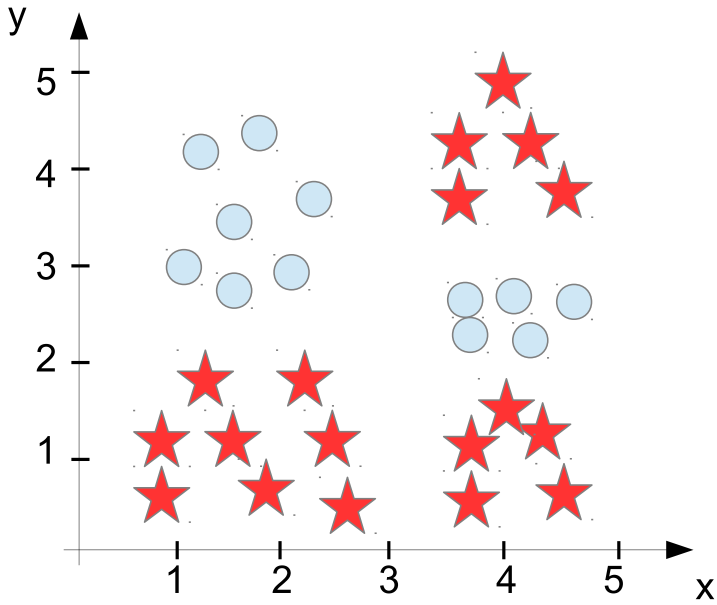
Solución: *Al eliminar los efectos negativos cualquier solución que antes resolvía el problema también lo resuelve ahora. En ese sentido, funciona como una relajación.*

c. (*4 pts*) Suponga que para obtener una heurística para un estado s resolvemos el problema relajado desde s y luego retornamos el costo de la solución obtenida. Diga por qué usar P^{-}, en general, permite obtener mucha más información que usar P^{pre}.

Solución: *Con P^{pre} se obtiene muy poca información, porque para todo problema hay acciones cuyos efectos resuelven el problema de inmediato. La heurística siempre sería igual al número de acciones necesario para cumplir cada objetivo en G. En el ejemplo $lanzarse(piscina)$ tendría como efecto $en(piscina)$. Al elimnar precondiciones esta acción es siempre posible en todo estado, por lo que con la relajación siempre se encuentra que $h = 1$. El efecto que esto tiene es que la heurística discrimina muy poco entre pares de estados.*

5. (*12 puntos*) (*Árboles de decisión*)

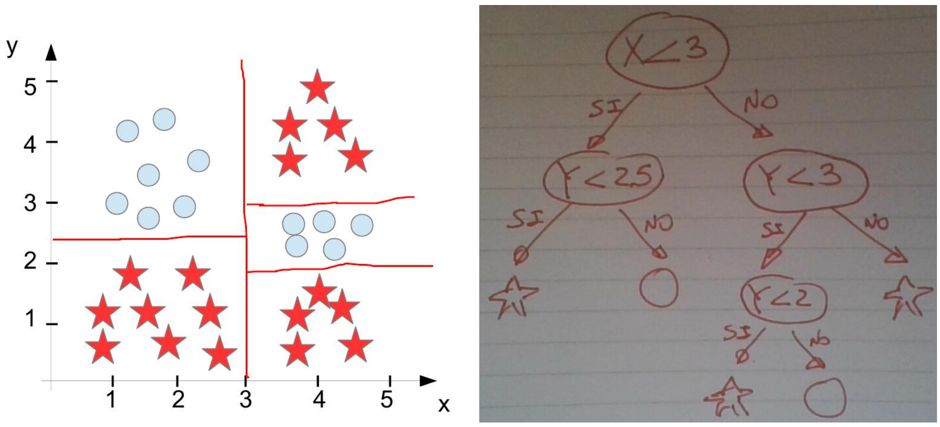
Los datos de la figura corresponden al set de entrenamiento para un problema de clasificación binaria.



a. (*4 pts*) Indique la estructura de un árbol de decisión que clasifique estos datos de entrenamiento con un 100% de exactitud (ayuda: no es necesario calcular ganancias de información, la estructura se puede determinar directamente por inspección visual de la figura).

Solución:

Existen varias posibles soluciones, la figura muestra la superficie de decisión y árbol correspondiente para una posible solución:



b. (*4 pts*) En base a la estructura anterior, indique una regla lógica que represente el mismo sistema de decisión que representa el árbol indicado.

Solución:

Basta con calcular la disyunción de las conjunciones de cada rama del árbol:

$if\ [(x < 3) \wedge (y < 2.5)] \rightarrow \star\ \vee$

$if\ [(x < 3) \wedge \neg(y < 2.5)] \rightarrow \bigcirc\ \vee$

$if\ [\neg(x < 3) \wedge (y < 3) \wedge (y < 2)] \rightarrow \star\ \vee$

$if\ [\neg(x < 3) \wedge (y < 3) \wedge \neg(y < 2)] \rightarrow \bigcirc\ \vee$

$if\ [\neg(x < 3) \wedge \neg(y < 3)] \rightarrow \star$

c. (*4 pts*) Al momento de test, que clasificación entregaría el árbol para las siguientes instancias:

1. $\{x = 4, y = 4\}$?
2. $\{x = 2, y = 4\}$?

Solución:

1. $\{x = 4, y = 4\}$ Clase $\star$
2. $\{x = 2, y = 4\}$ Clase $\bigcirc$

6. (*12 puntos*) (*SVMs*)

Modifique la formulación del clasificador SVM binario para que pueda cumplir con las siguientes condiciones:

1. En lugar de usar un hiperplano, la superficie de decisión está determinada por una función no lineal $f_\theta(\vec{x})$, donde $\vec{x}$ corresponde al vector con los atributos (variables) de entrada y θ es el espacio paramétrico de $f(\cdot)$.
2. El mínimo margen de clasificación debe ser superior a un valor constante γ previamente definido.

Como antecedente, se conoce que la distancia de cualquier punto $\vec{x_k}$ a la superficie de decisión $f(\vec{x})$ está dada por $d(\vec{x_k}, f(\vec{x}))$.

a. (*8 pts*) Asuma que los datos de entrenamiento E son separables por una función de tipo $f(\vec{x})$. Así, según las condiciones requeridas, indique las ecuaciones para el problema de optimización de SVM modificado. En otras palabras, el problema de optimización debe entregar como solución un clasificador de máximo margen, pero esta vez en el espacio funcional $f_\theta(\cdot)$.

Solución:

$$\underset{\theta}{\operatorname{argmax}} \left\{ \min_k \quad d(\vec{x_k}, f_\theta(\vec{x})) \right\}$$

$$\text{sujeto a: } z_i(f(x_i)) \geq \gamma, \quad \forall (x_i, y_i) \in TS,$$

$$z_i = \begin{cases} 1, & \text{if } x_k \in C_1 \\ -1, & \text{if } x_k \in C_2 \end{cases}$$

TS: training set.

b. (*4 pts*) Si se conoce que existe en E sólo un punto $\vec{x_e}$ que es imposible de clasificar correctamente por una función en el espacio parámetrico de $f_\theta(\vec{x})$, indique las ecuaciones para el problema de optimización de SVM modificado, tal que la solución sigue minimizando el número de puntos mal clasificados en E.

Solución:

$$\underset{\theta}{\operatorname{argmax}} \left\{ \min_k \quad d(\vec{x_k}, f_\theta(\vec{x})) \right\}$$

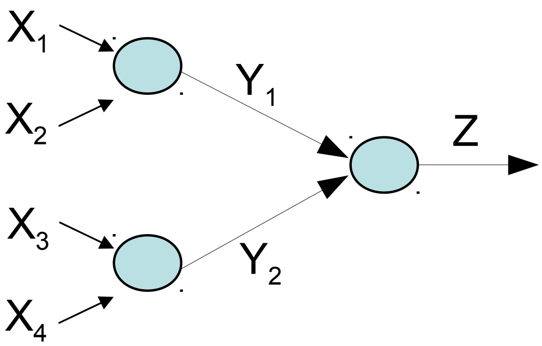
$$\text{sujeto a: } z_i(f(x_i)) \geq \gamma, \quad \forall (x_i, y_i) \in TS, \quad i \neq e,$$

$$z_i = \begin{cases} 1, & \text{if } x_k \in C_1 \\ -1, & \text{if } x_k \in C_2 \end{cases}$$

TS: training set.

7. (*12 puntos*) (*Redes neuronales*)

La red neuronal en la siguiente figura consiste de un modelo tipo feed-forward, donde en cada nodo la función de activación tiene la forma $f(x) = \alpha x$, donde α es un valor constante pre-establecido, por ejemplo, $\alpha = 0.5$.



a. (*4 pts*) Indique el número de parámetros que es necesario entrenar para ajustar esta red a un set de datos de entrenamiento. Para cada neurona, recuerde considerar el parámetro de la entrada constante o bias.

Solución:
Hay 9 parámetros, 3 pesos asociados a las conexiones a cada una de las 3 neuronas.

b. (*4 pts*) Si después del entrenamiento se obtiene una red en que el valor de todos los pesos tiene un valor 1. ¿Cuál es la salida Z de la red para la siguiente entrada?:
$\{X_1 = 4, X_2 = 0, X_3 = 2, X_4 = 1\}$, asuma un valor $\alpha = 0.5$?

Solución:
$0.5(0.5 * 4 + 0.5 * 3) = 1.75$

c. (*4 pts*) Demuestre que esta red puede reducirse a una red equivalente pero sin capa oculta. Indique el número de parámetros de esta red.

Solución:
La función de activación es lineal, por tanto, la capa oculta es innecesaria pues una combinación lineal de funciones lineales sigue siendo una combinación lineal. Matemáticamente:
$\alpha(w_1^2\alpha(w_1^1x1 + w_2^1x2) + w_2^2\alpha(w_3^1x3 + w_4^1x4))$

$(\alpha w_1^2\alpha(w_1^1x1 + w_2^1x2) + \alpha w_2^2\alpha(w_3^1x3 + w_4^1x4))$

$(\alpha w_1^2\alpha w_1^1x1 + \alpha w_1^2\alpha w_2^1x2) + (\alpha w_2^2\alpha w_3^1x3 + \alpha w_2^2\alpha w_4^1x4)$

$\beta_1x1 + \beta_2x2 + \beta_3x3 + \beta_4x4$

Por tanto, la red es equivalente a una red sin capa oculta con 4 parámetros. A lo cual hay que agregar el parámetro de bias, para un total de 5 parámetros.

8. (*12 puntos*) (*Bayes*)

Un panel de 3 jueces determina el resultado de una elección política. Cada juez puede votar por sólo uno de 2 posibles candidatos y el resultado final es determinado por el candidato con más votos. Los jueces reciben como entrada la argumentación de 1 representante por cada candidato. Durante la argumentación, dos de los jueces muestran alta imparcialidad y escuchan atentamente al representante de cada candidato. El tercer juez no posee la misma imparcialidad y sólo pone atención al representante de uno de los candidatos. Los representantes de cada candidato realizan sus representaciones en forma independiente, no hay ningun tipo de debate entre ellos. Asimismo, los jueces actuan en forma independiente, no teniendo ningun tipo de fase de discusión conjunta.

a. (*6 pts*) Identifique las variables relevantes a la situación descrita. Para cada variable indique que valores puede tomar. En su análisis use un criterio de máxima simplicidad, es decir, prefiera la solución de menor complejidad (Ockham's razor).
Solución:
E: $\in [1, 2]$, resultado de la elección favorable a candidato 1 o 2.
J1: $\in [1, 2]$, juez 1 imparcial, escucha a ambos representantes.
J2: $\in [1, 2]$, juez 2 imparcial, escucha a ambos representantes.
J3: $\in [1, 2]$, juez 3 parcial, escucha sólo a representante 1.
R1: $\in [1, 2]$, representante 1, tiene un efecto positivo o negativo en juez.
R2: $\in [1, 2]$, representante 2, tiene un efecto positivo o negativo en juez.

b. (*6 pts*) Dibuje una red de Bayes que capture la interacción entre las variables de interés. Evite cualquier tipo de supuesto que no sea directamente derivable de la descripción del problema. En su esquema de la red indique con claridad la variable que representa cada nodo.
Solución: