

Estudiantes: Agustin Díaz, Aldana Zarate

Enunciado

Nuestra amiga Clara nos recomendó el juego “Ori and the blind forest”, que dice que es un juego imperdible. Por otro lado nuestro amigo Alejandro estuvo jugando mucho al Hades y está muy ansioso por que lo probemos. Nosotros, Aldana y Agustín, estamos en busca de un juego metroidvania, y dado que queremos relajarnos en estas épocas buscamos un juego fácil.

Un experto gamer conoce las reglas que relaciona un videojuego con el género metroidvania. Utilizaremos esto y la dificultad para obtener un recomendador e inferir los juegos que nos gustan.

Los metroidvania se distinguen por:

- mapas grandes
- muchas habilidades distintas
- mucha variedad de enemigos distintos

Todas estas propiedades son lineales, i.e. mapas más pequeños harán que el juego sea menos metroidvania.

No olvidemos que nosotros somos muy exquisitos con los juegos que nos gustan, por lo cual un buen juego tiene que ser valorado mayor a 90 puntos, mientras que uno interesante ronda los 70. Jugar un juego con puntaje menor a 50 sería un desperdicio de nuestro tiempo.

Al estar en búsqueda de un metroidvania para relajarnos, lo que más nos gustaría sería un metroidvania fácil, pero sin embargo nos parece interesante algo que se asemeje, es decir hasta moderadamente metroidvania y dificultad media. Si el juego es muy difícil o no es metroidvania no nos interesará.

Variables y Reglas

Con estos datos e intuición podemos definir las variables lingüísticas:

x	T(x)	X	M
nivel de dificultad (d)	{baja, media, alta}	[0, 10]	d baja = (1/0, 1/3, 0.5/4, 0/5) d media = (0/3, 1/5, 0/8) d alta = (0/5, 0.5/6.5, 1/8, 1/10)
tamaño de mapa (m)	{pequeño, mediano, grande}	[0, 100]	m pequeño = (1/0, 1/30, 0.5/40, 0/50) m mediano = (0/30, 1/50, 0/80) m grande = (0/50, 0.5/65, 1/80, 1/100)
cantidad de habilidades (h)	{pocas, normal, muchas}	[0, 10]	h pocas = (1/0, 1/2, 0.5/3.5, 0/5) h normal = (0/2, 1/5, 0/8) h muchas = (0/5, 0.5/6.5, 1/8, 1/10)
especies de enemigos (e)	{no variado, variado, muy variado}	[0, 30]	e poco variado = (1/0, 1/3, 0.5/6.5, 0/10) e variado = (0/3, 1/10, 0/20) e muy variado = (0/10, 0.5/15, 1/20, 1/30)
metroidvania	{para nada, moderadamente, fuertemente}	[0, 10]	metr para nada = (1/0, 1/2, 0.5/3.5, 0/5) metr moderadamente = (0/2, 1/5, 0/8) metr fuertemente = (0/5, 0.5/6.5, 1/8, 1/10)
gusto	{no, interesante, mucho}	[0, 100]	g no = (1/0, 1/50, 0.5/60, 0/70) g interesante = (0/50, 1/70, 0/90) g mucho = (0/70, 0.5/80, 1/90, 1/100)

Con la ayuda del experto obtenemos las reglas para la primera y segunda etapa de inferencia.

Etapa 1:

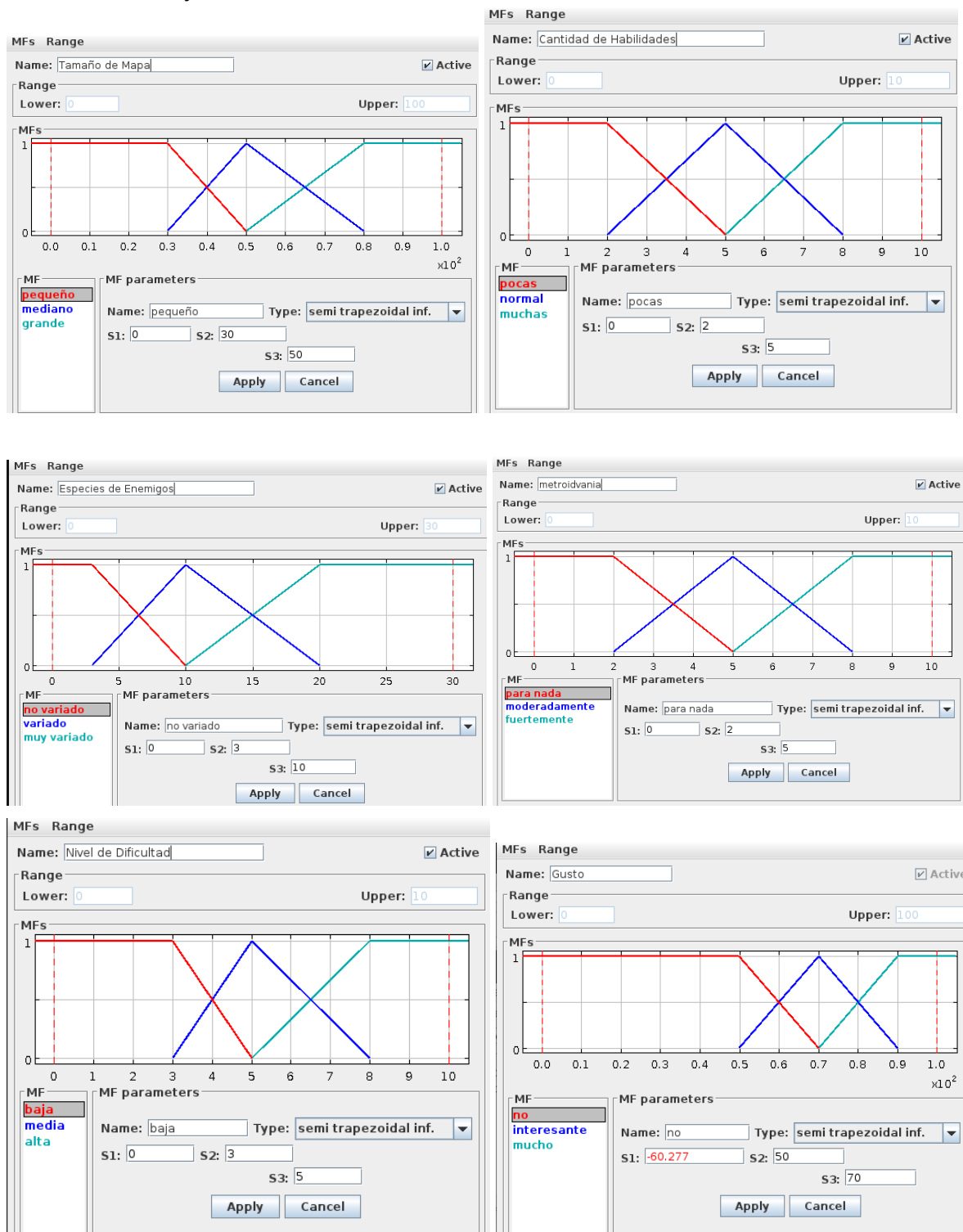
- Si el mapa es grande o las especies de enemigos son muy variadas o la cantidad de habilidades son muchas entonces el juego es fuertemente metroidvania.
- Si el mapa es mediano o las especies de enemigos son variadas o la cantidad de habilidades son normales entonces el juego es moderadamente metroidvania.
- Si el mapa es pequeño o las especies de enemigos son poco variadas o la cantidad de habilidades son pocas entonces el juego es para nada metroidvania.

Etapa 2:

- Si la dificultad es baja y el juego es fuertemente metroidvania entonces nos gusta mucho
- Si la dificultad es media y el juego es fuertemente metroidvania entonces nos interesa
- Si la dificultad NO es alta y el juego es moderadamente metroidvania entonces el juego nos interesa.
- Si la dificultad es alta o el juego es para nada metroidvania entonces no nos gusta

Pasaje a FisPro

Pasamos los conjuntos borrosos a FisPro:



Traducimos las reglas a FisPro. Notemos que hay reglas en lenguaje natural que se traducen a múltiples reglas en FisPro, ya sea en base a negaciones o disyunciones.

Etapas 1:

Rule	Active	IF Tamaño de Ma...	AND Cantidad de...	AND Especies de...	THEN metroid...
1	<input checked="" type="checkbox"/>	pequeño			para nada
2	<input checked="" type="checkbox"/>		pocas		para nada
3	<input checked="" type="checkbox"/>			no variado	para nada
4	<input checked="" type="checkbox"/>	mediano			moderadamente
5	<input checked="" type="checkbox"/>		normal		moderadamente
6	<input checked="" type="checkbox"/>			variado	moderadamente
7	<input checked="" type="checkbox"/>	grande			fuertemente
8	<input checked="" type="checkbox"/>		muchas		fuertemente
9	<input checked="" type="checkbox"/>			muy variado	fuertemente

Etapas 2:

Rule	Active	IF metroidvania	AND Nivel de Dificultad	THEN Gusto
1	<input checked="" type="checkbox"/>	para nada		no
2	<input checked="" type="checkbox"/>	moderadamente	media	interesante
3	<input checked="" type="checkbox"/>	fuertemente	baja	mucho
4	<input checked="" type="checkbox"/>	fuertemente	media	interesante
5	<input checked="" type="checkbox"/>	moderadamente	baja	interesante
6	<input checked="" type="checkbox"/>		alta	no

Inferencia en FisPro

Utilizando FisPro generamos los conjuntos borrosos y las reglas, los mismos se pueden ver en los archivos metroidvania.fis(Etapa 1) y gusto.fis(Etapa 2)

Notas:

- Utilizamos la funcionalidad automática de FisPro para los conjuntos de salida que normaliza las variables según su modo de inferencia (para que puedan utilizar todos sus valores posibles)
- Tomamos principalmente los valores por conjunción de Mamdani y defuzzificación por área
- Los demás resultados los mostramos a modo ilustrativo para poder analizar distintos enfoques.

Ori and the blind Forest

Etapa 1:

Ver qué tan metroidvania es este juego sabiendo:

- $m=100$, $h=5$, $e=15$

Defuzzificamos el resultado con los distintos métodos que nos provee FisPro:

Área: Obtenemos 8.516 puntos, lo que nos dice que es fuertemente metroidvania

Rules	Tamaño de Mapa	Cantidad de Habilidades	Especies de Enemigos	metroidvania
	100	5	15	8.516
1				
2				
3				
4				
5				
6				
7				
8				
9				

Mean Max: Obtenemos 5 puntos, lo que nos dice que es moderadamente metroidvania

Rules	Tamaño de Mapa	Cantidad de Habilidades	Especies de Enemigos	metroidvania
	100	5	15	5
				No connex
1				
2				
3				
4				
5				
6				
7				
8				
9				

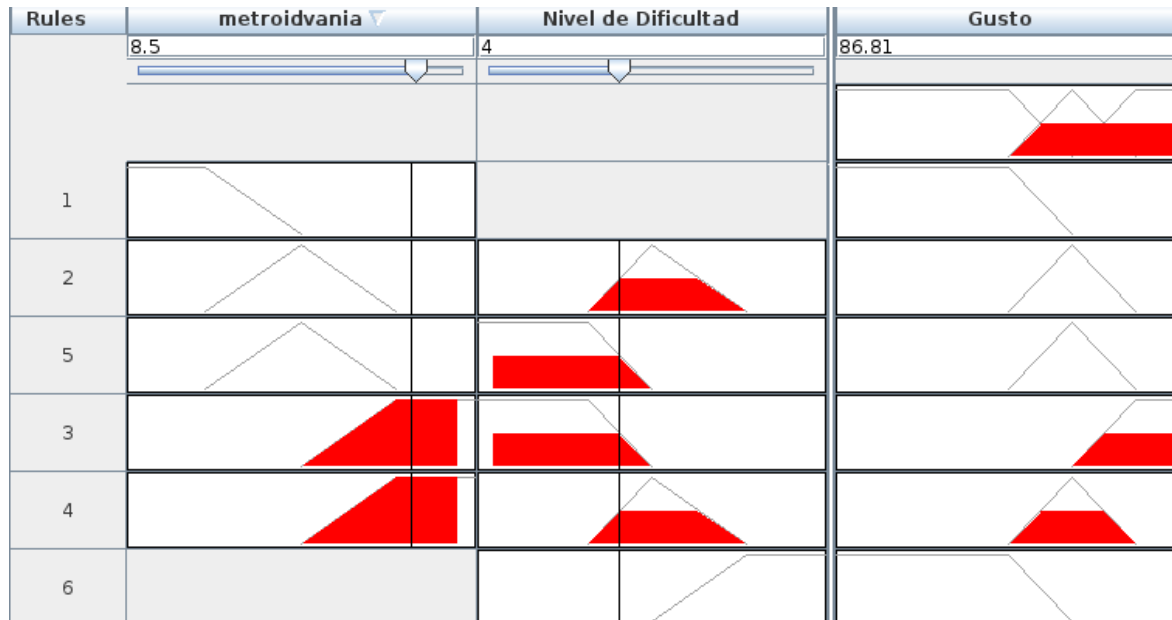
Sugeno: obtenemos 7.5, lo cual significa que es en gran parte fuertemente metroidvania, pero no del todo ya que en parte es moderadamente metroidvania

Rules	Tamaño de Mapa	Cantidad de Habilidades	Especies de Enemigos	metroidvania
	100	5	15	5
				No connex
1				
2				
3				
4				
5				
6				
7				
8				
9				

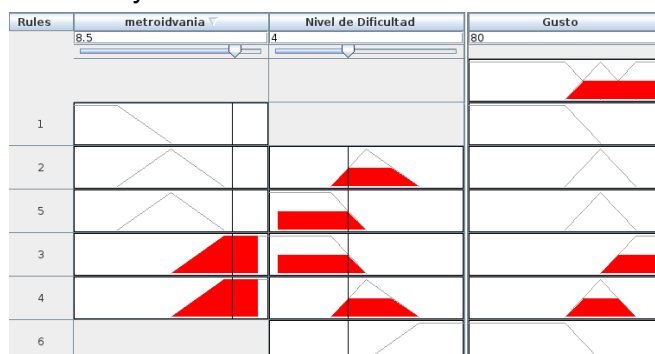
Etapa 2:

Buscamos reviews en metacritic y encontramos que la dificultad del Ori es 4/10, con esto y el resultado anterior(utilizamos el valor del área: 8.516) inferimos en FisPro:

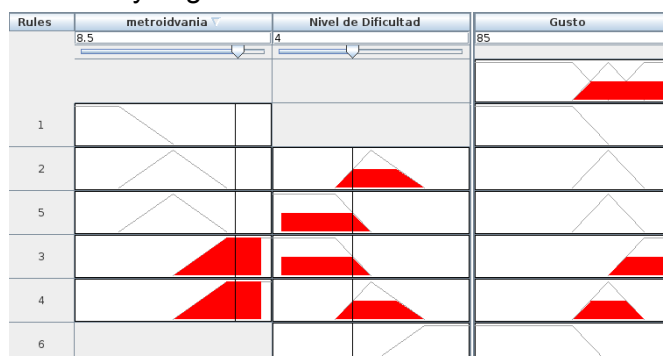
Minimum y área:



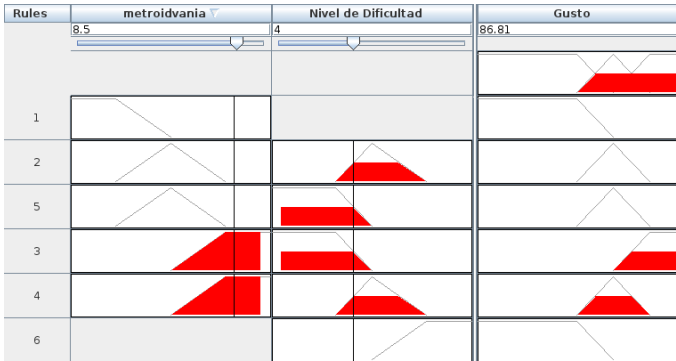
Minimum y mean max:



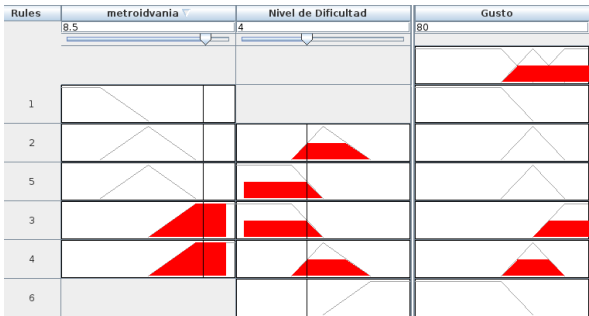
Minimum y sugeno:



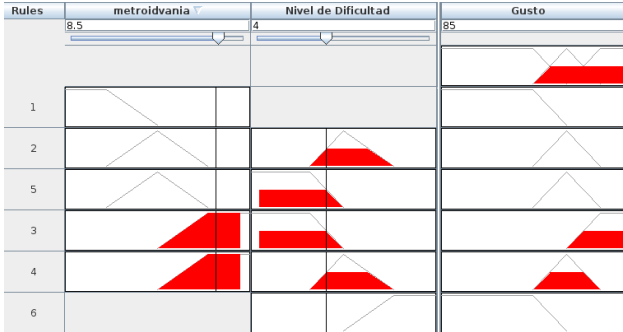
Product y área:



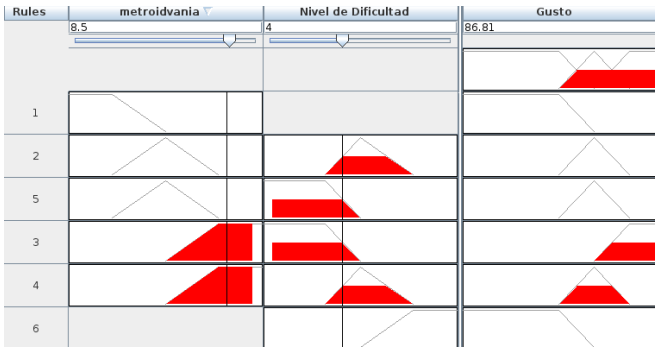
Product y mean max:



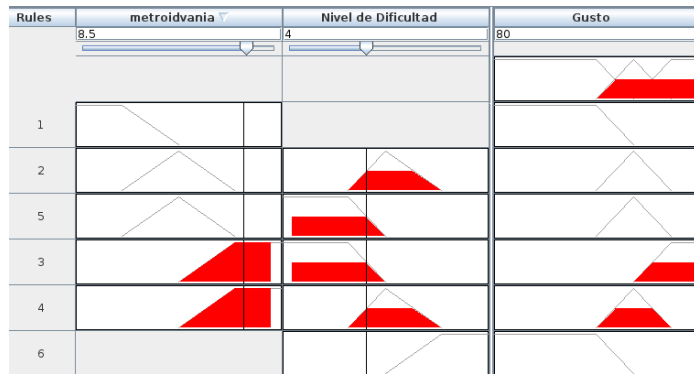
Product y sugeno:



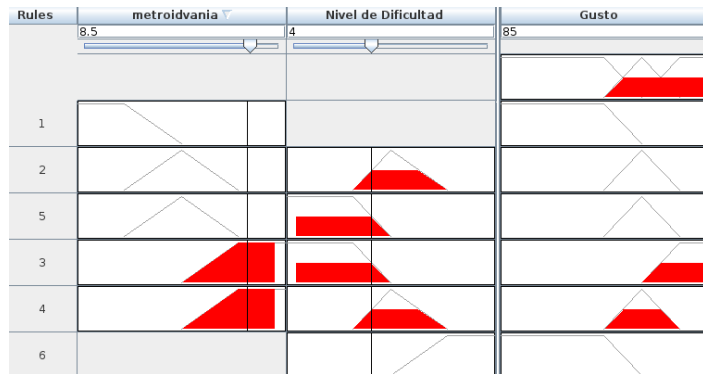
Lukasiewicz y área:



Lukasiewicz y mean max:



Lukasiewicz y sugeno:



En este caso particular, todos los métodos de conjunción activan de la misma manera las reglas. Esto se debe a que el valor de metroidvania es 1, haciendo que el valor final de la conjunción sea el valor de la dificultad.

Finalmente, nosotros utilizamos el método de área para defuzzificar (por la naturaleza de los datos) y conseguimos 86.81/100, lo cual es en pequeña parte interesante y en gran parte nos gusta mucho! Parece que ya tenemos planes para las vacaciones de invierno.

Hades

Etapa 1:

Ver qué tan metroidvania es este juego sabiendo:

- $m=40, h=10, e=7$

Área:

Rules	Tamaño de Mapa	Cantidad de Habilidades	Especies de Enemigos	metroidvania
	40	10	7	6.276
1				
2				
3				
4				
5				
6				
7				
8				
9				

Mean max:

Rules	Tamaño de Mapa	Cantidad de Habilidades	Especies de Enemigos	metroidvania
	40	10	7	10
1				
2				
3				
4				
5				
6				
7				
8				
9				

Sugeno:

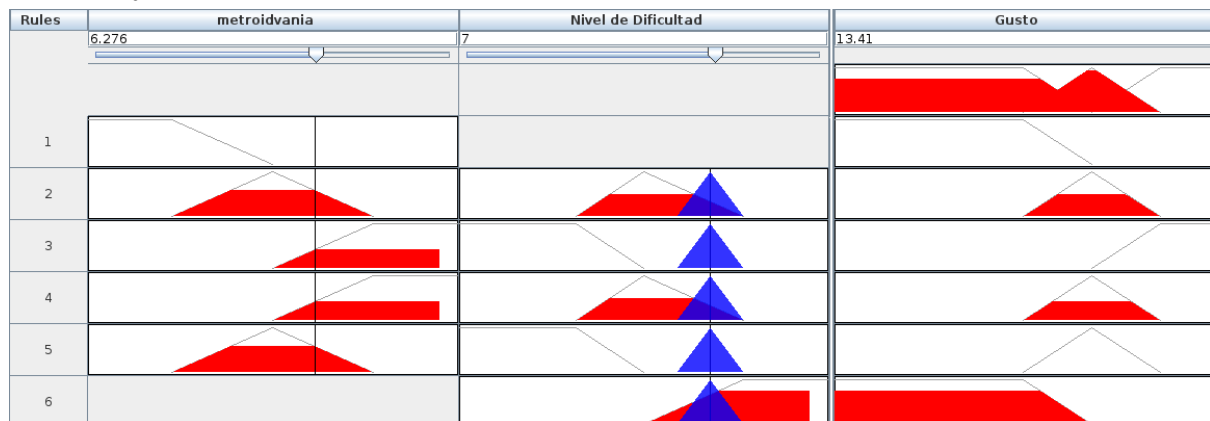
Rules	Tamaño de Mapa	Cantidad de Habilidades	Especies de Enemigos	metroidvania
	40	10	7	6.207
1				
2				
3				
4				
5				
6				
7				
8				
9				

Tomando el valor del área de 6.276, el juego todavía tiene chance ya que es en parte moderadamente y en parte fuertemente metroidvania.

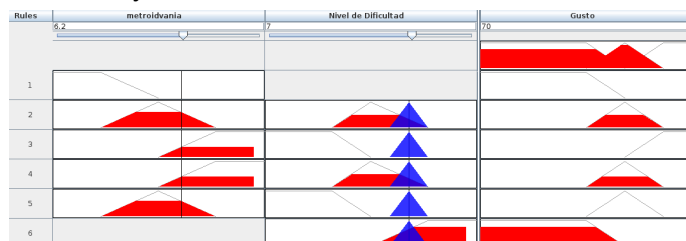
Etapa 2:

Utilizando el valor de metroidvania de la etapa anterior de 6.276, ahora necesitamos el valor de dificultad para inferir el gusto. Para ello, le consultamos a Alejandro qué puntaje le pondría de dificultad, y él nos respondió que le pondría un 6. Además, buscamos en metacritic y la dificultad que le asignan los críticos es de 8! Esto debe ser porque Ale es muy bueno jugando, pero vamos a tomar su opinión y la de metacritic para generar una entrada borrosa: el tamaño de la base será de 2 puntos de dificultad (la distancia entre el 6 y el 8) y tendrá su máximo en el 7 (punto medio entre 6 y 8), asignándole así igual importancia a la crítica de Alejandro y a la de metacritic.

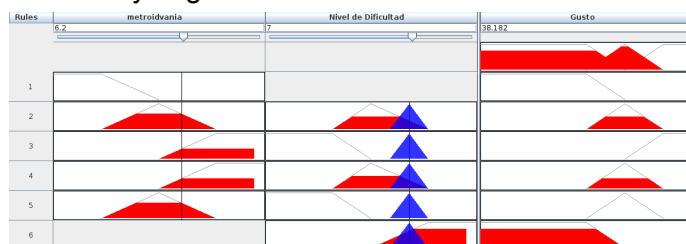
Minimum y área:



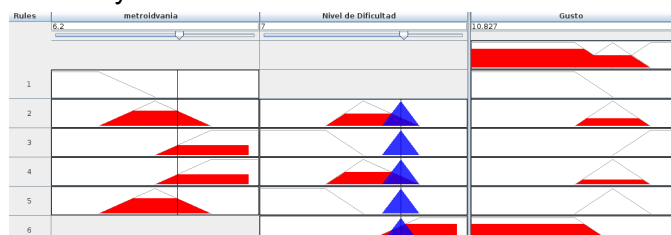
Minimum y mean max:



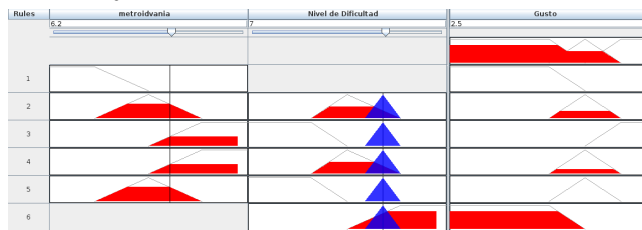
Minimum y sugeno:



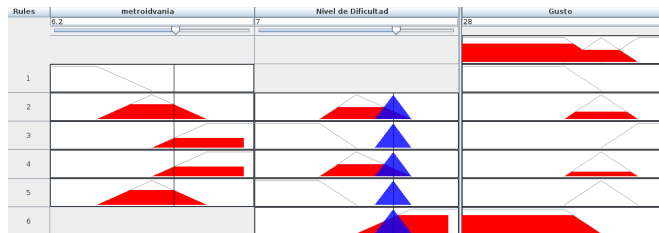
Product y área:



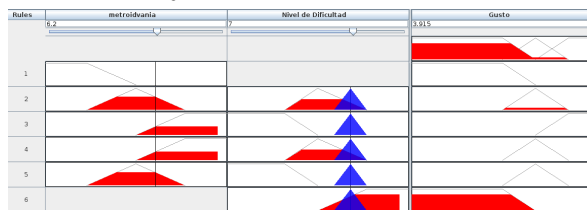
Product y mean max:



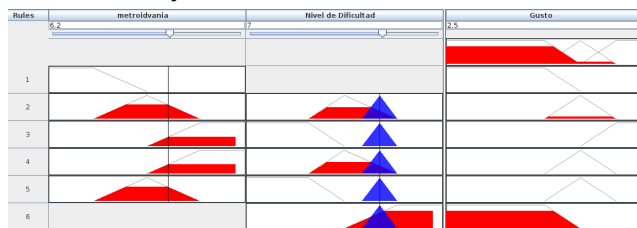
Product y sugeno:



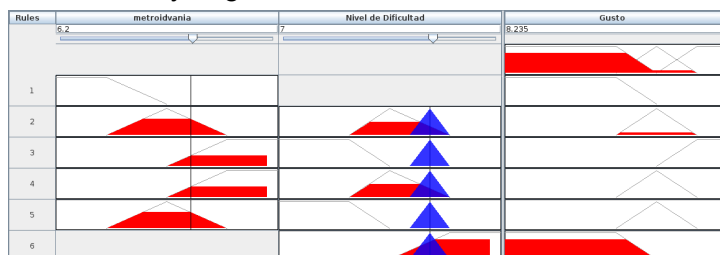
Lukasiewicz y área:



Lukasiewicz y mean max:



Lukasiewicz y sugeno:



Finalmente, nosotros utilizamos Mandami para la conjunción y el método de área para defuzzificar (por la naturaleza de los datos) y conseguimos 13.41/100, lo cual lo categoriza en los juegos que no nos gustan! El juego tenía potencial pero al tener tanta dificultad, nos quedará pendiente para cuando tengamos la motricidad fina más desarrollada.

Conclusiones

- Las variables difusas pueden tener rangos infinitos, pero con utilizar un rango finito de interés podemos capturar los comportamientos importantes y trabajar sobre ellos.
- A la hora de decidir cómo manejamos las implicancias, podemos elegir entre el min (de Mamdani), el prod (Larsen) y Luckasiewicz. Por la semántica que le queremos dar a las reglas, nos inclinamos por el modelo de inferencia de Mamdani. Esto es porque pensamos en un acercamiento pesimista, donde causa y efecto se vean bien reflejadas, donde la implicancia sea verdadera sólo si antecedente y consecuente siguen la misma línea. El modelo de Lukasiewicz no es compatible con esta idea y el de Larsen ya demasiado pesimista.
- Para la defuzzificación utilizamos el método de área, ya que sentimos que es el que más representación le da a cada conjunto borroso de salida en base a su relevancia. Tiene la semejanza con un promedio ponderado sobre los resultados obtenidos.
- Al armar las reglas, fuimos implementando varias ideas a la hora de formularlas, viendo los resultados que obteníamos, aumentando/disminuyendo la complejidad, etc. Para las diferentes etapas en un principio habíamos hecho muchas más reglas, pero analizando el problema y usando los operadores lógicos vistos (and, or, not) llegamos a esta última manera de expresarlas que nos pareció la más concisa y clara.
- Los alcances de las diferentes categorías en las variables también fueron mutando, en un principio había algunos rangos que se extendían más de lo necesario cuando veíamos los resultados obtenidos.
- En la etapa 1 usamos el operador “o” ya que cada variable de entrada tiene una relación directa con el valor de salida, independientemente de las otras variables de entrada. Esto no fue así en la etapa 2, ya que (como dice el enunciado) no alcanza con que un juego sea metroidvania o no, sino que también hay que tener en cuenta la dificultad del mismo. Por esto, en este caso usamos conjunciones para varias de las reglas.
- Los resultados obtenidos son los esperados a partir de la intuición que nos dan las reglas y los datos para cada juego (siguiendo los modelos y métodos de defuzzificación comentados anteriormente). Ya que si cambiamos alguna de estas elecciones, obtenemos resultados considerablemente diferentes y hasta inesperados.