

Universidad Autónoma de Baja California

INSTITUTO DE INGENIERÍA

MAESTRÍA Y DOCTORADO EN CIENCIAS E INGENIERÍA



**METODOLOGÍA PARA LA OBTENCIÓN DE  
PROPIEDADES TERMODINÁMICAS DE  
REFRIGERANTES MEDIANTE INTELIGENCIA  
ARTIFICIAL**

TESIS PARA OBTENER EL GRADO DE:

**DOCTOR EN INGENIERÍA**

Presenta:

Jesús Eduardo Mora Ramírez

Director: Dr. Carlos Pérez Tello

Mexicali, Baja California, México. Diciembre de 2014.

## **DEDICATORIA**

A mi preciosa esposa, por su fiel compañía, amor, comprensión y apoyo incondicional.

## AGRADECIMIENTOS

Agradezco de todo corazón a Dios Todopoderoso por haberme dado el tiempo y la vida necesarios para poder estudiar un poquito más de su maravillosa creación, también al Señor Jesucristo que me tendió su mano a lo largo de estos años y al Espíritu Santo que me dió la paciencia y alegría para concluir este estudio. *Es gloria de Dios ocultar un asunto, es gloria de reyes averiguarlo (Prov 25:2)*

A mi esposa por ser mi aliento y apoyo, mi amiga y consejera, y por compartir las alegrías y batallas en este proyecto de vida. Agradezco a mis hijos porque con su alegría adornan los días que Dios me ha dado.

A mis padres por haberme cuidado y apoyado con cariño y dedicación, por su consejo y dirección.

A mis hermanos y amigos que hacen más divertidos y alegres mis días.

A todos aquéllos que me ayudaron directa o indirectamente, que por haber sacrificado algunas neuronas en la comprensión de un poco de termodinámica, en este momento no los recuerdo en su totalidad.

A la Universidad Autónoma de Baja California que me brindó el espacio necesario para terminar este proyecto.

## RESUMEN

El presente proyecto presenta una metodología que emplea redes neuronales artificiales (RNAs) para determinar las propiedades termodinámicas de sustancias refrigerantes. Para ello se desarrolló un algoritmo capaz de encontrar la mejor estructura de RNA para cada refrigerante analizado.

Se utilizaron datos de los refrigerantes R12 (diclorodifluorometano), R22 (clorodifluorometano), R134a (1,1,1,2-tetrafluoroetano), R290 (propano), R717 (amoníaco), y R410a (difluorometano-pentafluoroetano). Se llevó a cabo un análisis comparativo entre los resultados obtenidos por los modelos neuronales y los valores reales reportados y se obtuvieron desviaciones máximas de 6.6 % para la presión, 0.07 % para la densidad de líquido, 0.015 % para la entalpía de líquido, 0.025 % para la entalpía de vapor, 0.52 % para la entropía de líquido, 0.02 % para la entropía de vapor y de hasta 94 % para el volumen específico de vapor.

Para comprobar el grado de ajuste se estimó el error cuadrático medio (MSE), la raíz cuadrada del error cuadrático medio (RMSE) y el coeficiente de regresión ( $R^2$ ). El MSE tuvo valores entre  $9.12E^{-9}$  hasta  $6.79E^{-5}$  para la presión, de  $1.48E^{-7}$  a  $5.45E^{-5}$  para la densidad de líquido, de  $5.70E^{-7}$  a  $3.96E^{-5}$  para el volumen de vapor, de  $8.18E^{-8}$  a  $1.00E^{-3}$  para la entalpía de líquido, de  $4.13E^{-8}$  a  $1.84E^{-5}$  para la entalpía de vapor, de  $1.04E^{-7}$  a  $3.37E^{-5}$  para la entropía de líquido y de  $2.29E^{-7}$  a  $1.19E^{-5}$  para la entropía de vapor.

De acuerdo con las EoS de Redlich-Kwong, Soave y Peng Robinson, se calcularon la presión, el volumen de vapor y la densidad de líquido para el R12, obteniéndose desviaciones máximas en el cálculo de presión de 40.44 %, 1.5 % y 1.88 % respectivamente,

comparados con 1.73 % de la RNA. Para la densidad de líquido se obtuvo 21 %, 18.75 % y 9.15 % respectivamente, en comparación con  $7E^{-14}$  de la RNA. Finalmente, el error máximo obtenido en el cálculo de volumen específico de vapor fue de 29.25 %, 2.42 % y 2.02 % respectivamente, mientras que la RNA mostró un 1.5 %.

# Índice general

DEDICATORIAS . . . . .	I
AGRADECIMIENTOS . . . . .	II
RESUMEN . . . . .	IV
1. ANTECEDENTES . . . . .	5
2. MARCO TEÓRICO . . . . .	10
2.1. Propiedades termodinámicas . . . . .	10
2.2. Refrigeración . . . . .	11
2.3. Ecuaciones de Estado (EoS) . . . . .	16
2.3.1. Aplicación de las Ecuaciones de Estado al ciclo de refrigeración	22
2.3.2. Limitaciones de las Ecuaciones de Estado . . . . .	23
2.4. Redes Neuronales Artificiales . . . . .	25
2.4.1. El modelo del cerebro humano . . . . .	28
2.4.2. Aprendizaje Artificial . . . . .	30
2.4.3. El Perceptrón . . . . .	31
2.4.4. Redes multicapa de perceptrones . . . . .	32
2.4.5. Modelado de datos con redes multicapa no-lineales . . . . .	34
2.4.6. Entrenamiento de una RNA . . . . .	36
3. REDES NEURONALES ARTIFICIALES Y EQUILIBRIO LÍQUIDO-VAPOR . . . . .	39
3.1. Notación propuesta . . . . .	39
3.2. Metodología para estructurar la RNA . . . . .	40
3.2.1. Preparación de los datos . . . . .	40
3.2.2. Construcción de la Red Neuronal Artificial . . . . .	42
3.2.3. Algoritmo . . . . .	48
3.3. Resultados . . . . .	52
3.3.1. Comparación entre Ecuaciones de Estado y Redes Neuronales . . . . .	62
4. CONCLUSIONES . . . . .	66
4.1. Recomendaciones . . . . .	68
4.2. Trabajo futuro . . . . .	70

A. ANÁLISIS DE LOS MODELOS DE RNA DE LOS REFRIGERANTES . . . . .	71
A.1. Análisis estadístico de la RNA del R12 . . . . .	71
A.2. Análisis estadístico de la RNA del R22 . . . . .	80
A.3. Análisis estadístico de la RNA del R134a . . . . .	89
A.4. Análisis estadístico de la RNA del R290 . . . . .	98
A.5. Análisis estadístico de la RNA del R717 . . . . .	107
A.6. Análisis estadístico de la RNA del R410a . . . . .	116
A.7. Esquema del algoritmo general . . . . .	125
A.8. Algoritmo desarrollado en Matlab . . . . .	126
A.9. Parámetros (pesos y bias) de las redes neuronales obtenidas. . . . .	142
BIBLIOGRAFÍA . . . . .	146

# Índice de figuras

1.	Refrigerador bifásico de Carnot. . . . .	12
2.	Ciclo de refrigeración por compresión de vapor. . . . .	13
3.	Ciclo de refrigeración por compresión de vapor en diagrama $P-H$ . . . . .	14
4.	Ejemplo de ciclo de refrigeración en diagrama $P-H$ . . . . .	16
5.	Diagrama $P-H$ del R22. . . . .	17
6.	Ejemplo de RNA construida de dos capas y tres neuronas. . . . .	26
7.	Estructura fundamental de una neurona biológica. . . . .	29
8.	Esquema del Perceptrón de Rosenblatt. . . . .	32
9.	Esquema de una Red de perceptrones multicapa. . . . .	33
10.	Función tangente hiperbólica. . . . .	34
11.	Regresión final con datos de validación para el R134a. . . . .	47
12.	Gráfica de caja-bigotes para MSE de los modelos del R717. . . . .	54
13.	Gráfica de caja-bigotes para $R^2$ de los modelos del R717. . . . .	54
14.	Variación del MSE y $R^2$ promedio entre los modelos del R717. . . . .	55
15.	Regresión de datos reales vs calculados para la RNA del R290. . . . .	57
16.	Regresión de datos de validación para la RNA del R290. . . . .	58
17.	Presión real vs calculada para el R134a. . . . .	58
18.	Densidad de líquido real vs calculada para el R134a. . . . .	59

19.	Volumen de vapor real vs calculado para el R134a. . . . .	59
20.	Entalpía de líquido real vs calculada para el R134a. . . . .	60
21.	Entalpía de vapor real vs calculada para el R134a. . . . .	60
22.	Entropía de líquido real vs calculada para el R134a. . . . .	61
23.	Entropía de vapor real vs calculada para el R134a. . . . .	61
24.	Boxplot de MSE en modelos del R12. . . . .	72
25.	Boxplot de $R^2$ en modelos del R12. . . . .	72
26.	Variación del MSE promedio de los modelos del R12. . . . .	73
27.	Variación del $R^2$ promedio de los modelos del R12. . . . .	73
28.	Regresión final general de la RNA para el R12. . . . .	75
29.	Regresión final en validación de la RNA para el R12. . . . .	75
30.	Presión real vs RNA para el R12. . . . .	76
31.	$D_l$ real vs RNA para el R12. . . . .	76
32.	$D_v$ real vs RNA para el R12. . . . .	77
33.	$H_l$ real vs RNA para el R12. . . . .	77
34.	$H_v$ real vs RNA para el R12. . . . .	78
35.	$S_l$ real vs RNA para el R12. . . . .	78
36.	$S_v$ real vs RNA para el R12. . . . .	79
37.	Boxplot de MSE en modelos del R22. . . . .	80
38.	Boxplot de $R^2$ en modelos del R22. . . . .	81
39.	Variación del MSE promedio de los modelos del R22. . . . .	81
40.	Variación del $R^2$ promedio de los modelos del R22. . . . .	82

41.	Regresión final general de la RNA para el R22. . . . .	83
42.	Regresión final en validación de la RNA para el R22. . . . .	84
43.	Presión real vs RNA para el R22. . . . .	85
44.	$D_l$ real vs RNA para el R22. . . . .	85
45.	$D_v$ real vs RNA para el R22. . . . .	86
46.	$H_l$ real vs RNA para el R22. . . . .	86
47.	$H_v$ real vs RNA para el R22. . . . .	87
48.	$S_l$ real vs RNA para el R22. . . . .	87
49.	$S_v$ real vs RNA para el R22. . . . .	88
50.	Boxplot de MSE en modelos del R134a. . . . .	89
51.	Boxplot de $R^2$ en modelos del R134a. . . . .	90
52.	Variación del MSE promedio de los modelos del R134a. . . . .	90
53.	Variación del $R^2$ promedio de los modelos del R134a. . . . .	91
54.	Regresión final general de la RNA para el R134a. . . . .	92
55.	Regresión final en validación de la RNA para el R134a. . . . .	93
56.	Presión real vs RNA para el R134a. . . . .	94
57.	$D_l$ real vs RNA para el R134a. . . . .	94
58.	$D_v$ real vs RNA para el R134a. . . . .	95
59.	$H_l$ real vs RNA para el R134a. . . . .	95
60.	$H_v$ real vs RNA para el R134a. . . . .	96
61.	$S_l$ real vs RNA para el R134a. . . . .	96

62.	$S_v$ real vs RNA para el R134a. . . . .	97
63.	Boxplot de MSE en modelos del R290. . . . .	98
64.	Boxplot de $R^2$ en modelos del R290. . . . .	99
65.	Variación del MSE promedio de los modelos del R290. . . . .	99
66.	Variación del $R^2$ promedio de los modelos del R290. . . . .	100
67.	Regresión final general de la RNA para el R290. . . . .	101
68.	Regresión final en validación de la RNA para el R290. . . . .	102
69.	Presión real vs RNA para el R290. . . . .	103
70.	$D_l$ real vs RNA para el R290. . . . .	103
71.	$D_v$ real vs RNA para el R290. . . . .	104
72.	$H_l$ real vs RNA para el R290. . . . .	104
73.	$H_v$ real vs RNA para el R290. . . . .	105
74.	$S_l$ real vs RNA para el R290. . . . .	105
75.	$S_v$ real vs RNA para el R290. . . . .	106
76.	Boxplot de MSE en modelos del R717. . . . .	107
77.	Boxplot de $R^2$ en modelos del R717. . . . .	108
78.	Variación del MSE promedio de los modelos del R717. . . . .	108
79.	Variación del $R^2$ promedio de los modelos del R717. . . . .	109
80.	Regresión final general de la RNA para el R717. . . . .	110
81.	Regresión final en validación de la RNA para el R717. . . . .	111
82.	Presión real vs RNA para el R717. . . . .	112
83.	$D_l$ real vs RNA para el R717. . . . .	112

84.	$D_v$ real vs RNA para el R717. . . . .	113
85.	$H_l$ real vs RNA para el R717. . . . .	113
86.	$H_v$ real vs RNA para el R717. . . . .	114
87.	$S_l$ real vs RNA para el R717. . . . .	114
88.	$S_v$ real vs RNA para el R717. . . . .	115
89.	Boxplot de MSE en modelos del R410a. . . . .	116
90.	Boxplot de $R^2$ en modelos del R410a. . . . .	117
91.	Variación del MSE promedio de los modelos del R410a. . . . .	118
92.	Variación del $R^2$ promedio de los modelos del R410a. . . . .	119
93.	Regresión final general de la RNA para el R410a. . . . .	119
94.	$T_b$ real vs RNA para el R410a. . . . .	120
95.	$T_r$ real vs RNA para el R410a. . . . .	120
96.	$D_l$ real vs RNA para el R410a. . . . .	121
97.	$D_v$ real vs RNA para el R410a. . . . .	122
98.	$H_l$ real vs RNA para el R410a. . . . .	122
99.	$H_v$ real vs RNA para el R410a. . . . .	123
100.	$S_l$ real vs RNA para el R410a. . . . .	123
101.	$S_v$ real vs RNA para el R410a. . . . .	124
102.	Diagrama general de la metodología para elegir el mejor modelo. . . . .	125

# Índice de tablas

3.1. Datos de propiedades termodinámicas del refrigerante R410a. . . . .	41
3.2. Número de neuronas en la capa interna de cada modelo. . . . .	43
3.3. MSE y $R^2$ promedios para los modelos del R134a. . . . .	44
3.4. Prueba de Wilcoxon sobre los MSE de los modelos del R134a. . . . .	46
3.5. Valores de pesos y umbrales para la RNA del R410a. . . . .	52
3.6. Modelos elegidos para cada refrigerante. . . . .	53
3.7. Prueba de Wilcoxon para el R410a. . . . .	56
3.8. Errores porcentuales en cálculo de presión para el R12. . . . .	63
3.9. Errores porcentuales en cálculo de densidad de líquido para el R12. . . . .	64
3.10. Errores porcentuales en cálculo de volumen de vapor para el R12. . . . .	64
A.1. Prueba de Wilcoxon para el R12. . . . .	74
A.2. Prueba de Wilcoxon para el R22. . . . .	83
A.3. Prueba de Wilcoxon para el R134a. . . . .	91
A.4. Prueba de Wilcoxon para el R290. . . . .	100
A.5. Prueba de Wilcoxon para el R717. . . . .	110
A.6. Prueba de Wilcoxon para el R410a. . . . .	118
A.7. Valores de pesos y umbrales para la RNA del R12. . . . .	142
A.8. Valores de pesos y umbrales para la RNA del R22. . . . .	142

A.9. Valores de pesos y umbrales para la RNA del R134a. . . . .	142
A.10.Valores de pesos y umbrales para la RNA del R290. . . . .	143
A.11.Valores de pesos y umbrales para la RNA del R410a. . . . .	143
A.12.Valores de pesos y umbrales para la RNA del R717. . . . .	143

## INTRODUCCIÓN

El estudio de la naturaleza es por excelencia el campo de acción de la ciencia, en particular, la física y la química se enfocan en el comportamiento de la materia y las leyes que la rigen. La matemática como herramienta fundamental, les permite a ambas expresar características y magnitudes en términos de variables y cantidades. La elegancia que las matemáticas otorgan a la física radica no solamente en la expresión de ecuaciones, sino en que estas últimas brindan la posibilidad de descubrir nuevas relaciones entre las variables de un fenómeno dado. Tal es el caso, por ejemplo, del teorema de Green de la divergencia [20], cuyo planteamiento comienza en argumentaciones matemáticas y al aplicarse a una situación física se encuentra que describe perfectamente el flujo de distintos tipos de fluidos como los líquidos o el flujo magnético. Así pues, lo máspreciado para un estudioso de los fenómenos naturales se convirtió en llegar a expresar en una fórmula matemática la relación oculta que hay entre las variables del objeto de análisis. Las ecuaciones matemáticas que se asocian a un fenómeno son también una maquinaria potente que ayuda a desarrollar un aspecto muy importante del quehacer científico: la predicción. De hecho una ecuación prueba su “poder” a partir de la exactitud con la que predice los valores de las variables involucradas en cierto fenómeno.

En teoría es posible construir una ecuación que describa cualquier fenómeno físico. Pero a medida que se profundiza en el análisis de un acontecimiento dado, generalmente se encuentran nuevas variables o relaciones entre ellas que antes no se habían manifestado. Esto introduce nuevos términos en las ecuaciones y en muchos casos conducen a expresiones que no se pueden resolver, ya sea por su complejidad o por no existir algún método apropiado. También existen casos en los que existen tantas variables que influyen en el fenómeno, que es imposible conocer con certeza cómo afecta cada una a

las demás. Todo lo anterior conduce a la idea de que debe haber otra forma de plantear matemáticamente las relaciones entre las variables de un acontecimiento físico. Esta forma de la que se habla es precisamente la Inteligencia Artificial. Esta nueva herramienta matemática tiene la ventaja de que incorpora procesos de “aprendizaje” que son llevados a cabo dentro de “neuronas” artificiales que forman redes organizadas en capas y se interconectan de la misma forma en la que las conexiones de las neuronas en nuestro cerebro lo hacen. Si bien se siguen utilizando ecuaciones, estas ya no son el centro del modelado del problema, sino que ahora lo que se busca es encontrar parámetros de ajuste para dichas ecuaciones, de tal forma que los valores de salida correspondan con los valores “reales” del fenómeno. Se puede pensar en la Red Neuronal Artificial (RNA) como una “caja negra” donde ocurren ciertos procesos que transforman los datos de entrada en los de salida. Lo importante entonces es encontrar la estructura apropiada de dicha RNA para que pueda representar correctamente las relaciones entre las variables de entrada y las de salida. En este trabajo se aplican las Redes Neuronales Artificiales a la predicción de propiedades termodinámicas de gases refrigerantes y se plantea una metodología para la correcta construcción de dichas redes.

El conocimiento del valor de las propiedades termodinámicas de refrigerantes es importante debido a que permite obtener parámetros de diseño de sistemas y equipos de climatización centralizados. También permite realizar la simulación y optimización de dichos sistemas. Para calcular las propiedades termodinámicas de un compuesto es común el uso de ecuaciones de estado (EoS, por sus siglas en inglés), y generalmente, llevar a cabo un proceso iterativo donde también es común encontrar dificultades de convergencia. Las RNAs son capaces de reproducir con gran aproximación los valores de las propiedades termodinámicas a partir del “aprendizaje” de datos reales, que son utilizados para obtener constantes de una ecuación, que si bien no tiene relación alguna con ningún modelo físico, sirve para reproducir adecuadamente dichos datos reales e

incluso predecir aquellos para los que no hay información. Además pueden utilizarse dentro de dispositivos programables (como los PLC, «programable logic controller») para medir variables de un proceso y optimizarlo en tiempo real, lo cual puede ser más difícil si se usa una EoS.

Por lo anteriormente mencionado, en este trabajo se propone sustituir las EoS por RNAs en el proceso de cálculo de las propiedades termodinámicas de refrigerantes, para lo cual, se desea encontrar la metodología que permita estructurar dichas redes, de forma que su capacidad predictiva sea la mejor posible.

Al utilizar la inteligencia artificial se abre la posibilidad de la optimización de equipos de climatización centralizada en tiempo real. Todo esto se traduce en ahorro energético y la reducción del impacto al medio ambiente.

Derivado del análisis del estado del arte se puede notar que los diferentes investigadores que han aplicado RNAs al cálculo de propiedades termodinámicas de sustancias no han propuesto o documentado (al menos públicamente) la metodología que siguieron para encontrar los modelos de RNAs que les funcionaron mejor. Es por ello que en este estudio de doctorado se plantea definir la metodología que permita obtener el modelo óptimo de RNAs para calcular las propiedades termodinámicas de refrigerantes.

Por lo anteriormente mencionado, se propone el siguiente objetivo: Desarrollar una metodología que permita aplicar herramientas de inteligencia artificial en la obtención de propiedades termodinámicas de refrigerantes.

Como objetivos específicos se plantean los siguientes:

Encontrar el mejor tratamiento previo de los datos para su correcta aplicación dentro de la RNA;

Establecer el mecanismo que permita encontrar el óptimo número de capas y de neuronas;

Realizar un análisis estadístico de los datos de las RNAs que permita verificar que son adecuadas para predecir las propiedades termodinámicas de los refrigerantes elegidos.

# Capítulo 1

## Antecedentes

Los sistemas de acondicionamiento de aire son de enorme importancia en regiones cálidas debido a que permiten tener un ambiente confortable en casas habitación y edificaciones públicas. El proceso de refrigeración también se aplica en la conservación de alimentos, la producción de hielo, algunos procesos industriales y hasta en los automóviles. Los equipos que se utilizan con este propósito funcionan gracias a la compresión y expansión cíclica de una sustancia con propiedades especiales comúnmente conocida como «refrigerante» o «fluido de trabajo». A pesar de la utilidad que representa el uso de refrigerantes, hace décadas se descubrió que algunos de ellos destruían la capa de ozono que cubre nuestra atmósfera y nos mantiene protegidos de la radiación ultravioleta proveniente del sol. Este hecho fue el punto de partida para la búsqueda de nuevos refrigerantes e hizo transitar del uso de hidrocarburos halogenados puros hacia combinaciones de ellos, de manera que alcancen un factor cero de daño al ozono de acuerdo con el Protocolo de Montreal firmado en 1987. Este es el caso de la sustitución del R-12 y R-22, diclorodifluoro metano y clorodifluoro metano respectivamente, por otros refrigerantes como el R-134a (1,1,1,2-tetrafluoroetano) o el R410a, formado por R-32 (difluorometano) y R-125 (pentafluoroetano) mezclados en proporciones iguales.

Para el diseño de los equipos de refrigeración se requieren conocer las propiedades termodinámicas de los refrigerantes, tales como presión (P), temperatura (T), volu-

men ( $V$ ), entalpía ( $H$ ) y entropía ( $S$ ). Para conocer dichas propiedades se disponen de ecuaciones de estado, que generalmente son ecuaciones cúbicas cuya resolución puede volverse complicada debido a que es difícil garantizar la correcta convergencia de los procesos iterativos involucrados. Por tal motivo, se presenta una forma alternativa de calcular dichas propiedades termodinámicas sin el uso de ecuación de estado: utilizar redes neuronales artificiales.

Las redes neuronales artificiales (RNAs) son capaces de imitar en cierta medida a las redes neuronales biológicas que se encuentran en nuestro cerebro, sólo que lo hacen dentro de una computadora. A partir de datos experimentales, se estructura una RNA para que «aprenda» la relación que guardan las diferentes variables y así, pueda reproducir los datos de entrada y salida tan acertadamente como lo haría una ecuación. Se puede pensar en las redes neuronales como algo semejante a una regresión multivariable donde puede haber relaciones no-lineales y los parámetros se adaptan dinámicamente para reflejar mejor la relación entre las distintas variables. Al final del proceso de estructuración de una RNA, se obtienen coeficientes que se aplican a una función sencilla, que no tiene nada que ver con una ecuación determinista donde se aplican leyes o restricciones físicas propias del fenómeno. A pesar de no tener como resultado una ecuación acorde con las leyes de la física, la función obtenida de la RNA puede representar, incluso mejor que una ecuación, la relación entre las variables. Así pues, el uso de una RNA en lugar de una ecuación de estado presenta las siguientes ventajas:

- Proveen una forma sencilla de obtener información sobre nuevos escenarios a partir de valores iniciales de las variables;
- Poseen la capacidad de adaptarse internamente para «aprender» cómo se relacionan las diferentes variables;

- Los cálculos pueden realizarse en forma paralela, permitiendo controlar dispositivos «en tiempo real» por lo que es posible la optimización constante de procesos.

Debido al éxito que ha representado la aplicación de las RNA's, en años recientes se han utilizado en diferentes áreas de la ingeniería tales como la predicción de propiedades termodinámicas [11], [22], [19], predicción de presiones de vapor de compuestos orgánicos e inorgánicos [13], [10], la estimación de propiedades de sustancias en equilibrio líquido-vapor [2], también [23] y [1], así como evaluar instalaciones centralizadas de climatización [5]. Algunos otros casos involucran la aplicación de las RNAs a situaciones más complejas como el análisis exergético [25], optimización en tiempo real de turbinas de gas para sistemas de propulsión [3] y predicción de la demanda eléctrica horaria [14]. Así pues, se puede decir que continuar con la investigación en la implementación de redes neuronales es una tarea innovadora e importante ya que permitirá facilitar el cálculo de propiedades termodinámicas de compuestos, optimizar sistemas y, en algunos casos, mejorar la predicción de propiedades en comparación con las EoS. La única desventaja que podría mencionarse, es el hecho de que una RNA no brinda información nueva que contribuya a entender cómo las propiedades de un átomo y la geometría molecular contribuyen al comportamiento físico y termodinámico de una sustancia y plasmar todo esto en una ecuación determinista y elegante.

Un documento clave es el artículo de Adnan Sözen [22], *Calculation for the thermodynamic properties of an alternative refrigerant (R508b) using artificial neural network*, en el cual se muestra cómo estructurar diferentes RNAs para obtener las propiedades termodinámicas del R508b a partir de datos experimentales. Debido a la cantidad de datos disponibles en dicho experimento, no fue necesario emplear una técnica de remuestreo. Cabe resaltar que Sözen utiliza la misma estructura de RNA que se empleó en este estudio, pero la forma de encontrar el mejor modelo de RNA es diferente. En el artículo *Estimating the Saturation Thermodynamic Properties of Propene*

*Using a Feed Forward Neural Network* [19], Sheik Setareh utiliza una red neuronal tipo «*feedforward*» con dos capas ocultas para calcular la capacidad calorífica, el coeficiente de Joule-Thompson, la viscosidad, la conductividad térmica y la tensión superficial del propeno. La estructura más compleja de dicha RNA permite ver su flexibilidad de adaptación, sin embargo, los resultados mostraron que sólo las tres primeras propiedades mencionadas son calculadas satisfactoriamente por la red. En el artículo de Martin Mandischer, *Neural Networks and Evolutionary Algorithms for the prediction of thermodynamic properties for chemical engineering* [15] se expone la forma de aplicar los algoritmos genéticos al cálculo de la entalpía de vaporización del etano, comparando los resultados con complicados modelos matemáticos motivados en análisis físico. Mandischer concluye que la aplicación de la inteligencia artificial en su caso de estudio puede generar incluso mejores resultados que el modelo físico disponible. En el trabajo publicado por Farhad Gharagheizi [11] titulado *Prediction of Enthalpy of Fusion of Pure Compounds Using an Artificial Neural Network-Group Contribution Method*, se utiliza una RNA desarrollada en Matlab para calcular la entalpía de fusión de 4,157 compuestos químicos. Para ello separa dichos compuestos en 130 grupos de similar estructura química y después busca la relación entre el grupo funcional químico y el valor de su entalpía de fusión. Al final restringe la utilización del modelo generado a la predicción de la entalpía de fusión de compuestos químicos similares a los que se manejan en su estudio. Respecto al cálculo de propiedades en el equilibrio líquido-vapor, se destaca el trabajo de Vashishtha Manish [23] y los de Moghadassi Abdolreza [1] [2] donde se comparan los resultados de utilizar las ecuaciones de estado de Lee Kesler, NRTL, Soave-Redlich-Kwong y Peng Robinson con los obtenidos por una RNA. Al final de su trabajo, Moghadassi concluye que las RNAs predicen los datos experimentales reportados para los compuestos que él seleccionó, tan satisfactoriamente como lo

hacen las ecuaciones de estado. Lo anterior sirve como buena referencia para el presente estudio.

En cuanto a la metodología para elegir el mejor modelo de RNA, en este estudio se propone el Análisis de Rangos con Signo de Wilcoxon. Esto es en base a diferentes estudios publicados. El artículo de Pizarro titulado *A statistical model selection strategy applied to neural networks* [16] se manifiesta la necesidad de comparar las medias de error de los modelos de RNA que en él se comparan. Para ello propone un análisis de varianza. De igual modo Rezequi en su artículo titulado *Statistical Model Selection Based on Resampling Procedure for Neural Network Classification* [17], se aborda el estudio de la mejor técnica para elegir el mejor modelo de RNA del tipo feedforward, y particularmente se enfoca en una estructura similar a la de este estudio, es decir, con una sola capa oculta. Además aplica remuestreo de datos y aplica una hipótesis y prueba estadística para llevarlo a cabo. Un elemento clave es que menciona que es posible aplicar también pruebas estadísticas no paramétricas en la selección de modelos de RNA, dando pie a que en el presente estudio se propusiera la prueba no paramétrica de Wilcoxon.

Con los estudios anteriormente mencionados se establece la base necesaria y suficiente para sustentar el desarrollo teórico de los modelos de RNA en este trabajo, así como la correspondiente metodología de selección estadística.

# Capítulo 2

## Marco Teórico

### 2.1. Propiedades termodinámicas

La termodinámica estudia los fenómenos que involucran la transferencia o transformación de energía. Para ello, analiza las variables que caracterizan dichos fenómenos y estudia la relación que hay entre ellas definiendo un *sistema*, que es una cantidad de materia o región del espacio separada de su entorno por una barrera real o imaginaria. Dicha barrera se le denomina *frontera*, y puede ser fija o móvil. Los sistemas pueden ser *abiertos* si la materia y energía puede entrar a ellos, pueden ser *cerrados*, si solo la energía puede penetrar y por último, pueden ser *aislados* si no pueden intercambiar ni materia ni energía con sus alrededores. Cuando se lleva a cabo un análisis termodinámico, la elección del sistema es muy importante debido a que las relaciones entre las variables termodinámicas son diferentes entre los sistemas cerrados y abiertos.

Los sistemas poseen diferentes características, denominadas *propiedades*, tales como la presión ( $P$ ), temperatura ( $T$ ), volumen ( $V$ ), masa ( $m$ ), entre otras. Algunas de estas propiedades son independientes entre ellas y otras dependen de algún valor de las demás. Por ejemplo, en un sistema que se encuentra en estado de saturación, al establecer un valor para la presión, automáticamente se obtienen los valores de temperatura, volumen específico y las demás propiedades termodinámicas. El número de variables  $F$  que se deben especificar para poder conocer todas las demás, cuando exis-

ten  $N$  especies químicas y  $\pi$  fases, se determina por la *regla de las fases* [21] reflejada en la Ec. 2.1:

$$F = 2 - \pi + N \quad (2.1)$$

De acuerdo con esta regla, si por ejemplo, en un contenedor se tiene agua líquida (una especie química,  $N = 1$ ) en equilibrio con su vapor (2 fases,  $\pi = 2$ ), se requiere  $F = 2 - 2 + 1 = 1$ , es decir, fijar arbitrariamente o conocer una sola propiedad, para obtener los valores de todas las demás. Esto significa que el sistema queda completamente determinado.

Hay propiedades que se definen en términos de otras, por ejemplo, el volumen específico es el volumen por unidad de masa  $v = V/m$  y la densidad es la masa por unidad de volumen  $\rho = m/V$ . Cuando un sistema no experimenta ningún cambio, se dice que se encuentra en un **estado de equilibrio** y es posible obtener los valores de sus propiedades. Si el sistema cambia, el estado será diferente y se definirá por nuevos valores de sus propiedades. El conocer las propiedades en cierto estado permite conocer el comportamiento físico y químico de un sistema, por ejemplo, si un sistema contiene agua y se encuentra a una temperatura de  $100^\circ\text{C}$ , automáticamente queda determinado el valor para la presión, volumen específico y otras propiedades. Un sistema puede experimentar cambios y cuando pasa de un estado a otro, se dice que ha ocurrido un **proceso**. Si el proceso regresa a su estado inicial, entonces se trata de un **ciclo**.

## 2.2. Refrigeración

Un ciclo en particular que interesa en este estudio es el que permite extraer calor de cierto espacio para disminuir su temperatura. El ciclo ideal para refrigerar fue propuesto por Carnot como puede verse en la Fig. 1.

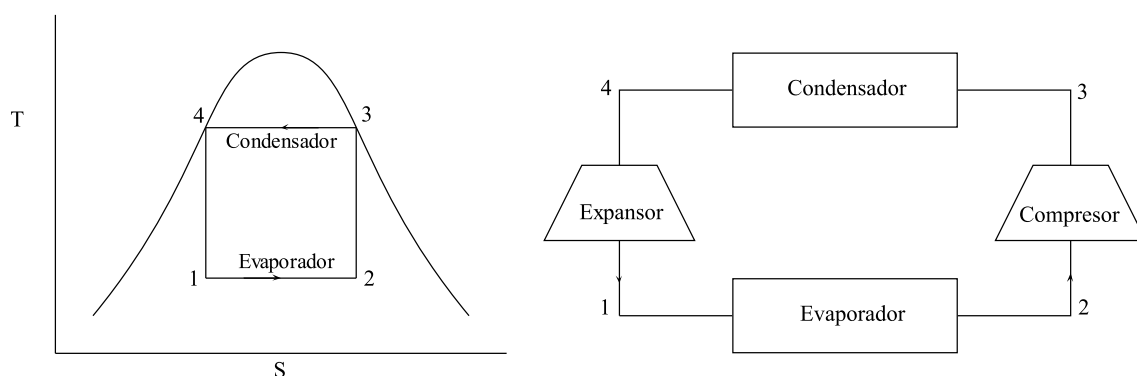


Figura 1: Refrigerador bifásico de Carnot.

En teoría se puede utilizar cualquier sustancia para absorber y expulsar el calor, a dicha sustancia se le denomina «fluido de trabajo». En él se puede observar que el fluido de trabajo absorbe calor en el proceso 1-2, aumentando su entropía ( $S$ ). La entropía es una medida cuantitativa del desorden microscópico de un sistema [9]. Posteriormente, en el proceso 3-4 se desecha ese calor. Ambos procesos ocurren a temperatura y presión constantes. Dado a que la evaporación y condensación son excelentes mecanismos de transferencia de calor, se escoge un fluido de trabajo que se pueda evaporar y condensar fácilmente en las condiciones de operación del ciclo. Así pues, cuando el fluido de trabajo absorbe calor, cambia a su fase vapor y después se le vuelve a comprimir para condensar todo lo que se pueda, trabajo que termina en el condensador, donde rechaza calor y sale en estado totalmente líquido. Aunque se podría operar este ciclo con una mezcla bifásica de líquido-vapor en todos sus procesos, la compresión de dicha mezcla sería un tanto problemática y generalmente se evita. Este problema se puede eliminar si el refrigerante se deja evaporar totalmente alcanzando la condición de saturación en el evaporador, tal como se muestra en la Fig. 2.

En comparación con la Fig. 1, se puede ver que el punto 2 se ha movido a la derecha hasta que toca la curva de saturación. Posteriormente, en el proceso 2-3, de la misma Fig. 2, se observa una compresión a entropía constante acompañada de un aumento de

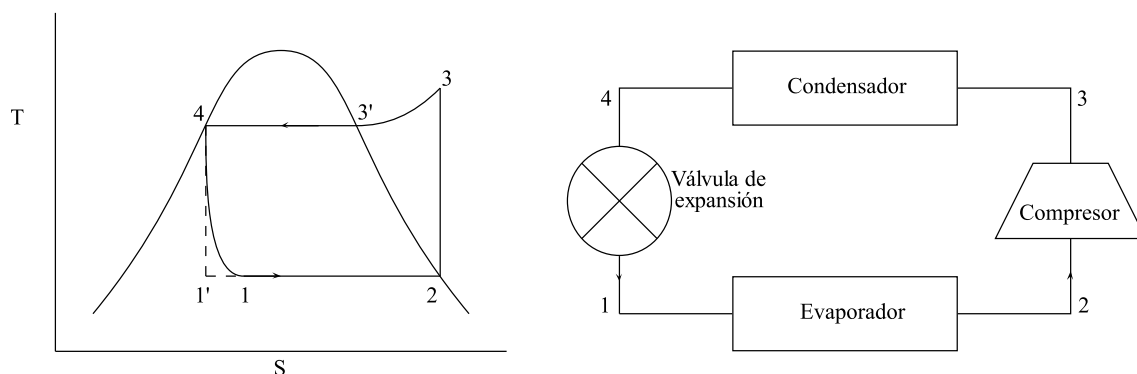


Figura 2: Ciclo de refrigeración por compresión de vapor.

la temperatura del refrigerante, que termina en 3 como un vapor sobre calentado. Esto es necesario para que el proceso de enfriamiento 3-3' se efectúe a presión constante sin requerimiento de trabajo. Esto ocurre en el primer tramo del condensador y dado a que este proceso de eliminación de calor no involucra cambio de fase, es menos eficiente y se busca que se produzca al mínimo. En la expansión que ocurre en el proceso 4-1, se produce trabajo útil. Sin embargo, el aprovechamiento involucraría la operación de un expansor reversible, que requiere mantenimiento y genera un costo adicional de operación. Por ello, en vez de aprovechar esta pequeña cantidad de trabajo, se utiliza una válvula de expansión isentrópica o un tubo capilar, que son más baratos, no requieren mantenimiento y pueden operar con mezcla líquido-vapor sin problemas.

El proceso de estrangulamiento es irreversible, por lo tanto, hay un incremento de entropía, tal como se ve en la Fig. 2 en la diferencia entre la trayectoria de la línea isentrópica 4-1' y la línea real que es la que va de 4 a 1. Dado que la distancia 1-2 es menor que la 1'-2, la cantidad de calor que puede retirar el proceso es menor que la ideal, sin embargo, esta pérdida de eficiencia es compensada por las consideraciones prácticas mencionadas con anterioridad. Debido a que en varios puntos del ciclo se requiere conocer los valores de la entalpía, es común representar este ciclo en un diagrama Presión-Entalpía (P-H), como el que se aprecia en la Fig. 3.

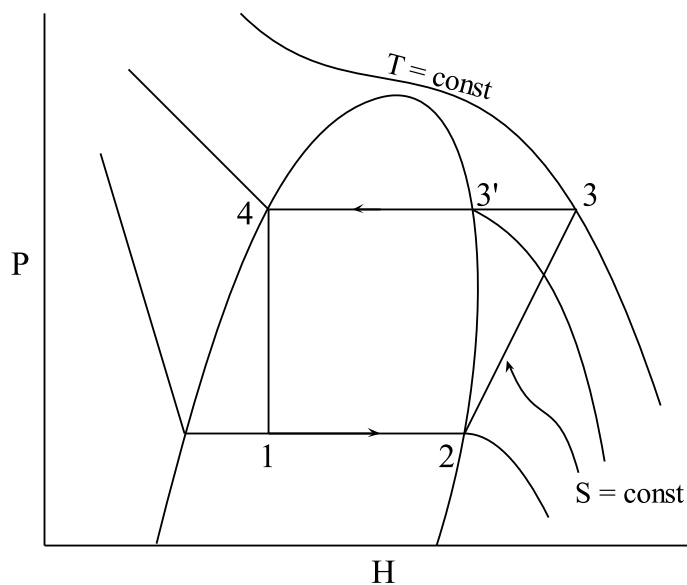


Figura 3: Ciclo de refrigeración por compresión de vapor en diagrama  $P-H$ .

El ciclo de refrigeración por compresión de vapor se puede resumir de la siguiente forma [6]:

1. Entrada del refrigerante como vapor al compresor, y sale a una temperatura relativamente alta.
2. Entra al condensador para enfriarse y condensarse liberando calor del espacio refrigerado.
3. Entra a un tubo capilar, donde se estrangula, por lo que su temperatura y presión descienden drásticamente.
4. Entra al serpentín del evaporador, donde absorbe calor del medio a enfriar para evaporarse.
5. Por último, el refrigerante sale del evaporador y entra al compresor, completando así el ciclo.

En este ciclo se requiere conocer los valores de las entalpías en diferentes puntos, a partir de los valores de presión o temperatura. Para ello, normalmente se consulta un diagrama  $P-H$  para el fluido de trabajo dado. Sin embargo, estos datos pueden ser obtenidos a partir de valores reales reportados por diferentes fuentes en la literatura abierta, o utilizando una red neuronal artificial. Esto se tratará más adelante.

De acuerdo con la Fig. 3, para obtener la cantidad de calor (por unidad de masa) que se puede retirar del espacio a refrigerar  $\bar{Q}_L$  se requiere conocer  $\bar{H}_1$  y  $\bar{H}_2$  (entalpías inicial y final del refrigerante a través del evaporador), de acuerdo con la siguiente ecuación:

$$\bar{Q}_L = \bar{H}_2 - \bar{H}_1 \quad (2.2)$$

Para conocer el trabajo que hay que suministrar al compresor  $\dot{W}$  es necesario conocer también la entalpía en el punto 3, ya que está dado por la siguiente ecuación:

$$\dot{W} = \dot{M}\bar{W} = \dot{M}\Delta\bar{H} = \dot{M}(\bar{H}_3 - \bar{H}_2) \quad (2.3)$$

Finalmente, el calor que debe ser expulsado en el condensador  $\dot{Q}_H$  se puede obtener por la diferencia entre el trabajo suministrado  $\dot{W}$  y el calor que se debe absorber del espacio refrigerado  $\dot{Q}_L$  con la siguiente ecuación:

$$\dot{Q}_H = \dot{W} - \dot{Q}_L \quad (2.4)$$

En la Fig. 4 se puede apreciar un diagrama  $P-H$  con los valores de las propiedades termodinámicas del refrigerante Freón 22 que se requieren para obtener los anteriores datos y poder seleccionar un equipo. Las unidades en esta Fig. están expresadas en psia para la presión, grados Fahrenheit para la temperatura, Btu/lb<sub>m</sub> para la entalpía y Btu/lb<sub>m</sub> °R para la entropía.

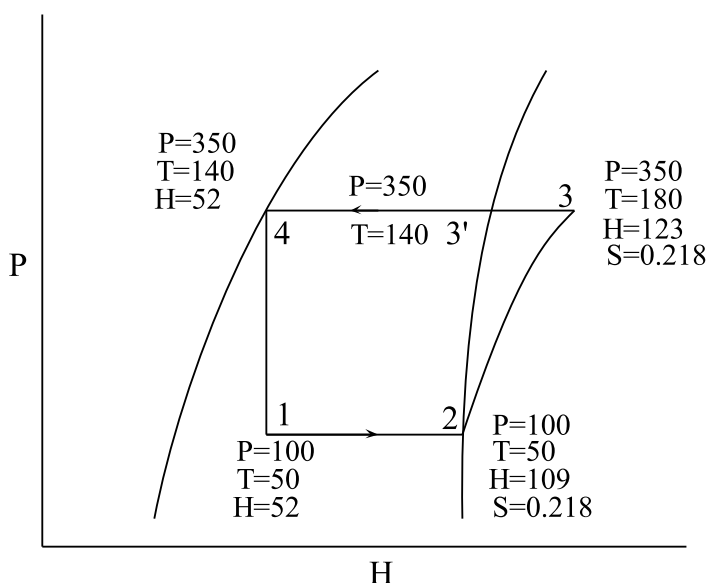


Figura 4: Ejemplo de ciclo de refrigeración en diagrama  $P-H$ .

Los valores mostrados en la Fig. 4 se obtienen directamente de la lectura del diagrama de Mollier para el R22 que se aprecia en la Fig. 5. Para utilizar dicho diagrama, es necesario conocer los valores de dos variables intensivas, por ejemplo, la temperatura y la presión, posteriormente se dibuja un punto en el diagrama y se localizan las líneas de entalpía, entropía, volumen específico y calidad que pasen cerca de ese punto. Este procedimiento es sencillo, pero a veces puede ser confuso ubicar los valores correctos de las propiedades. Para evitar errores de lectura, se pueden usar ecuaciones de estado o bien, como se propone en el presente estudio, emplear RNAs.

### 2.3. Ecuaciones de Estado (EoS)

Como ya se mencionó en la sección 2.2, es necesario calcular los valores de las propiedades termodinámicas para un ciclo de refrigeración si se desea obtener parámetros de diseño como la potencia necesaria en el compresor o la cantidad de calor que puede retirar el sistema desde el espacio a refrigerar. Además de las tablas de datos de los refrigerantes y diagramas de Mollier, se pueden emplear EoS.

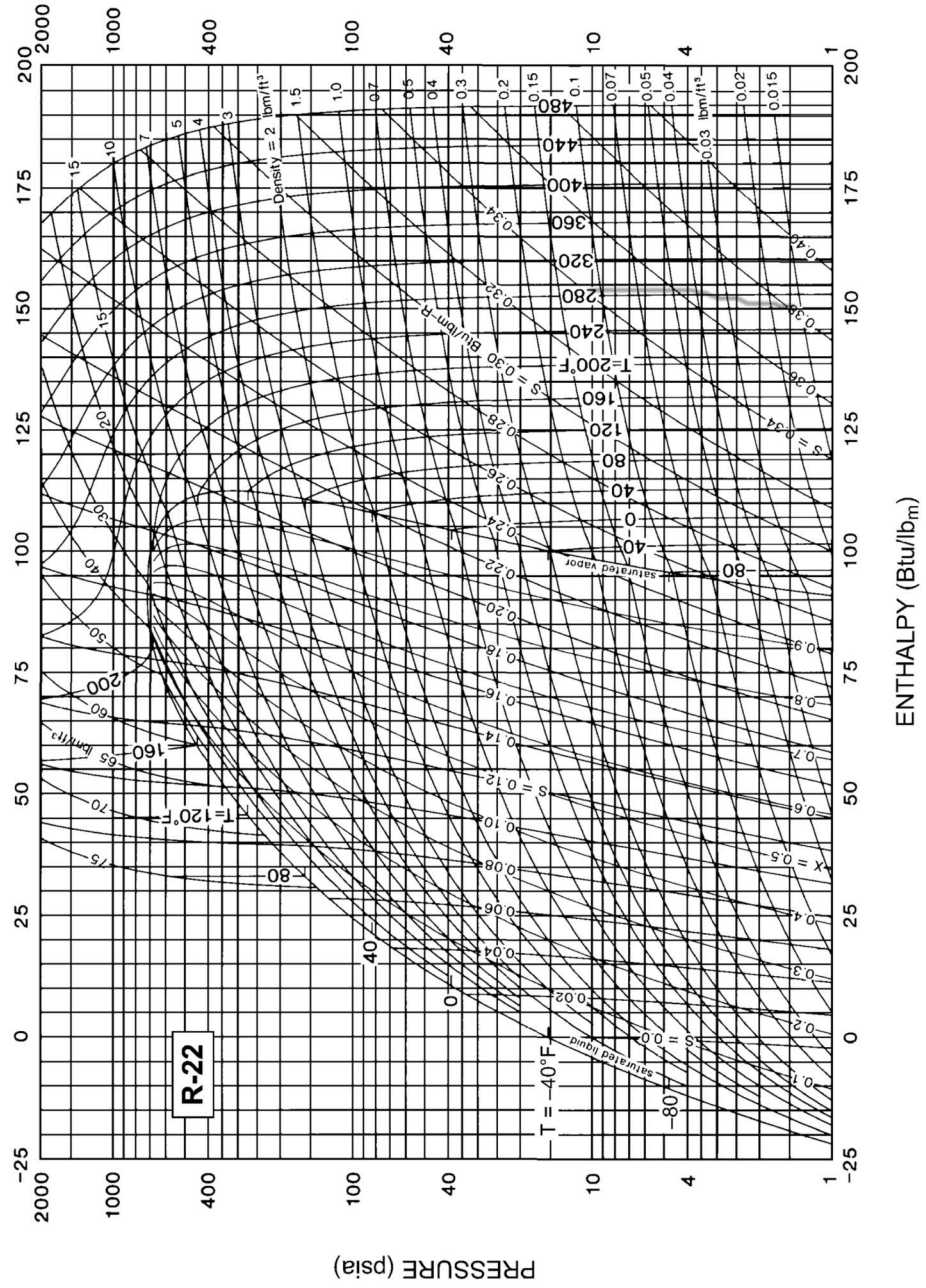


Figura 5: Diagrama P-H del R22.

Una EoS representa la relación que existe entre la presión, temperatura y volumen específico de una sustancia. La ecuación más sencilla es la *ecuación del gas ideal*:  $P\bar{V} = RT$ . En ella se incluye la constante universal de los gases  $R$  que es independiente del gas en cuestión. A menos que haya condiciones de baja presión o altas temperaturas, la ecuación del gas ideal no podrá ser conveniente para calcular correctamente la presión, temperatura o volumen específico. Debido a ello, se propuso agregarle un término adicional conocido como el *factor de compresibilidad*  $Z$  que representa la desviación de la idealidad. Así pues la nueva ecuación, conocida como *ecuación del gas real*, toma la forma de la Ec. 2.5.

$$P\bar{V} = ZRT \quad (2.5)$$

Con el fin de calcular los valores de  $Z$  para cualquier gas, se emplean normalmente las *propiedades críticas* ( $\bar{V}_{cr}, T_{cr}, P_{cr}$ ) del compuesto dado, que se reportan en la literatura [9]. Dichas propiedades críticas son la base para el establecimiento del *teorema de estados correspondientes* con dos parámetros [21, pág.98], que se enuncia como sigue:

*Todos los fluidos, cuando se les compara a la misma temperatura y presión reducidas, tienen aproximadamente el mismo factor de compresibilidad y todos se desvían del comportamiento de gas ideal de manera semejante.*

Las *propiedades reducidas* ( $\bar{V}_r, T_r, P_r$ ) a las que se refiere dicho teorema se obtienen con las siguientes fórmulas:

$$\bar{V}_r = \bar{V}/V_{cr} \quad (2.6a)$$

$$T_r = T/T_{cr} \quad (2.6b)$$

$$P_r = P/P_{cr} \quad (2.6c)$$

Es importante notar que aunque los resultados mejoran cuando se utilizan ecuaciones basadas en el teorema de estados correspondientes, solamente fluidos simples como el argón, kriptón y xenón, se ajustan a dicho teorema. Para los demás gases se presentan desviaciones importantes, lo que pone de manifiesto la necesidad de buscar otro parámetro de ajuste. Los estudios de K. S. Pitzer lo llevaron a proponer un tercer parámetro, conocido como *factor acéntrico* ( $\omega$ ), que se define en base a la presión de vapor de una especie pura dada. Pitzer encontró que el logaritmo de la presión de vapor de un fluido puro ( $\log P_r^{sat}$ ) variaba en forma aproximadamente lineal con respecto al recíproco de la temperatura reducida y que de acuerdo con el teorema de los estados correspondientes, esta relación se debería cumplir en general para todos los fluidos puros (Ec. 2.7). Sin embargo, Pitzer notó que cuando se hace una gráfica de  $\log P_r^{sat}$  contra el inverso de la temperatura reducida, los fluidos simples (Ar, Kr y Xe) los datos de los tres se encuentran sobre la misma línea y que en  $T_r = 0.7$  se tenía que  $\log P_r^{sat} = -1.0$ .

$$a = \frac{d \log P_r^{sat}}{d(1/T)} \quad (2.7)$$

El factor acéntrico se define precisamente como la diferencia del valor del  $\log P_r^{sat}$  entre un fluido puro (-1.0) y el fluido en cuestión (Ec. 2.8), por lo que puede determinarse para cualquier fluido a partir de  $T_c$ ,  $P_c$  y una sola medición del valor de la presión de vapor realizada a  $T_r = 0.7$ . Todo esto sienta las bases para postular el teorema de estados correspondientes con tres parámetros:

*Todos los fluidos que tienen el mismo valor de  $\omega$  cuando se les compara a la misma  $T_r$  y  $P_r$ , tienen el mismo valor de  $Z$  y todos se alejan del comportamiento de gas ideal de manera parecida.*

$$\omega \equiv -1.0 - \log(P_r^{sat})_{T_r=0.7} \quad (2.8)$$

Con las propiedades críticas calculadas, se pueden calcular las demás propiedades a partir de gráficas para obtener  $Z$  que se haya reportado en la literatura [9]. El factor acéntrico se utilizar para el cálculo del factor de compresibilidad, tal como se ejemplifica con la ecuación de Soave-Redlich-Kwong (Ec. 2.9), donde  $\epsilon$ ,  $b$  y  $\sigma$  son constantes específicas de dicha ecuación y la constante  $\alpha$  se obtiene a partir del factor de compresibilidad con la Ec. 2.10.

$$Z_i = \frac{V}{V - b} - \frac{\Omega\alpha VR^2T_c^2}{RTP_c(V + \epsilon b)(V + \sigma b)} \quad (2.9)$$

$$\alpha = [1 + (0.480 + 1.574\omega - 0.176\omega^2)(1 - T_r^{1/2})]^2 \quad (2.10)$$

Las ecuaciones del gas ideal, gas real (Ec. 2.5) y Soave-Redlich-Kwong (Ec. 2.9), no son las únicas que sirven para calcular la presión, volumen o temperatura. También se encuentra la *ecuación virial*, que está expresada en series de potencias de los *coeficientes viriales* ( $B, C, D, \dots$ ) y puede tener la forma de la Ec. 2.11a o Ec. 2.11b.

$$Z \equiv \frac{P\bar{V}}{RT} = 1 + B'P + C'P^2 + D'P^3 + \dots \quad (2.11a)$$

$$Z = 1 + \frac{B}{V} + \frac{C}{V^2} + \frac{D}{V^3} + \dots \quad (2.11b)$$

A medida que se desea aplicar la ecuación virial a un intervalo más amplio de presiones, el número de coeficientes viriales empleados debe ser mayor, sin embargo, se sabe poco sobre los terceros coeficientes viriales ( $D$ ). Normalmente la ecuación virial se utiliza truncada al segundo término, de manera que esta toma la forma de la Ec.

2.12. Adicionalmente, la ecuación virial truncada en su tercer coeficiente reproduce aceptablemente el comportamiento PVT de una gran cantidad de sustancias hasta presiones moderadas.

$$Z = \frac{PV}{RT} = 1 + \frac{B}{V} + \frac{C}{V^2} \quad (2.12)$$

Si se desea que sea explícita en presión, toma la forma de la Ec. 2.13a donde también se puede ver que es una ecuación cúbica en volumen. Sin embargo, es necesario encontrar una expresión para conocer el volumen específico. Para ello se podría resolver mediante el procedimiento común para las ecuaciones cúbicas, sin embargo, por simplicidad, se prefiere escribirla en la forma de la Ec. 2.13b de manera que se pueda llevar a cabo un proceso iterativo.

$$P = RT \left( \frac{1}{V} + \frac{B}{V^2} + \frac{C}{V^3} \right) \quad (2.13a)$$

$$V_{i+1} = \frac{RT}{P} \left( 1 + \frac{B}{V_i} + \frac{C}{V_i^2} \right) \quad (2.13b)$$

Aunque la ecuación virial proporciona una aproximación mucho más adecuada que la ecuación del gas ideal para representar el comportamiento PVT de una sustancia, es necesario encontrar ecuaciones más completas, también cúbicas en volumen, que sean suficientemente generales para aplicarlas a las fases líquido y vapor, así como intervalos amplios de temperatura y presión. Así pues, se han propuesto una serie de ecuaciones de estado, comenzando por Johannes Diderik van der Waals (Ec. 2.14), pasando por la ecuación de Redlich/Kwong (Ec. 2.15) y concluyendo con el modelo general para las ecuaciones cúbicas de estado (Ec. 2.16). Los parámetros  $b, \theta, \delta, \epsilon,$  y  $\eta$

generalmente dependen de la temperatura y en caso de mezclas de especies químicas, de la composición.

$$\text{(van der Waals)} \quad P = \frac{RT}{V-b} - \frac{a}{V^2} \quad (2.14)$$

$$\text{(Redlich/Kwong)} \quad P = \frac{RT}{V-b} - \frac{a}{T^{1/2}V(V+b)} \quad (2.15)$$

$$\text{(Modelo general)} \quad P = \frac{RT}{V-b} - \frac{\theta(V-\eta)}{(V-b)(V^2 + \delta V + \epsilon)} \quad (2.16)$$

Existen ecuaciones más complejas y más precisas, tal como la ecuación de Benedict/Webb/Rubin:

$$P = \frac{RT}{V} + \frac{B_0RT - A_0 - C_0/T^2}{V^2} + \frac{bRT - a}{V^3} + \frac{a\alpha}{V^6} + \frac{c}{V^3T^2} \left(1 + \frac{\gamma}{V^2}\right) \exp \frac{-\gamma}{V^2} \quad (2.17)$$

Esto apunta a pensar que si se desea más exactitud en el cálculo de propiedades de sustancias, será necesario generar nuevas ecuaciones que involucren más términos y esto hace más difícil su resolución y aplicación. Por este motivo, se requiere encontrar otro método que sea tan efectivo y útil como las ecuaciones de estado. Este método es precisamente la aplicación de redes neuronales artificiales.

### 2.3.1. Aplicación de las Ecuaciones de Estado al ciclo de refrigeración

El ciclo de refrigeración que se mencionó en la sección 2.2 se diseña en base a los valores de  $P, T, H$  y  $S$ . Dichos valores pueden ser calculados a partir de EoS. Para ello se definen las *propiedades residuales* de acuerdo a la siguiente ecuación:

$$M^R \equiv M - M^{gi} \quad (2.18)$$

En donde la propiedad residual  $M^R$  es la diferencia entre el valor real  $M$  y el valor calculado para el gas ideal  $M^{gi}$ . Así pues se pueden obtener expresiones para la entalpía y entropía residuales [21, p.216] como las que se aprecian en la Ec. 2.19 y Ec. 2.20 considerando la temperatura constante.

$$\frac{H^R}{RT} = -T \int_0^P \left( \frac{\partial Z}{\partial T} \right)_P \frac{dP}{P} \quad (2.19)$$

$$\frac{S^R}{RT} = -T \int_0^P \left( \frac{\partial Z}{\partial T} \right)_P \frac{dP}{P} - \int_0^P (Z - 1) \frac{dP}{P} \quad (2.20)$$

Las integrales se pueden evaluar analíticamente cuando  $Z$  se expresa mediante una ecuación de estado, y así obtener los valores reales de la entalpía y entropía para una  $P, T$  o  $V$  dados. También es necesario expresar la entalpía del gas ideal  $H^{gi}$  y la entropía también para el gas ideal  $S^{gi}$ . Esto se puede observar en la Ec. 2.21 y Ec. 2.22.

$$H^{gi} = H_0^{gi} + \int_{T_0}^T C_P^{gi} dT \quad (2.21)$$

$$S^{gi} = S_0^{gi} + \int_{T_0}^T C_P^{gi} \frac{dT}{T} - R \ln \frac{P}{P_0} \quad (2.22)$$

### 2.3.2. Limitaciones de las Ecuaciones de Estado

A pesar de que las ecuaciones de estado son muy eficaces para el cálculo de propiedades, su aplicación al cálculo del equilibrio líquido-vapor está sujeto a un procedimiento

iterativo que puede no converger si no se elige o estima correctamente el valor de inicio del cálculo. Por ejemplo, para el cálculo de la presión de burbuja o de rocío, es necesario estimar un valor inicial de la temperatura. Para ello se utiliza comúnmente [21, p.225] la ecuación de Antoine (Ec. 2.23) que expresa la presión de saturación en función de la temperatura y tres constantes que pueden obtenerse fácilmente para un gran número de compuestos [7]. Sin embargo, esta ecuación está limitada a un rango de temperatura específico y no se debe aplicar fuera de él. Esta restricción también limita la obtención de propiedades termodinámicas para procesos de refrigeración.

$$\ln P^{sat} = A - \frac{B}{T + C} \quad (2.23)$$

La ecuación de Redlich/Kwong en un sentido práctico, no es muy buena para utilizarse en el cálculo del equilibrio líquido-vapor, por lo tanto se han desarrollado otras ecuaciones como las de Soave/Redlich/Kwong y la de Peng/Robinson [21, p.551]. Esto indica que no existe una ecuación de estado «general» que pueda ser aplicada a todo tipo de especies químicas y esto revela la necesidad de especializar las ecuaciones para distintos tipos de compuestos. La idea fundamental es encontrar una ecuación que pudiera contemplar todos los casos, pero el desarrollo de las ecuaciones de estado apunta precisamente en el sentido opuesto.

Otro aspecto importante a considerar en el uso de las ecuaciones de estado, en el caso particular del equilibrio líquido-vapor es el hecho de que cuando se presentan varios componentes en una mezcla, se deben utilizar «reglas de mezclado» que permiten obtener las constantes  $a$  y  $b$  para las ecuaciones de estado como la de Redlich/Kwong, tal como se puede ver en la Ec. 2.24 y Ec. 2.25. Los parámetros  $a_{ii}$  y  $b_i$  corresponden a las constantes para las especies puras, mientras que los parámetros  $a_{ij}$  con  $i \neq j$  son los parámetros de interacción entre los diferentes componentes.

$$a = \sum_i \sum_j y_i y_j a_{ij} \quad (2.24)$$

$$b = \sum_i y_i b_i \quad (2.25)$$

Las anteriores reglas de mezclado sirvieron para el cálculo de propiedades de sistemas que presentaban desviaciones modestas del comportamiento ideal en la fase líquida. Pero como antes, mayor precisión requiere ecuaciones más complejas, como las de Wong/Sandler [21, p.553].

De acuerdo con lo mencionado en esta sección, es posible decir que la aplicación de ecuaciones de estado al cálculo de propiedades termodinámicas es útil y requiere perfeccionamiento. No obstante, a medida que se requieren resultados más precisos, el número de constantes atribuidas a especies químicas y la complejidad de las ecuaciones aumentan. Otro hecho que cabe resaltar es que al aumentar el número de constantes propias de cada compuesto, en realidad no se está contribuyendo a entender cómo el comportamiento atómico y molecular afecta las propiedades macroscópicas de una sustancia y por lo tanto solo se está «ajustando» la ecuación para que concuerde con los datos reales. Esto es lo mismo que hace una red neuronal artificial.

## 2.4. Redes Neuronales Artificiales

La inteligencia es una característica de los humanos y ciertos animales que siempre ha sido motivo de asombro. La forma en la que un ser vivo aprende, almacena y utiliza la información que recibe de su entorno es todavía objeto de estudio para muchos científicos. Las redes neuronales artificiales (RNAs) son una forma matemática de imitar los procesos que se supone que ocurren en el aprendizaje y son capaces de

almacenar, agrupar y clasificar la información que reciben. Las RNAs se aplican en una variedad de tareas, tales como la predicción, ajuste de funciones, clasificación, agrupamiento, reconocimiento de caracteres o patrones, entre otras. La flexibilidad de su estructura interna las convierte en una excelente herramienta para modelar fenómenos complejos, donde las ecuaciones no pueden ser definidas o resueltas fácilmente. Básicamente una RNA está conformada de *neuronas* que son las unidades de procesamiento de información y cada neurona posee una *función de activación* a través de la cual convierte un valor de entrada en uno de salida. Las neuronas se interconectan entre ellas de acuerdo a la estructura elegida, así que la salida de una neurona se convierte en la entrada para otra y a la fuerza con la que una conexión influye en la neurona se le conoce como *peso*. Cada neurona también posee un *umbral* o *bias* que es un parámetro que permite un ajuste fino en el aprendizaje y hace que la RNA se adapte mejor a los datos que está procesando. Dicho *bias* no está conectado a otras neuronas ni es parte de la información de entrada a la RNA. Los grupos de neuronas se pueden acomodar en *capas*, las cuales se pueden configurar de acuerdo al tipo de problema. En la Fig. 6 se puede ver un ejemplo de una RNA conformada por dos capas, una de dos neuronas y la otra por una neurona. En dicha Fig. se puede apreciar como se conectan unas neuronas con otras.

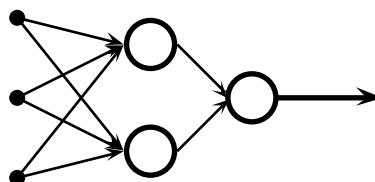


Figura 6: Ejemplo de RNA construida de dos capas y tres neuronas.

Para tener un panorama general del desarrollo de las RNAs a través del tiempo, a continuación se presenta un resumen de los eventos más importantes de acuerdo a la década en la que se presentaron [12].

- Década de los 40's y 50's:

- ⊕ **Warren McCulloch y Walter Pitts** realizaron el primer modelo matemático de las RNA's bajo el modelo de impulsos binarios para las neuronas.

- ⊕ **Donald Hebb** desarrolló un modelo matemático de aprendizaje que ahora lleva su nombre.

- ⊕ **Marvin Minsky** construyó una máquina de 40 neuronas que modelaba con éxito el comportamiento de una rata buscando comida en un laberinto.

- ⊕ **Albert Uttley** creó una máquina compuesta de informes (elementos de proceso) que utilizaban la medida de entropía de Shannon, las cuales han sido utilizadas para simular fenómenos atmosféricos.

- ⊕ **Frank Rosenblatt** integró el aprendizaje al modelo de neuronas propuestas por McCulloch y Pitts para generar el *Perceptrón*.

- ⊕ **Bernard Widrow** diseñó la *Adaptative Linear Element*, una RNA parecida al Perceptrón que ha sido utilizada principalmente en el procesamiento de señales y sistemas de control.

- Década de los 60's:

- ⊕ **Steinbuch** fue el primero en desarrollar métodos de codificación de información en RNA's aplicándolas al reconocimiento de escritura a mano, entre otras aplicaciones.

- ⊕ **Stephen Grossberg** realizó importantes estudios sobre procesos y fenómenos y procesos psicológicos y biológicos del procesamiento de la información por el humano. Esto lo convirtió en el más influyente y formal de los investigadores de RNA's.

⊕ **Shun-Ichi Amari** resolvió el famoso problema de la asignación de créditos y estudió sobre el aprendizaje competitivo.

⊕ **Willshaw y Buneman** junto con **Longuet y Higgins** introdujeron los principios holográficos como posibles mecanismos en la memoria humana.

■ Década de los 70's y 80's:

⊕ **Teuvo Kohonen** Desarrolló un sistema de aprendizaje competitivo denominado *Learning Vector Quantization*.

⊕ **Leon Cooper**, ganador del Premio Nobel y su colega **Charles Elbaum** comenzaron la explotación comercial de las RNA's.

⊕ **Terence Sejnowski** junto con **Geoff Hinton** construyeron la primera RNA que reconocía un algoritmo de aprendizaje para una red de tres niveles, aplicado a distintas áreas de visión. Además aplicaron el algoritmo de retropropagación principalmente al reconocimiento de voz.

⊕ **Robert Hecht-Niesel** fue el principal diseñador del neuro-computador *TWR MARK III*, dedicado al procesamiento de paradigmas de RNA's.

⊕ **John Hopfield** Introdujo una función de energía en sistemas de ecuaciones no lineales, la cual es capaz de describir la actividad de una RNA monocapa en tiempo discreto o continuo y ésta puede irse disipando hasta converger a un mínimo. Esto hace posible que se puedan resolver problemas que los computadores convencionales no pueden resolver.

### 2.4.1. El modelo del cerebro humano

Nuestro cerebro es la principal unidad central de procesamiento de nuestro cuerpo. Se encarga de recibir los impulsos eléctricos de los diferentes órganos y decide qué hacer

en respuesta a ellos. Además tiene una enorme cantidad de tareas programadas, que se ejecutan sin que uno siquiera se dé cuenta. El número de neuronas que se encuentran en él va del orden de 100 mil a 500 mil millones, organizadas en alrededor de 1,000 agrupamientos principales [18] conectados de tal forma que cada neurona puede enviar diferentes señales a cientos o hasta miles de otras neuronas. A medida que las neuronas reciben señales, crean nuevas conexiones o dan más fuerza a las que ya poseía. Es así como se cree que actúa el proceso de memorización y aprendizaje. Es precisamente la característica del aprendizaje el objeto de estudio de la Inteligencia Artificial, dado a que en nuestro entorno existen fenómenos que no pueden ser modelados matemáticamente en forma precisa. Además en muchos casos, aunque se conozca el modelo matemático, no puede resolverse y por ello se simplifica o *idealiza*.

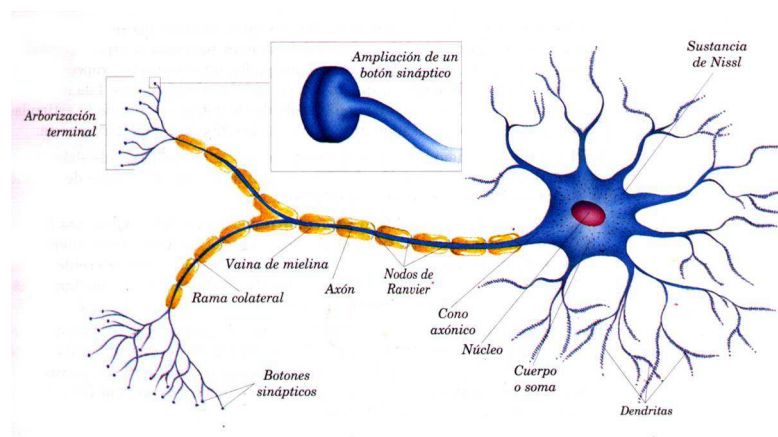


Figura 7: Estructura fundamental de una neurona biológica.

Una neurona está estructurada por tres componentes principales: dendritas, cuerpo celular y axón. Las dendritas se encargan de recibir las señales de entrada, el cuerpo celular las procesa de acuerdo a la suma de la fuerza de cada una y genera la señal de salida que se conduce a través del axón. En esto, una neurona se parece mucho a una función matemática, que de acuerdo a los valores de las variables de entrada, produce un valor de salida. Pero las neuronas pueden recibir miles de valores de entrada y pro-

ducir miles de valores de salida. En la Fig. 7 se pueden apreciar las partes principales de la neurona biológica, entre las que se encuentran las anteriormente mencionadas.

### 2.4.2. Aprendizaje Artificial

El modelo más sencillo de neurona artificial fue desarrollado por Warren McCulloch y Walter Pitts en 1943 [12]. Esta neurona solo tenía dos estados posibles ( $f(u)$ ): encendida (1) y apagada (0). El valor 1 dependía de que la señal de entrada compuesta ( $u$ ), obtenida como la suma de los productos de los pesos ( $w_i$ ) por la señal de entrada simple ( $x_i$ ), superara un valor de umbral establecido ( $\theta$ ), tal como se muestra en la siguiente ecuación:

$$f(u) = \begin{cases} 1 & \text{si } u = \sum w_i x_i > \theta \\ 0 & \text{si } u = \sum w_i x_i \leq \theta \end{cases}$$

Con este modelo de neurona se pueden programar fácilmente las funciones lógicas NOT, AND y OR. Sin embargo, los pesos  $w_i$  de la función  $u$  se mantienen fijos y por ello, la neurona no tiene capacidad de *aprender*, es decir, adaptarlos. Esto confirma la necesidad de un mecanismo que los pueda adaptar dinámicamente.

Donald Hebb postuló en 1949 [18] que la información de una red neuronal se encuentra en los «pesos» de las conexiones entre las neuronas. Dichos pesos cambian su valor ( $\Delta w$ ) proporcionalmente al producto de sus valores de activación (Ec. 2.26) de acuerdo a la fuerza de excitación que una neurona produce sobre otra y que precisamente el aprendizaje consiste en la continua adaptación de dichos pesos (Ec. 2.27). A este tipo de aprendizaje se le denominó *Aprendizaje Hebbiano*, y puede expresarse matemáticamente como:

$$\Delta w \propto x \cdot y \therefore \Delta w = \beta x \cdot y \quad (2.26)$$

$$w_i = w_{i-1} + \Delta w = w_{i-1} + \beta x \cdot y \quad (2.27)$$

Posteriormente Rosenblatt propuso dos tipos más de aprendizaje: el aprendizaje competitivo y el supervisado. El primero, también denominado *no supervisado* consiste en que la neurona desarrolla la capacidad de responder correctamente por sí misma, sin tener como referencia el valor de la señal de salida correcta. Este tipo de aprendizaje se aplica en los Mapas Auto-Organizados, principalmente para tareas de clasificación. Por otro lado, el aprendizaje supervisado implica que cada neurona recibe información sobre los valores correctos de la señal de salida y en respuesta a ellos adapta sus pesos. Así se puede pensar en que se está aplicando el *reforzamiento* propuesto por la psicología. Para ello se lleva a cabo un proceso cíclico en el que se muestran repetidamente los valores correctos de salida y con el tiempo, los pesos alcanzan su valor óptimo.

Una forma más especializada del aprendizaje supervisado se denomina *aprendizaje por corrección de error*, el cual los valores reales son expuestos a la red hasta que la diferencia entre el valor real y el calculado llega a un valor aceptable. Este tipo de aprendizaje ha permitido el desarrollo de técnicas basadas en el gradiente del error, mismas que permiten un mejor aprendizaje.

### 2.4.3. El Perceptrón

Rosenblatt se dio cuenta que la neurona de McCulloch-Pitts no era útil para representar los procesos de cognición debido a que no es suficientemente flexible para adaptarse y aprender a partir de la nueva información que recibe. Así pues, postuló

el primer modelo neuronal, llamado *Perceptrón*. Este modelo utiliza las neuronas de McCulloch-Pitts en una sola capa, pero agrega un parámetro a la neurona: el *umbral* o *bias*. Este umbral permite hacer más flexible todavía el cálculo de la señal de salida, ya que durante el proceso de aprendizaje se adaptan tanto los pesos como el umbral. La función que permite calcular la salida de la red se puede ver en la Ec. 2.28. El esquema del Perceptrón se puede ver en la Fig. 8.

$$y = f \left( \sum_{i=1}^n w_i x_i + \theta \right) \quad (2.28)$$

Cuando la ecuación anterior se compara con la Ecuación 2.27, se nota inmediatamente cómo el umbral influye en la señal de salida  $y$ . Es un sencillo cambio que permite incrementar notablemente la capacidad de adaptación de la red.

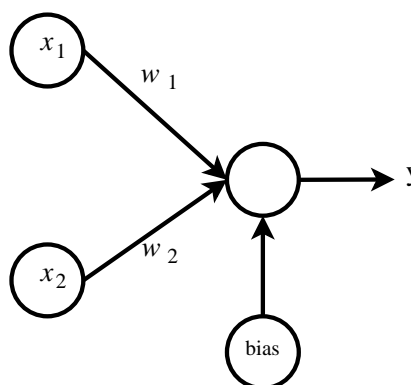


Figura 8: Esquema del Perceptrón de Rosenblatt.

#### 2.4.4. Redes multicapa de perceptrones

Las Redes de Perceptrones Multicapa (RPM) se estructuran con una capa de neuronas en la entrada, una o varias capas internas u «ocultas», y una capa de salida que brinda la señal de salida. Los perceptrones se pueden ordenar de forma que la señal de salida de las neuronas de una capa, sirva como dato de entrada para la capa subsecuente. Las neuronas de las capas exteriores están expuestas a los datos de entrada y

salida. Un bosquejo de este tipo de redes se puede ver en la Fig. 9.

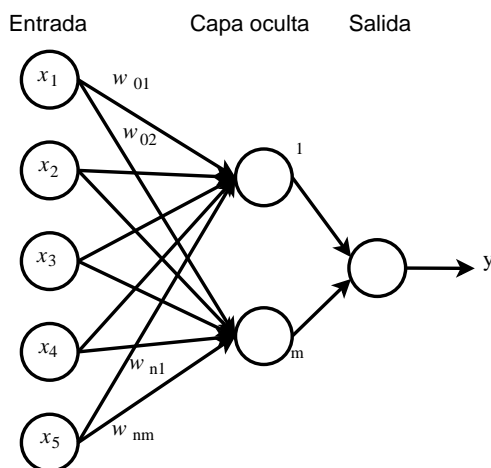


Figura 9: Esquema de una Red de perceptrones multicapa.

El aprendizaje en las RPM se supervisa mostrando a la red continuamente los datos de salida deseados, para posteriormente calcular el error y utilizarlo para actualizar los pesos y umbrales de las neuronas a través de la red. La rapidez con la que se ajusten dichos parámetros depende del método de corrección de error que se aplique. Uno comúnmente utilizado es la *Regla Delta* o *Gradiente Descendente*, pero existen otros.

Las neuronas no necesariamente deben generar un valor de salida 0 o 1, sino que también pueden generar otros valores de acuerdo al fenómeno que se desea modelar con la red. Para ello, en vez de usar un simple producto de pesos en las neuronas, se puede utilizar una función no lineal, llamada *función de activación* o *función de transferencia*. Entre las funciones de transferencia se pueden mencionar la función *logística*, la *tangente hiperbólica*, la *gausiana*, la *gausiana complementaria*, entre otras. Por ejemplo, la fórmula para la función tangente hiperbólica, o simplemente *tansig*, se puede apreciar en la Ec. 2.29.

$$\tanh(u) = \frac{1 + e^{-u}}{1 - e^{-u}} \quad (2.29)$$

En la Fig. 10 se puede apreciar la gráfica de la función *tansig*, donde se muestra que los valores de salida están en el rango de -1 a 1, por esta razón, los valores de salida de la RNA deben redimensionarse para que coincidan con la salida esperada. A este cambio de escala se le conoce como «normalización».

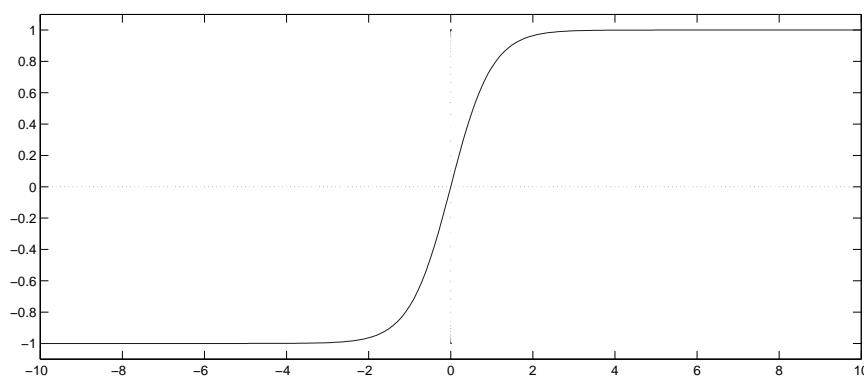


Figura 10: Función tangente hiperbólica.

Un tipo especial de RPM es la llamada *red de alimentación adelante con retropropagación* o ***feedforward-backpropagation***. Este tipo de red es particularmente útil en el caso de ajuste de datos a curvas y predicción de datos en función del tiempo. Es muy flexible en su construcción y los métodos para minimizar el error de salida son muy eficientes.

#### 2.4.5. Modelado de datos con redes multicapa no-lineales

En muchos fenómenos se presenta una relación desconocida entre sus diferentes variables. Es posible entonces construir una RPM que a partir de las variables de entrada, calcule las variables de salida. En este caso se puede decir que se construirá una RPM con  $n$  datos de entrada,  $m$  neuronas ocultas y  $k$  neuronas de salida. A continuación se presentan las fórmulas que aplican para este caso.

Para la neurona  $j$ -ésima, se tiene la entrada  $u_j$  y salida  $y_j$  cuya relación es  $y_j = f(u_j)$  donde  $f(u_j)$  es la función de activación y se pueden calcular a través de la Ec. 2.30.

$$u_j = a_{0j} + \sum_{i=1}^n a_{ij}x_i \quad (2.30)$$

donde  $x_i$  es la  $i$ -ésima entrada,  $a_{ij}$  es el peso asociado a la entrada  $i$  y neurona  $j$ . De la misma manera, para la capa de salida se tiene como entrada la suma de pesos  $v_k$  y la correspondiente salida  $z_k$  relacionadas por  $z_k = f(v_k)$ , donde  $f(v_k)$  es la función de transferencia. Para calcular las salidas se puede escribir la siguiente ecuación:

$$v_j = b_{0j} + \sum_{j=1}^m b_{jk}y_j \quad (2.31)$$

Así pues, definiendo las variables de entrada  $x_i$  y las de salida  $t_k$ , se puede construir una RPM que se ajuste dinámicamente y pueda predecir los valores de salida  $t_k$ . Los valores de salida de la red,  $z_k$  son una aproximación a los valores deseados de salida o «targets»,  $t_k$ . La diferencia entre estos dos últimos, define el error  $\epsilon$  de la red, mediante la Ec. 2.32.

$$\epsilon_k = t_k - z_k \quad (2.32)$$

La sencilla definición del error en la ecuación anterior, permite posteriormente calcular el error cuadrático medio (*MSE*, por sus siglas en inglés) y así obtener el error global de la predicción de la red. Su fórmula, para  $n$  datos de entrada, se expresa en la Ec. 2.33.

$$MSE = \frac{1}{2n} \sum_{i=1}^n \epsilon_i^2 \quad (2.33)$$

### 2.4.6. Entrenamiento de una RNA

Al proceso de presentar los datos de salida (target) a la red para que esta adapte los pesos y umbrales, se le denomina «entrenamiento». Para llevar a cabo el entrenamiento se elabora un algoritmo iterativo que minimice el error optimizando los pesos y umbrales. Un parámetro a observar es el MSE definido en la Ec. 2.33. Durante el entrenamiento de una red tipo *feedforward-backpropagation*, se lleva a cabo el aprendizaje con retropropagación debido principalmente al ajuste simultáneo de todos los pesos y umbrales aprovechando el conocimiento de una superficie de errores, formada por los errores y los valores de las entradas y targets de la red. A partir de esta superficie es posible conocer el gradiente, que apunta siempre en dirección del aumento global de una función multivariable determinada. Esto se aprovecha para buscar lo opuesto al gradiente, es decir, el *gradiente descendente* para poder encontrar la dirección donde la función multivariable disminuye más rápidamente. Así pues se puede saber en qué dirección el error es menor y ajustar los pesos y umbrales en ese sentido. Si se denota por  $\partial\epsilon/\partial a$  las derivadas del error con respecto a los pesos de la capa oculta y  $\partial\epsilon/\partial b$  a las derivadas del error con respecto a los pesos de la capa de entrada, se pueden obtener expresiones usando la regla de la cadena, que permitan obtener su valor y usarlo para ajustar los pesos y umbrales. A este concepto se le llama *retropropagación* o *backpropagation*. Su formulación matemática se expresa en la Ec. 2.34.

$$\frac{\partial\epsilon}{\partial b} = \frac{\partial\epsilon}{\partial z} \cdot \frac{\partial z}{\partial v} \cdot \frac{\partial v}{\partial b} \quad (2.34)$$

La forma de obtener los valores de cada una de estas derivadas parciales se puede consultar en la bibliografía [18]. Lo importante es que se puede construir una relación matemática que tome en cuenta el error y lo utilice para adaptar los pesos. Así pues se puede escribir:

$$\Delta w_m = -\epsilon d_m \quad (2.35)$$

$$w_{m+1} = w_m + \Delta w_m \quad (2.36)$$

Donde  $-d_m$  representa la dirección en la que el error disminuye más rápido, o sea, el gradiente descendente. Este puede obtenerse a partir de las siguientes ecuaciones:

$$d_i^{a_i} = \sum \frac{\partial \epsilon}{\partial a_i} \quad (2.37)$$

$$d_i^{b_i} = \sum \frac{\partial \epsilon}{\partial b_i} \quad (2.38)$$

donde  $d_i^{a_i}$  y  $d_i^{b_i}$  son los gradientes descendentes para la capa interna y la capa de salida respectivamente, para el  $i$ -ésimo ciclo del algoritmo de iteración.

Aunque esta forma de adaptar los pesos es muy efectiva, se puede proponer otra forma de modificar el valor de cambio de cada peso mediante lo que se conoce como métodos de segundo orden. De manera general se expresan en las siguientes ecuaciones:

$$\Delta w = -\frac{\partial \epsilon / \partial w}{\partial^2 \epsilon / \partial w^2} \quad (2.39)$$

$$w_m = w_{m-1} - \epsilon \mathbf{R} d_m \quad (2.40)$$

donde  $w_m$  es el peso en el ciclo  $m$ -ésimo y se ha introducido un nuevo parámetro  $R$  y se han desarrollado varias formas de obtenerlo.

El *Método de Levenverg-Marquardt* incorpora tanto las primeras como segundas derivadas del error, y puede resolver problemas de solución que otros algoritmos re-

suelven con dificultad. En este método, el cambio en los pesos se calcula mediante las Ecuaciones 2.41 y 2.42. En ellas se destaca el uso del *Hessiano*  $\mathbf{H}$ , que es una matriz de segundas derivadas relacionadas con el error; así como también se utiliza la matriz identidad  $\mathbf{I}$  modificada por un nuevo parámetro introducido especialmente en este método:  $e^\lambda$ . La modificación en el error se calcula como de costumbre con la Ec. 2.40.

$$\mathbf{R} = \frac{1}{\mathbf{H} + e^\lambda \mathbf{I}} \quad (2.41)$$

$$\mathbf{H} = \frac{\partial^2 \epsilon}{\partial w_i \partial w_j} \approx \frac{2}{n} \sum_{i=1}^n \left\{ \left( -\frac{\partial z_i}{\partial w_i} \right) \left( -\frac{\partial z_i}{\partial w_j} \right) \right\} \quad (2.42)$$

En estas ecuaciones  $n$  es el número de datos de entrada y  $z_i$  son los valores de salida calculados por la red (outputs).

Por último, el valor de  $\lambda$  se ajusta en cada iteración del entrenamiento de manera que el MSE vaya *cuesta abajo*. Al principio del entrenamiento, se fija un valor de  $\lambda$  y va disminuyendo en saltos de  $\Delta\lambda$ . Si el MSE no disminuye,  $\lambda$  aumenta hasta que se presenta una disminución del MSE. Cuando  $\lambda$  es pequeño, este método se asemeja al de *Gauss-Newton* y cuando es grande, al método de gradiente descendente. Así pues el método de Levenverg-Marquardt combina las ventajas de ambos y por ello es más eficiente.

## Capítulo 3

# Redes Neuronales Artificiales y Equilibrio Líquido-Vapor

### 3.1. Notación propuesta

Como se mencionó en el capítulo anterior, las RNAs están estructuradas en capas dentro de las cuales se encuentran las neuronas. Actualmente no existe una convención para referirse a las características de una RNA, por lo tanto, cuando es necesario comentar la estructura interna de una RNA normalmente se describe en forma textual. Por tal motivo, en este trabajo se comienza por definir una forma de identificar las propiedades de una RNA en términos similares a la notación matemática, cuya forma general es la siguiente:

$$RedX_{(ff/bp)}\{C_1 : n_1 * f_{func\text{transf}} - C_2 : n_2 * f_{func\text{transf}} - \dots\} \quad (3.1)$$

En esta expresión  $RedX$  es el nombre de la RNA,  $ff/bp$  denota el tipo de algoritmo de aprendizaje, en este estudio se utiliza el tipo *feedforward/backpropagation*. Dentro de las llaves se encuentran las capas  $C_1, C_2, \dots$  que tienen un número de neuronas  $n_1, n_2, \dots$  activadas por una función de transferencia  $*f_{func\text{transf}}$ . El guión (-) se emplea solamente para separar la descripción de cada capa. Por ejemplo, una de las RNAs para el R134a tiene la siguiente estructura:

$$Red134a_{(ff/bp)}\{C_1 : 8 * f_{tansig} - C_2 : 7 * f_{purelin}\} \quad (3.2)$$

La expresión anterior representa la estructura de la RNA del refrigerante 134a que utiliza un algoritmo de aprendizaje tipo feedforward-backpropagation y tiene 2 capas ( $C_1$  y  $C_2$ ), la primera con ocho neuronas activadas por la función de transferencia *tansig* ( $C_1 : 8 * f_{tansig}$ ). La segunda capa tiene siete neuronas activadas por la función *purelin* ( $C_2 : 7 * f_{purelin}$ ).

Con esta notación es más fácil expresar rápidamente las estructuras de las RNAs no importando si estas son sencillas o complejas.

## 3.2. Metodología para estructurar la RNA

En esta sección se aborda el método seguido para estructurar las diferentes RNAs de las sustancias analizadas. Se comienza con el tratamiento de los datos que ingresan a la RNA, después se explica el proceso para elegir el mejor número de capas, número de neuronas y función de transferencia. Al final se muestra el procedimiento estadístico para analizar la información que arroja la RNA y comprobar que es un modelo válido para representar el comportamiento real de la sustancia en cuestión.

### 3.2.1. Preparación de los datos

Los datos de entrada para desarrollar la RNA se obtienen de los valores reportados en la literatura abierta [4] y se transcriben o «exportan» a una hoja de cálculo. Después se elige una porción de ellos para usarlos en la estructuración de la RNA. Al final del proceso se utilizarán los que no se usaron para compararlos con los resultados predichos por la RNA. En este caso, se eligieron los datos de las propiedades termodinámicas

correspondientes a la temperatura ( $T$ ), presión ( $P$ ), densidad de líquido ( $D_l$ ), volumen específico de vapor ( $\bar{V}_v$ ), entalpía de líquido y de vapor ( $H_l$  y  $H_v$ ), y entropía de líquido y de vapor ( $S_l$  y  $S_v$ ). Dado a que la mayoría de las veces se requiere conocer las propiedades termodinámicas a partir de la temperatura, esta se definió como la variable de entrada y por consiguiente,  $P, D_l, \bar{V}_v, H_l, H_v, S_l$  y  $S_v$  son las variables de salida. Un ejemplo de la presentación de los datos se puede ver en la Tabla 3.1 [4]. En esta tabla se muestra un extracto de los datos reportados para las propiedades termodinámicas del refrigerante 410a saturado. De tablas similares se extrajeron los datos reales de los refrigerantes seleccionados para elaborar las RNAs.

T ( C)	P (MPa)	$D_l$ ( $kg/m^3$ )	$V_v$ ( $m^3/kg$ )	$H_l$ (kJ/Kg)	$H_v$ (kJ/Kg)	$S_l$ (kJ/Kg*K)	$S_v$ (kJ/Kg*K)
-100.00	0.00119	1679.1	10.004	113.32	306.09	0.6077	1.721
-90.00	0.00286	1652.8	4.3948	121.53	310.59	0.6538	1.6861
-80.00	0.00619	1626.3	2.1355	129.81	315.19	0.6978	1.6576
-70.00	0.01228	1599.5	1.1286	138.17	319.87	0.74	1.6344
-60.00	0.02261	1572.3	0.63992	146.62	324.61	0.7806	1.6156
-50.00	0.03911	1544.7	0.38494	155.18	329.39	0.8197	1.6004
-40.00	0.06409	1516.5	0.24342	163.86	334.18	0.8577	1.5882
-30.00	0.10026	1487.7	0.16057	172.67	338.94	0.8946	1.5784
-20.00	0.1507	1458.1	0.10978	181.62	343.65	0.9305	1.5706
-10.00	0.21878	1427.6	0.07737	190.72	348.29	0.9656	1.5644
0	0.30815	1396.1	0.05595	200	352.81	1	1.5594
12	0.44895	1356.5	0.03901	211.38	358.03	1.0404	1.5547
24	0.63305	1314.6	0.0279	223.07	362.97	1.0801	1.5509
30	0.74365	1292.7	0.02377	229.04	365.31	1.0997	1.5492
40	0.95882	1254.3	0.01838	239.22	368.96	1.1322	1.5465
50	1.2166	1213	0.01433	249.71	372.24	1.1645	1.5437
60	1.5219	1168.1	0.01124	260.58	375.05	1.1969	1.5404
70	1.8802	1118.3	0.00883	271.94	377.22	1.2295	1.5363
80	2.2975	1061.4	0.00691	283.94	378.48	1.2629	1.5306
90	2.7808	993.2	0.00533	296.91	378.35	1.2978	1.522
100	3.3399	903.8	0.00396	311.58	375.6	1.336	1.5076
110	3.9924	742.7	0.00252	331.82	363.95	1.3874	1.4712
111.97	4.1361	565	0.00177	347.76	347.76	1.4283	1.4283

Tabla 3.1: Datos de propiedades termodinámicas del refrigerante R410a.

Una vez seleccionados los datos de entrada, estos deben «normalizarse», es decir, se redimensionan de tal manera que los valores se reduzcan sin perder su magnitud

relativa. Existen varias formas para normalizar, pero en este caso se utiliza una que permite que el mayor valor de la variable  $x$  sea 1 (Eq. 3.3), de acuerdo a lo recomendado en la literatura [8].

$$x_{norm} = x_i/x_{max} \quad (3.3)$$

En la Ec. 3.3,  $x_i$  representa el valor de cualquier propiedad termodinámicas, por ejemplo, la temperatura o la presión. Asimismo  $x_{max}$  corresponde al máximo valor que tiene la misma variable que se está normalizando.

Los datos normalizados se incorporan al algoritmo desarrollado en el software Matlab. Para iniciar el cálculo los datos deben tener forma de vector renglón, por lo tanto se procede a generar variables simplemente aplicando la transpuesta de los vectores que se importan desde las hojas de cálculo donde los datos reales se han normalizado previamente. Así se genera la variable de entrada, por ejemplo  $VENT=[T]'$ . También se genera una matriz que contiene todas las variables de salida de la RNA, en este caso mediante el comando  $META=[P \ Dl \ Dv \ Hl \ Hv \ Sl \ Sv]'$ .

### 3.2.2. Construcción de la Red Neuronal Artificial

Se llama «modelo» a determinada estructura de RNA, es decir, un número definido de capas, neuronas y funciones de transferencia en cada capa. Debido a que se busca simplificar al máximo el cálculo de las propiedades termodinámicas, se busca un modelo que además de reproducir satisfactoriamente los datos reales, tenga el menor número posible de capas y de neuronas. Por ello, todas los modelos que se utilizaron son del tipo *feedforward-backpropagation* y constan de sólo dos capas: una interna activada por la función de transferencia *tansig* y una externa activada por una función *purelin*. Esta estructura se puede expresar como:

$$\text{Red } X_{(ff/bp)} \{ C_1 : n * f_{tansig} - C_2 : Z * f_{purelin} \}$$

donde  $n$  es el número de neuronas de la capa oculta y  $Z$  el número de variables de salida de la red («outputs»), es decir, el total de propiedades termodinámicas que se desean calcular. En este caso es  $Z = 7$  para todos los refrigerantes con excepción del R410a, donde se calculan 8 propiedades de salida. Así pues, cada modelo tiene esta estructura fundamental.

Para encontrar el mejor modelo se hace variar el número de neuronas en la capa oculta. Para ello se construye un ciclo que permite generar 11 modelos diferentes, donde el primer modelo tiene 5 neuronas en la capa oculta y el undécimo, tiene 15. Esto se puede ver en la Tabla 3.2. Al número de neuronas de la capa oculta que genera un modelo con el menor MSE y mayor  $R^2$ , se le denominará  $n_{best}$ .

Número de Modelo	neuronas en la capa oculta
I	5
II	6
III	7
IV	8
V	9
VI	10
VII	11
VIII	12
IX	13
X	14
XI	15

Tabla 3.2: Número de neuronas en la capa interna de cada modelo.

Se escogieron los refrigerantes R12 (diclorodifluorometano), R22 (clorodifluorometano), R717 (amoníaco), R134a (1,1,1,2 tetrafluoroetano) y R410a [R-32/125 (50/50)] para estructurar 11 modelos de RNA. Para todas se utilizó la función de entrenamiento *trainlm*, o sea, el algoritmo de optimización de pesos y umbrales de Levenberg-

Marquardt. Este algoritmo es el que utilizan varios autores sobre RNAs aplicadas al cálculo de propiedades termodinámicas [22].

Debido a que se cuenta con relativamente pocos datos reales, es necesario utilizar un método de remuestreo. Por ello se eligió el Monte Carlo Cross-Validation [24]. Al aplicar este método, se generan 100 versiones diferentes de cada uno de los 11 modelos. Cada una tiene asociado un MSE y una  $R^2$ . Si estos dos últimos parámetros se mantienen razonablemente constantes a través de 100 ciclos, entonces se puede decir que el modelo es *estable*. Después de la generación de cada uno de los 100 modelos se obtiene un MSE promedio y una  $R^2$  del modelo en general. Esto ayudará a elegir el modelo que tenga el menor valor de MSE y mayor  $R^2$  promedio. En la Tabla 3.3 se encuentran los valores obtenidos para estos parámetros en el caso del R134a. Allí se han acomodado en orden ascendente los modelos de acuerdo a su MSE y posteriormente en orden descendente de acuerdo a su  $R^2$ . Se puede ver claramente que el modelo IV, con 8 neuronas en la capa oculta, es el que tiene el menor MSE promedio y mayor  $R^2$ . Así pues, se podría concluir que el modelo IV es el mejor para reproducir los datos reales de las propiedades termodinámicas para el R134a.

Neuronas	MSE promedio	Neuronas	$R^2$
8	0.0003	8	0.9990
9	0.0005	9	0.9985
7	0.0005	5	0.9980
5	0.0006	7	0.9979
6	0.0009	6	0.9973
11	0.0014	11	0.9946
12	0.0020	12	0.9934
10	0.0024	10	0.9926
14	0.0044	14	0.9880
13	0.0045	15	0.9872
15	0.0047	13	0.9853

Tabla 3.3: MSE y  $R^2$  promedios para los modelos del R134a.

Para analizar el comportamiento de cada modelo, también se construyen diagramas de caja-bigotes así como gráficas de barras para los procesos de entrenamiento, prueba y validación. Esto se hace con el fin de visualizar mejor la tendencia de cada modelo. Estos gráficos se encuentran en los apéndices correspondientes de este documento.

La metodología para seleccionar el mejor modelo de RNA es crucial. Debe observarse la validez estadística de los resultados para asegurar la confiabilidad de dicho modelo. En este sentido, el parámetro más importante es el MSE. Sin embargo, es posible que dentro de los 100 ciclos haya algunos cuyo MSE contribuya a aumentar mucho el MSE promedio. Así pues, estos criterios deben ser acompañados por otro análisis, para ello, en este estudio se utiliza el Análisis de Rangos con Signo de Wilcoxon de acuerdo con lo recomendado por Pizarro [16] y Rezeki [17]. En este tipo de prueba, se hace una comparación de los 100 MSE de cada modelo con los 100 MSE de cada uno de los modelos restantes. Por ejemplo, los MSE del modelo II, se comparan con los del modelo I, IV, V...XI. Como resultado de esta comparación, es posible decir cuáles modelos no difieren significativamente en sus MSE y así establecer cuáles son estadísticamente «equivalentes». Por ejemplo, para el R134a parecía adecuado elegir el modelo IV, con  $n_{best} = 8$ , de acuerdo con la Tabla 3.3. Pero al generar la prueba de Wilcoxon para cada modelo, la cual se observa en la Tabla 3.4, se puede apreciar que el modelo II no difiere significativamente del IV. Debido a lo anterior y aplicando el criterio propuesto por el fraile franciscano Guillermo de Ockham (*la Navaja de Ockham*), que establece que «en igualdad de condiciones, la explicación más sencilla suele ser la correcta», se elige el modelo II. Así pues, después de esta prueba se tiene que  $n_{best} = 6$ .

La prueba de Wilcoxon es de gran importancia en la elección de los modelos y de no haberla aplicado, se hubiera elegido otro modelo más complejo.

Una vez reconocido el mejor modelo, se estructura y entrena nuevamente una RNA, con el  $n_{best}$  elegido. Esta será la RNA que se tomará como el mejor modelo capaz de

MSE	Modelo	Modelos no significativamente diferentes
0.000321162	4	2,3,5
0.000543773	5	3,4,6,7
0.000556923	3	2,4,5
0.000572772	1	2
0.000939539	2	1,3,4
0.001445927	7	5,6,8
0.002073589	8	6,7,9,10
0.002412017	6	5,7,8
0.00441938	10	8,9,11
0.004501433	9	8,10,11
0.004745306	11	9,10

Tabla 3.4: Prueba de Wilcoxon sobre los MSE de los modelos del R134a.

reproducir los datos reales para las propiedades termodinámicas. El MSE y  $R^2$  de este último modelo es el que se toma para conocer el desempeño de la red. También se guardan los pesos y umbrales de sus capas los cuales pueden ser utilizados posteriormente en una hoja de cálculo junto con la función de activación, y así calcular los valores de salida de la RNA sin necesidad de usar Matlab.

Para conocer las desviaciones entre la salida de la RNA y los datos reales se preparan gráficos para cada una de las variables de salida definidas. Es útil particularmente conocer el ajuste de los datos usados en la validación, es decir, los que no se utilizaron en el entrenamiento. Para visualizar su comparación con los datos reales, se prepara una gráfica como la que se puede ver en la Fig. 11.

Al final del algoritmo, para los 11 modelos de RNA del refrigerante en cuestión, se han exportado los resultados a una hoja de cálculo con 10 «pestañas» diferentes. Dichas pestañas corresponden a los siguientes resultados:

- ⊕ Los 100 MSE de entrenamiento, prueba y validación de cada modelo (pestañas 1 a 3).

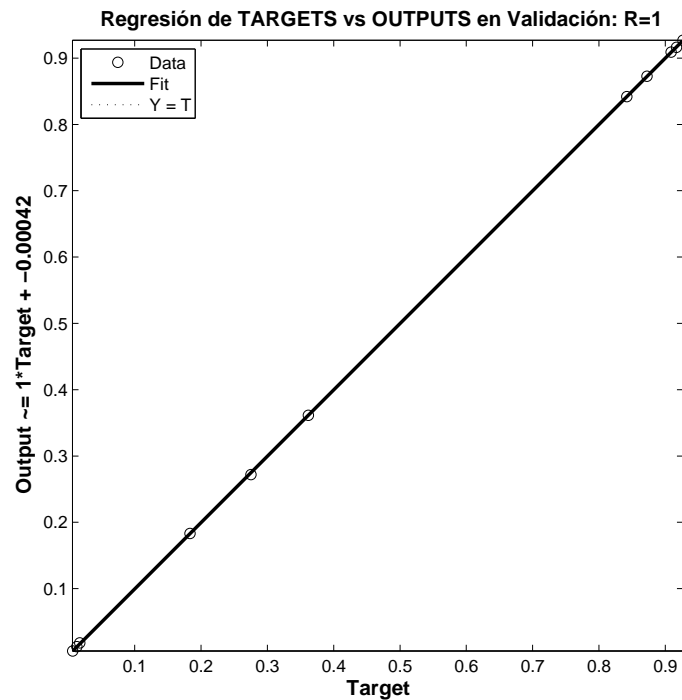


Figura 11: Regresión final con datos de validación para el R134a.

- ⊕ Los 100  $R^2$  de entrenamiento, prueba y validación de cada modelo (pestañas 4 a 6).
- ⊕ El orden del número de neuronas en la capa oculta, de menor a mayor, de acuerdo con su MSE promedio durante las 100 ejecuciones. Así también orden del número de neuronas en la capa oculta, de mayor a menor, de acuerdo con su  $R^2$  promedio durante las 100 ejecuciones (pestaña 7).
- ⊕ El MSE,  $R^2$ , pesos y umbrales del último modelo generado de acuerdo al  $n_{best}$  (pestaña 8).
- ⊕ Datos de prueba de hipótesis de la prueba de Wilcoxon para los MSE y  $R^2$  de cada uno de los 11 modelos (pestaña 9).
- ⊕ Comparación de los 11 modelos generados con la prueba de Wilcoxon, en relación a su MSE (pestaña 10).

En la parte final del algoritmo se generan gráficas que permiten conocer el grado de ajuste del mejor modelo con relación a los datos reales. Así pues se generan las siguientes gráficas, mismas que aparecen para los refrigerantes utilizados en este estudio en los primeros siete apéndices de este documento:

- ⊕ Regresión de «targets» contra «outputs» del modelo, incluyendo todas variables de salida.
- ⊕ Regresión de «targets» contra «outputs» utilizando los datos no incluidos en el entrenamiento, o sea, en la validación.
- ⊕ Gráficos de cada variable simulada junto con la real, con la variable de entrada en el eje  $x$ . Pueden ser 7 u 8 gráficas.

### 3.2.3. Algoritmo

La metodología para estructurar un modelo de RNA para un refrigerante dado, se siguió el siguiente procedimiento general:

1. Normalizar los datos reales de acuerdo a la Ec. 3.3 dentro de una hoja de cálculo.
2. Extraer al azar el 30 % de los datos reales para generar un conjunto para el proceso de prueba y validación con el 15 % en cada conjunto. Estos datos extraídos no se utilizan en el entrenamiento.
3. Generar las matrices de «inputs» y «targets», o sea, entradas y salidas reales.
4. Construir los 11 modelos de RNA. Cada modelo se ha remuestreado 100 veces de acuerdo con el método Monte Carlo Cross Validation.
5. Realizar la prueba de Wilcoxon para encontrar los modelos que no son significativamente diferentes.

6. Mostrar los valores de los MSE y  $R^2$  promedio así como los resultados de la prueba de Wilcoxon para poder elegir el  $n_{best}$ .
7. Construir un modelo final de RNA con el  $n_{best}$  y se exportan los datos importantes, así como los pesos y umbrales a una hoja de cálculo.
8. Graficar los datos reales contra los calculados por el modelo elegido en función de la variable de entrada, de manera que se vean claramente la capacidad de predicción de cada variable de salida.

La parte más importante del algoritmo es la que corresponde a la selección del mejor modelo de RNA. Es importante mencionar que es precisamente en esta etapa donde el presente estudio propone algo que no se ha realizado anteriormente: aplicar la prueba de rangos con signo de Wilcoxon. Este proceso de selección de la mejor RNA se bosqueja con pseudocódigo en el siguiente algoritmo:

En el proceso bosquejado en el Algoritmo 1, la prueba de rangos con signo de Wilcoxon se aplica comparando cada modelo con los otros 10 restantes. Por esta razón se creó la matriz  $MSE(s, m)$  que hace referencia al modelo  $m$  cuyo MSE es  $s$ , quedando una matriz de 100 renglones por 11 columnas. Posteriormente se usan los MSE de cada modelo, o sea, se comparan las columnas en la prueba de Wilcoxon. Es necesario crear la matriz  $ComparisonMatrix(m, r)$ , 11 renglones por 11 columnas, donde están alojados los resultados de la prueba de Wilcoxon. Cada modelo  $m$  se compara con todos los 11 modelos, incluyéndose a sí mismo. Si la prueba rechaza la hipótesis nula de que los valores MSE del modelo  $i$  son iguales a las del modelo  $j$ , entonces la matriz de comparación toma valor de 1, y de 0 en caso contrario. Así pues, se toma un modelo a la vez y se compara con los demás, para así saber cuáles no son significativamente diferentes. De hecho, lo más importante de esta prueba es saber si existe algún modelo con menos

---

**Algorithm 1 Selección del mejor modelo de RNA**


---

```

for  $m = 1$  to 11 (once modelos diferentes) do
   $n = i + 4$ 
  for  $s = 1$  to 100 do
    1.- Generar los conjuntos de entrenamiento, prueba y validación aleatoriamente
    desde el conjunto de datos reales.
    2.- Construir una RNA tipo Feedforward-Backpropagation con  $n$  neuronas en
    la capa interna.
    3.- Guardar los MSE y  $R^2$  en una matriz columna:  $MSE(s, m)$ 
  end for
end for
Realizar la prueba de rangos con signo Wilcoxon:  $H_0$  es  $MSEs(\text{modelo } m) =$ 
 $MSEs(\text{modelo } s)$ 
for  $i = 1$  to 11 do
  for  $j = 1$  to 11 do
    if Existe diferencia significativa entre el modelo  $i$  y el  $j$  then
      ComparisonMatrix( $i, j$ ) = 1
    else
      ComparisonMatrix( $i, j$ ) = 0
    end if
  end for
end for
print Desplegar ComparisonMatrix( $i, j$ ) para seleccionar el mejor modelo.
Require:  $n = n_{best}$ 
  ▷Construir una nueva RNA Feedforward-Backpropagation con  $n$  neuronas en la capa
  interna.
  ▷Guardar los valores de pesos y umbrales.

```

---

neuronas significativamente parecido a aquel que tuvo el menor MSE promedio, y en base a ello, se elige  $n_{best}$ .

El MSE y la  $R^2$  de cada ciclo se obtienen utilizando las Ecuaciones 3.4 y 3.5 respectivamente. Donde  $N_t$  es el número total de renglones de datos que se presentan a la RNA para su entrenamiento,  $x_i$  es el valor real de la propiedad termodinámica y  $z_i$  es el valor estimado por la RNA para el correspondiente  $x_i$ . La función de activación para las neuronas de la capa oculta es la función *tansig*, que se puede ver en la Ec. 3.6, misma que es utilizada internamente por Matlab para obtener los parámetros del modelo de RNA, por lo que no se programa explícitamente en el algoritmo. Los pesos y umbrales de las capas de cada modelo varían de acuerdo al refrigerante en cuestión. Para obtener las salidas de la capa oculta ( $Y_i$ ) y de la capa de salida («outputs»  $z_i$ ) se utilizan las Ecuaciones 3.7 y 3.8. En estas ecuaciones los pesos de la capa oculta se denotan por  $w_i^\alpha$  y los de la capa de salida  $w_i^\beta$ . De igual forma,  $b_i^\alpha$  y  $b_i^\beta$  son los umbrales de las capas oculta y la de salida, respectivamente. El índice  $i$  varía desde 1 hasta el número de neuronas  $n_{best}$  y  $j$  varía desde 1 hasta el número de variables de salida, usualmente 7.

$$MSE = \frac{1}{N_t} \sum_{i=1}^{N_t} (x_i - z_i)^2 \quad (3.4)$$

$$R^2 = 1 - \left( \frac{\sum_i (x_i - z_i)^2}{\sum_i z_i^2} \right) \quad (3.5)$$

$$tansig(x) = \frac{2}{1 + e^{-2x}} - 1 \quad (3.6)$$

$$Y_i = \frac{2}{1 + e^{-2(x_i \cdot w_i^\alpha + b_i^\alpha)}} - 1 \quad (3.7)$$

$$z_i = \frac{2}{1 + e^{-2[(\sum_{j=1}^n Y_j \cdot w_{ji}^\beta) + b_i^\beta]}} - 1 \quad (3.8)$$

La Tabla 3.5 muestra los valores de los pesos y umbrales obtenidos de la RNA para el refrigerante R410a. Estos valores, junto con las Ecuaciones 3.4 a la 3.8 pueden ser utilizados en una hoja de cálculo para obtener las salidas de la RNA sin utilizar ningún software adicional especializado. También podrían usarse en el algoritmo de un Controlador Lógico Programable (PLC, por sus siglas en inglés) y así optimizar un proceso industrial en tiempo real. De allí la importancia práctica de la metodología aquí presentada.

	$b_i^\alpha$	$w_i^\alpha$	$b_j^\beta$	$w_{1j}^\beta$	$w_{2j}^\beta$	$w_{3j}^\beta$	$w_{4j}^\beta$	$w_{5j}^\beta$	$w_{6j}^\beta$	$w_{7j}^\beta$	$w_{8j}^\beta$	$w_{9j}^\beta$
1	13.91	-11.29	-1.26	-1.08	-0.05	-0.05	0.07	0.08	0.1	0.15	-0.54	-2.29
2	9.56	-12.53	-1.54	-0.8	-0.05	-0.05	0.07	0.08	0.1	0.15	-0.54	-2.29
3	6.13	-11.87	0.99	0.17	0.3	0.07	-0.09	-0.08	-0.08	-0.11	0.32	1.12
4	-2.48	11.74	5.3	4.23	-0.02	0	0	0	0.01	-0.03	0	10.47
5	1.01	10.86	-1.05	-0.71	-0.16	-0.06	0.08	0.08	0.09	0.13	-0.44	-1.71
6	3.52	9.68	-2.36	-1.08	0.86	0.06	-0.07	-0.02	0.02	0.08	-0.57	-2.77
7	4.79	7.34	-1.3	-1.18	-0.12	-0.05	0.07	0.07	0.08	0.13	-0.49	-2.47
8	-6.5	-6.67	0.85	1.87	0.3	0.04	-0.05	-0.05	-0.05	-0.07	0.34	2.8
9	-34.75	-33.51										

Tabla 3.5: Valores de pesos y umbrales para la RNA del R410a.

A manera de resumen, en la Fig. 102 del Anexo A.7 se muestra un diagrama general del proceso completo del algoritmo de creación de modelos y selección del mejor. Hay detalles relacionados con la programación del algoritmo que se omitieron mencionar en esta sección. Estos se pueden ver en el Anexo A.8.

### 3.3. Resultados

Para cada refrigerante se generaron 11 modelos de RNA y se sometieron a un proceso de selección. Los resultados finales de los modelos elegidos para cada refrigerante se pueden ver en la Tabla 3.6. En los apéndices se muestran las tablas y figuras que corresponden al análisis de cada uno de los refrigerantes. El método para estructurar

una RNA capaz de predecir los datos reales en cuestión, fue probado con 6 diferentes refrigerantes para probar su generalidad y validez de aplicación. Los resultados indican que los modelos de RNA generados tienen un bajo MSE y alto coeficiente de regresión, por lo que pueden reproducir adecuadamente los datos reales.

Refrigerante	Modelo elegido	$n_{best}$	MSE	$R^2$
R12	1	5	6.51 E-07	0.999998255
R22	1	5	7.42 E-07	0.999997585
R134a	2	6	1.51 E-07	1
R290	1	5	7.33 E-07	0.999998292
R717	1	5	6.04 E-08	1
R410a	5	9	0.0028	0.995862217

Tabla 3.6: Modelos elegidos para cada refrigerante.

Como apoyo al proceso de selección del mejor modelo, se preparan gráficas de caja-bigotes como las que se pueden ver en las Figuras 12 y 13. Así se puede tener idea de la variabilidad de los MSE y  $R^2$  durante los 100 ciclos de cada modelo. Es particularmente importante el gráfico de caja-bigotes para el proceso de validación. Además con la información que se exporta a la hoja de cálculo, se puede preparar una gráfica que presente como varía el MSE en validación a través de las 100 ejecuciones.

También es necesario hacer una gráfica de barras de los valores promedio de MSE en validación para visualizar cuál modelo tiene menor MSE y mayor  $R^2$  en promedio. En las Figuras 14a y 14b se muestran ejemplos de este tipo de gráficas. Con los datos que se exportan a la hoja de cálculo es posible construir gráficas de barras para los MSE promedio en entrenamiento y prueba, así como  $R^2$  promedio para entrenamiento, prueba y validación. Sin embargo, el proceso de validación es el que determina en mayor medida cuál será el modelo ganador.

Una vez analizados las gráficas de caja-bigotes y las de barras se procede con la prueba de rangos con signos de Wilcoxon. El algoritmo muestra al usuario una ventana

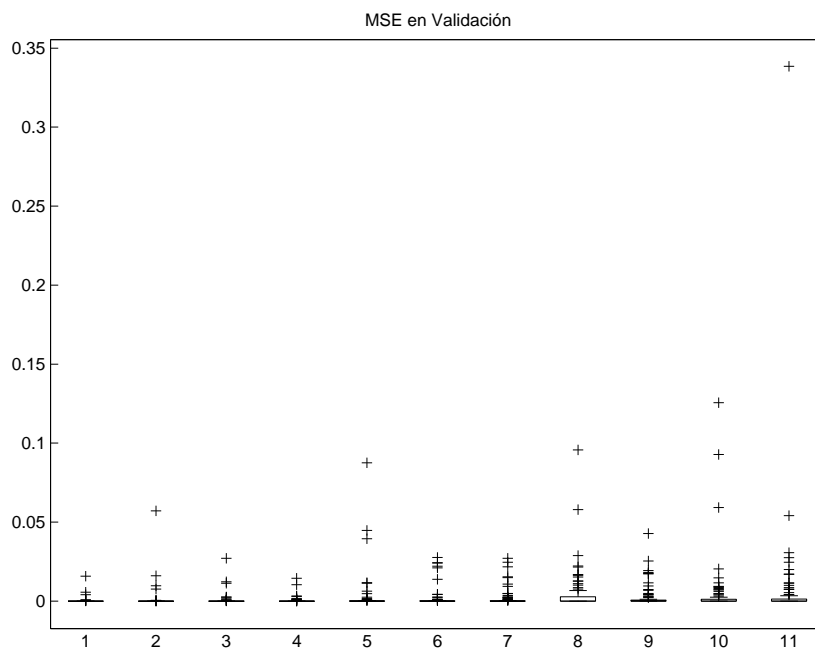


Figura 12: Gráfica de caja-bigotes para MSE de los modelos del R717.

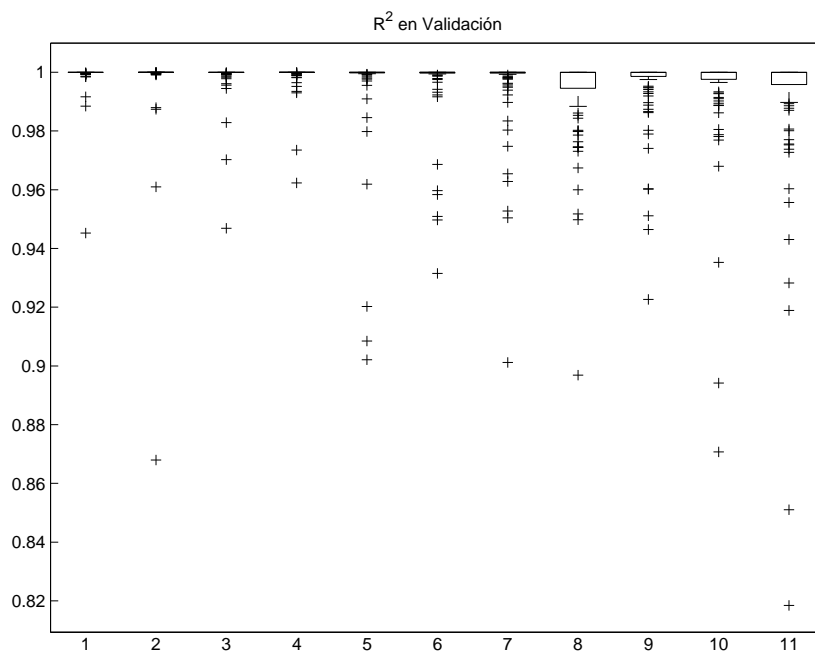


Figura 13: Gráfica de caja-bigotes para  $R^2$  de los modelos del R717.

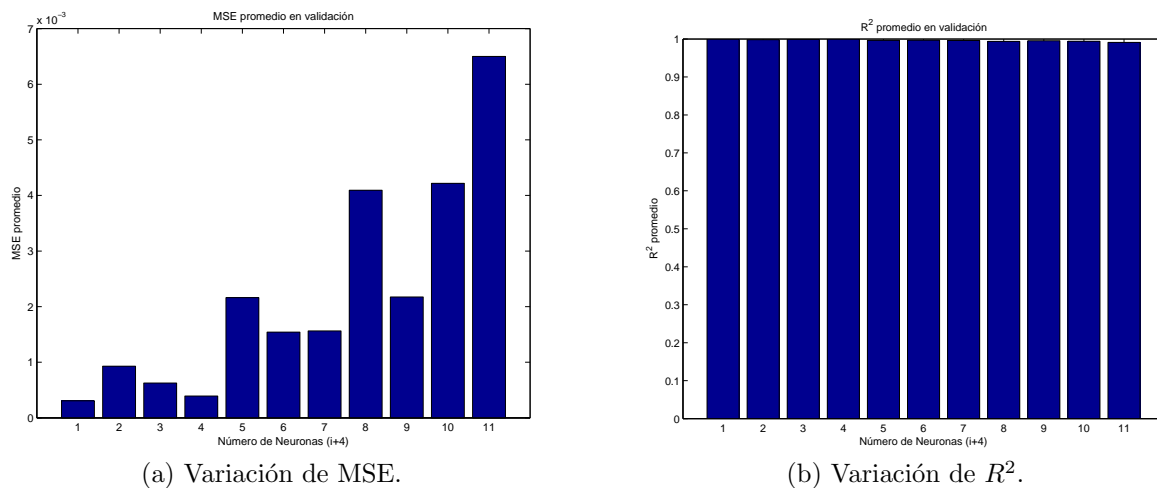


Figura 14: Variación del MSE y  $R^2$  promedio entre los modelos del R717.

con la comparación de modelos, especificando cuáles de ellos tienen MSEs que no son significativamente diferentes. Un ejemplo de esta comparación se puede ver en la Tabla 3.7, donde se puede ver que el modelo XI tiene el menor MSE promedio y por lo tanto, parece el mejor modelo. Sin embargo, en la misma tabla se observa que el modelo V no es significativamente diferente del XI. Así pues, se tiene una base estadística para descartar el modelo XI y utilizar el V, con la ventaja de que este último tiene menos neuronas. Al elegir modelos con pocas neuronas, se simplifica su aplicación, ya que se requieren menos parámetros de ajuste para las ecuaciones que permiten calcular las variables de salida de la RNA (Ecs. 3.4 a 3.8). Por eso es muy importante esta prueba dentro del proceso de selección del modelo. Para todos los demás refrigerantes, la prueba indicó que el modelo con menor MSE promedio era el mejor modelo, ya que los modelos no significativamente diferentes tenían más neuronas. Las tablas con los resultados de la prueba de Wilcoxon aplicada a cada refrigerante estudiado, se pueden encontrar en los apéndices.

La observación de los promedios de MSE y  $R^2$  de cada modelo seleccionado reportados en la Tabla 3.6 permite apreciar que los MSE son del orden de  $10^7$ . Este valor

MSE promedio	Modelo	Modelos sin diferencia significativa
0.0022	11	5,7,8,9,10
0.0030	9	1,4,5,6,7,8,10,11
0.0041	5	1,2,3,4,6,7,8,9,10,11
0.0041	4	1,2,3,5,6,7,9,10
0.0044	1	2,3,4,5,6,7,8,9,10
0.0045	10	1,2,3,4,5,6,7,8,9,11
0.0049	7	1,4,5,6,8,9,10,11
0.0056	8	1,2,5,6,7,9,10,11
0.0057	2	1,3,4,5,6,8,10
0.0071	6	1,2,3,4,5,7,8,9,10
0.0077	3	1,2,4,5,6,10

Tabla 3.7: Prueba de Wilcoxon para el R410a.

es suficientemente bajo para considerar el modelo aceptable. De igual forma, los  $R^2$  están muy cercanos a 1 en todos los casos, reafirmando que el modelo predice los datos reales satisfactoriamente. El modelo elegido para el R410a tiene un MSE mayor a los de los demás refrigerantes, sin embargo, su  $R^2$  está muy próximo a 1. Por esta razón el modelo se consideró aceptable. El proceso de validación es crítico en la decisión del mejor modelo. Las gráficas de regresión de los datos reales y calculados utilizados en la validación, indican un aceptable ajuste de los datos. Dichas gráficas se encuentran en el apéndice.

Con el fin de visualizar la capacidad predictiva del modelo elegido, se prepara una gráfica de regresión que muestra el ajuste de los datos reales («targets») con respecto a los calculados por la RNA («outputs»). En dicha gráfica se incluyen todos los valores y por ello el conjunto de datos puede mostrar cierta dispersión. Debido a esto, se obtiene también una gráfica de regresión similar a la anterior, pero esta vez con los datos que se usaron para la validación. En esta se usa el grupo de datos reales que no se incluyeron en el entrenamiento y los correspondientes calculados por la RNA. Este ajuste es el más significativo del proceso. Un ejemplo de estas dos gráficas se puede

ver en las Figuras 15 y 16. En general, para todos los modelos se observa un excelente ajuste en la validación, con lo cual es posible decir que las RNAs elegidas reproducen satisfactoriamente los datos reales.

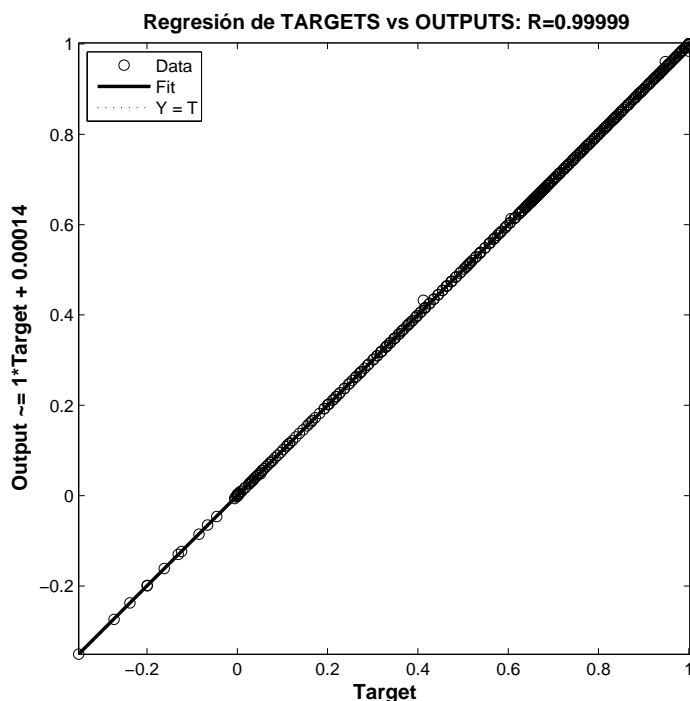


Figura 15: Regresión de datos reales vs calculados para la RNA del R290.

Finalmente, para conocer la capacidad de predicción de la RNA para cada una de las variables termodinámicas, se construyen gráficas donde se muestra el valor real de la propiedad contra el obtenido por la RNA en función de la variable de entrada. Por ejemplo, en las Figuras 17 a 23, se observa la comparación datos reales y calculados por la RNA del R134a para la presión, densidad de líquido, volumen específico de vapor, entalpía de líquido, entalpía de vapor, entropía de líquido y entropía de vapor, respectivamente. La temperatura está en el eje  $x$  ya que es la variable de entrada.

Los datos presentados en estas gráficas están normalizados, pues así se utilizan dentro de la RNA. En general, después de analizar este tipo de gráficas en todos los refrigerantes seleccionados, se encuentra un ajuste muy satisfactorio de los datos reales

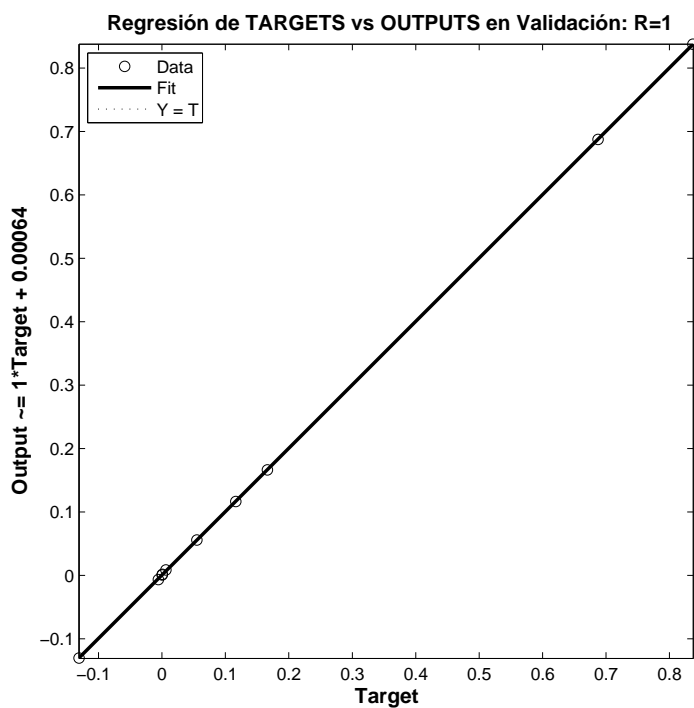


Figura 16: Regresión de datos de validación para la RNA del R290.

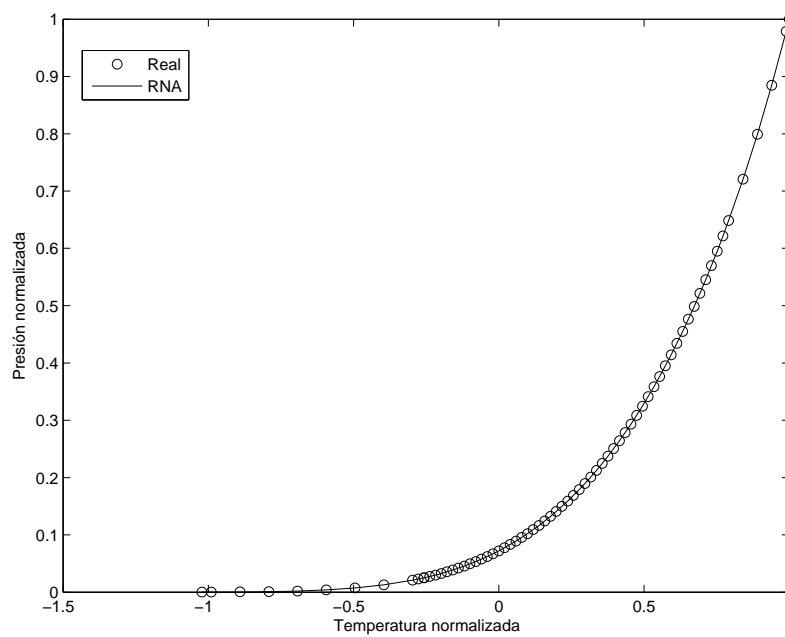


Figura 17: Presión real vs calculada para el R134a.

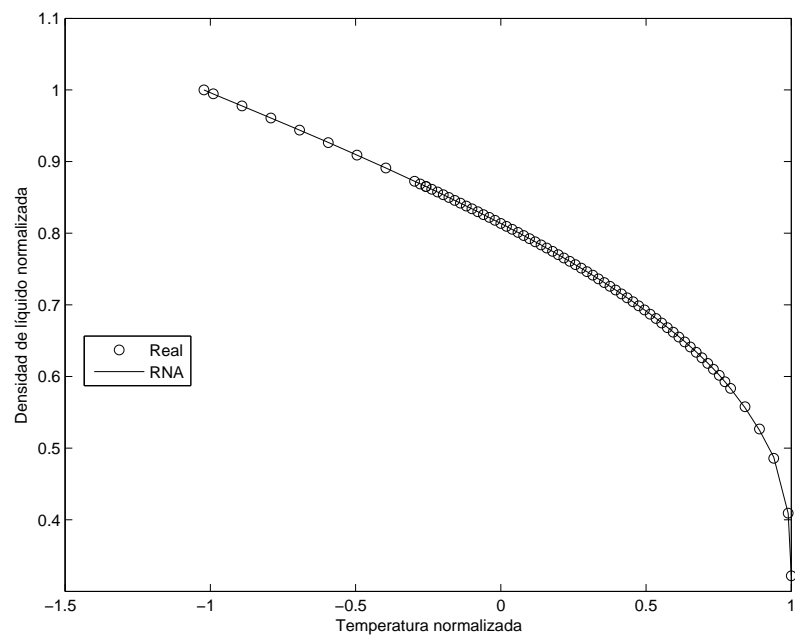


Figura 18: Densidad de líquido real vs calculada para el R134a.

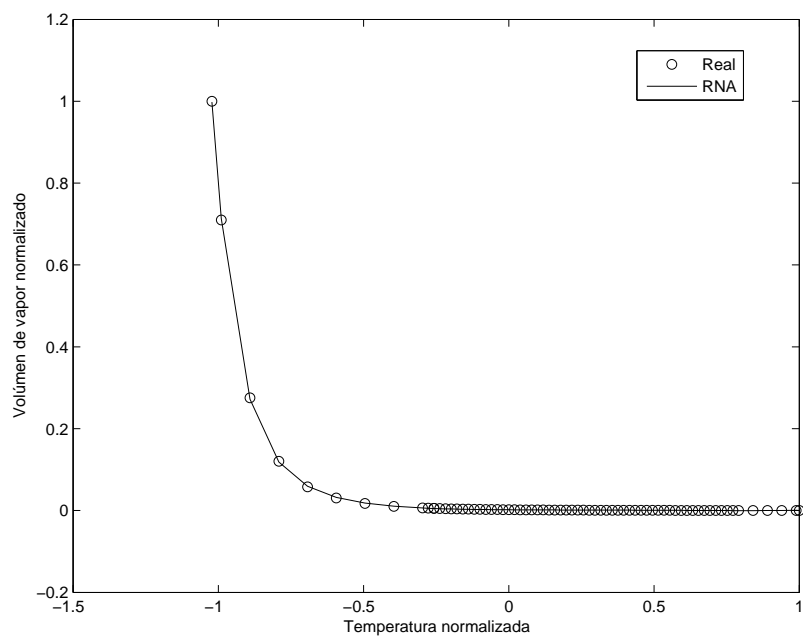


Figura 19: Volumen de vapor real vs calculado para el R134a.

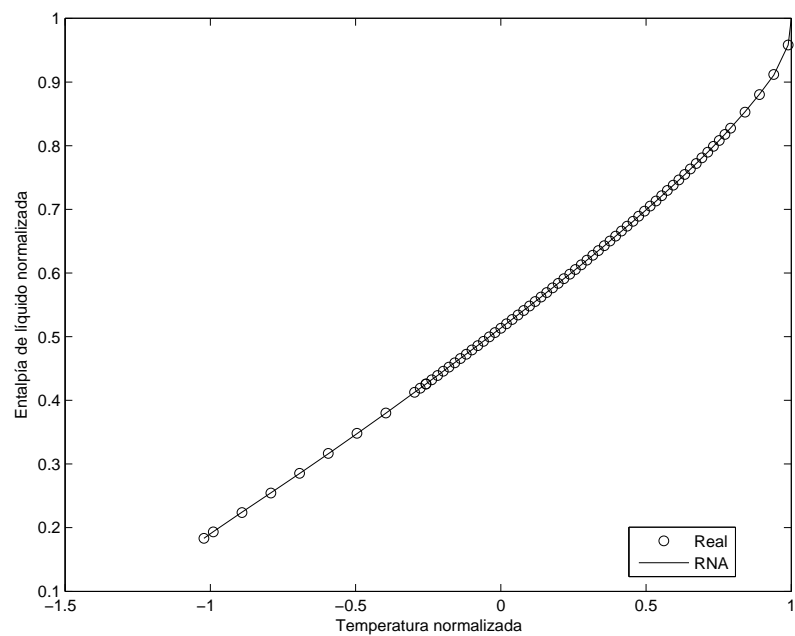


Figura 20: Entalpía de líquido real vs calculada para el R134a.

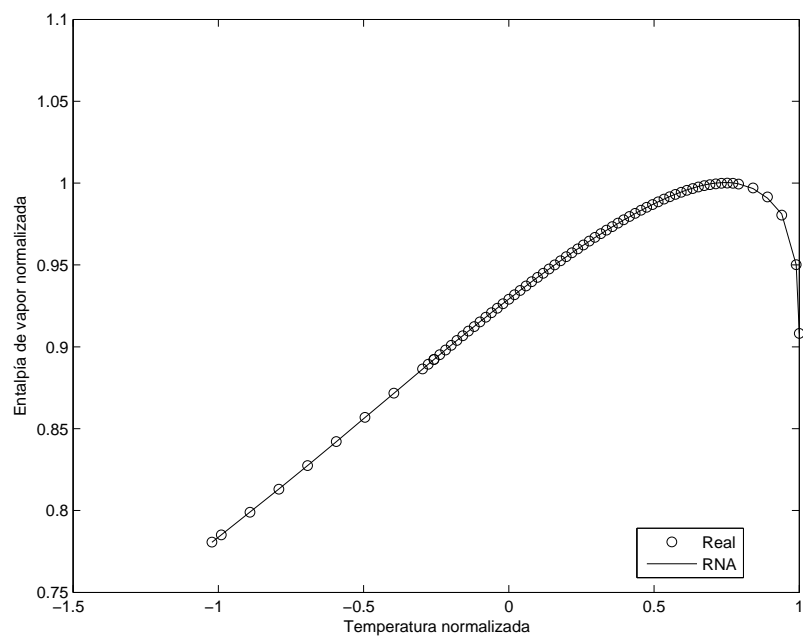


Figura 21: Entalpía de vapor real vs calculada para el R134a.

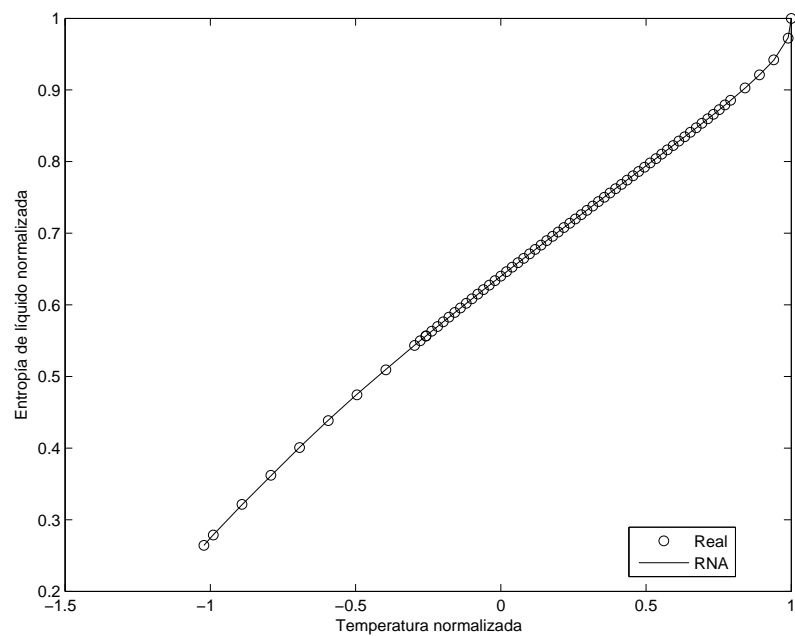


Figura 22: Entropía de líquido real vs calculada para el R134a.

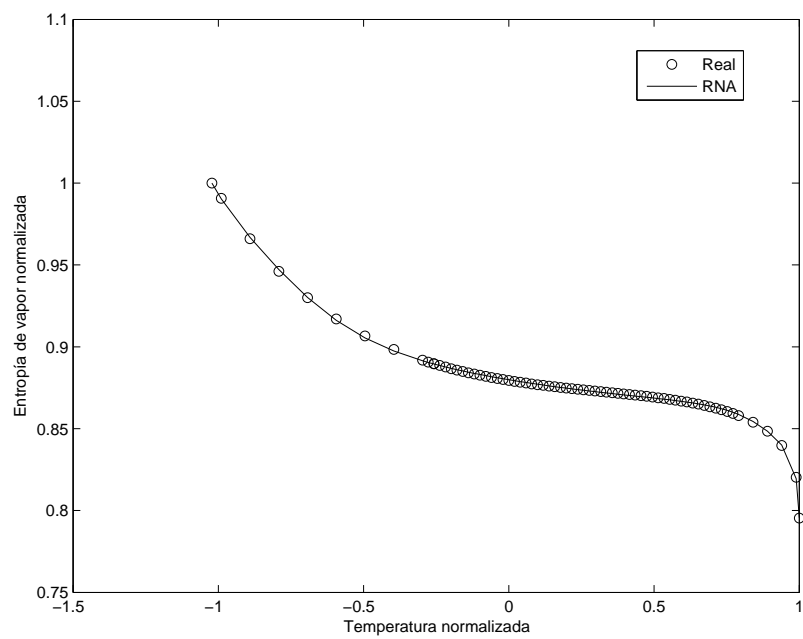


Figura 23: Entropía de vapor real vs calculada para el R134a.

contra los que se obtienen con cada RNA. De esta manera se confirma la validez de aplicación de las RNAs a la predicción de propiedades termodinámicas. Las demás gráficas para todos los refrigerantes analizados se encuentran en el apéndice A.

### 3.3.1. Comparación entre Ecuaciones de Estado y Redes Neuronales

Es importante tener un punto de comparación en el desempeño de una RNA y una EoS. Para ello se tomaron datos del R12, y se realizó el cálculo de la presión de saturación, la densidad de líquido ( $D_l$ ) y el volumen de vapor ( $V_v$ ), utilizando la ecuación de Redlich-Kwong (R-K), Soave y la de Peng-Robinson (P-R). Posteriormente se obtuvieron los errores porcentuales absolutos ( $\varepsilon_i$ ) y se compararon con los errores que presenta la RNA. La Eq. 3.9 se utilizó para obtener dichos errores porcentuales, donde  $y_i$  es el valor real de la propiedad termodinámica en cuestión y  $z_i$  es el valor estimado por la RNA o la EoS.

$$\varepsilon_i = \frac{y_i - z_i}{y_i} \times 100 \quad (3.9)$$

Los resultados mostraron que la RNA predice las propiedades termodinámicas con errores mucho menores que los generados por las EoS. Es importante notar que para algunos puntos, la RNA presenta errores considerables, sin embargo, solamente son dos o tres puntos de entre más de 65 puntos en general. Por esta razón, también es necesario definir el rango de temperatura o presión para el cual la RNA es válida.

En la Tabla 3.8 se pueden observar los errores para la presión de saturación del R12, que para la ecuación de R-K están entre el 1 y 33 %, para la ecuación de Soave están entre 1 y 40 %, para la ecuación de P-R entre el 0.027 y 1.89 %, mientras que la RNA muestra errores entre  $1.91E^{-14}$  y 1.74 %. Un comparativo similar se puede

ver en la Tabla 3.9, en la que se puede ver que los errores obtenidos en el cálculo de la densidad de líquido ( $D_l$ ) con la ecuación de R-K son de entre 7 y 21 %, para la ecuación de Soave están entre 5 y 18 %, para la ecuación de P-R son desde 0.48 hasta 9.1 %, mientras que la RNA muestra errores entre  $1.91E^{-14}$  y 1.73 %. Finalmente, para el volumen de vapor ( $V_v$ ) se preparó la Tabla 3.10, en donde se ve que los errores obtenidos para la ecuación de R-K están entre 3 y 29 %, para la ecuación de Soave son de entre 0.2 y 2 %, para la ecuación de P-R son del orden de 0.5 y 2 % y para la RNA están entre 0 y  $3.39E^{-13}$ . Se puede ver de estas tablas que para las estimaciones obtenidas con la RNA, la mayoría de los errores son prácticamente cero, lo cual es muy difícil con una ecuación de estado. Aquí solo se presentan resultados para algunos puntos, como ejemplo comparativo solamente, ya que lo más importante es la certeza de que las RNAs pueden calcular las propiedades termodinámicas con un error muy pequeño.

Temperatura ( C )	Error ( $\varepsilon$ )			
	R-K	Soave	P-R	RNA
-30	-40.442	0.983	-0.027	$5.60E^{-14}$
-10	-33.35	0.45	0.45	$4.11E^{-14}$
0	-29.851	0.393	0.393	0.097
10	-27.023	0.059	0.538	$-4.26E^{-13}$
30	-20.691	-0.258	0.559	$-1.81E^{-13}$
50	-15.322	-0.667	0.415	$7.39E^{-14}$
70	-11.58	-1.02	0.165	$-1.91E^{-14}$
90	-5.314	-0.615	0.041	$3.88E^{-14}$
100	-0.942	1.501	1.889	1.74

Tabla 3.8: Errores porcentuales en cálculo de presión para el R12.

Cuando se utilizan EoS para calcular la presión, densidad de líquido y volumen de vapor, se requiere reescribir la ecuación de manera que tome la forma de un polinomio cúbico, que después se resuelve mediante un proceso iterativo. El valor inicial del proceso de iteración puede influir notablemente en la convergencia, por lo que debe escogerse

Temperatura ( C)	Error ( $\varepsilon$ )			
	R-K	Soave	P-R	RNA
-30	7.674	5.805	-6.432	$5.608E^{-14}$
-10	8.353	6.126	-6.236	$4.112E^{-14}$
0	8.813	6.416	-5.985	0.0971
10	9.351	6.793	-5.627	$-4.257E^{-13}$
30	10.74	7.901	-4.469	$-1.814E^{-13}$
50	12.613	9.591	-2.556	$7.395E^{-14}$
70	15.134	12.109	0.486	$-1.914E^{-14}$
90	18.658	15.966	5.4	$3.882E^{-14}$
100	21.006	18.759	9.155	1.738

Tabla 3.9: Errores porcentuales en cálculo de densidad de líquido para el R12.

Temperatura ( C)	Error ( $\varepsilon$ )			
	R-K	Soave	P-R	RNA
-30	29.253	-1.501	-0.312	$-2.25E^{-14}$
-10	25.524	-1.115	-0.834	0
0	23.977	-1.139	-0.742	-1.51
10	22.51	-0.845	-0.841	$-1.17E^{-13}$
30	18.798	-0.623	-0.642	$1.17E^{-13}$
50	15.385	-0.221	-0.054	$3.39E^{-13}$
70	12.031	0.252	0.962	$-3.93E^{-14}$
90	7.009	-0.627	2.026	$-6.51E^{-14}$
100	3.176	-2.042	1.995	$2.19E^{-14}$

Tabla 3.10: Errores porcentuales en cálculo de volumen de vapor para el R12.

con cuidado. Se puede decir que esta es la primera *dificultad* que se debe considerar al utilizar una EoS. Después de obtener los valores del factor de compresibilidad para líquido y vapor mediante la solución de la EoS, se pueden obtener la densidad de líquido y vapor. Sin embargo, para obtener la entalpía y entropía, es necesario llevar a cabo un procedimiento más complicado. Se comienza por calcular la entalpía y entropía residual con una ecuación apropiada, posteriormente se deben estimar estas mismas propiedades para un gas ideal a la misma temperatura. Al llevar a cabo la adición de las propiedades residuales con las del gas ideal, se obtiene finalmente el valor de la propiedad termodinámica en cuestión (entalpía o entropía). Estimar las propiedades del gas ideal involucra a su vez la aplicación de ecuaciones que incluyen el cálculo de capacidades caloríficas, las cuales requieren generalmente integración numérica. Así pues, se puede ver que este proceso de aplicación de las EoS para obtener las propiedades termodinámicas requiere un esfuerzo considerable para resolver ecuaciones, así como la programación en computadora. Es por esto que en el presente estudio se propone reemplazar la aplicación de las ecuaciones de estado por redes neuronales artificiales, de forma que el esfuerzo se reduzca, el cálculo sea más preciso y que el tiempo empleado sea actividad de la computadora. Los resultados comentados en esta sección, que se amplían con los que se muestran en los apéndices, permiten confirmar que las RNAs pueden reemplazar de forma satisfactoria una ecuación de estado, reduciendo además la complejidad matemática en el cálculo. Otro beneficio de utilizar las RNAs aquí presentadas, es que es posible incorporar las ecuaciones generadas en un dispositivo programable, de manera que se puedan optimizar procesos en tiempo real. Esto último sería muy difícil si se utilizara una ecuación de estado.

# Capítulo 4

## Conclusiones

Las gráficas que muestran los datos reales contra los obtenidos por la RNA, corroboran el aceptable ajuste de la red para cada variable de salida. Dichas gráficas se pueden consultar en los apéndices correspondientes para cada refrigerante. Los valores bajos de MSE obtenidos para cada red, que están entre  $1.57E^{-07}$  y 0.0028, así como los coeficientes de regresión que oscilan entre 0.996 y 1.00, indican que los modelos se ajustan aceptablemente a los datos reales. La estabilidad del proceso para estructurar las RNAs y lograr su óptima adaptación a los datos reales de las propiedades termodinámicas, se garantizó mediante la técnica del re-muestreo (*Monte Carlo Cross-Validation*). Esto puede confirmarse analizando las gráficas de variación del MSE promedio y de  $R^2$  en cada ciclo, mismas que se detallan en los correspondientes apéndices. Sin embargo, el hecho que los modelos de RNA generados para un refrigerante dado muestren estabilidad, no es criterio suficiente para elegir cuál es el más adecuado para representar los datos reales de las propiedades termodinámicas. La prueba de rangos con signo de Wilcoxon proporcionó satisfactoriamente la base estadística necesaria para descartar modelos y finalmente elegir aquel que tuviera el menor número de neuronas. Esta prueba estadística, junto con la técnica del re-muestreo, pueden ser aplicadas al proceso de selección de modelos de RNA ya que presenta las siguientes ventajas:

- La metodología es suficientemente general para poder aplicarse a sistemas termodinámicos más complejos.
- La base estadística para la selección del mejor modelo de RNA permite la aplicación de esta metodología a datos de compuestos químicos sin importar si son polares o no, eliminando la necesidad de realizar experimentos para determinar parámetros especiales de ajuste para las ecuaciones de estado.
- Una vez obtenidos los pesos y umbrales del modelo de RNA, el cálculo de las propiedades termodinámicas es matemáticamente más simple, comparado con los esquemas iterativos empleados por las ecuaciones de estado.
- La ecuación resultante del modelo de RNA puede utilizarse en un dispositivo de control, como un PLC (controlador lógico programable).
- El error obtenido con la aplicación de la RNA, es mucho menor que el que puede alcanzar una EoS. De hecho, la mayoría de los errores que presenta la RNA, son prácticamente cero. Lo que incrementa la confiabilidad en su aplicación.

Así pues, se puede concluir que la metodología propuesta se puede aplicar con éxito a los refrigerantes seleccionados, generando modelos de RNA capaces de predecir propiedades termodinámicas en el equilibrio líquido-vapor. El análisis de resultados muestra que esta metodología es suficientemente general como para ser aplicada a la predicción de propiedades de otros refrigerantes, incluso otros compuestos de naturaleza química diferente.

## 4.1. Recomendaciones

La metodología aquí presentada fue aplicada solo a 6 refrigerantes y probó su generalidad, sin embargo sería bueno usarla en más refrigerantes y comprobar si tiene el mismo éxito.

El algoritmo propuesto no fue aplicado a la zona de sobrecalentamiento de los refrigerantes, por lo tanto no se sabe si puede ser utilizada para predecir propiedades termodinámicas en ese estado físico. En teoría, la RNA no notará la diferencia entre los datos en ELV y la zona de sobre calentamiento, simplemente tratará de aprender. Una vez que se construyan RNAs para predecir los datos de vapor sobrecalentado, se podrán sumar a las RNAs de la zona de equilibrio líquido-vapor que se obtuvieron en este estudio. Así será posible sustituir los diagramas comúnmente usados en el diseño de sistemas de refrigeración, como el Diagrama de Mollier.

Hay ciertas mezclas de sustancias químicas que presentan azeotropía. La metodología desarrollada aquí tampoco se ha probado para estos casos. Esta aplicación sería especialmente útil en casos de refinación de hidrocarburos o procesos de destilación.

En la actualidad es de enorme importancia encontrar nuevas formas tanto de producir como de aprovechar la energía. Los sistemas de aire acondicionado pueden representar la mayor parte del consumo de energía eléctrica en zonas con clima extremo. Esto plantea un nuevo reto: ¿cómo construir equipos de aire acondicionado que sólo utilicen la cantidad de energía que se requiere en un momento determinado? Normalmente los equipos están diseñados para un flujo determinado de refrigerante, una sola velocidad de ventiladores, una sola capacidad para el compresor. Esto fija las cantidades de energía que el equipo consume y limita las posibilidades de optimización. Sin embargo, hay ocasiones en que varios dispositivos como bombas y compresores, trabajan en conjunto. Es entonces cuando se puede optimizar el sistema ya sea encendiendo

o apagándolos según se requiera. Precisamente en este punto es donde la aplicación de la inteligencia artificial se convierte en una herramienta esencial, pues es la que brinda la información oportuna y necesaria para que los dispositivos sean activados o no. El conocimiento de las propiedades termodinámicas permite obtener los requerimientos energéticos de un sistema de acondicionamiento de aire, es por ello que el cálculo de dichas propiedades es de gran importancia en la optimización de sistemas de climatización. Las ecuaciones de estado permiten calcular las propiedades termodinámicas de los refrigerantes a través de un algoritmo iterativo que puede consumir un tiempo considerable, si bien no es demasiado, no es tan rápido si se deseara ajustar *en tiempo real* un sistema. Al utilizar una red neuronal artificial, el cálculo de propiedades se reduce a una ecuación casi lineal, donde los parámetros ajustados previamente en dicha ecuación, permiten obtener el valor de las propiedades termodinámicas sin requerir un proceso iterativo. Si bien es cierto que al principio, para obtener la mejor estructura de la RNA se requiere tiempo computacional y también humano, también es cierto que la ecuación resultante permite obtener en un solo paso, las propiedades termodinámicas deseadas. Así pues, solo se requiere un poco esfuerzo en el inicio, pero los resultados se pueden utilizar no solo en un proceso, sino en varios.

Los pesos y umbrales obtenidos, junto con la función de transferencia, pueden utilizarse en controladores programables que actúen sobre los dispositivos que conforman un sistema de acondicionamiento de aire. Esto permite construir equipos «inteligentes» que son capaces de adaptarse a las condiciones climáticas, es decir, que dependiendo de las variables climatológicas y necesidades de remoción de calor, pueden encender o apagar dispositivos y así reducir el consumo energético. Cabe mencionar que disminuir el consumo energético es sinónimo de reducir el impacto al medio ambiente, producto de los procesos de la explotación de los recursos energéticos. Así pues, la aplicación de la Inteligencia Artificial al cálculo de propiedades termodinámicas de refrigerantes se

suma a los esfuerzos por preservar el medio ambiente y la óptima explotación de los recursos naturales.

## 4.2. Trabajo futuro

De acuerdo con los resultados obtenidos en este estudio, se pueden definir por lo menos tres tópicos de investigación por abordar posteriormente. El primero es aplicar la metodología aquí presentada a la predicción de los datos de la zona de sobrecalentamiento de los refrigerantes estudiados, tal como se mencionó en la sección anterior. Esto será útil para reproducir los datos de las propiedades termodinámicas para cualquier valor de presión y temperatura. El segundo aspecto que se puede abordar es el referente a la explotación del algoritmo que permite encontrar la mejor estructura de RNA, mismo que se pretende patentar o en todo caso, incorporarlo a la *Fundación R* para contribuir a la investigación y enseñanza de cómputo estadístico a nivel mundial. Dicha Fundación es la de mayor prestigio internacional y la publicación de código en ella se considera como de alto prestigio e impacto. Un tercer tema comprende el relativo a la aplicación de las ecuaciones obtenidas en las RNAs directamente a dispositivos de control de procesos. El autor de este trabajo tiene posibilidad de aplicarlo a un par de industrias productoras de hielo.

# Apéndice A

## Análisis de los Modelos de RNA de los refrigerantes

### A.1. Análisis estadístico de la RNA del R12

El algoritmo de generación de modelos de RNA posibles, produjo gráficas que permiten comparar los MSE y  $R^2$  para el proceso de validación del R12. En las Figuras 24 y 25 podemos ver la variación del MSE y el  $R^2$  a través de los 100 ciclos de cada modelo generado. En las Figuras 26 y 27 se puede el valor del MSE y  $R^2$  promedios de cada modelo y se puede comparar con los demás. De acuerdo con las dos gráficas anteriores, se puede ver que el modelo II tiene el menor MSE y tiene de los más altos valores de  $R^2$ . Por lo tanto, se puede decir que el modelo II, con  $n_{best} = 6$  es el mejor candidato hasta el momento.

En la Tabla A.1 se pueden observar los resultados de la Prueba de Rangos con Signo de Wilcoxon, la cual indica que el Modelo II, con 6 neuronas, no es significativamente diferente del Modelo III. Por lo tanto, se confirma la elección del Modelo II, con 6 neuronas en la capa oculta, como el mejor modelo de RNA para predecir las propiedades termodinámicas del R12.

Una vez que se eligió definitivamente el Modelo II, se construyeron las gráficas que confirman el óptimo ajuste de la RNA a los datos reales. Al observar la Figura 28 se aprecia el ajuste de todos los datos reales con respecto a los que predice el Modelo.

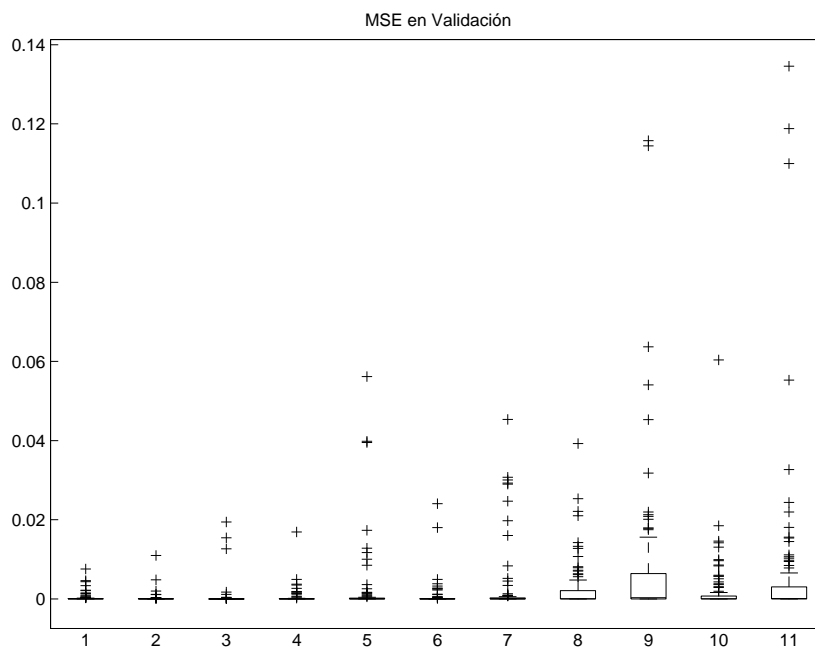


Figura 24: Boxplot de MSE en modelos del R12.

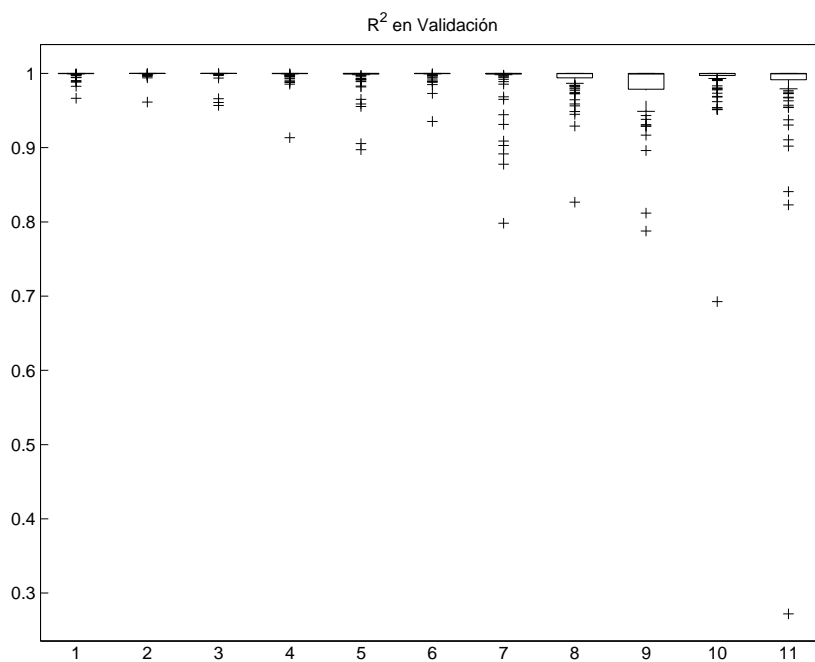


Figura 25: Boxplot de  $R^2$  en modelos del R12.

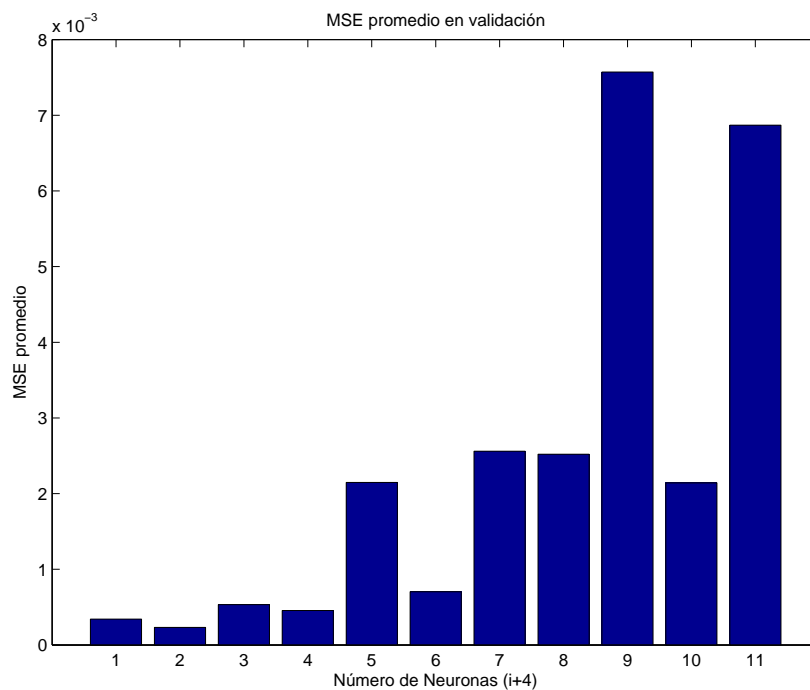


Figura 26: Variación del MSE promedio de los modelos del R12.

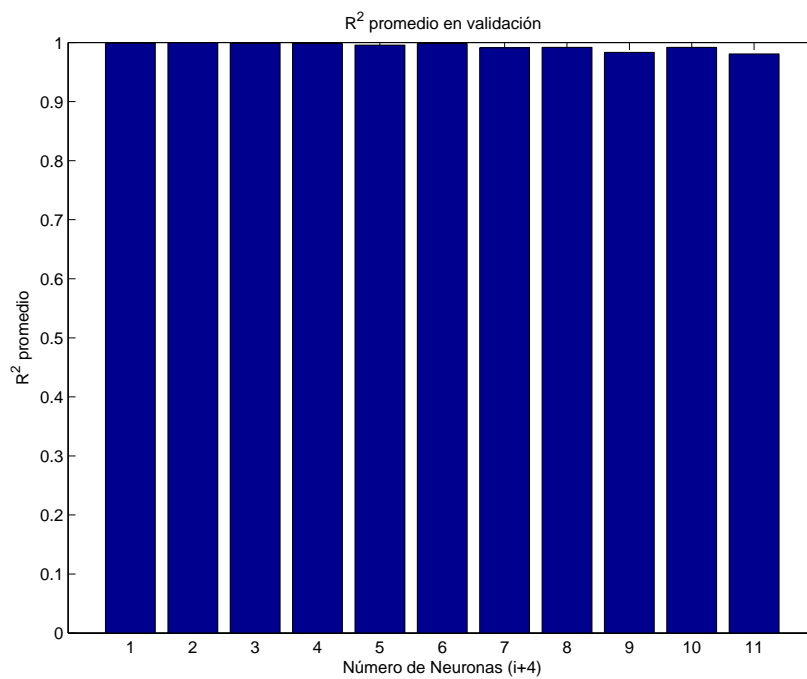


Figura 27: Variación del  $R^2$  promedio de los modelos del R12.

MSE promedio	Modelo	Modelos sin diferencia significativa
0.00023	2	3
0.00034	1	3,4,5,6,7
0.00046	4	1,3,5,6,7
0.00053	3	1,2,4
0.00070	6	1,4,5,7
0.00214	10	7,8
0.00215	5	1,4,6,7
0.00252	8	7,10,11
0.00256	7	1,4,5,6,8,10
0.00687	11	8,9
0.00757	9	11

Tabla A.1: Prueba de Wilcoxon para el R12.

La cercanía a la línea recta indica que el Modelo predice satisfactoriamente los datos reales. En la Figura 29 se puede ver que durante el proceso de validación, los datos predichos por el Modelo, se solapan prácticamente a la línea recta. Esto refleja que la RNA posee una excelente capacidad de predicción de los datos reales que no se incluyeron en el entrenamiento. Los valores de los pesos y umbrales se encuentran en la Tabla A.7.

Para confirmar que el Modelo elegido es el mejor, se construyen gráficas que muestran como varía cada propiedad termodinámica con respecto a la variable de entrada, en este caso, la temperatura. En las Figuras 30 a la 36 se pueden observar dichas gráficas, corroborando que las propiedades predichas por la RNA se acercan satisfactoriamente a los valores reales.

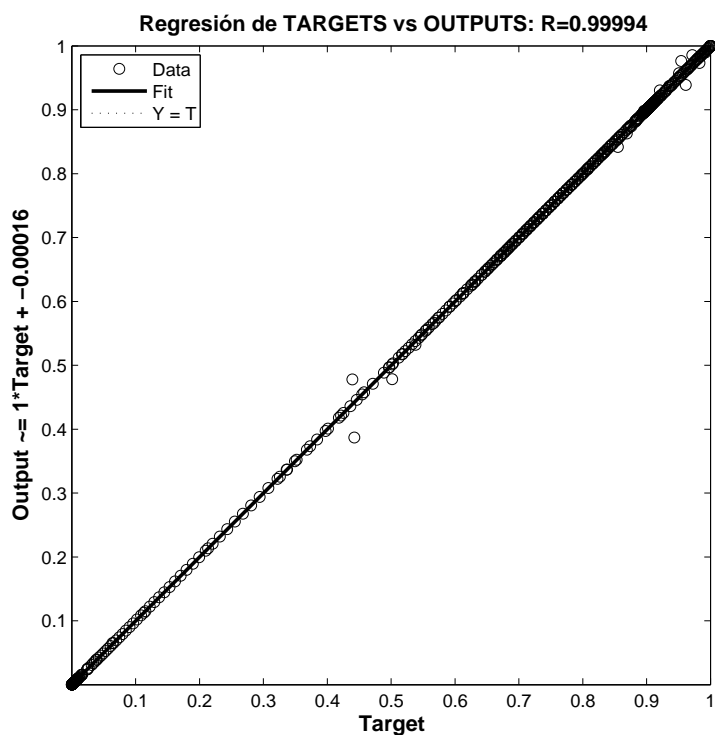


Figura 28: Regresión final general de la RNA para el R12.

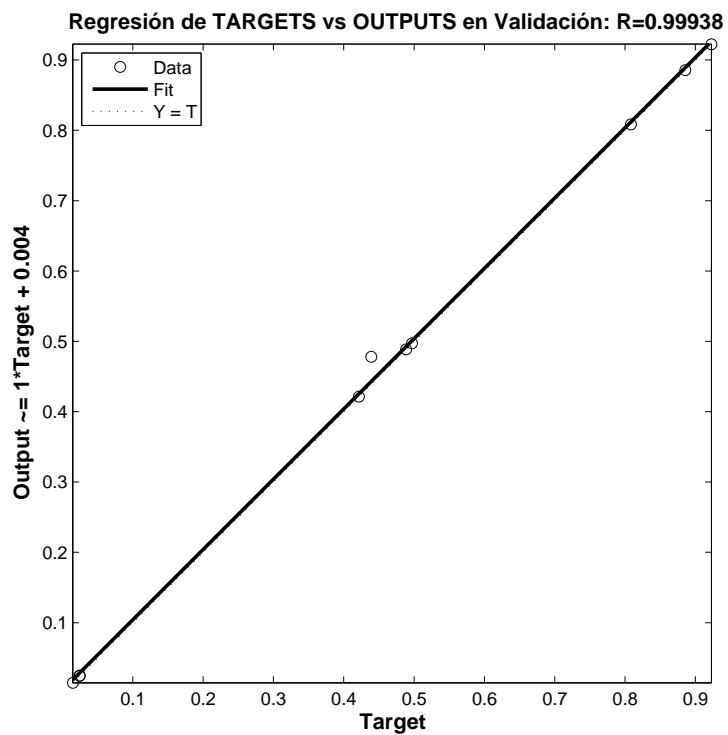


Figura 29: Regresión final en validación de la RNA para el R12.

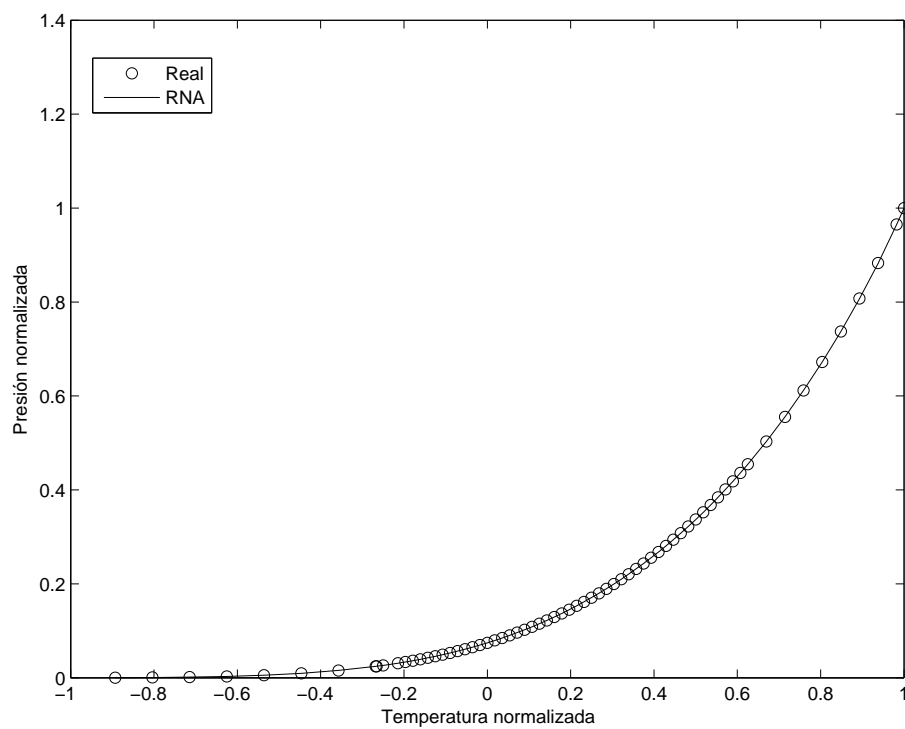
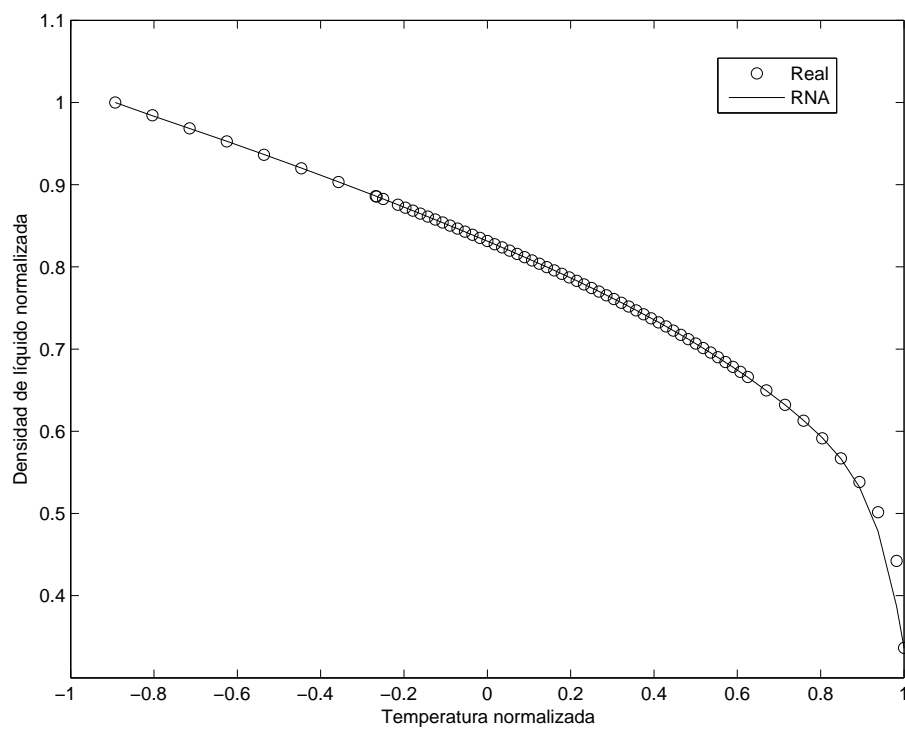
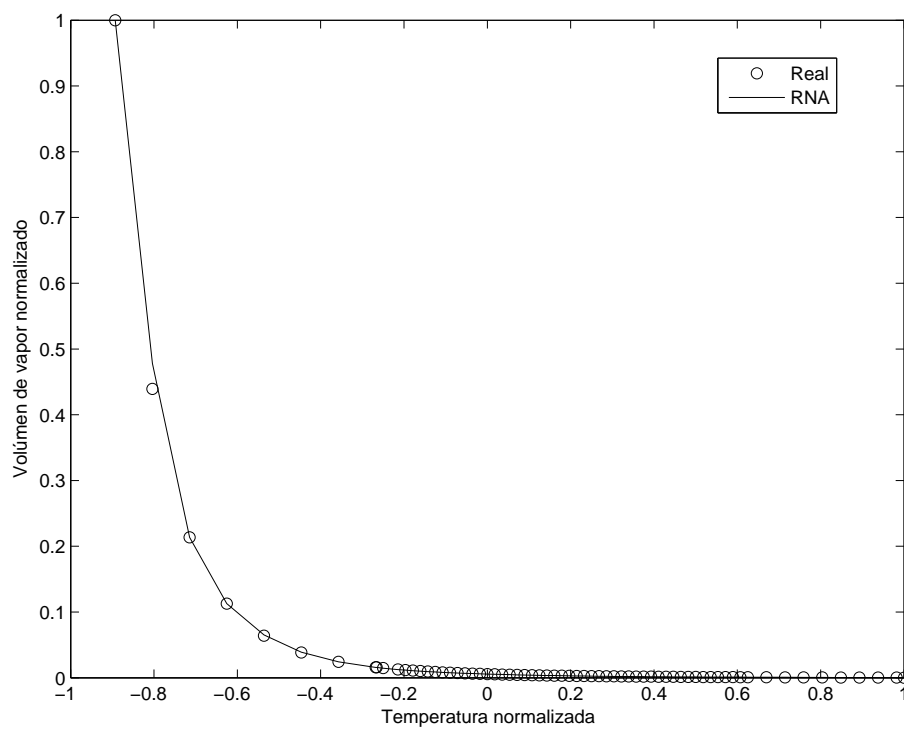
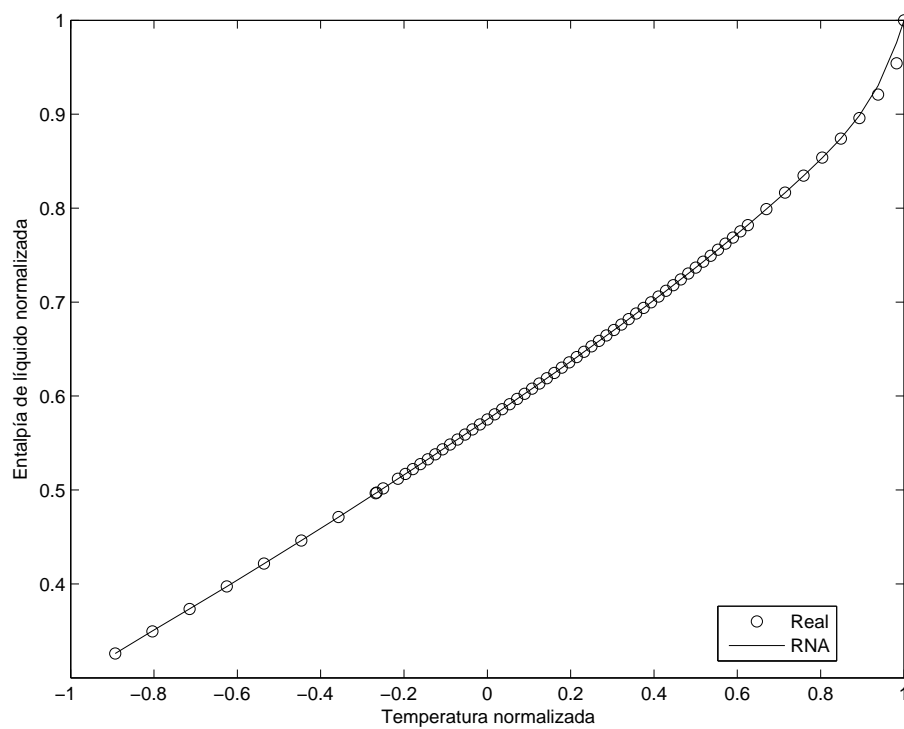
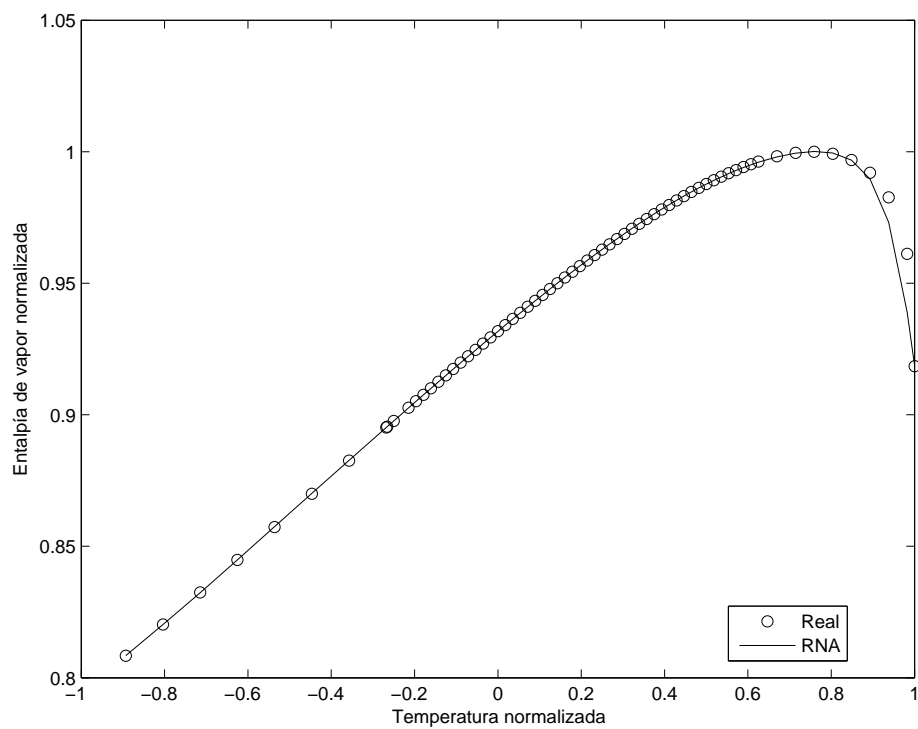
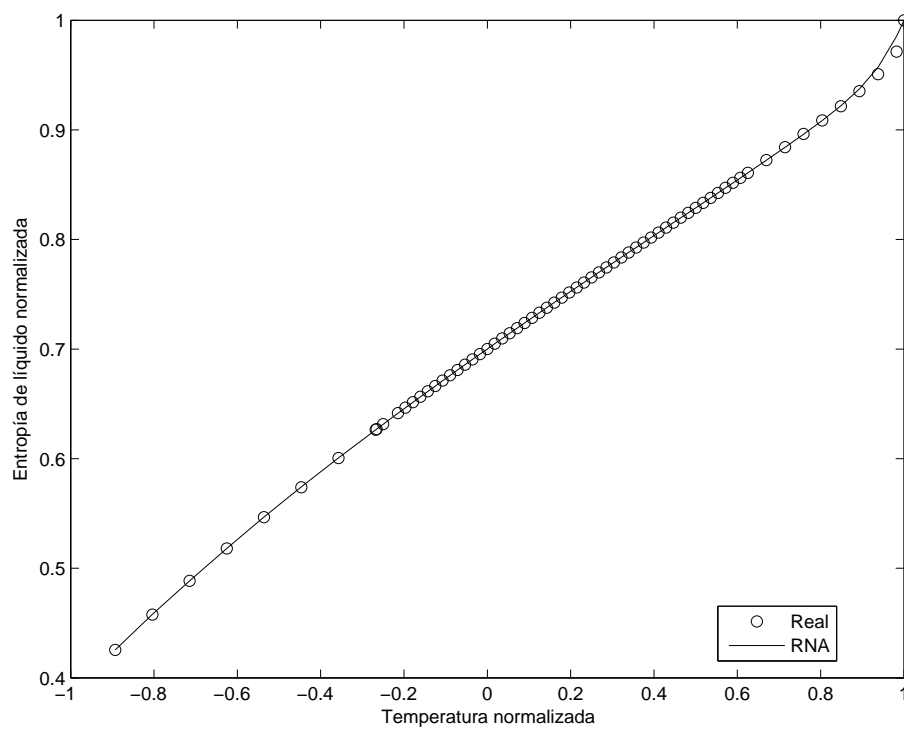


Figura 30: Presión real vs RNA para el R12.

Figura 31:  $D_l$  real vs RNA para el R12.

Figura 32:  $D_v$  real vs RNA para el R12.Figura 33:  $H_l$  real vs RNA para el R12.

Figura 34:  $H_v$  real vs RNA para el R12.Figura 35:  $S_l$  real vs RNA para el R12.

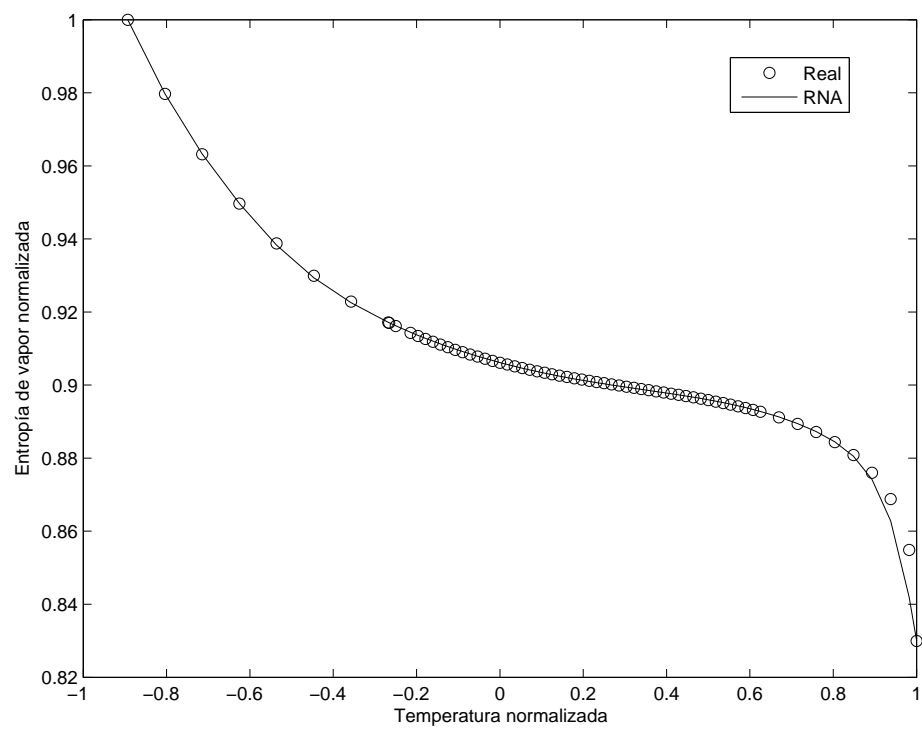


Figura 36:  $S_v$  real vs RNA para el R12.

## A.2. Análisis estadístico de la RNA del R22

Al igual que para el R12, al finalizar el algoritmo se produjeron gráficas que permiten comparar los MSE y  $R^2$  para el proceso de validación del R22. En las Figuras 37 y 38 podemos ver la variación del MSE y el  $R^2$  a través de los 100 ciclos de cada modelo generado. En las Figuras 39 y 40 se puede el valor del MSE y  $R^2$  promedios de cada modelo y se pueden comparar con los demás. De acuerdo con estas dos últimas gráficas, se puede ver que el modelo I tiene el menor MSE y tiene de los más altos valores de  $R^2$ . Así pues, es posible decir que el modelo I, con  $n_{best} = 5$  es el mejor candidato hasta el momento.

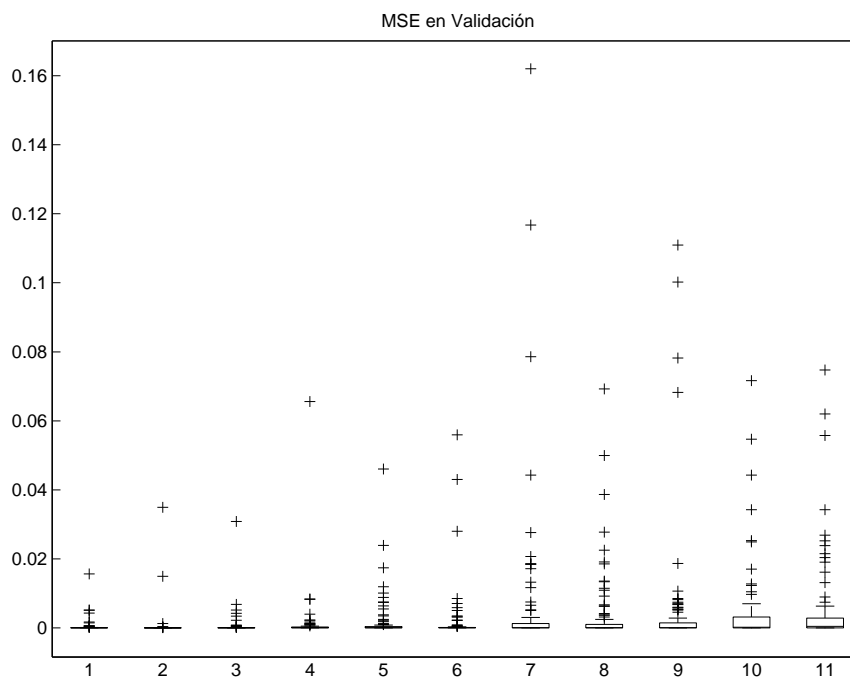


Figura 37: Boxplot de MSE en modelos del R22.

En la Tabla A.2 se muestran los resultados de la Prueba de Rangos con Signo de Wilcoxon, la cual indica que el Modelo I, con 5 neuronas, no es significativamente diferente de los Modelos III y IV, pero sí es más sencillo. Por lo tanto, se confirma la

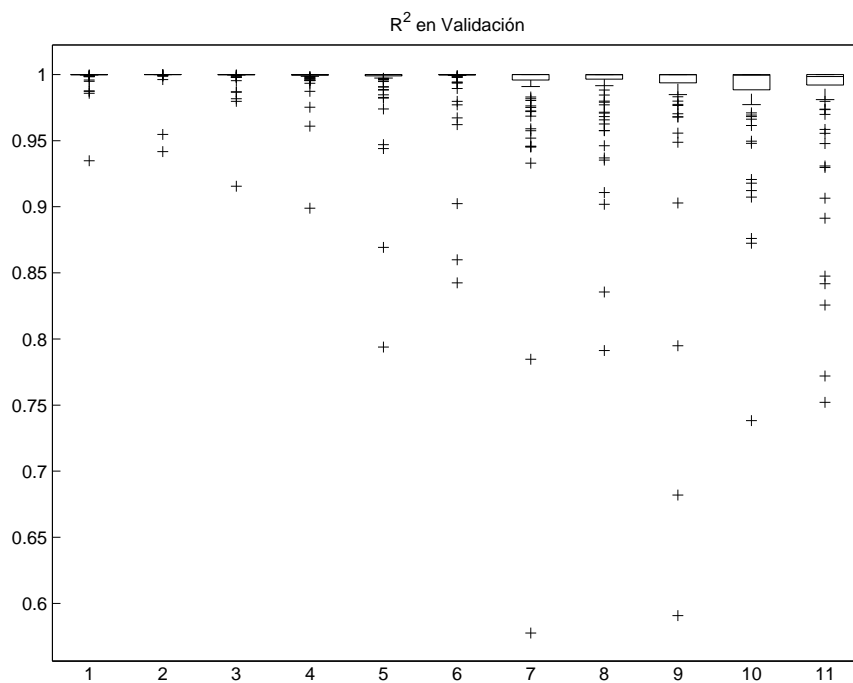


Figura 38: Boxplot de  $R^2$  en modelos del R22.

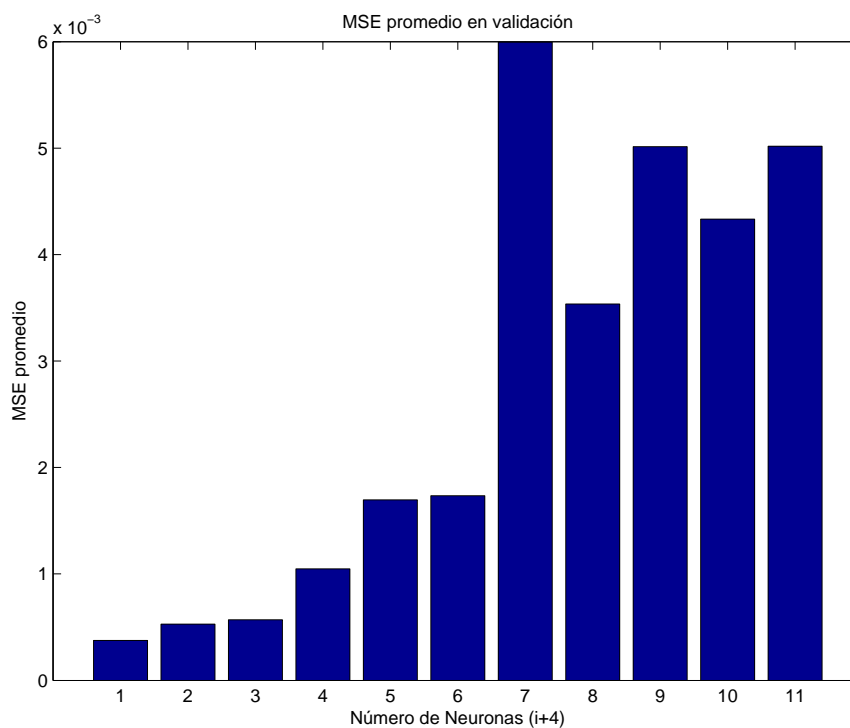


Figura 39: Variación del MSE promedio de los modelos del R22.

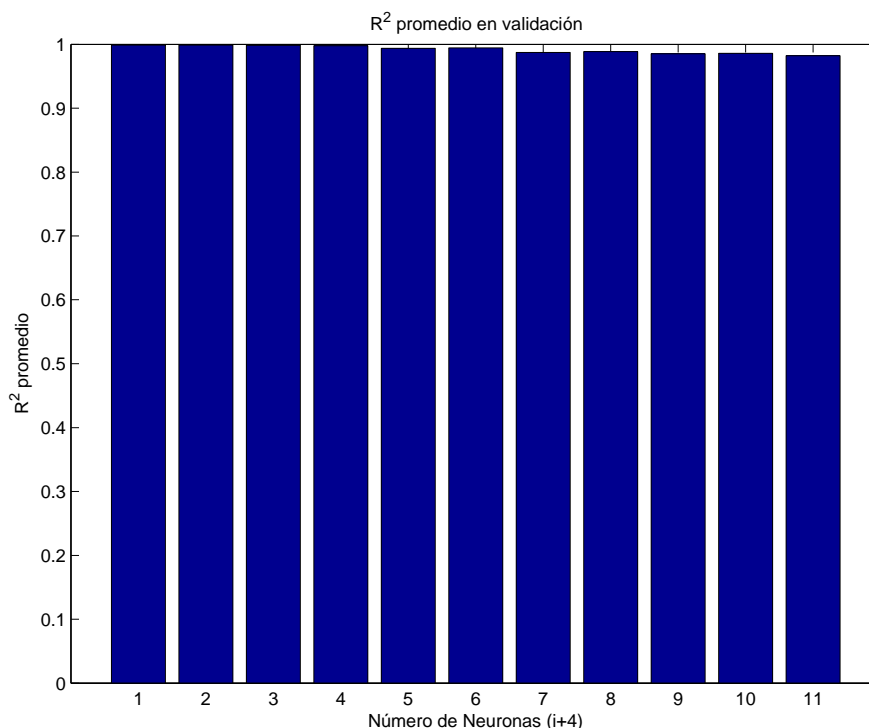


Figura 40: Variación del  $R^2$  promedio de los modelos del R22.

elección del Modelo I, con 5 neuronas en la capa oculta, como el mejor modelo de RNA para predecir las propiedades termodinámicas del R22.

Una vez elegido definitivamente el Modelo I, se construyen las gráficas que confirman que la RNA predice valores cercanos satisfactoriamente a los datos reales. En la Fig. 41 se muestra un gráfico de regresión de todos los datos reales con respecto a los obtenidos por el Modelo. Nuevamente, la cercanía a la línea recta indica que el Modelo puede predecir los datos reales con bastante exactitud. En la Fig. 42 se puede ver que durante el proceso de validación, los datos predichos por el Modelo, se acercan mucho a la línea recta. Debido a que los datos mostrados en esta última gráfica no se incluyeron en el entrenamiento, se puede concluir que el Modelo elegido predice muy bien las propiedades termodinámicas del refrigerante en cuestión. Los valores de los pesos y umbrales se encuentran en la Tabla A.8.

MSE promedio	Modelo	Modelos sin diferencia significativa
0.00038	1	3,4
0.00053	2	3
0.00057	3	1,2
0.00105	4	1,5,6,7
0.00170	5	4,6,7
0.00173	6	4,5,7
0.00353	8	7,9,10
0.00433	10	7,8,9,11
0.00501	9	7,8,10,11
0.00502	11	9,10
0.00600	7	4,5,6,8,9,10

Tabla A.2: Prueba de Wilcoxon para el R22.

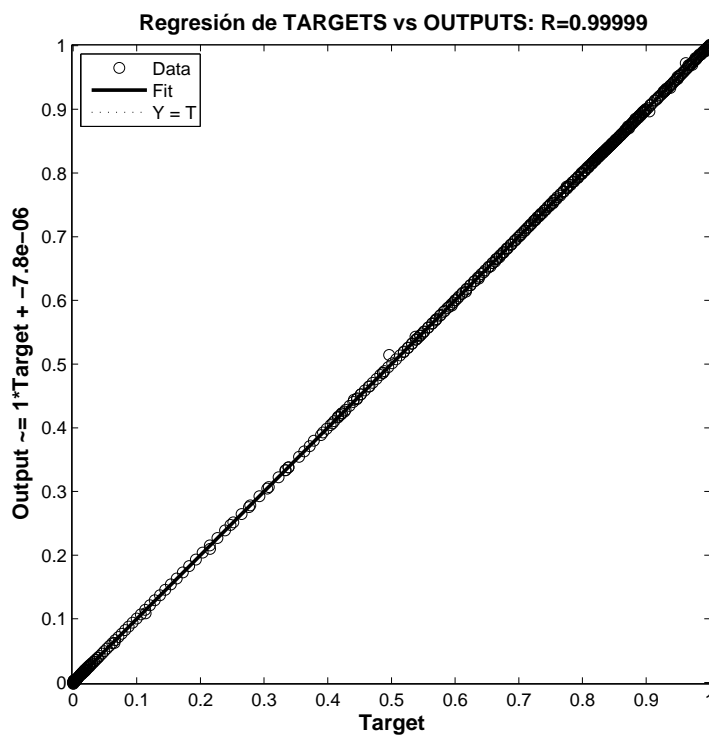


Figura 41: Regresión final general de la RNA para el R22.

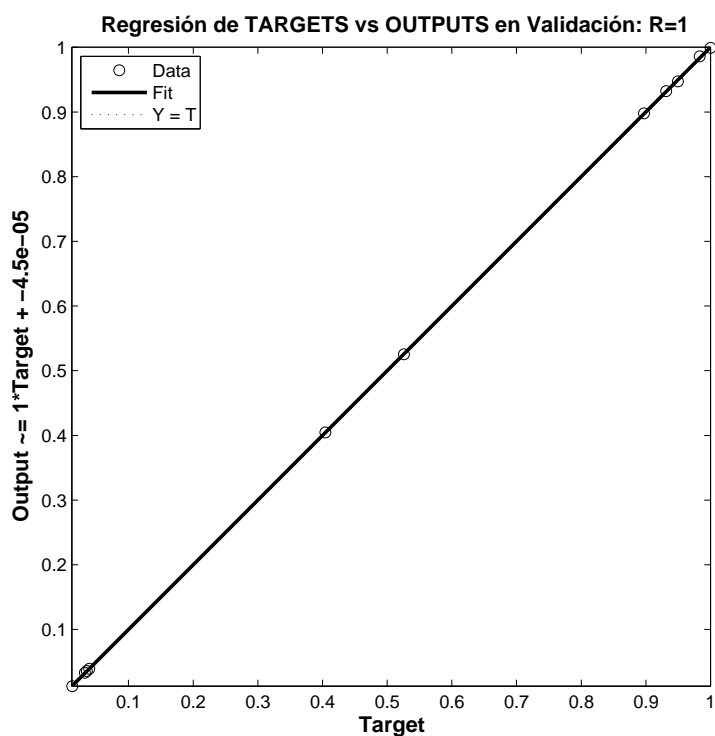


Figura 42: Regresión final en validación de la RNA para el R22.

Posteriormente se construyeron las gráficas que muestran como varía cada propiedad termodinámica con respecto a la variable de entrada, en este caso, la temperatura. Esto es con el fin de confirmar que el Modelo elegido es el mejor. En las Figuras 43 a la 49 se pueden observar dichas gráficas, corroborando que las propiedades predichas por la RNA se acercan satisfactoriamente a los valores reales.

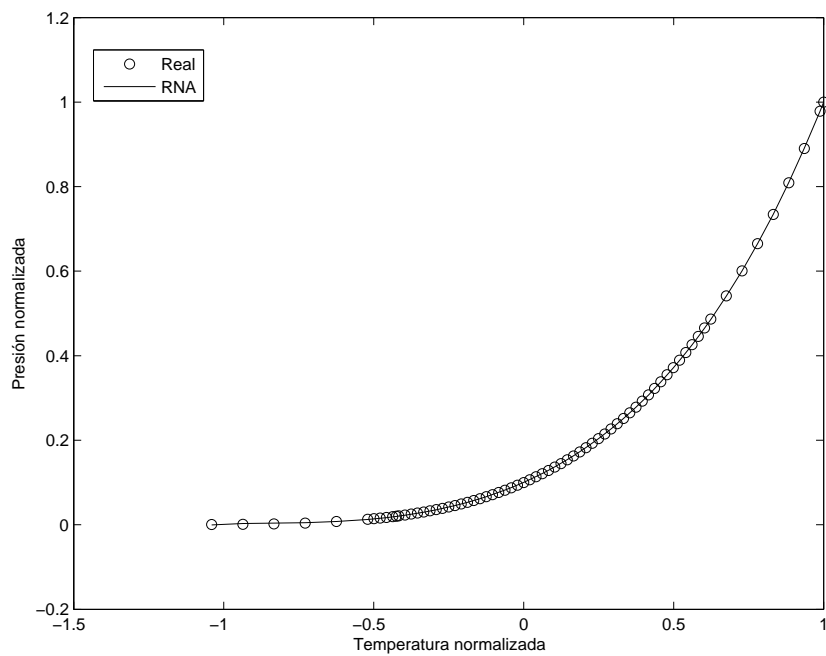
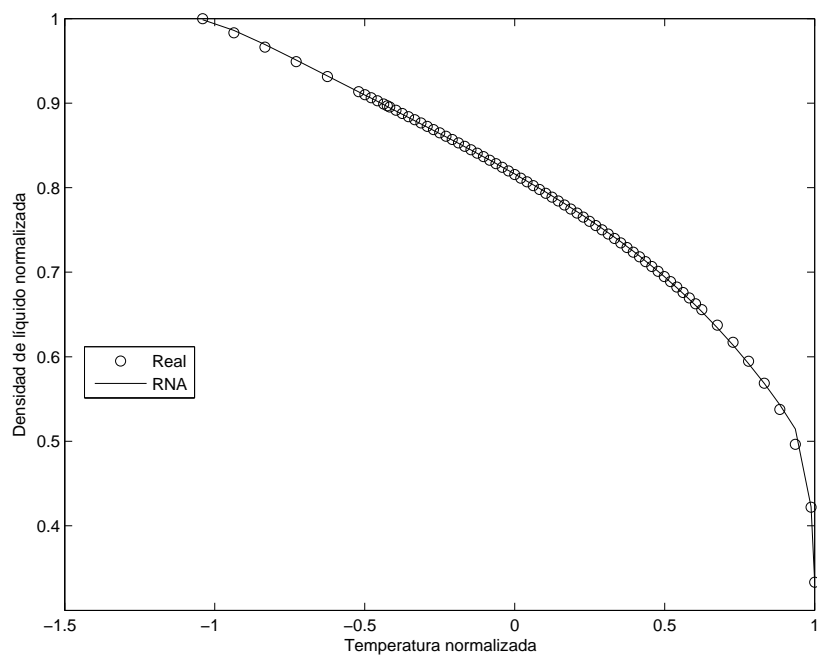
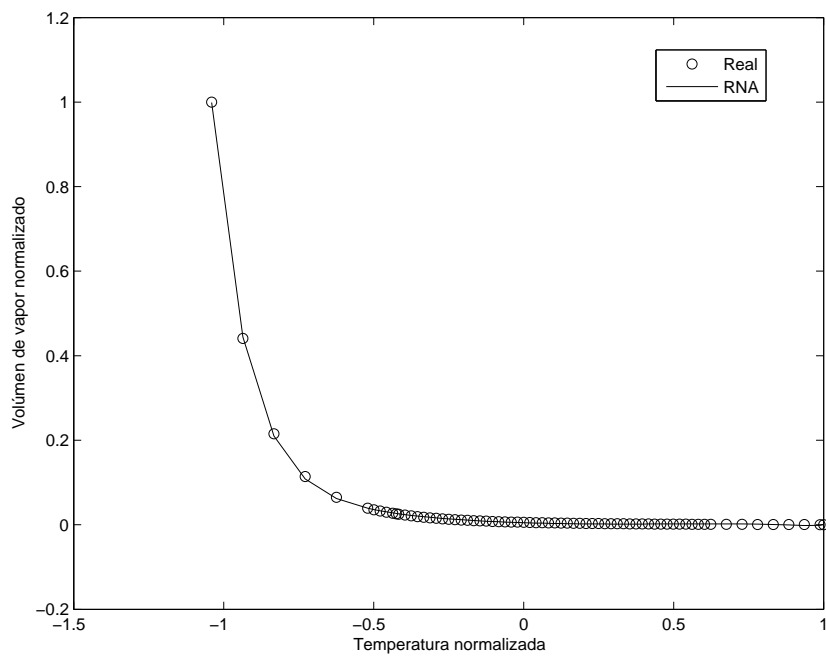
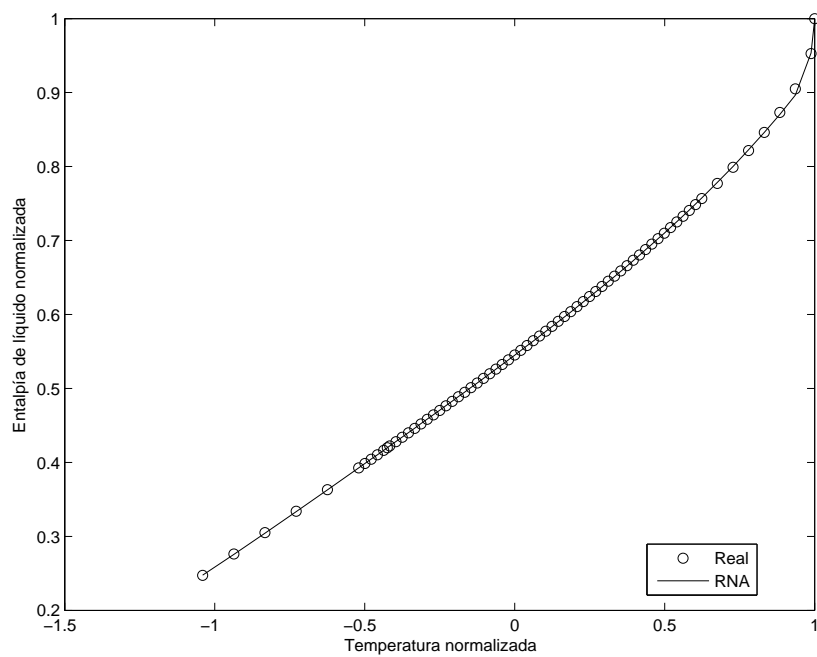


Figura 43: Presión real vs RNA para el R22.

Figura 44:  $D_l$  real vs RNA para el R22.

Figura 45:  $D_v$  real vs RNA para el R22.Figura 46:  $H_l$  real vs RNA para el R22.

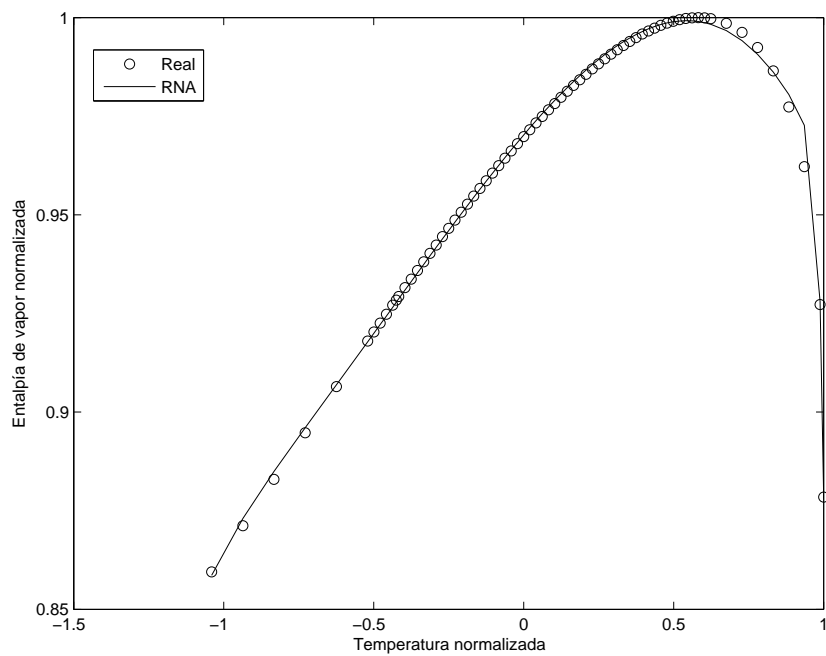


Figura 47:  $H_v$  real vs RNA para el R22.

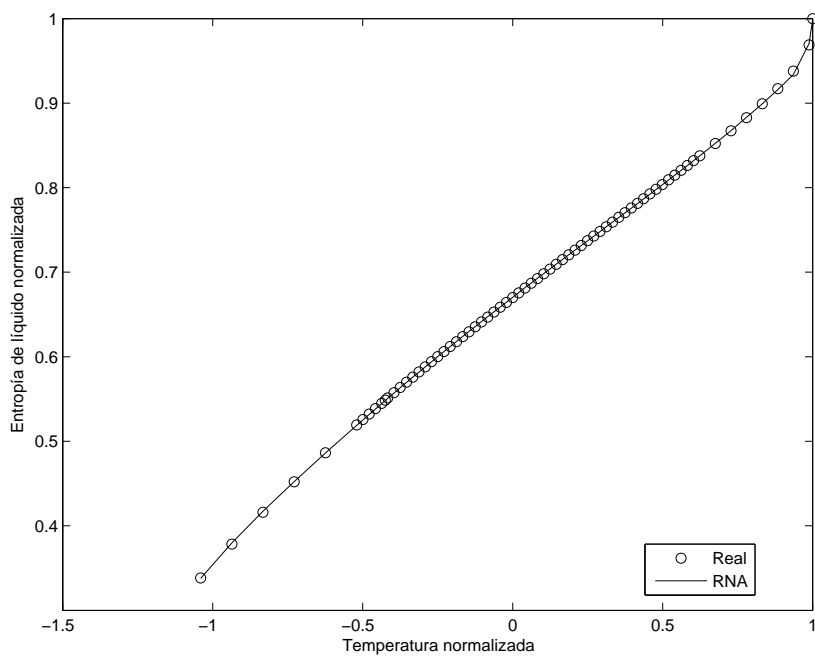


Figura 48:  $S_l$  real vs RNA para el R22.

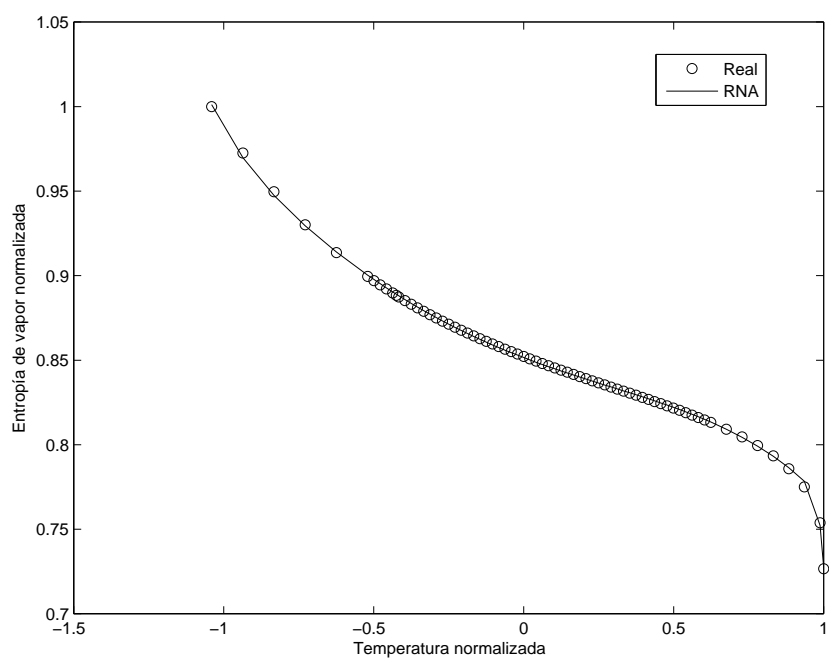


Figura 49:  $S_v$  real vs RNA para el R22.

### A.3. Análisis estadístico de la RNA del R134a

Para este refrigerante se sigue el mismo análisis que los anteriores, al final del algoritmo se obtuvieron la gráficas que permiten comparar los MSE y  $R^2$  para el proceso de validación del R134a. En las Figuras 50 y 51 se muestra la variación del MSE y el  $R^2$  a través de los 100 ciclos de cada modelo generado. En las Figuras 52 y 53 se puede el valor del MSE y  $R^2$  promedios de cada modelo y así es posible compararlos con los demás. De acuerdo con estas dos últimas gráficas, se puede ver que el modelo IV tiene el menor MSE y tiene de los más altos valores de  $R^2$ . Así pues, es posible decir que este modelo, con  $n_{best} = 8$  es el mejor candidato hasta el momento.

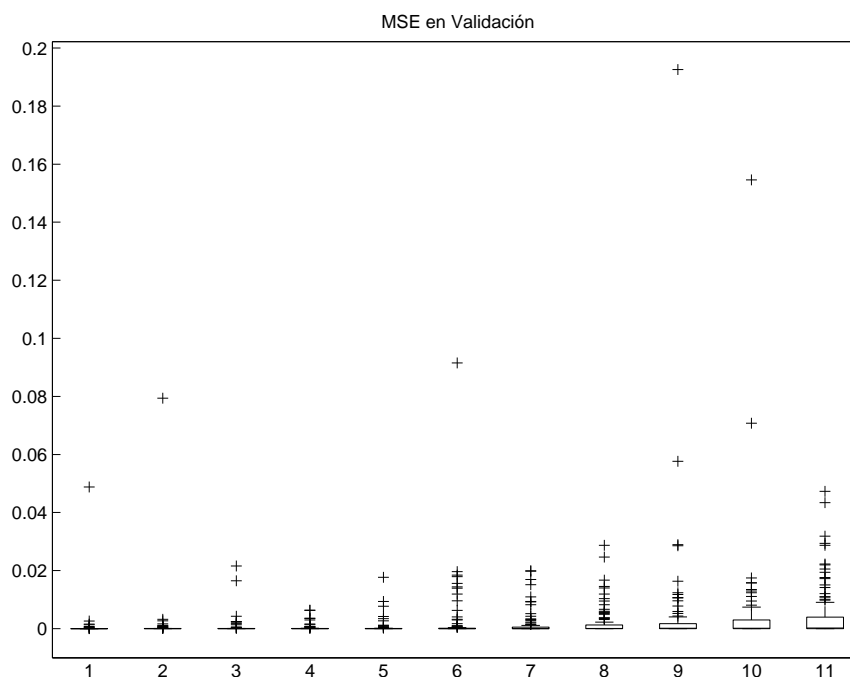


Figura 50: Boxplot de MSE en modelos del R134a.

En la Tabla A.3 se muestran los resultados de la Prueba de Rangos con Signo de Wilcoxon, la cual indica que el Modelo IV, con 8 neuronas, no es significativamente diferente de los Modelos II, III y V. Por esta razón, en vez de elegir al Modelo IV, se elige al modelo II, con  $n_{best} = 6$ .

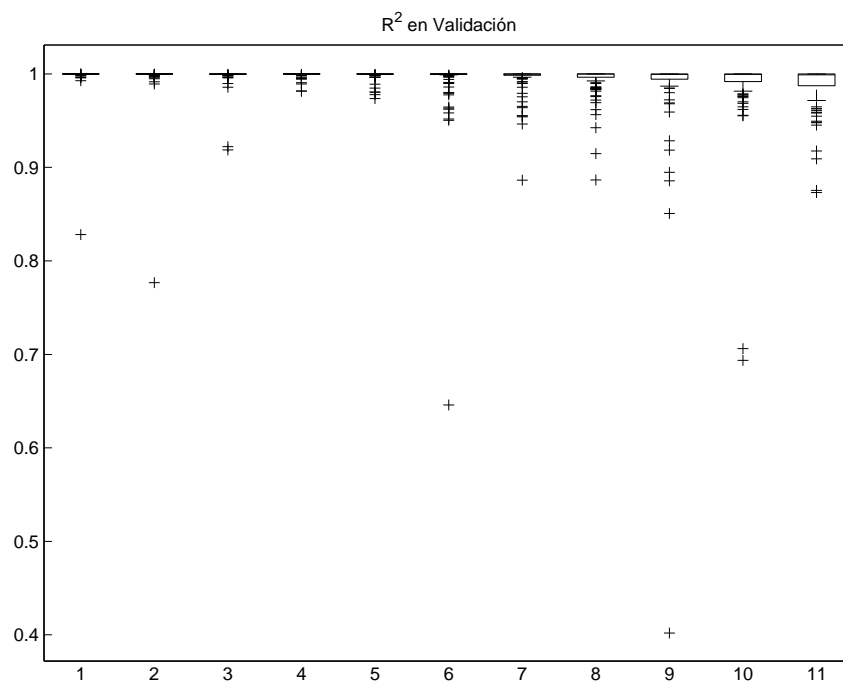


Figura 51: Boxplot de  $R^2$  en modelos del R134a.

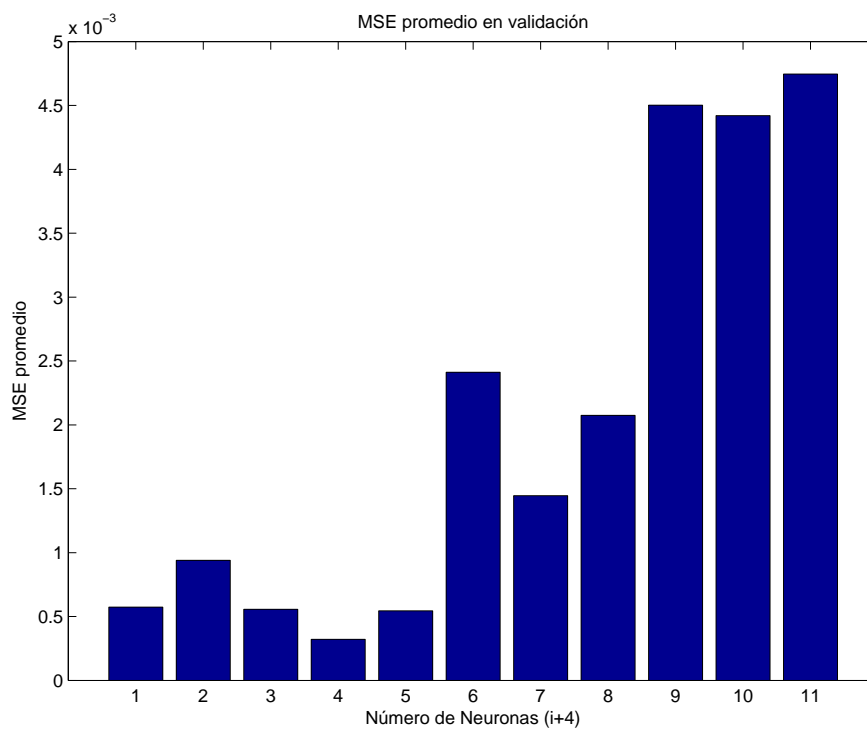


Figura 52: Variación del MSE promedio de los modelos del R134a.

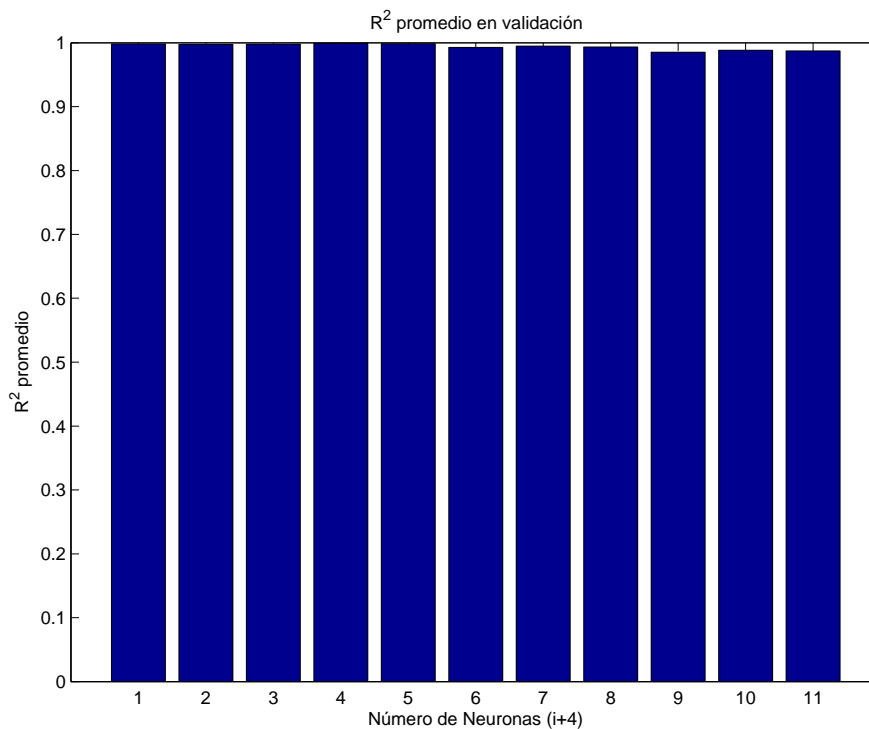


Figura 53: Variación del  $R^2$  promedio de los modelos del R134a.

MSE promedio	Modelo	Modelos sin diferencia significativa
0.00032	4	2,3,5
0.00054	5	3,4,6,7
0.00056	3	2,4,5
0.00057	1	2
0.00094	2	1,3,4
0.00145	7	5,6,8
0.00207	8	6,7,9,10
0.00241	6	5,7,8
0.00442	10	8,9,11
0.00450	9	8,10,11
0.00475	11	9,10

Tabla A.3: Prueba de Wilcoxon para el R134a.

Una vez elegido definitivamente el Modelo II, nuevamente se construyen las gráficas que confirman que la RNA predice valores cercanos satisfactoriamente a los datos reales. En la Fig. 54 se muestra un gráfico de regresión de todos los datos reales con respecto a los obtenidos por el Modelo. La cercanía a la línea recta indica que el Modelo puede predecir los datos reales con bastante exactitud. En la Fig. 55 se puede ver que durante el proceso de validación, los datos predichos por el Modelo, se acercan mucho a la línea recta. Debido a que los datos mostrados en esta última gráfica no se incluyeron en el entrenamiento, se puede concluir que el Modelo elegido predice muy bien las propiedades termodinámicas del refrigerante en cuestión. Los valores de los pesos y umbrales se encuentran en la Tabla A.9.

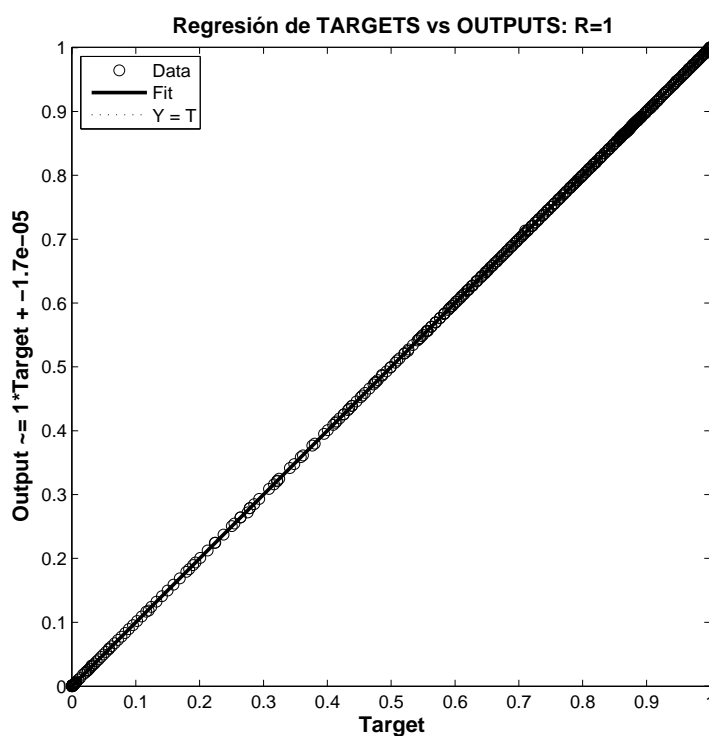


Figura 54: Regresión final general de la RNA para el R134a.

Posteriormente se construyeron las gráficas que muestran como varía cada propiedad termodinámica con respecto a la variable de entrada, en este caso, la temperatura. Esto

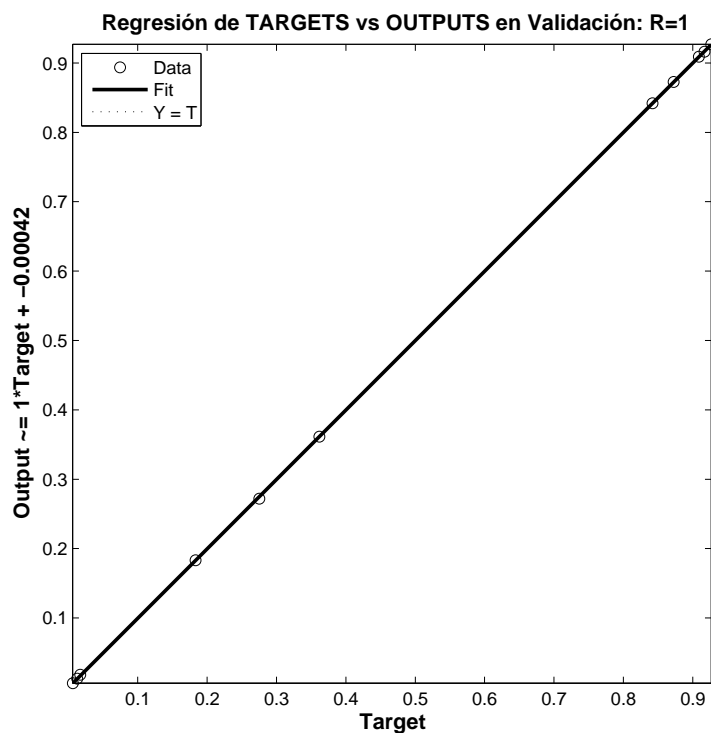


Figura 55: Regresión final en validación de la RNA para el R134a.

es con el fin de confirmar que el Modelo elegido es el mejor. En las Figuras 56 a la 62 se pueden observar dichas gráficas, corroborando que las propiedades predichas por la RNA se acercan satisfactoriamente a los valores reales.

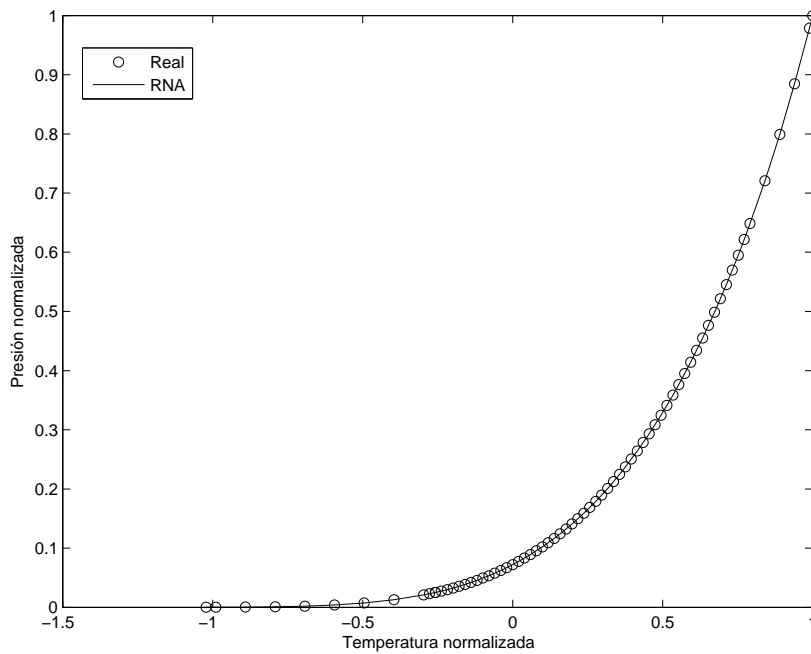
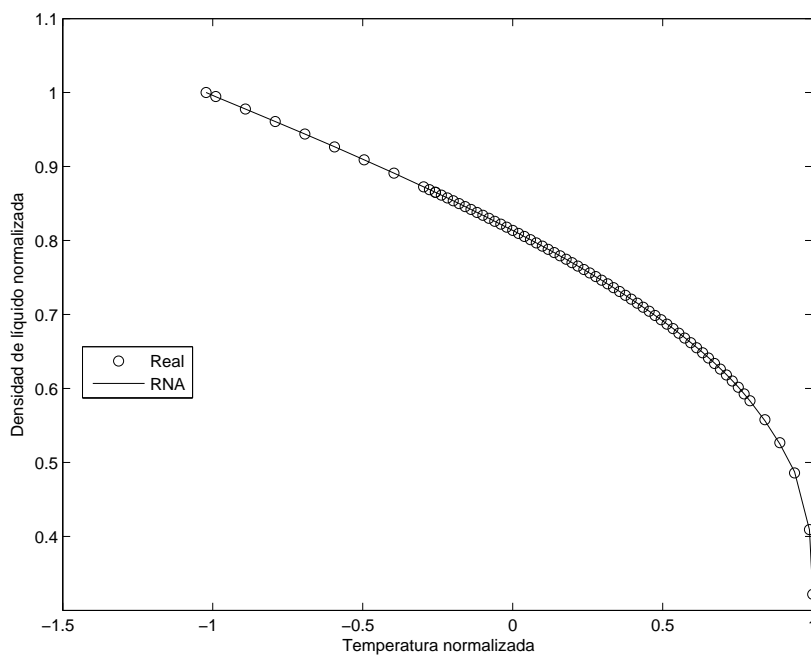
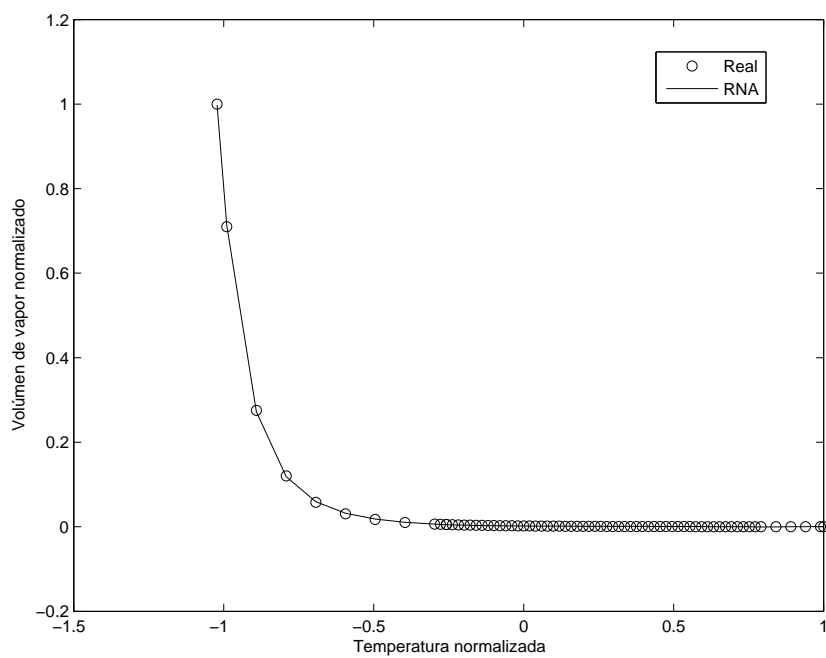
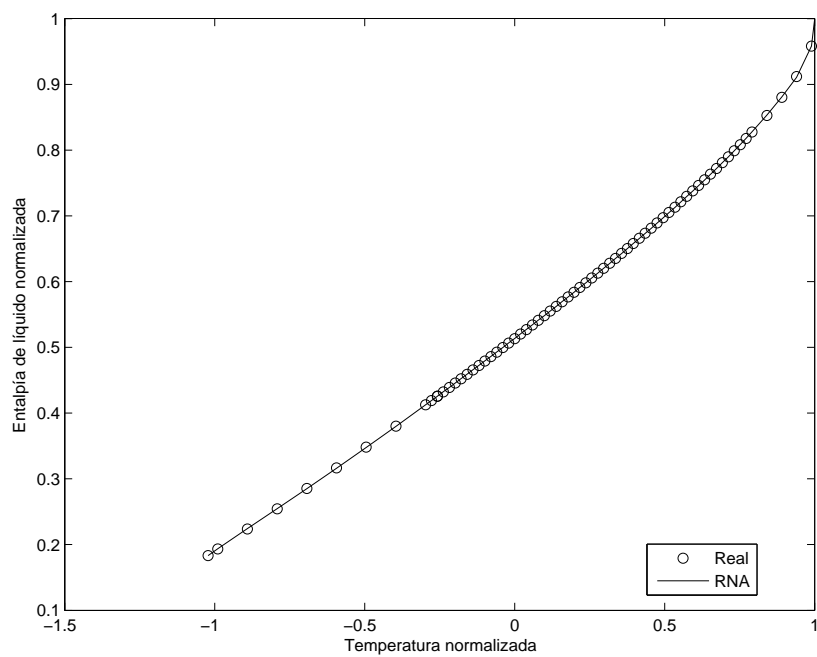


Figura 56: Presión real vs RNA para el R134a.

Figura 57:  $D_l$  real vs RNA para el R134a.

Figura 58:  $D_v$  real vs RNA para el R134a.Figura 59:  $H_l$  real vs RNA para el R134a.

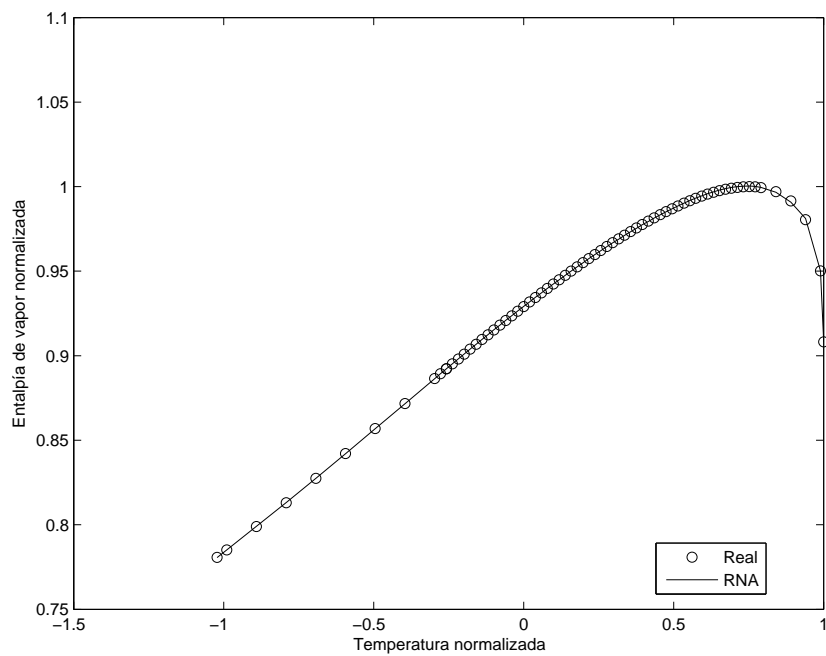


Figura 60:  $H_v$  real vs RNA para el R134a.

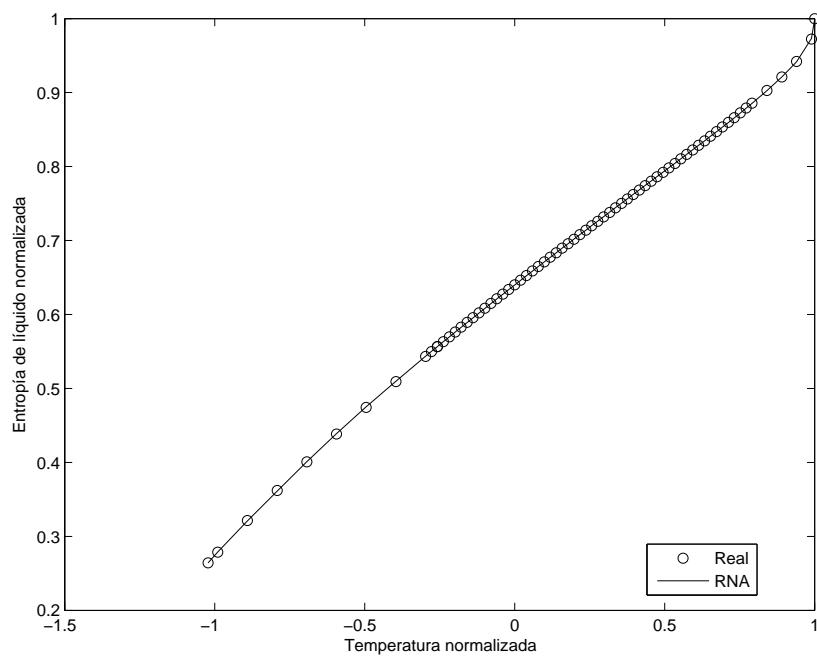


Figura 61:  $S_l$  real vs RNA para el R134a.

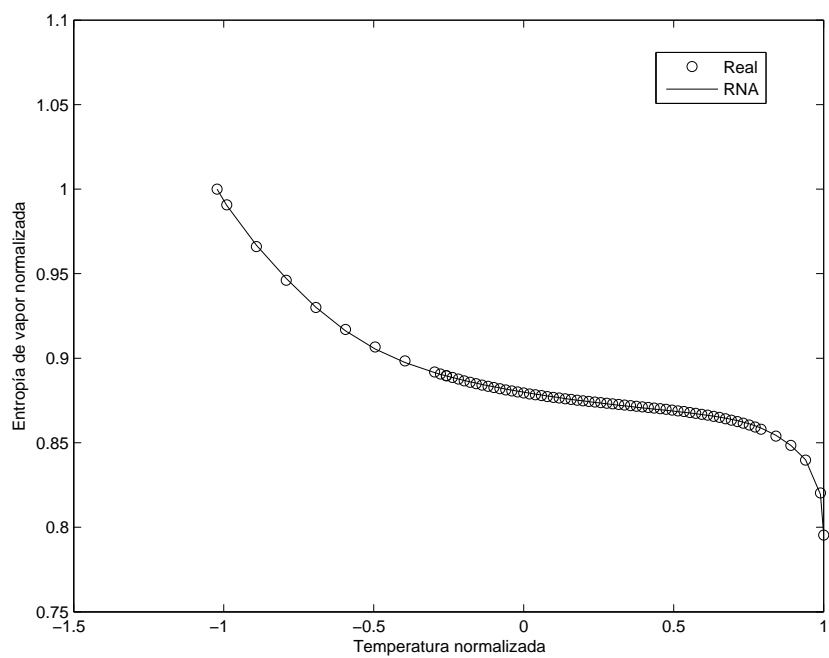


Figura 62:  $S_v$  real vs RNA para el R134a.

## A.4. Análisis estadístico de la RNA del R290

Se comienza con el análisis de los MSE y  $R^2$  para el proceso de validación del R290. En las Figuras 63 y 64 podemos ver la variación del MSE y el  $R^2$  a través de los 100 ciclos de cada modelo generado. En las Figuras 65 y 66 se puede el valor del MSE y  $R^2$  promedios de cada modelo y se pueden comparar con los demás modelos. De acuerdo con estas dos últimas gráficas, se puede ver que el modelo I tiene el menor MSE y tiene de los más altos valores de  $R^2$ . Así pues, es posible decir que el modelo I, con  $n_{best} = 5$  es el mejor candidato hasta el momento.

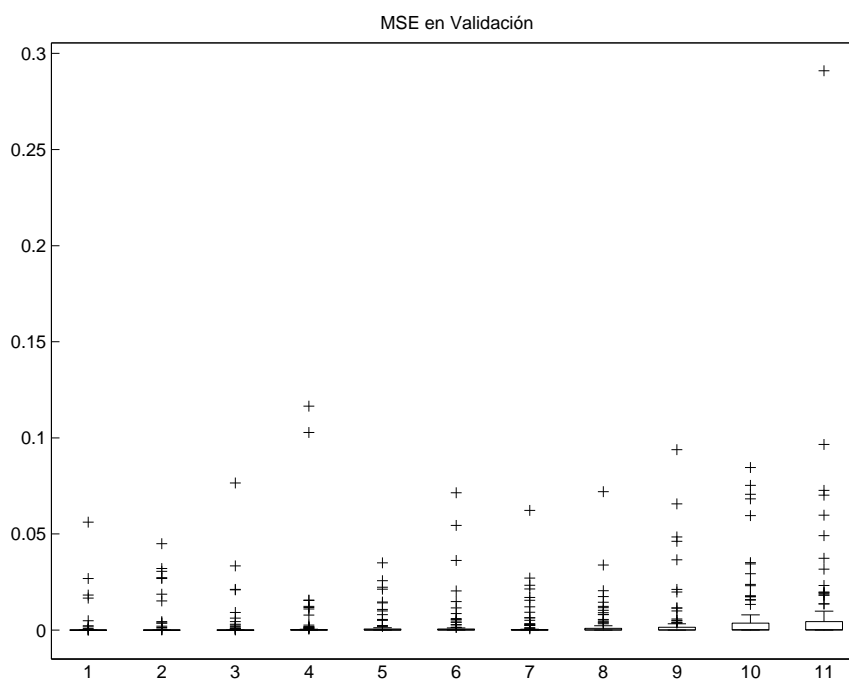


Figura 63: Boxplot de MSE en modelos del R290.

En la Tabla A.4 se muestran los resultados de la Prueba de Rangos con Signo de Wilcoxon, la cual indica que el Modelo I, con 5 neuronas, no es significativamente diferente de los Modelos II, III y IV, pero sí es más sencillo. Por lo tanto, se confirma la elección del Modelo I, con 5 neuronas en la capa oculta, como el mejor modelo de RNA para predecir las propiedades termodinámicas del R290.

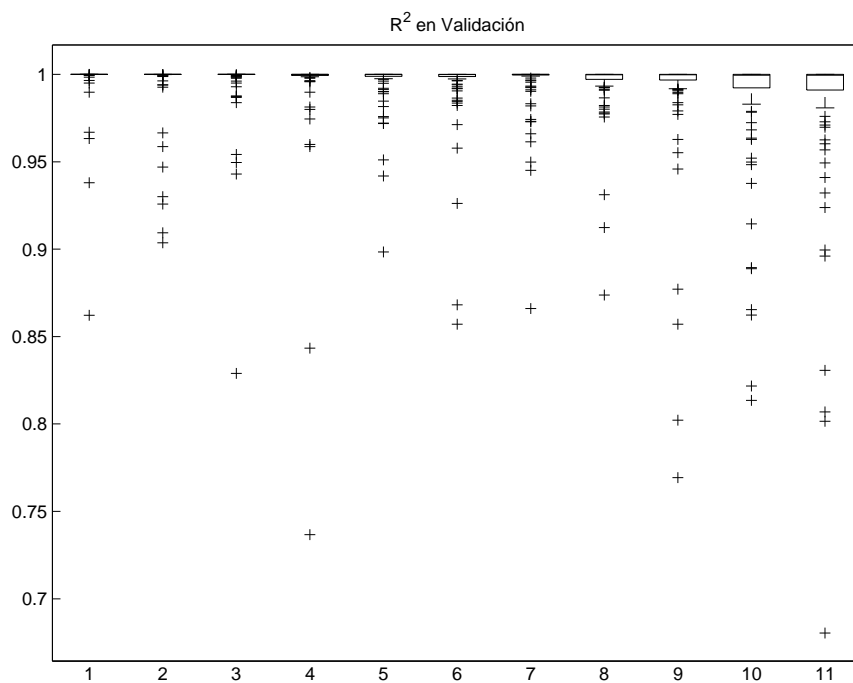


Figura 64: Boxplot de  $R^2$  en modelos del R290.

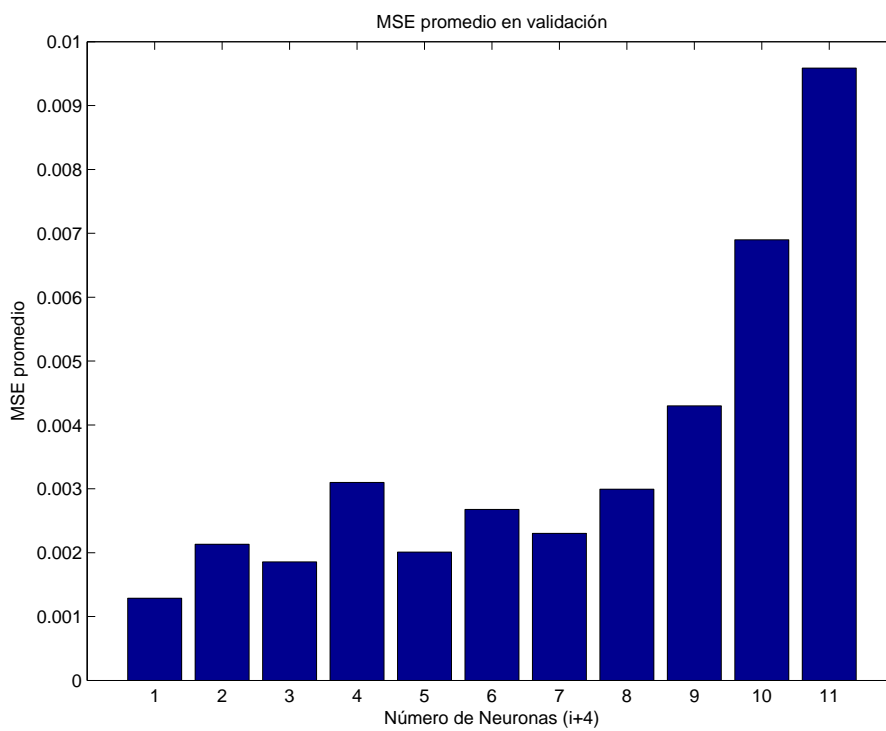


Figura 65: Variación del MSE promedio de los modelos del R290.

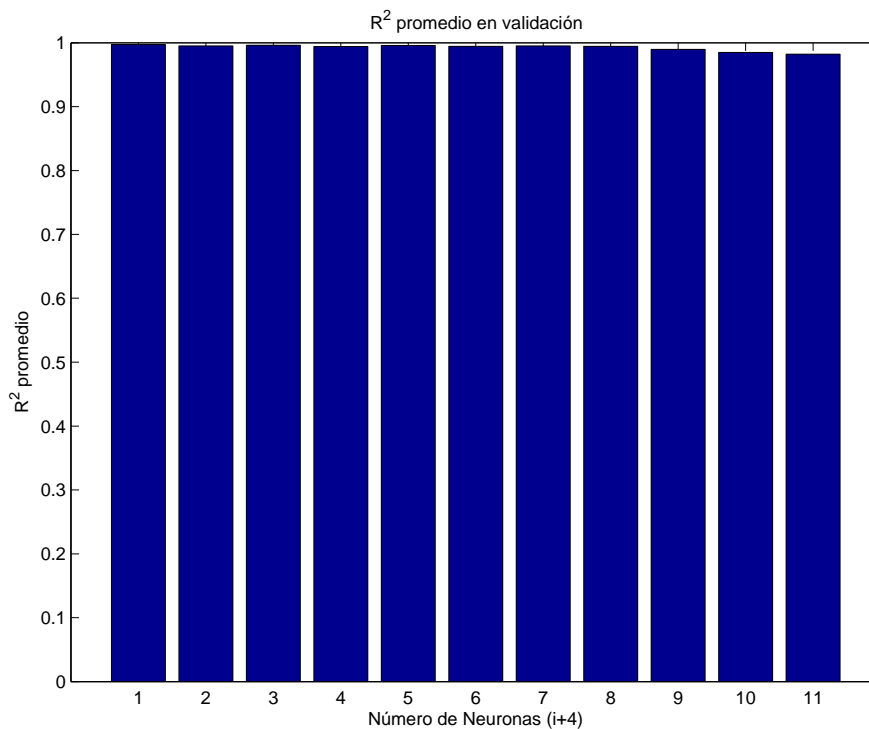


Figura 66: Variación del  $R^2$  promedio de los modelos del R290.

MSE promedio	Modelo	Modelos sin diferencia significativa
0.00129	1	2,3,4
0.00185	3	1,2,4
0.00201	5	4,6,7,8
0.00213	2	1,3,4
0.00230	7	4,5,6,9
0.00268	6	5,7,8,9
0.00299	8	5,6,9
0.00310	4	1,2,3,5,7
0.00430	9	6,7,8,10,11
0.00690	10	9,11
0.00959	11	9,10

Tabla A.4: Prueba de Wilcoxon para el R290.

Una vez elegido definitivamente el Modelo I, se construyen las gráficas que confirman que la RNA predice valores cercanos satisfactoriamente a los datos reales. En la Fig. 67 se muestra un gráfico de regresión de todos los datos reales con respecto a los obtenidos por el Modelo. Nuevamente, la cercanía a la línea recta indica que el Modelo puede predecir los datos reales con bastante exactitud. En la Fig. 68 se puede ver que durante el proceso de validación, los datos predichos por el Modelo, se acercan mucho a la línea recta. Debido a que los datos mostrados en esta última gráfica no se incluyeron en el entrenamiento, se puede concluir que el Modelo elegido predice muy bien las propiedades termodinámicas del refrigerante en cuestión. Los valores de los pesos y umbrales se encuentran en la Tabla A.10.

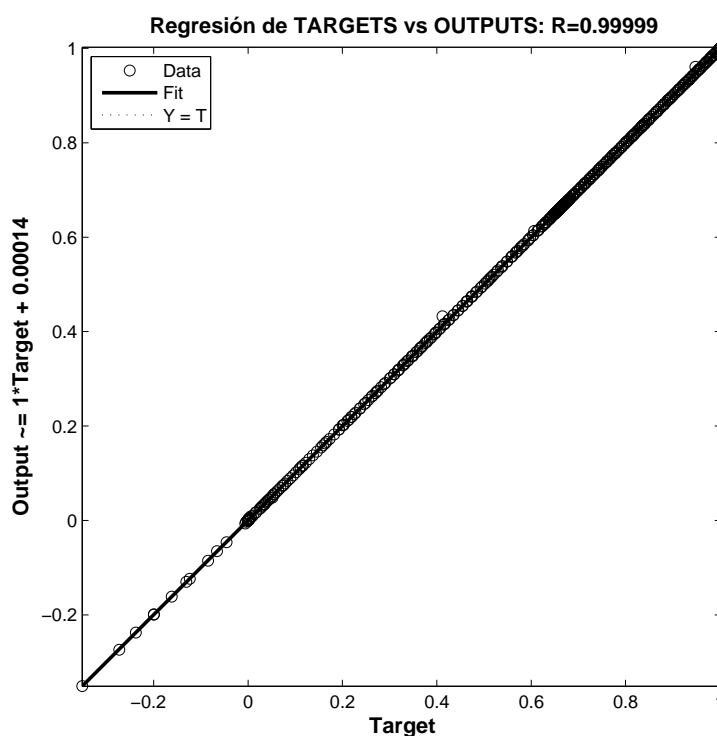


Figura 67: Regresión final general de la RNA para el R290.

Posteriormente se construyeron las gráficas que muestran como varía cada propiedad termodinámica con respecto a la variable de entrada, en este caso, la temperatura. Esto

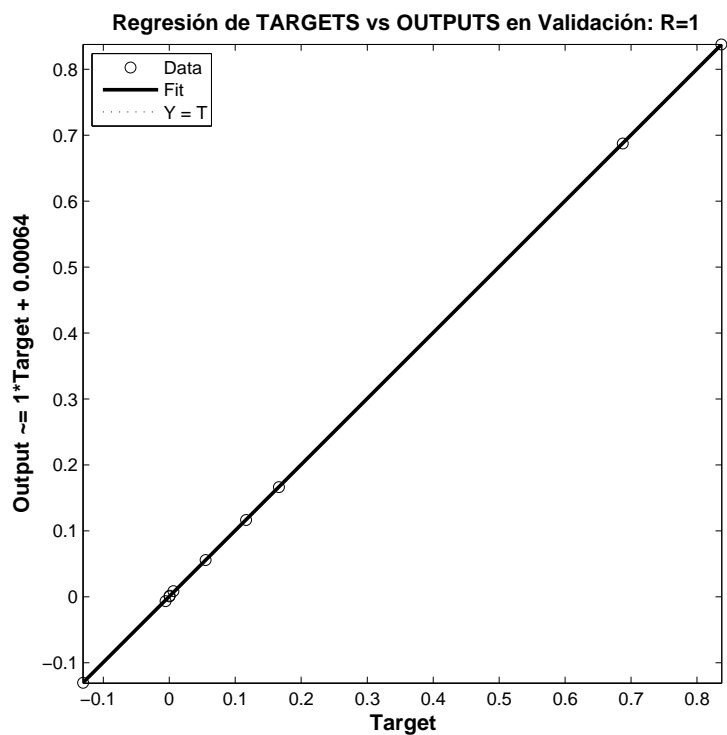


Figura 68: Regresión final en validación de la RNA para el R290.

es con el fin de confirmar que el Modelo elegido es el mejor. En las Figuras 69 a la 75 se pueden observar dichas gráficas, corroborando que las propiedades predichas por la RNA se acercan satisfactoriamente a los valores reales.

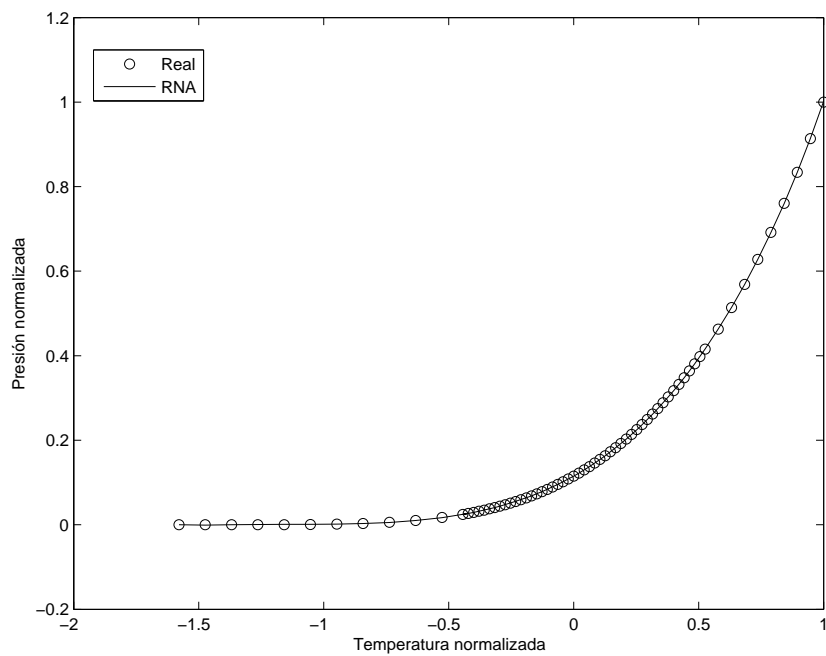
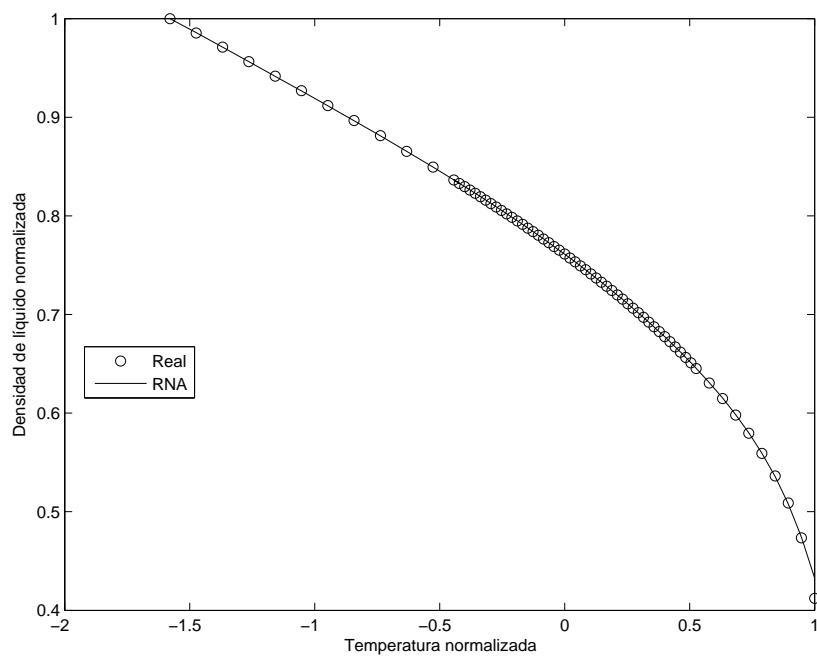
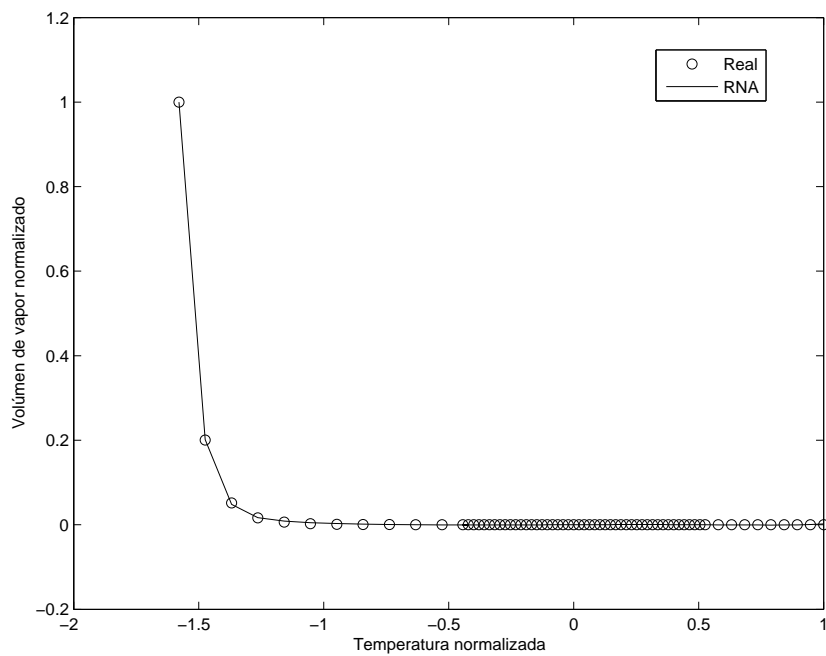
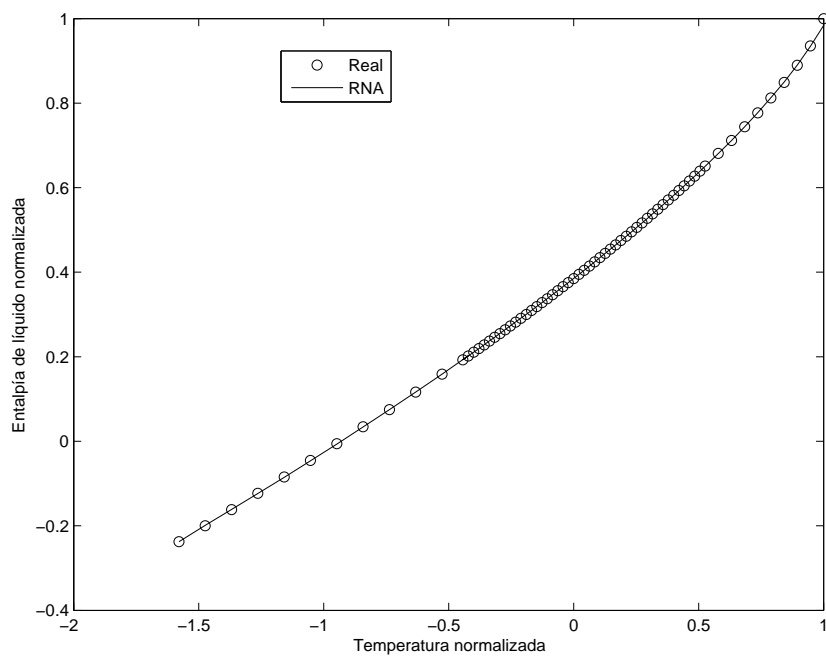
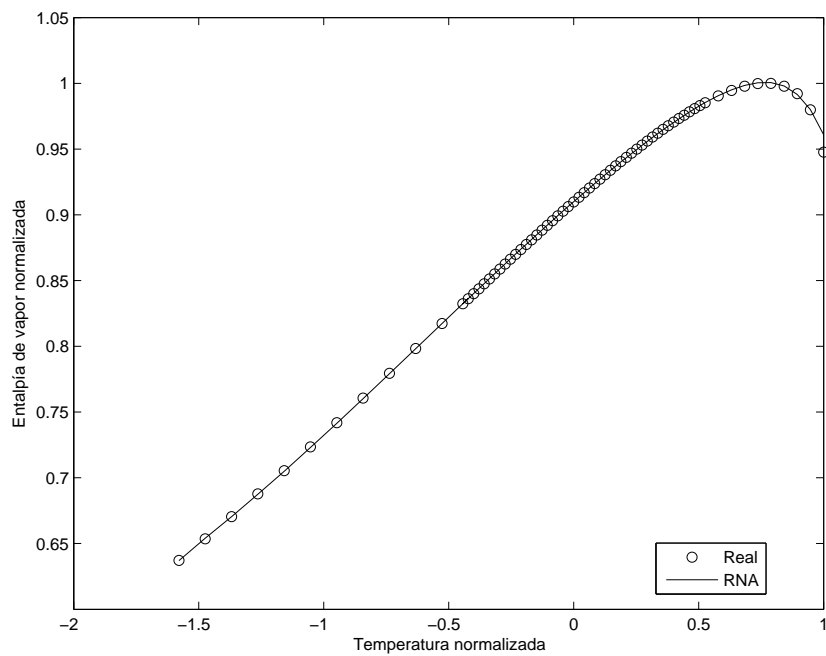
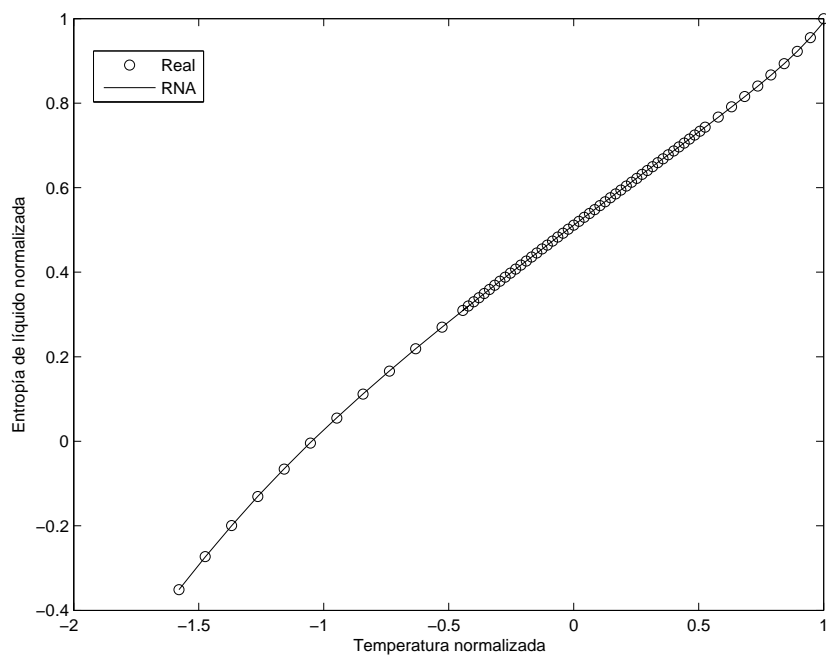


Figura 69: Presión real vs RNA para el R290.

Figura 70:  $D_l$  real vs RNA para el R290.

Figura 71:  $D_v$  real vs RNA para el R290.Figura 72:  $H_l$  real vs RNA para el R290.

Figura 73:  $H_v$  real vs RNA para el R290.Figura 74:  $S_l$  real vs RNA para el R290.

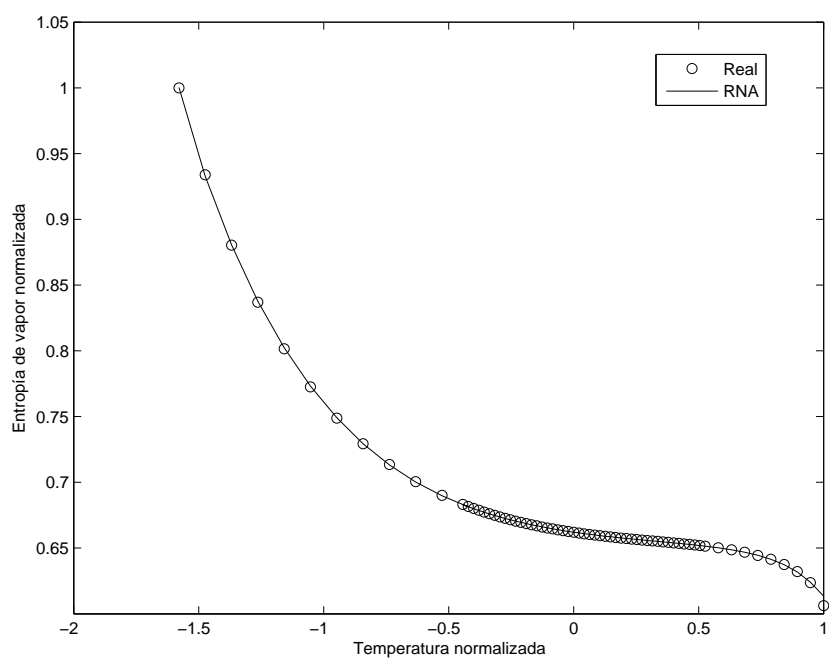


Figura 75:  $S_v$  real vs RNA para el R290.

## A.5. Análisis estadístico de la RNA del R717

Se comienza con el análisis de los MSE y  $R^2$  para el proceso de validación del R717. En las Figuras 76 y 77 podemos ver la variación del MSE y el  $R^2$  a través de los 100 ciclos de cada uno de los modelos generados. En las Figuras 78 y 79 se puede el valor del MSE y  $R^2$  promedios de cada modelo y se pueden comparar con los demás modelos. De acuerdo con estas dos últimas gráficas, se puede ver que el modelo I tiene el menor MSE y de los más altos valores de  $R^2$ . Así pues, es posible decir que el modelo I, con  $n_{best} = 5$  es el mejor candidato y no puede haber otro mejor, ya que es el más sencillo y tiene el mayor valor de  $R^2$ .

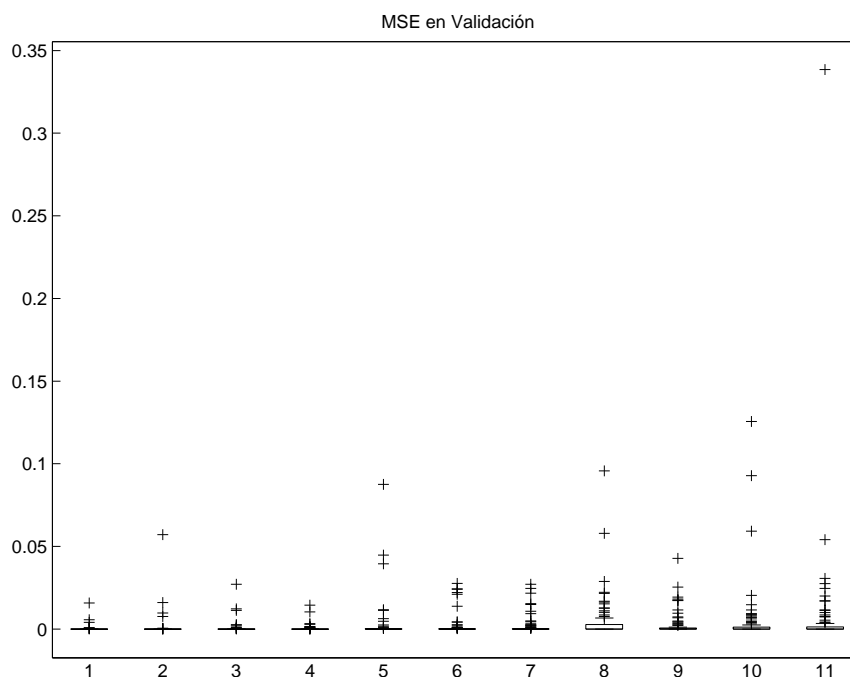


Figura 76: Boxplot de MSE en modelos del R717.

En la Tabla A.5 se muestran los resultados de la Prueba de Rangos con Signo de Wilcoxon, la cual indica que el Modelo I, con 5 neuronas, no es significativamente diferente de los Modelos III, IV y V, pero sí es el más sencillo. Siendo el modelo I el más sencillo, no era tan necesario hacer la prueba de Wilcoxon, ya que no se desea

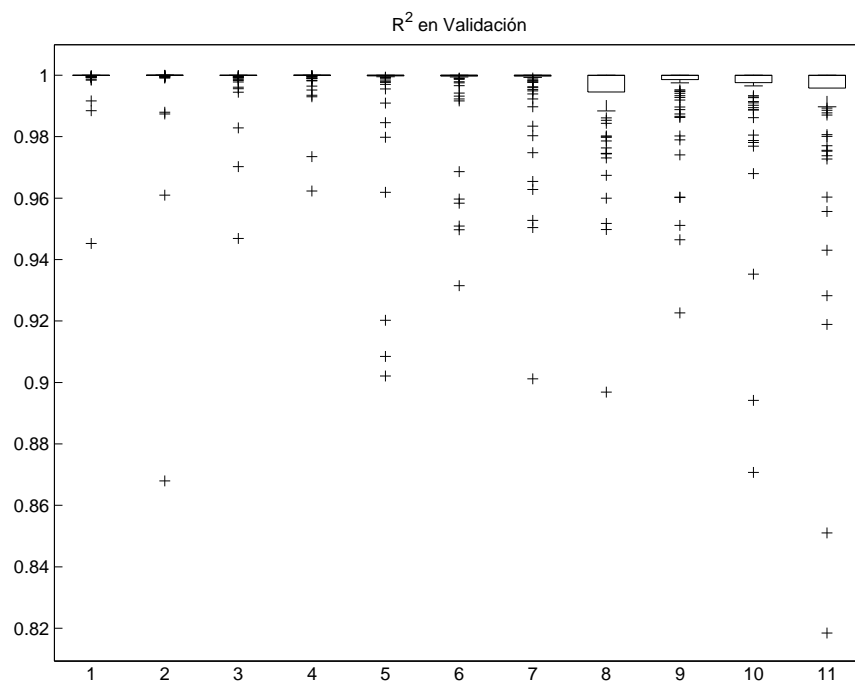


Figura 77: Boxplot de  $R^2$  en modelos del R717.

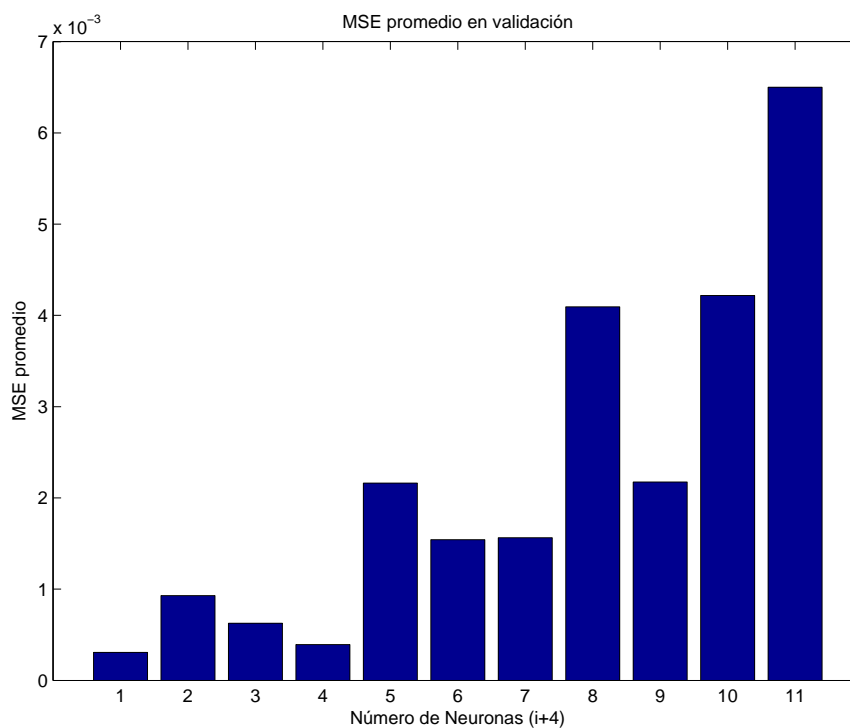


Figura 78: Variación del MSE promedio de los modelos del R717.

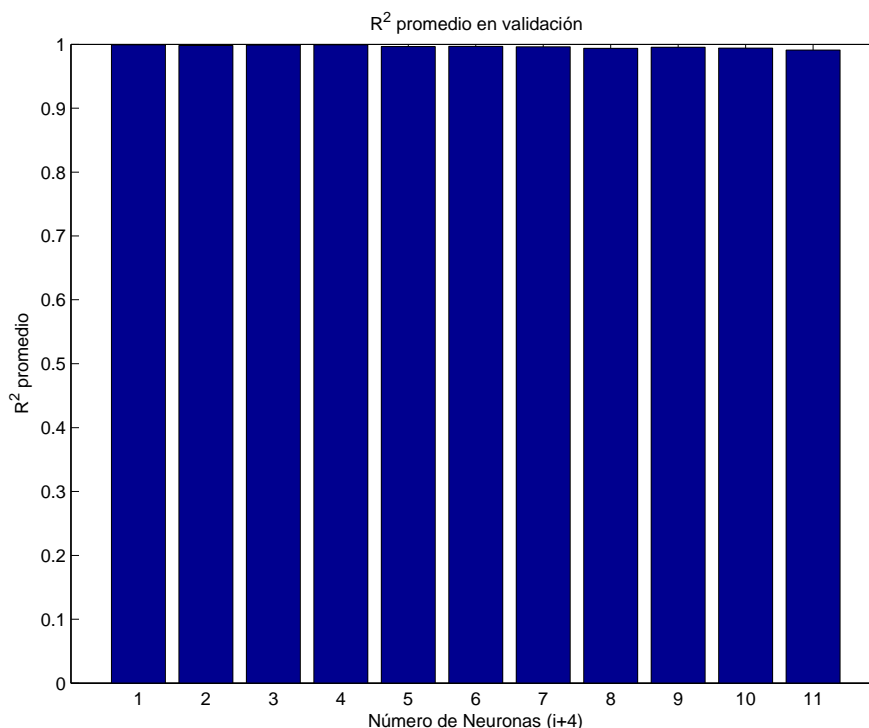


Figura 79: Variación del  $R^2$  promedio de los modelos del R717.

escoger otro modelo similar con más neuronas. Sin embargo, se aplicó de la misma forma que los refrigerantes anteriores.

Posteriormente se construyeron las gráficas que confirman que la RNA predice valores cercanos satisfactoriamente a los datos reales. En la Figura 80 se muestra un gráfico de regresión de todos los datos reales con respecto a los obtenidos por el Modelo. Nuevamente, la cercanía de los datos reales hacia la línea recta es señal de que el Modelo predice los datos reales con bastante exactitud. En la Fig. 81 se puede ver que durante el proceso de validación, los datos predichos por el Modelo, se acercan mucho a la línea recta. En el proceso de validación no se incluyeron los datos que aparecen en la figura anterior y de ella misma se puede concluir que el Modelo elegido predice satisfactoriamente las propiedades termodinámicas del R717. Los valores de los pesos y umbrales se encuentran en la Tabla A.12.

MSE promedio	Modelo	Modelos sin diferencia significativa
0.00031	1	3,4,5
0.00039	4	1,2,3,5
0.00063	3	1,2,4
0.00093	2	3,4
0.00154	6	5,7
0.00156	7	5,6,9,10
0.00216	5	1,4,6,7
0.00217	9	7,8,10,11
0.00409	8	9,10,11
0.00422	10	7,8,9,11
0.00650	11	8,9,10

Tabla A.5: Prueba de Wilcoxon para el R717.

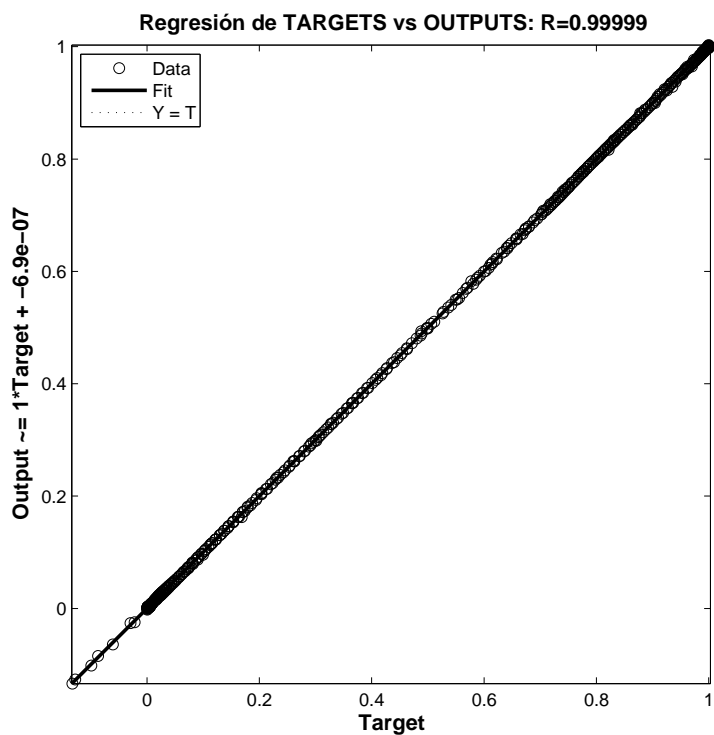


Figura 80: Regresión final general de la RNA para el R717.

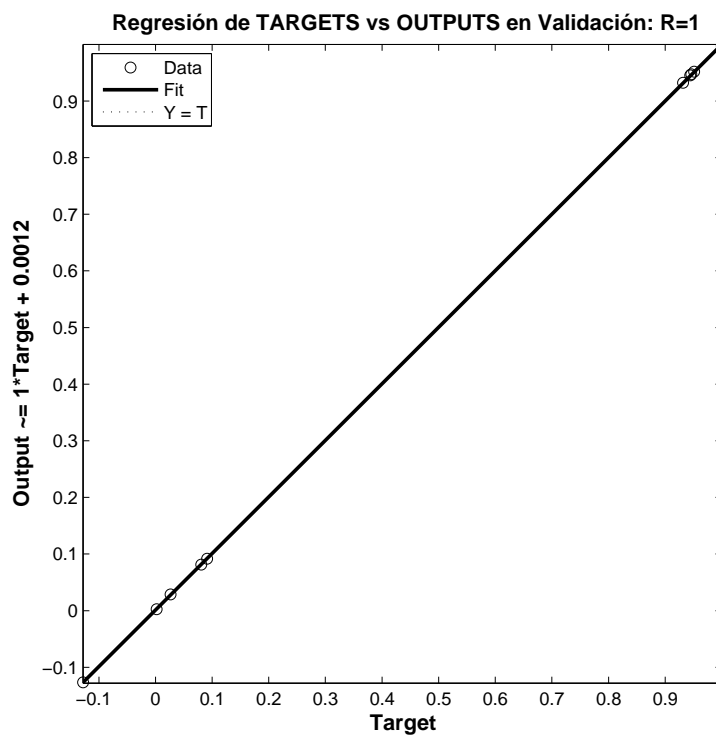


Figura 81: Regresión final en validación de la RNA para el R717.

Nuevamente se construyeron las gráficas que muestran como varía cada propiedad termodinámica con respecto a la temperatura. Esto permite confirmar que el Modelo elegido es el mejor. En las Figuras 82 a la 88 se pueden observar dichas gráficas, confirmando que el Modelo elegido puede predecir satisfactoriamente los valores reales.

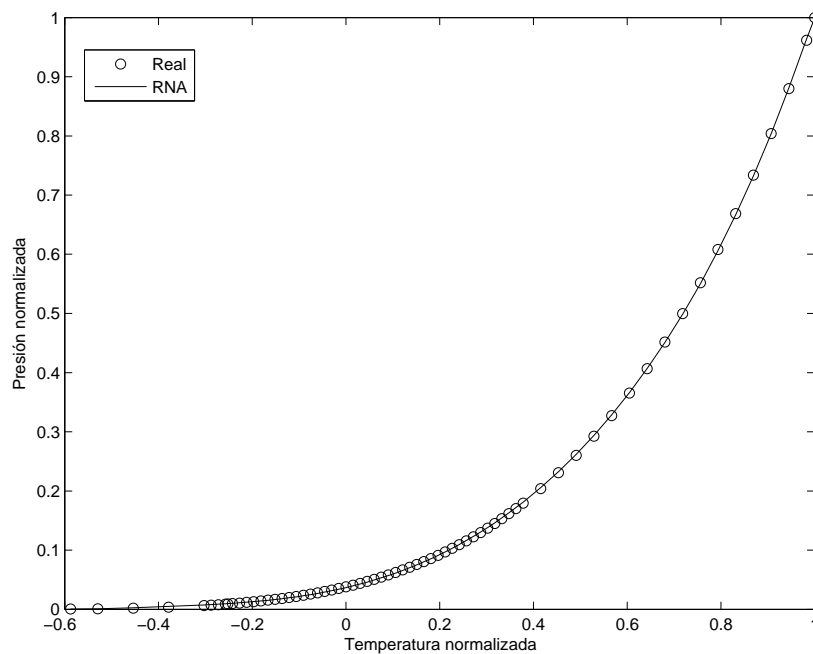
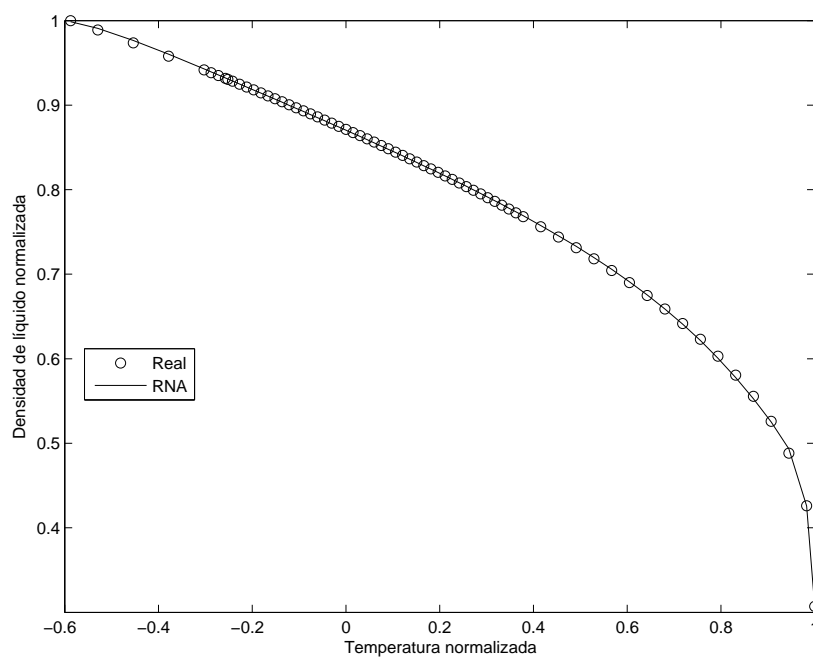
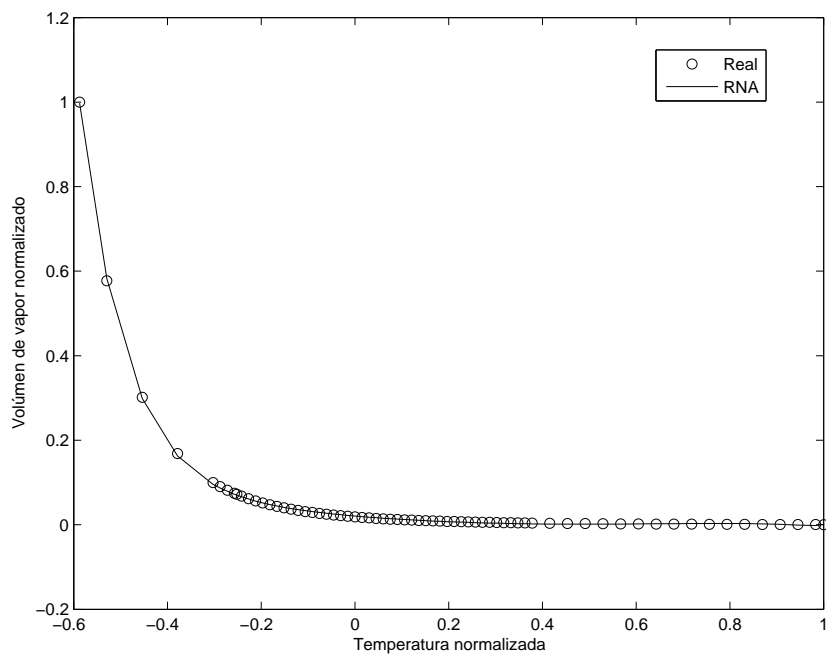
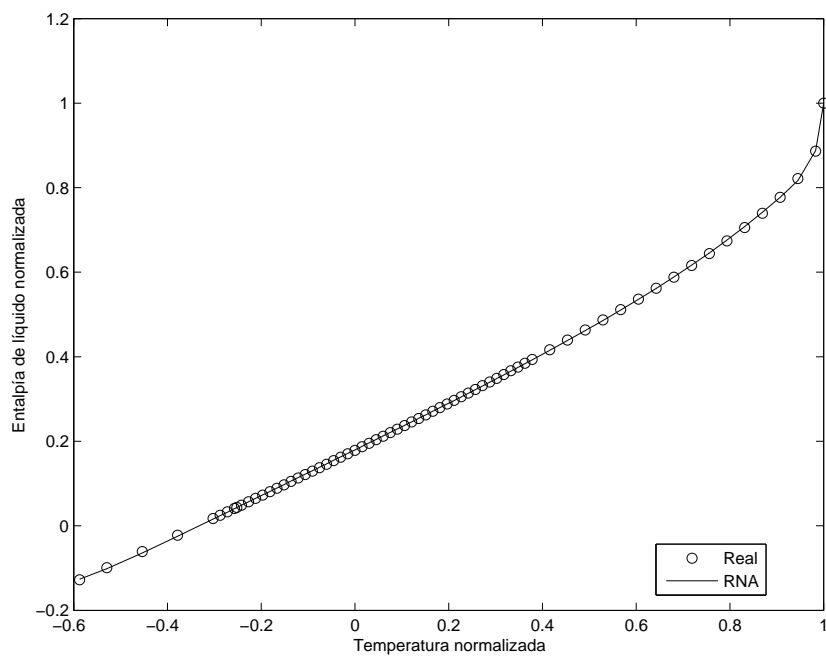
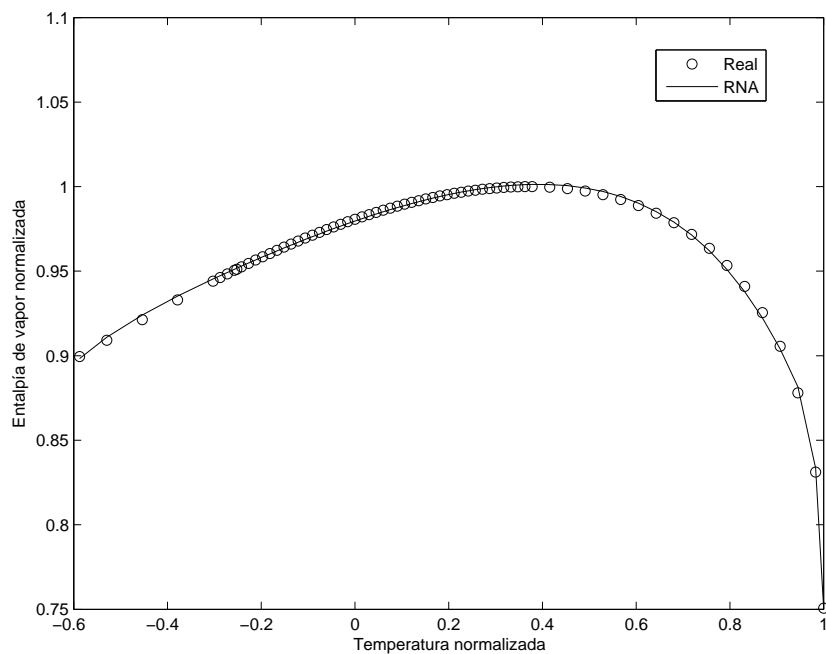
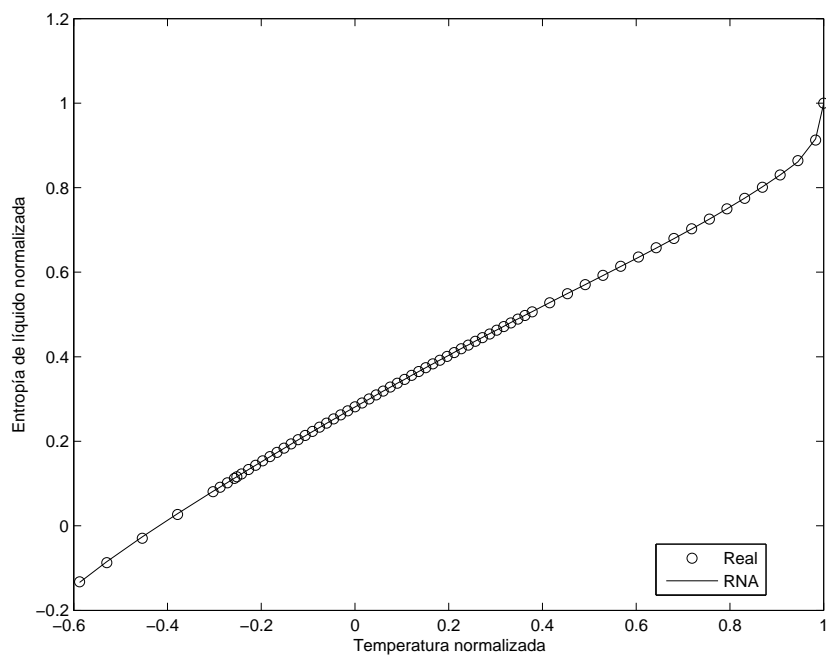


Figura 82: Presión real vs RNA para el R717.

Figura 83:  $D_l$  real vs RNA para el R717.

Figura 84:  $D_v$  real vs RNA para el R717.Figura 85:  $H_l$  real vs RNA para el R717.

Figura 86:  $H_v$  real vs RNA para el R717.Figura 87:  $S_l$  real vs RNA para el R717.

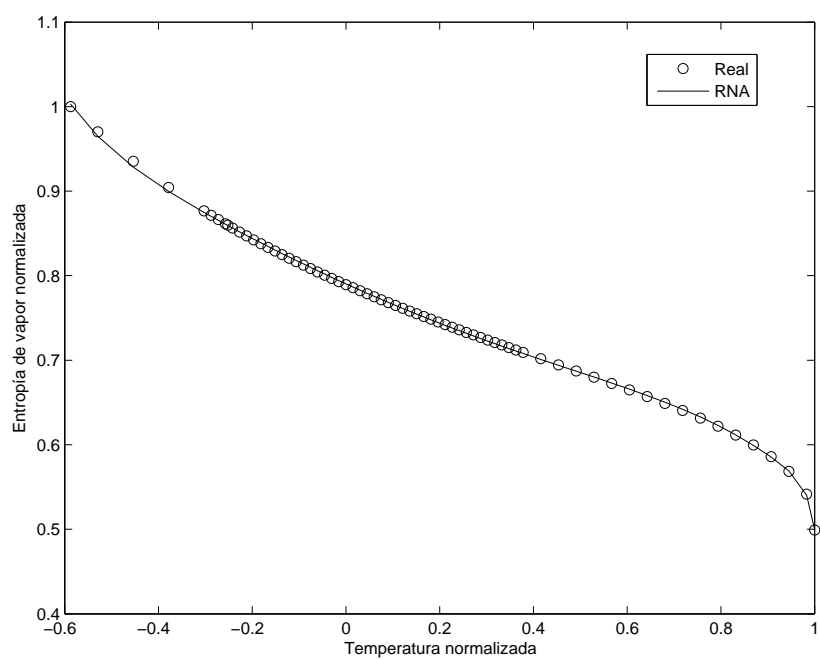


Figura 88:  $S_v$  real vs RNA para el R717.

## A.6. Análisis estadístico de la RNA del R410a

Se comienza con el análisis de los MSE y  $R^2$  para el proceso de validación del R410a. En las Figuras 89 y 90 podemos ver la variación del MSE y el  $R^2$  a través de los 100 ciclos de cada uno de los modelos generados. En las Figuras 91 y 92 se puede el valor del MSE y  $R^2$  promedios de cada modelo y se pueden comparar con los demás modelos. De acuerdo con estas dos últimas gráficas, se puede ver que el modelo XI tiene el menor MSE y de los más altos valores de  $R^2$ . Así pues, es posible decir que el modelo XI, con  $n_{best} = 15$  es el mejor candidato hasta este punto.

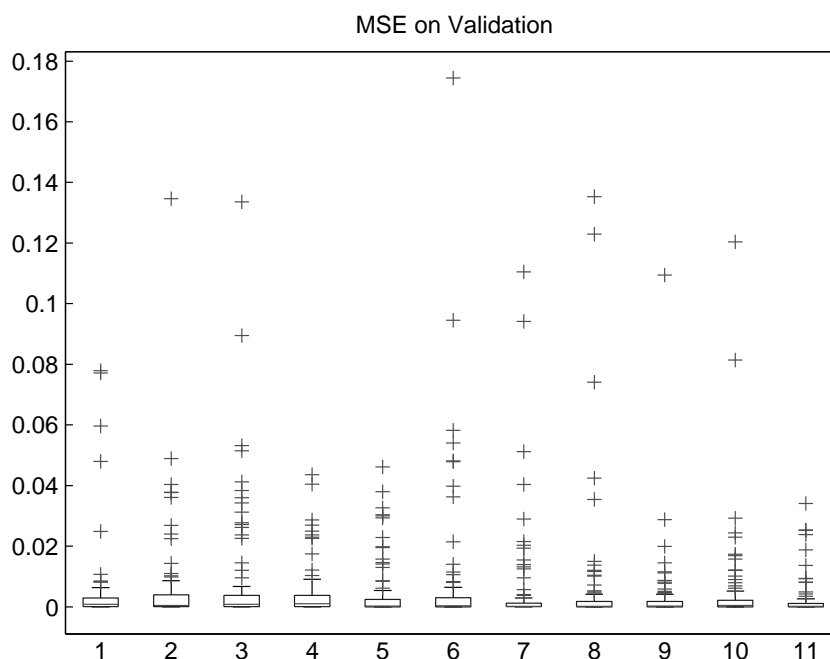


Figura 89: Boxplot de MSE en modelos del R410a.

En la Tabla A.6 se muestran los resultados de la Prueba de Rangos con Signo de Wilcoxon, la cual indica que el Modelo XI, con 15 neuronas, no es significativamente diferente de los Modelos V, VII, VIII, IX y X. Sin embargo, se puede ver que el modelo V es más sencillo, y de acuerdo con la prueba, se puede emplear como un modelo

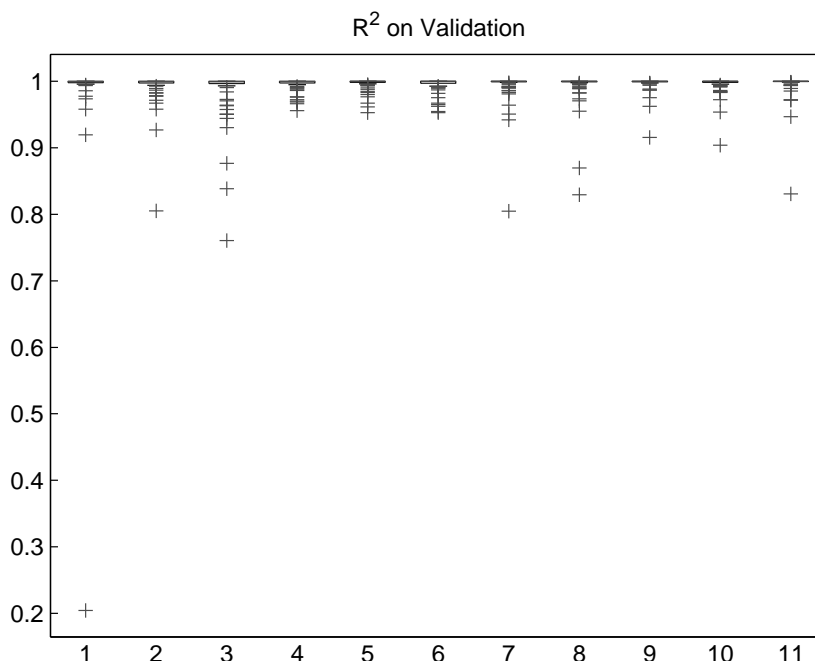


Figura 90: Boxplot de  $R^2$  en modelos del R410a.

equivalente. Por esta razón, se elige el Modelo V, con  $n_{best} = 9$  como el más adecuado para reproducir los datos reales del R410a.

Posteriormente se construyeron las gráficas que confirman que la RNA predice valores cercanos satisfactoriamente a los datos reales. En la Fig. 93 se muestra un gráfico de regresión de todos los datos reales con respecto a los obtenidos por el Modelo. Nuevamente, la cercanía de los datos reales hacia la línea recta es señal de que el Modelo predice los datos reales con bastante exactitud. Los valores de los pesos y umbrales se encuentran en la Tabla 3.5.

Nuevamente se construyeron las gráficas que muestran como varía cada propiedad termodinámica con respecto a la presión. Esto permite confirmar que el Modelo elegido es el mejor. En las Figuras 94 a la 101 se pueden observar dichas gráficas, confirmando

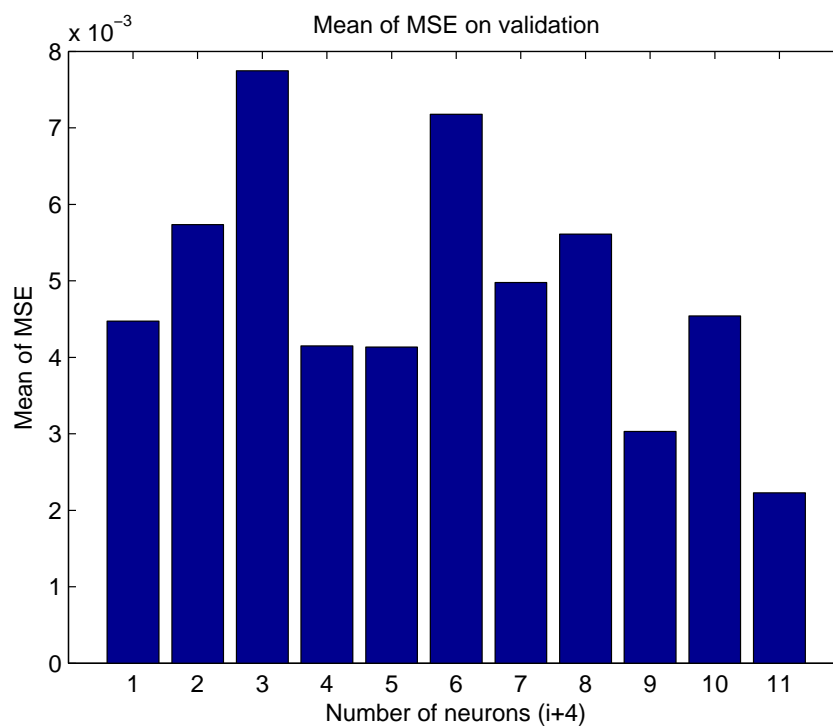


Figura 91: Variación del MSE promedio de los modelos del R410a.

MSE promedio	Modelo	Modelos sin diferencia significativa
0.00223	11	5,7,8,9,10
0.00303	9	1,4,5,6,7,8,10,11
0.00413	5	1,2,3,4,6,7,8,9,10,11
0.00415	4	1,2,3,5,6,7,9,10
0.00447	1	2,3,4,5,6,7,8,9,10
0.00454	10	1,2,3,4,5,6,7,8,9,11
0.00498	7	1,4,5,6,8,9,10,11
0.00561	8	1,2,5,6,7,9,10,11
0.00574	2	1,3,4,5,6,8,10
0.00718	6	1,2,3,4,5,7,8,9,10
0.00775	3	1,2,4,5,6,10

Tabla A.6: Prueba de Wilcoxon para el R410a.

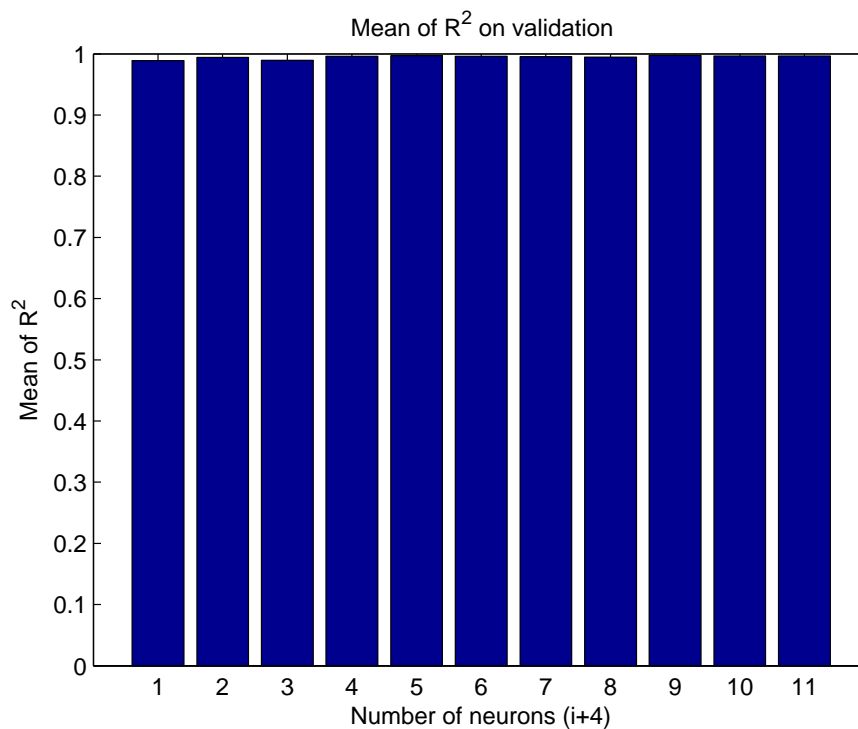


Figura 92: Variación del  $R^2$  promedio de los modelos del R410a.

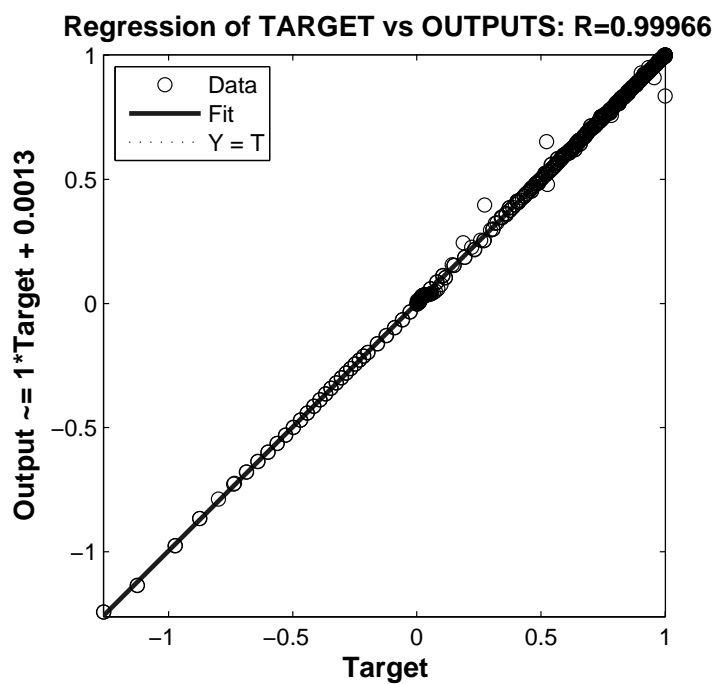
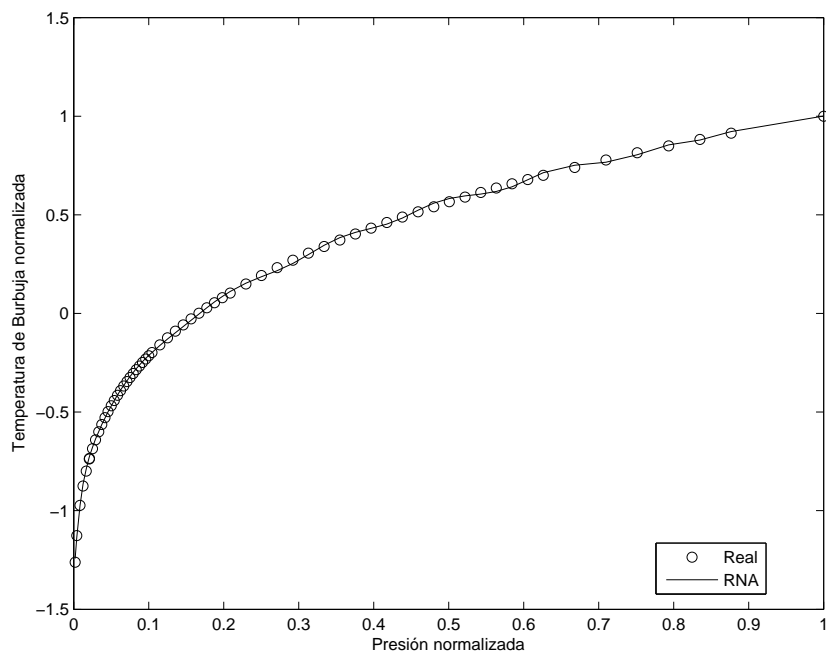
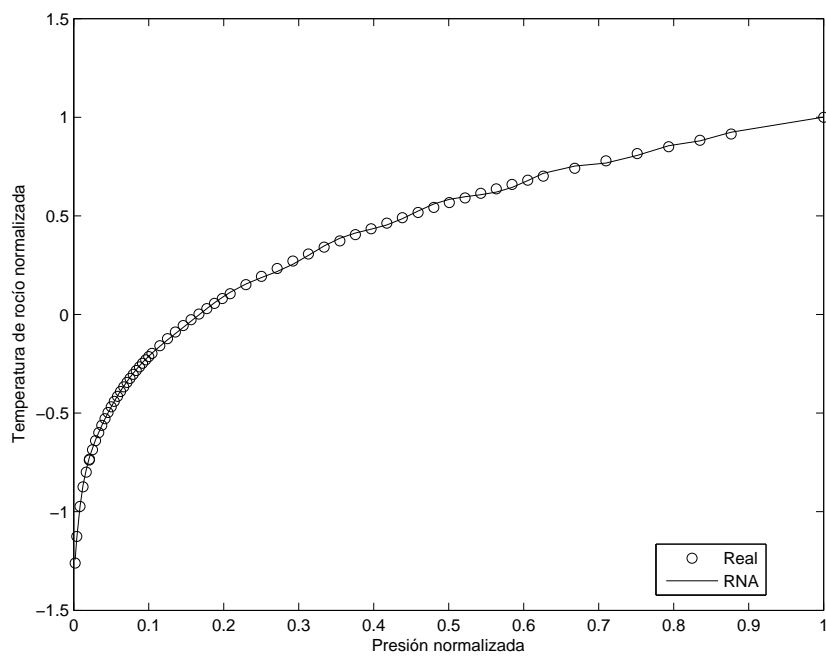


Figura 93: Regresión final general de la RNA para el R410a.

Figura 94:  $T_b$  real vs RNA para el R410a.Figura 95:  $T_r$  real vs RNA para el R410a.

que las propiedades calculadas por el Modelo elegido se acercan satisfactoriamente a los valores reales.

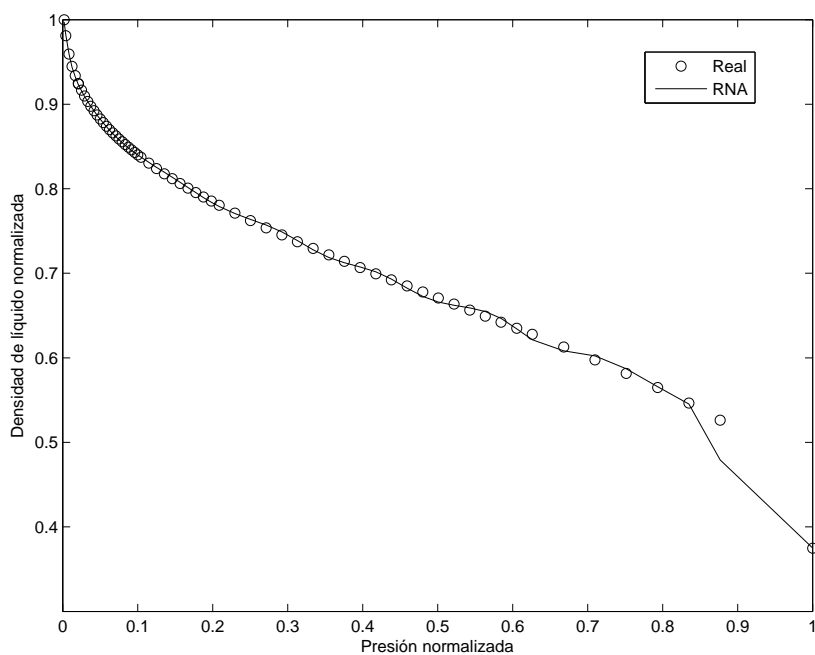
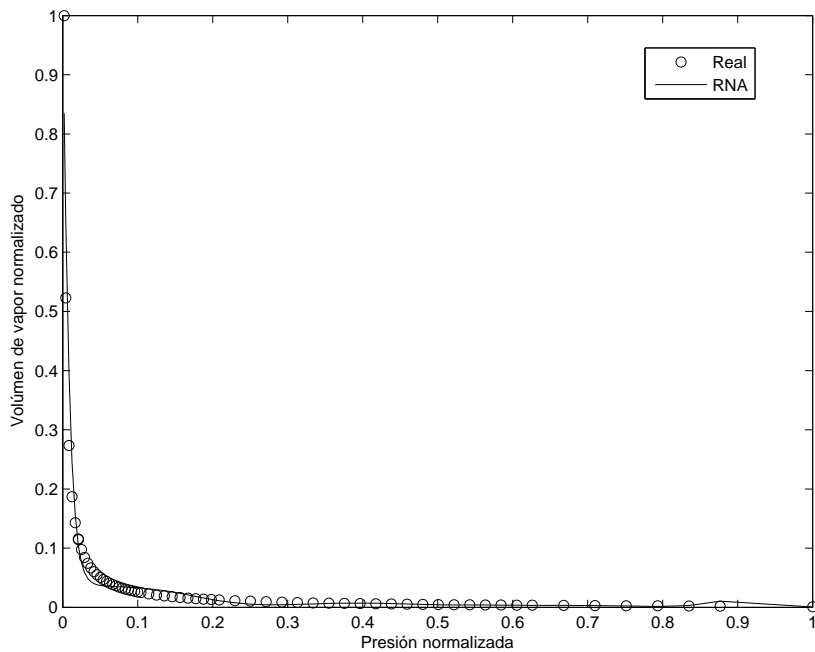
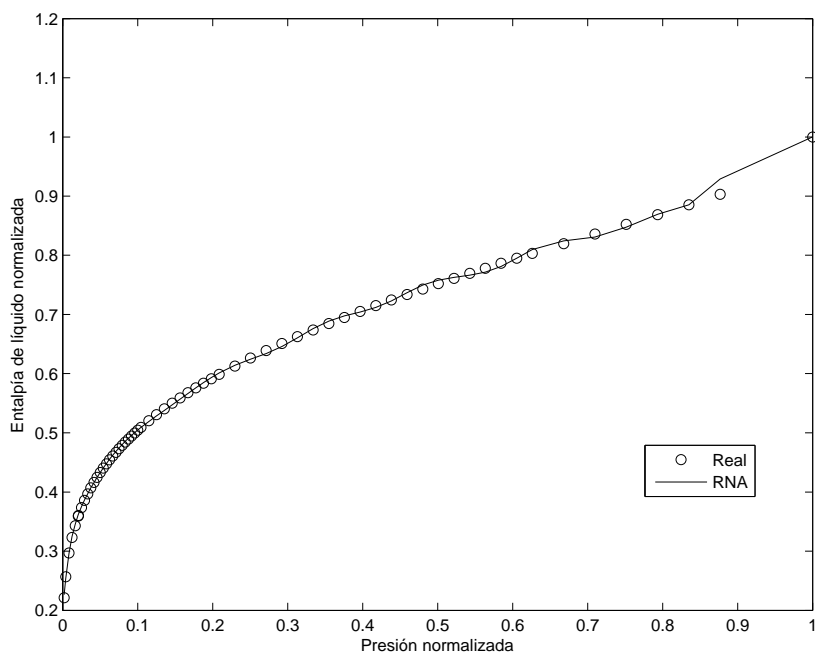
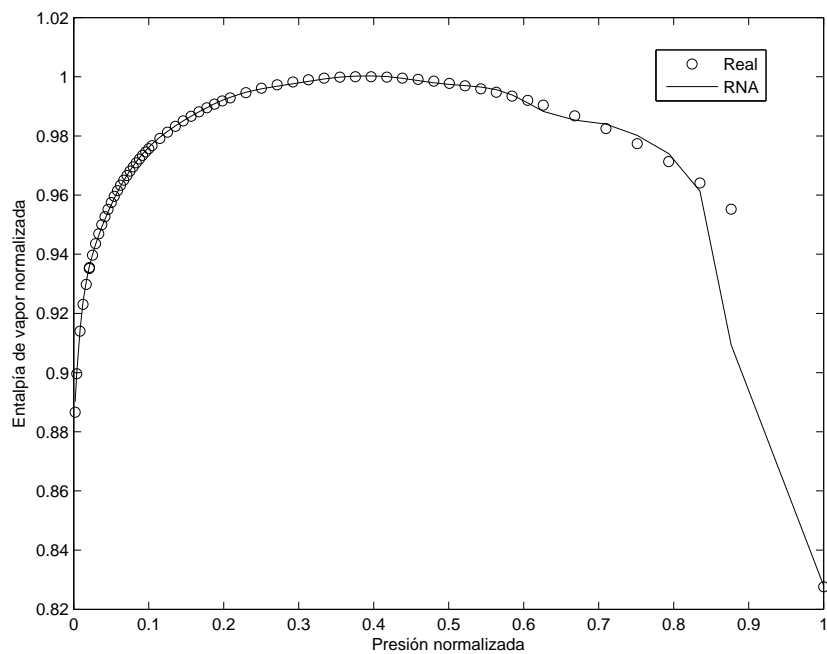
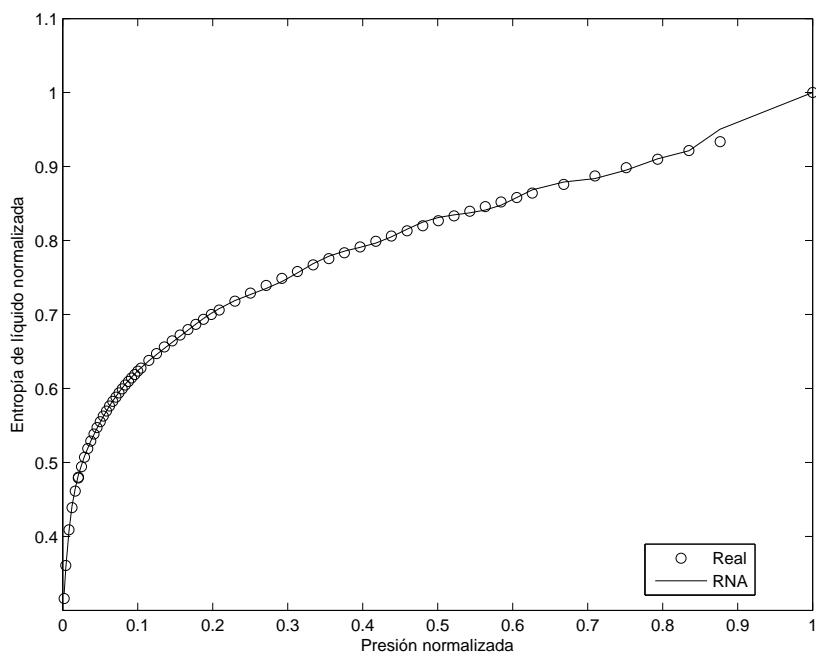


Figura 96:  $D_l$  real vs RNA para el R410a.

Figura 97:  $D_v$  real vs RNA para el R410a.Figura 98:  $H_l$  real vs RNA para el R410a.

Figura 99:  $H_v$  real vs RNA para el R410a.Figura 100:  $S_l$  real vs RNA para el R410a.

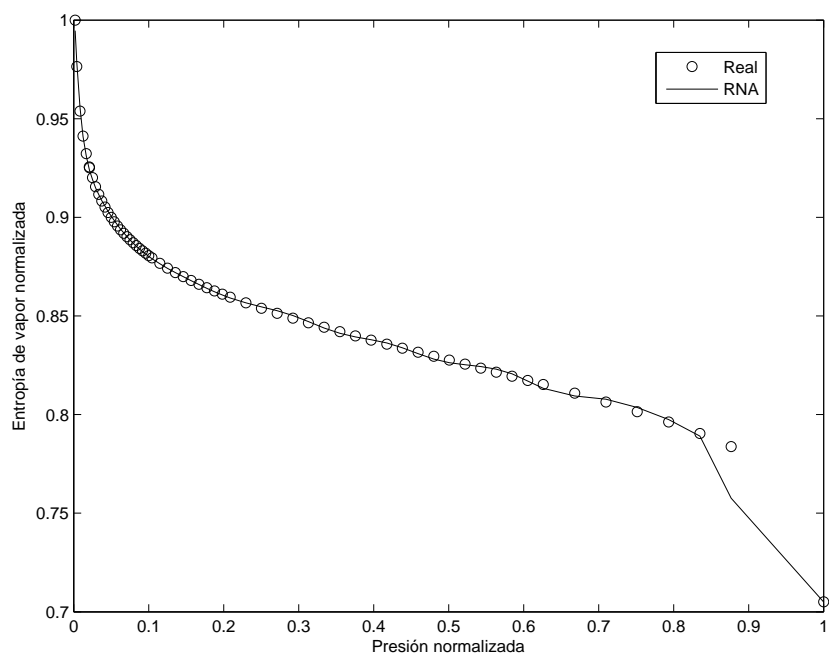


Figura 101:  $S_v$  real vs RNA para el R410a.

## A.7. Esquema del algoritmo general

En este diagrama de flujo se presenta el algoritmo general para construir y seleccionar modelos de RNA para los diferentes refrigerantes.

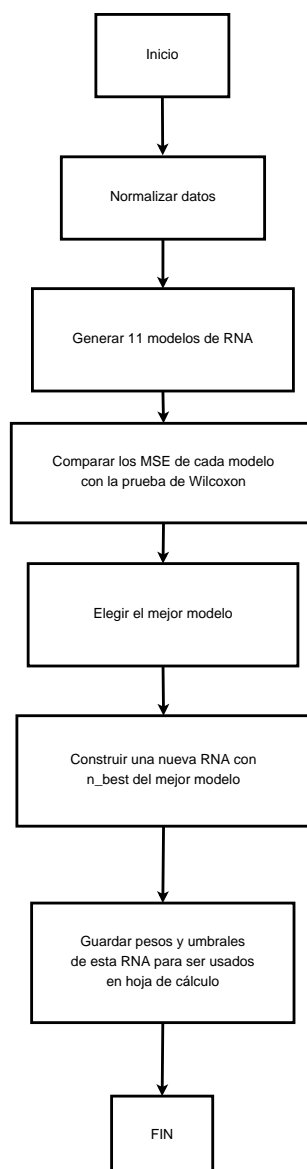


Figura 102: Diagrama general de la metodología para elegir el mejor modelo.

## A.8. Algoritmo desarrollado en Matlab

En esta sección se presenta el algoritmo desarrollado en Matlab para la construcción de los modelos de RNA de todos los refrigerantes, con excepción del R140a, ya que en este último las variables de entrada y salida son diferentes (ver Anexo A.6). La temperatura es la variable de entrada y la presión, densidad de líquido, volumen de vapor, entalpía de líquido, entalpía de vapor, entropía de líquido y entropía de vapor. Aquí se pueden ver los detalles de la programación que hace posible que se generen los modelos de red neuronal artificial, entrenarlos, obtener los errores, aplicar la prueba de Wilcoxon y seleccionar el mejor modelo.

### Secciones principales del código

- IMPORTAR LOS DATOS: SE USA EL SIGUIENTE CÓDIGO PARA ABRIR UN DIALOG BOX Y LOCALIZAR EL ARCHIVO DE EXCEL CON LOS DATOS A IMPORTAR.
- Generando el vector de entrada X y el vector de salida Y.
- VARIAR DE 5 A 15 NEURONAS: Se generan RNA con 5, 6, ... hasta 15 neuronas para poder decidir cual es la mejor RNA y cuántas neuronas producen la mejor salida.
- ESCOGER LA MEJOR RNA: El Segundo paso es escoger la RNA que tiene el menor MSE, el mayor  $R^2$  y el menor número de neuronas.
- COMPARAR MODELOS CON PRUEBA DE WILCOXON, DATOS PAREADOS.
- EXPORTAR MSE Y  $R^2$  DE CADA UNO DE LOS 100 CICLOS: En este paso se exportan los valores del MSE y  $R^2$  de cada una de las 100 corridas.
- CONSTRUIR UNA NUEVA RNA CON EL NÚMERO ÓPTIMO DE NEURONAS: Por último se corre una vez más una estructuración y entrenamiento de una RNA y guardar el MSE,  $R^2$ , y pesos y bias.
- GENERAR LAS GRÁFICAS PARA EL ANÁLISIS ESTADÍSTICO.

```
%%MENSAJE DE ENTRADA: Primero se muestra un dialog para
%preguntar si el usuario desea ejecutar
%%este programa, que es para obtener la mejor estructura
% de RNA para los datos de entrenamiento de
```

```

% cierto refrigerante.
% Construct a questdlg with three options
choice = questdlg('Programa para obtención de propiedades
  termodinámicas en el EQUILIBRIO LÍQUIDO-VAPOR
  para un refrigerante. Supone datos normalizados en la Hoja1
  de un archivo de Excel,
  con los nombres de las variables en el primer renglón.
  ¿Desea continuar?')
, ... 'Obteniendo la mejor estructura de RNA.', ...
'Salir de la ejecución','Continuar con el programa',...
'Continuar con el programa');
% Handle response
switch choice
  case 'Salir de la ejecución'
    error('Programa detenido por el usuario.')
  case 'Continuar con el programa'
    disp('Encontrando la mejor RNA del Refrigerante
  en el EQUILIBRIO LÍQUIDO-VAPOR...')
end

```

```

clc; %instrucción para limpiar la ventana de comandos.
clear all; %elimina todas las variables de la memoria.

```

```

%Para leer la hoja de excel con los datos y
%generar las variables:---

```

### **IMPORTANDO LOS DATOS: SE USA EL SIGUIENTE CÓDIGO PARA ABRIR UN DIALOG BOX Y LOCALIZAR EL ARCHIVO DE EXCEL CON LOS DATOS A IMPORTAR:**

```

[filename, pathname] = uigetfile( ...
{'*.xlsx', 'Archivos de excel (*.xlsx)'; ...
'*.xls', 'Archivos de excel (*.xls)'}), ...
'Selecciona el archivo:');
% If "Cancel" is selected then return
if isequal([filename,pathname],[0,0])
%return
error('No seleccionaste un archivo válido,
la ejecución del código

```

```

    se detendrá.', 'Error.')
```

error('No se seleccionó un archivo válido,  
la ejecución del código se  
interrumpió.')

```

% Otherwise construct the fullfilename and Check and load
% the file
else
File = fullfile(pathname,filename);%Se almacena la ruta del archivo
% en esta variable.
%Ahora se pregunta la hoja desde la cual se desea leer
%así como el rango de columnas en dicha hoja, con un InputBox.
prompt={'Hoja en la que estan los datos de entrenamiento:',...
        'Rango de celdas en las que se encuentran los datos:'};
name='Favor de elegir la hoja y el rango de celdas donde
     están los datos.';
numlines=1;
defaultanswer={'Sheet1','A2:H68'};
options.Resize='on';
options.WindowStyle='normal';
answer=inputdlg(prompt,name,numlines,defaultanswer,options);
Hoja=answer{1};
Celdas=answer{2};

%Falta validar que no esté vacío ningún campo.
    if isempty(Hoja)
        errordlg('No se definió un nombre válido de la hoja,
el código se detendrá.', 'Error Fatal')
        error('Nombre de la hoja inválido o vacío, el código
se detuvo.')
```

end

```

        if isempty(Celdas)
            errordlg('No se definió un rango de celdas válido
dentro de la hoja,
el código se detendrá.', 'Error Fatal')
```

error('Rango de celdas inválido o vacío, el código  
se detuvo.')

```

        end
    end

[~, ~, raw] = xlsread(File,Hoja,Celdas);
% Create output variable
data = reshape([raw{:}],size(raw));
```

```

% Allocate imported array to column variable names
%variables de entrada:
T = data(:,1);%Temperatura
%-----
%Variables de salida:

P = data(:,2);
Dl = data(:,3);
Vv = data(:,4);
Hl = data(:,5);
Hv = data(:,6);
Sl = data(:,7);
Sv = data(:,8);

%-----
% limpiar de variables temporales
clearvars data raw;
%-----

```

### Generando el vector de entrada X y el vector de salida Y.

```

X=T';
Y=[P Dl Vv Hl Hv Sl Sv]';

```

### VARIACIÓN DE 5 A 15 NEURONAS: Se generan RNA con 5, 6, ... hasta 15 neuronas

```

%para poder decidir cual es la mejor RNA y cuántas neuronas
%producen la mejor salida.

inputs = X;
targets = Y;
disp('Buscando la mejor RNA en el EQUILIBRIO LÍQUIDO-VAPOR...')

%Prealojando el valor de las variables MSE y Rcuadrada:
MSE_train = zeros(100,11); %Matriz de 100X10,
% contiene los MSE del train.
MSE_tst = zeros(100,11); %Matriz de 100X10,
% contiene los MSE del test.
MSE_val = zeros(100,11); %Matriz de 100X10,
% contiene los MSE de validación.

```

```

R_train = zeros(100,11);%Matriz de 100X10,
% contiene los R^2 del train.
R_tst = zeros(100,11);%Matriz de 100X10,
% contiene los R^2 del test.
R_val = zeros(100,11);%Matriz de 100X10,
% contiene los R^2 de validación.

promMSE_n=zeros(11,1); %almacena el promedio de los
%MSE de validación para el no. de neuronas "i".
promR_n=zeros(11,1);%almacena el promedio de los R^2
%de validación para el no. de neuronas "i".
medianaMSE_n=zeros(11,1);
medianaR_n=zeros(11,1);
hWilcoxon_mse=zeros(11,1);
hWilcoxon_r=zeros(11,1);

for i = 1:11 %variando desde 5 hasta 15 neuronas.
n=i+4;

for j=1:100 %Haciendo 100 ciclos con el mismo número
% n de neuronas.

[a,b,c,d,e,f]=CreaFitnet_DOS(n,inputs,targets);

MSE_train(j,i)=a;
MSE_tst(j,i)=b;
MSE_val(j,i)=c;
R_train(j,i)=d;
R_tst(j,i)=e;
R_val(j,i)=f;

end

promMSE_n(i)=mean(MSE_val(:,i)); %almacena el promedio
%de los MSE de validación para el no. de neuronas "i".
promR_n(i)=mean(R_val(:,i)); %almacena el promedio de
%los R^2 de validación para el no. de neuronas "i".
medianaMSE_n(i)=median(MSE_val(:,i));
medianaR_n(i)=median(R_val(:,i));

%A Continuación se realiza una prueba de rango con signo de
%WILCOXON para comprobar que en las 100 ejecuciones para la
%RNA con "i" neuronas en la capa oculta, hay un promedio de

```

```
%MSE y de  $R^2$ , y que los promedios de estas 100 ejecuciones
%no se alejan del promedio o mediana general, calculada a
%partir de las 100 ejecuciones.
```

```
[~,hW]=signrank(MSE_val(:,i),medianaMSE_n(i));
hWilcoxon_mse(i)=hW;
```

```
[~,hW]=signrank(MSE_val(:,i),medianaR_n(i));
hWilcoxon_r(i)=hW;
```

```
end
```

**ESCOGIENDO LA MEJOR RNA: El Segundo paso es escoger la RNA que tiene el menor MSE, el mayor R y el menor número de neuronas**

```
figure, boxplot(MSE_train);%despliega un boxplot para
%el MSE del Training
title('MSE en Entrenamiento');
%saveas(gcf,'1_MSEtrain.fig')
print -r600 -deps2 1_MSEtrain
```

```
figure, boxplot(MSE_tst);
title('MSE en Prueba');
%saveas(gcf,'2_MSEtest.fig')
print -r600 -deps2 2_MSEtest
```

```
figure, boxplot(MSE_val);
title('MSE en Validación');
%saveas(gcf,'3_MSEvalid.fig')
print -r600 -deps2 3_MSEvalid
```

```
figure, boxplot(R_train);
title('R2 en Entrenamiento');
%saveas(gcf,'4_Rtrain.fig')
print -r600 -deps2 4_Rtrain
```

```
figure, boxplot(R_tst);
title('R2 en Prueba');
%saveas(gcf,'5_Rtest.fig')
print -r600 -deps2 5_Rtest
```

```

f1=figure;
boxplot(R_val);
title('R^2 en Validación')
%saveas(f1,'6_Rvalid.fig')
print -r600 -deps2 6_Rvalid

waitfor(f1);

%Se mostrarán dos gráficas de barras donde se pueda apreciar
%la variación del promedio de MSE y de R^2 de validación,
%de acuerdo con el número de neuronas.

%f=figure;
%subplot (1,2,1)
bar(mean(MSE_val))
title('MSE promedio en validación')
xlabel('Número de Neuronas (i+4)')
ylabel('MSE promedio')
saveas(gcf,'7_MeanMSE_valid.fig')
print -r600 -deps2 7_MeanMSE_valid

%subplot (1,2,2)
f=figure;
bar(mean(R_val))
title('R^2 promedio en validación')
xlabel('Número de Neuronas (i+4)')
ylabel('R^2 promedio')
saveas(f,'8_MeanR_valid.fig')
print -r600 -deps2 8_MeanR_valid

waitfor(f);

%Elemento_i muestra el número de neuronas de la capa oculta,
%en orden acendente de valor de MSE, para saber el num de
%neuronas de la RNA que tuvo el menor MSE.
[menorMSE, Elemento_i]=sort(mean(MSE_val));
Elemento_i=Elemento_i+4;
Nneurona1=Elemento_i';

%Elementoj muestra el número de neuronas de la capa oculta,
%en orden ascendente de R^2, para saber el num. de neuronas

```

```

%de la RNA que tuvo el mayor R^2.
[mayorRcuadrada, Elementoj]=sort(mean(R_val),'descend');
Elementoj=Elementoj+4;
Nneurona2=Elementoj';

%-----
%Para desplegar los valores de MSE y R^2 para cada una de las RNA de
%acuerdo al número de neuronas, se mostrará una pantalla especial:
f = figure;
set(f, 'MenuBar', 'none');
h = uicontrol('Position',[50 120 200 40],'String','Continuar',...
             'Callback','uiresume(gcbf)');

h1=uicontrol('Style','text','String','valores de MSE:',...
            'Position',[50,340,150,150]);

h2=uicontrol('Style','ListBox','String',menorMSE,...
            'Position',[50,270,150,200]);

h3=uicontrol('Style','text','String','valores de R^2:',...
            'Position',[250,340,150,150]);

h4=uicontrol('Style','ListBox','String',mayorRcuadrada,...
            'Position',[250,270,150,200]);

h5=uicontrol('Style','text','String','Aqui se muestran los
valores promedio
de MSE y R^2 de validación, al aumentar el número de neuronas,
desde 5 hasta 15.',...
            'Position',[50 200 350 50]);

uiwait(gcf); %Instrucción para establecer que el codigo
% se detiene hasta que el formulario se cierra.
close(f);
%-----

```

## COMPARACIÓN DE MODELOS CON PRUEBA DE WILCOXON, DATOS PAREADOS.

```

%%En esta sección se genera la tabla de comparación de diferentes
%%modelos, para ver cual modelo tiene una media significativamente
%%diferente con respecto a todos los demás. Para ello se emplea
%%la prueba de rangos con signos de Wilcoxon.

```

```

%%En este algoritmo se compara el modelo M=1,2,3...11
%%con los demás modelos: el i con el 2, el 1 con el 3,
%%el 1 con el 4... Cada "modelo" corresponde a una RNA con un
%% número de neuronas, el Modelo1 tiene 5 neuronas en la
%%capa oculta, el Modelo 2 tiene 6 neuronas,... el
%%Modelo11 tiene 15 neuronas.
%%Así pues, el Modelo i-ésimo, tiene i+4 neuronas.

%Prealojando las variables:
comparaWilcoxon = eye(11);
comparaModelo = cell(11,1);

for i = 1:10
    for j = (i+1):11
        %if j~=i
            [~,hh]=signrank(MSE_val(:,i),MSE_val(:,j));
            comparaWilcoxon(i,j)=hh;
        %end
    end
end

kk=triu(comparaWilcoxon,0); %almacen la parte superior de la matriz
comparaWilcoxon = comparaWilcoxon + kk';%tiene en cuenta que el
%elemento (i,j) es
%igual que el elemento (j,i), es decir la comparación del modelo 3
%con el 5 es la misma que la comparación del modelo 5 con el 3.
%Así se contruye la matriz de comparaciones completa.

%Ahora se contruye una variable de tipo cadena que almacena
%la secuencia de modelos cuyas medias no son
%significativamente diferentes: "comparaModelo".

Modelo = Nneurona1-4; %Es la secuencia de los modelos, ordenados
% de menor a mayor promedioMSE_val, según
% el número de neuronas.

for i = 1:11 %comparando cada uno de los 11 modelos con los
% 10 diferentes.
    for j = 1:11
        monitor = Modelo(i);
        if comparaWilcoxon(Modelo(i),j)== 0
            j2=num2str(j);

```

```

                comparaModelo{i}=strcat(comparaModelo{i},j2,',');
            end
        end
    end
end

f = figure;
set(f, 'MenuBar', 'none');
hx = uicontrol('Position',[50 120 200 40],'String','Continuar',...
              'Callback','uiresume(gcbf)');

hx1=uicontrol('Style','text','String','Num. de neuronas:',...
              'Position',[30,340,150,150]);

hx2=uicontrol('Style','ListBox','String',Nneurona1,...
              'Position',[30,270,150,200]);

hx3=uicontrol('Style','text','String','MODELO:',...
              'Position',[190,340,100,150]);

hx4=uicontrol('Style','ListBox','String',Modelo,...
              'Position',[190,270,100,200]);

hx5=uicontrol('Style','text','String','Modelos con medias iguales:',...
              'Position',[350,340,230,150]);

hx6=uicontrol('Style','ListBox','String',comparaModelo,...
              'Position',[350,270,230,200]);

hx7=uicontrol('Style','text','String','Aquí se muestran
la comparación de modelos.', 'Position',[50 200 350 50]);

uiwait(gcf); %Instrucción para establecer que el código se detiene
% hasta que el formulario se cierra.
close(f);

%-----

%Ahora se muestra un InputBox mostrando el orden de número
% de neuronas con menor MSE y mayor R^2, para que el usuario
%pueda elegir cuántas neuronas quiere en la capa oculta
% de la RNA.
prompt={'Orden de neuronas de menor a mayor MSE:',...
        'Orden de neuronas de mayor a menor promedio

```

```

de R^2 en validación',...
'Elija el número de neuronas'};
name='Orden de número de neuronas según promedios de MSE
y R^2 en validación';
numlines=2;
defaultanswer={num2str(Nneurona1'),num2str(Nneurona2'), 'n'};
options.Resize='on';
options.WindowStyle='normal';
options.Interpreter='tex';
answer=inputdlg(prompt,name,numlines,defaultanswer,options);

if isempty(answer)
    error('Proceso interrumpido por el usuario')
end

n=str2double(answer{3});

```

## EXPORTANDO MSE Y R<sup>2</sup> DE CADA UNO DE LOS 100 CICLOS: En este paso

```

%se exportan los valores del MSE y R^2 de cada una
%de las 100 corridas:

%Se muestra un dialogbox para guardar:
[filename, pathname] = uiputfile( ...
{'*.xlsx'}, ...
'Guardar como...');
% If 'Cancel' was selected then exit subroutine
if isequal([filename,pathname],[0,0])
    error('Proceso interrumpido por el usuario')
else
% Construct the full path and save
File = fullfile(pathname,filename);

xlswrite(File,MSE_train,1)%exporta a excel los MSE de cada ciclo
xlswrite(File,MSE_tst,2)%exporta a excel los MSE de cada ciclo
xlswrite(File,MSE_val,3)%exporta a excel los MSE de cada ciclo
xlswrite(File,R_train,4)%exporta a excel la R^2 de cada ciclo
xlswrite(File,R_tst,5)%exporta a excel la R^2 de cada ciclo
xlswrite(File,R_val,6)%exporta a excel la R^2 de cada ciclo

xlswrite(File,{'Num. of neurons:', 'Average MSE sorted',...

```

```

'Num. of neurons:',...
'R^2 sorted'},7,'A1')
xlswrite(File,Nneurona1,7,'A2')%exporta a excel el orden del
% numero de neuronas
%en la capa oculta, de menor a mayor, de acuerdo con su MSE promedio
%durante las 100 ejecuciones.
xlswrite(File,menorMSE',7,'B2')%Exporta los valores de promedio
% de MSE
%en validación de los 100 ciclos para cada número de neuronas,
% para saber
%cual num. de neuronas tiene el menor MSE_val promedio.
xlswrite(File,Nneurona2,7,'C2')
xlswrite(File,mayorRcuadrada',7,'D2')%exporta a excel la Rcuadrada
% de cada ciclo.
end

```

```

msgbox ('Se alcanzaron las 100 iteraciones... El archivo de excel...
se escribió. Creando una nueva RNA...')

```

## CONSTRUCCIÓN DE UNA NUEVA RNA CON EL NÚMERO ÓPTIMO DE NEURONAS:

```

%Por último se corre una vez más una estructuración y
%entrenamiento de una RNA y guardar el MSE, R^2, y pesos y bias.---

```

```

net = fitnet(n);
net.inputs{1}.processFcns = {'removeconstantrows','mapminmax'};
net.outputs{2}.processFcns = {'removeconstantrows','mapminmax'};
net.divideFcn = 'dividerand'; % Divide data randomly
net.divideMode = 'sample'; % Divide up every sample
net.divideParam.trainRatio = 70/100;
net.divideParam.valRatio = 15/100;
net.divideParam.testRatio = 15/100;
net.trainFcn = 'trainlm';% Levenberg-Marquardt
net.performFcn = 'mse'; % Mean squared error
net.plotFcns = {'plotperform','plottrainstate','ploterrhist', ...
'plotregression', 'plotfit'};
net.trainParam.showWindow=0;
net.trainParam.showCommandLine=1;
[net,tr] = train(net,inputs,targets);
% % Test the Network
outputs = net(inputs);
%-----

```

```

tsOut=outputs(tr.valInd);%Salidas de los datos de validación

Ytrg=targets(tr.valInd);%ok, datos target del set de validación

[r]=regression(Ytrg,tsOut);%calcula el coeficiente de regresión
% entre los valores Target y los valores de salida de
% la red calculados en VALIDACIÓN.

disp('Coeficiente de regresión')
fprintf('R^2=%f\n',r)%despliega el valor de la R^2
%-----
disp('Error en VALIDACIÓN')
fprintf('MSE=%f\n',perform(net,Ytrg,tsOut))%despliega el MSE
MSEred=perform(net,Ytrg,tsOut);

Salida(1,1)=MSEred;%Se almacena el valor del MSE
Salida(2,1)=r;%Se almacena el valor de la R^2: el
%coeficiente de regresión.
VectorPesosyBias=getwb(net); %Se almacena el vector de pesos y bias

xlswrite(File,Salida,8,'A1')%exporta a excel el MSE, R^2 de la
% última RNA generada.
xlswrite(File,VectorPesosyBias,8,'C1')%exporta a excel los pesos
% y bias de la última RNA generada.

xlswrite(File,hWilcoxon_mse, 9,'A1')%exporta los valores de h de
% la prueba
% de rango con signo de Wilcoxon, para MSE
%de validación de cada una de las 100 ejecuciones de los 11
%modelos de RNA con número de neuronas 5 a 15.
xlswrite(File,hWilcoxon_r, 9,'B1')%exporta los valores de h de
% la prueba de rango con signo de Wilcoxon, para R^2
%de validación para cada una de las 100 ejecuciones de los 11
% modelos de RNA con número de neuronas de 5 a 15.

%Genera una hoja en excel con la tabla de comparación de modelos:
xlswrite(File,{'average MSE sorted', 'Model',...
'Models with equal mean'},10,'A1')
xlswrite(File,menorMSE', 10,'A2')%Genera una hoja en excel con
% la tabla de comparación de modelos.
xlswrite(File,Modelo, 10,'B2')
xlswrite(File,comparaModelo,10,'C2')

```

```

msgbox('Se ejecutó la estructuración de una RNA, su MSE, R^2,
      pesos y bias
      se guardaron en la Hoja 8 y 9 del mismo archivo de excel
      recientemente guardado.')
%Al final se grafica la regresión entre los datos target
%(datos reales)y los que se generan como
% salida de las RNA.

```

## GENERANDO LAS GRÁFICAS PARA EL ANÁLISIS ESTADÍSTICO

```

figure, plotregression(targets,outputs,...
'Regresión de TARGETS vs OUTPUTS')
%saveas(gcf,'RegresionFinal.fig')
print -r600 -deps2 9_RegresionFinalGeneral

%También se grafican la regresión de los datos no usados
% en trainin y que se usaron en la validación:
% tsOut y Ytrg:

figure, plotregression(Ytrg,tsOut,..
'Regresión de TARGETS vs OUTPUTS
en Validación')
%saveas(gcf,'RegresionFinal.fig')
print -r600 -deps2 10_RegresionFinalValidacion

%Después se generan los plots ocultos de comparación de cada variable
% simulada contra la real:
f = figure;
plot(inputs, targets(7,:), 'o', inputs, outputs(7,:))
legend('Real', 'RNA', 'Location', 'Best')
xlabel('Temperatura normalizada')
ylabel('Entropía de vapor normalizada')
saveas(gcf, 'Compara_Sv.fig')
print(gcf, '-deps2', '-tiff', '-r600', sprintf('%s.eps', 'Compara_Sv'));
system(sprintf('ps2pdf -deps2Crop %s.eps %s.pdf', 'Compara_Sv',
... 'Compara_Sv'));
close(f);

f = figure;
plot(inputs, targets(6,:), 'o', inputs, outputs(6,:))
legend('Real', 'RNA', 'Location', 'Best')

```

```

xlabel('Temperatura normalizada')
ylabel('Entropía de líquido normalizada')
saveas(gcf,'Compara_Sl.fig')
print(gcf,'-deps2','-tiff','-r600',sprintf('%s.eps',...
'Compara_Sl'));
system(sprintf('ps2pdf -deps2Crop %s.eps %s.pdf','Compara_Sl',...
'Compara_Sl'));
close(f);

f = figure;
plot(inputs, targets(5,:), 'o', inputs, outputs(5,:))
legend('Real', 'RNA', 'Location', 'Best')
xlabel('Temperatura normalizada')
ylabel('Entalpía de vapor normalizada')
saveas(gcf,'Compara_Hv.fig')
print(gcf,'-deps2','-tiff','-r600',sprintf('%s.eps','Compara_Hv'));
system(sprintf('ps2pdf -deps2Crop %s.eps %s.pdf','Compara_Hv',
...'Compara_Hv'));
close(f);

f = figure;
plot(inputs, targets(4,:), 'o', inputs, outputs(4,:))
legend('Real', 'RNA', 'Location', 'Best')
xlabel('Temperatura normalizada')
ylabel('Entalpía de líquido normalizada')
saveas(gcf,'Compara_Hl.fig')
print(gcf,'-deps2','-tiff','-r600',sprintf('%s.eps','Compara_Hl'));
system(sprintf('ps2pdf -deps2Crop %s.eps %s.pdf','Compara_Hl',
...'Compara_Hl'));
close(f);

f = figure;
plot(inputs, targets(3,:), 'o', inputs, outputs(3,:))
legend('Real', 'RNA', 'Location', 'Best')
xlabel('Temperatura normalizada')
ylabel('Volúmen de vapor normalizado')
saveas(gcf,'Compara_Dv.fig')
print(gcf,'-deps2','-tiff','-r600',sprintf('%s.eps','Compara_Dv'));
system(sprintf('ps2pdf -deps2Crop %s.eps %s.pdf','Compara_Dv',
...'Compara_Dv'));
close(f);

f = figure;

```

```
plot(inputs, targets(2,:), 'o', inputs, outputs(2,:))
legend('Real', 'RNA', 'Location', 'Best')
xlabel('Temperatura normalizada')
ylabel('Densidad de líquido normalizada')
saveas(gcf, 'Compara_Dl.fig')
print(gcf, '-deps2', '-tiff', '-r600', sprintf('%s.eps', 'Compara_Dl'));
system(sprintf('ps2pdf -deps2Crop %s.eps %s.pdf', 'Compara_Dl',
... 'Compara_Dl'));
close(f);

f = figure;
plot(inputs, targets(1,:), 'o', inputs, outputs(1,:))
legend('Real', 'RNA', 'Location', 'Best')
xlabel('Temperatura normalizada')
ylabel('Presión normalizada')
saveas(gcf, 'Compara_T.fig')
print(gcf, '-deps2', '-tiff', '-r600', sprintf('%s.eps', 'Compara_T'));
system(sprintf('ps2pdf -deps2Crop %s.eps %s.pdf', 'Compara_T',
... 'Compara_T'));
close(f);

%Por último se guardan todas las variables en un archivo .mat,
%por si se llegan a ocupar después.
save 11_variablesGeneradas.mat

msgbox('Cálculo concluido. Fue un gusto poder servirte.')
```

## A.9. Parámetros (pesos y bias) de las redes neuronales obtenidas.

En esta sección se presentan los valores de los pesos y umbrales, redondeados a 4 cifras decimales, para las diferentes RNAs que se desarrollaron en este estudio. Estos parámetros de la RNA se pueden utilizar en una hoja de cálculo con el fin de obtener las propiedades termodinámicas de los diferentes refrigerantes dados.

	$b_i^\alpha$	$w_i^\alpha$	$b_j^\beta$	$w_{1j}^\beta$	$w_{2j}^\beta$	$w_{3j}^\beta$	$w_{4j}^\beta$	$w_{5j}^\beta$	$w_{6j}^\beta$
1	-10.2344	9.6285	1.2411	0.0952	-1.7739	-0.4191	0.0521	0.0002	-0.0001
2	1.984	-1.7377	-1.2784	-1.0988	0.7538	0.2134	0.3234	-0.0215	0.0225
3	0.623	-1.4659	0.6709	0.001	0.0051	-0.007	0.0309	-0.2	1.4469
4	-0.5581	-1.2896	0.6203	0.4145	-0.5661	-0.326	-0.47	0.0382	-0.0312
5	3.0524	3.9463	-2.2586	-1.8071	0.5453	-0.5039	-0.8664	0.0821	-0.0447
6	-7.997	-8.115	0.3705	0.292	-0.397	-0.2518	-0.5868	0.0901	-0.0618
7	-	-	-0.974	-1.0969	0.5706	-0.1816	0.4212	-0.2749	0.2064

Tabla A.7: Valores de pesos y umbrales para la RNA del R12.

	$b_i^\alpha$	$w_i^\alpha$	$b_j^\beta$	$w_{1j}^\beta$	$w_{2j}^\beta$	$w_{3j}^\beta$	$w_{4j}^\beta$	$w_{5j}^\beta$
1	37.6412	-35.9807	26.9546	-0.0313	-37.5136	9.8853	2.1149	-2.0225
2	1.7595	-0.5537	-25.433	6.3311	25.1042	-16.3201	-10.5652	-3.2786
3	0.7292	-0.3754	154.7518	-0.1552	7.1027	-7.946	-5.397	158.7563
4	-0.27	0.2855	11.7373	-2.7047	-13.3513	9.4194	8.1353	0.3948
5	-6.9576	-4.3861	-72.2843	17.4221	67.683	-33.5533	-8.2977	-10.327
6	-	-	6.9059	-1.8933	-15.4593	15.967	13.4007	-3.1355
7	-	-	-12.28	4.292	43.7127	-42.6064	-28.6335	14.3066

Tabla A.8: Valores de pesos y umbrales para la RNA del R22.

	$b_i^\alpha$	$w_i^\alpha$	$b_j^\beta$	$w_{1j}^\beta$	$w_{2j}^\beta$	$w_{3j}^\beta$	$w_{4j}^\beta$	$w_{5j}^\beta$	$w_{6j}^\beta$
1	-59.4273	57.728	2.3304	0.0264	0.2517	-3.2156	-0.4724	0.0282	-0.0591
2	-5.0393	4.4574	-5.5907	-5.0559	-0.5746	0.7853	-1.1286	-0.0006	0.0369
3	1.4745	-1.1557	16.9015	-0.0283	0.0137	0.0228	0.0213	-0.1373	17.8029
4	0.227	0.4209	1.954	1.8868	0.1713	-0.6216	1.8134	-0.01	-0.0679
5	2.3602	2.8905	-9.8413	-7.8168	-1.1044	0.625	3.5296	-0.0427	0.0484
6	-7.4926	-6.0255	0.9026	1.3083	0.1692	-0.1764	2.0648	0.0809	-0.205
7	-	-	-3.2233	-4.5923	-0.8333	-0.2237	-0.7401	-0.4978	1.5173

Tabla A.9: Valores de pesos y umbrales para la RNA del R134a.

	$b_i^\alpha$	$w_i^\alpha$	$b_j^\beta$	$w_{1j}^\beta$	$w_{2j}^\beta$	$w_{3j}^\beta$	$w_{4j}^\beta$	$w_{5j}^\beta$
1	-6.8473	4.5968	5.3933	4.7847	3.3997	1.4161	-1.9047	-0.3207
2	-1.5581	1.2491	-12.7362	-12.7039	-0.7252	2.6674	0.4211	0.0978
3	0.018	-0.2131	72.6781	0.5858	-0.0635	-0.1963	1.3268	-71.8328
4	-2.7052	-1.0994	4.26	3.8116	0.5049	-3.7147	0.2977	0.1754
5	12.5107	10.3666	-17.3538	-18.8495	-0.7354	-6.8413	2.2798	0.4421
6	-	-	-1.3855	2.9667	0.0686	-3.601	-4.8282	-0.2221
7	-	-	19.7042	-7.1904	-0.2546	-1.043	26.5188	-1.4851

Tabla A.10: Valores de pesos y umbrales para la RNA del R290.

	$b_i^\alpha$	$w_i^\alpha$	$b_j^\beta$	$w_{1j}^\beta$	$w_{2j}^\beta$	$w_{3j}^\beta$	$w_{4j}^\beta$	$w_{5j}^\beta$	$w_{6j}^\beta$	$w_{7j}^\beta$	$w_{8j}^\beta$	$w_{9j}^\beta$
1	13.91	-11.29	-1.26	-1.08	-0.05	-0.05	0.07	0.08	0.1	0.15	-0.54	-2.29
2	9.56	-12.53	-1.54	-0.8	-0.05	-0.05	0.07	0.08	0.1	0.15	-0.54	-2.29
3	6.13	-11.87	0.99	0.17	0.3	0.07	-0.09	-0.08	-0.08	-0.11	0.32	1.12
4	-2.48	11.74	5.3	4.23	-0.02	0	0	0	0.01	-0.03	0	10.47
5	1.01	10.86	-1.05	-0.71	-0.16	-0.06	0.08	0.08	0.09	0.13	-0.44	-1.71
6	3.52	9.68	-2.36	-1.08	0.86	0.06	-0.07	-0.02	0.02	0.08	-0.57	-2.77
7	4.79	7.34	-1.3	-1.18	-0.12	-0.05	0.07	0.07	0.08	0.13	-0.49	-2.47
8	-6.5	-6.67	0.85	1.87	0.3	0.04	-0.05	-0.05	-0.05	-0.07	0.34	2.8
9	-34.75	-33.51	-	-	-	-	-	-	-	-	-	-

Tabla A.11: Valores de pesos y umbrales para la RNA del R410a.

	$b_i^\alpha$	$w_i^\alpha$	$b_j^\beta$	$w_{1j}^\beta$	$w_{2j}^\beta$	$w_{3j}^\beta$	$w_{4j}^\beta$	$w_{5j}^\beta$
1	33.9132	-32.1382	-7.7137	0.2464	-6.2004	-47.5449	-19.0659	0.8235
2	3.142	-1.6674	-10.2481	7.1406	5.1859	8.4352	2.1301	-1.3726
3	-0.4954	-0.1971	52.6793	-0.2043	1.1429	-7.5596	-3.4109	56.6102
4	0.4562	0.4399	3.3898	-3.9673	-2.5091	-10.9074	-2.6062	1.0634
5	-6.1059	-4.0579	-27.8991	13.7256	14.9569	12.4164	6.8955	-2.9857
6	-	-	3.2407	-2.9131	-2.5011	-0.258	2.1113	-1.0976
7	-	-	-6.1928	3.0742	5.8524	-14.6787	-8.0338	6.04

Tabla A.12: Valores de pesos y umbrales para la RNA del R717.

# Bibliografía

- [1] ABDOLREZA, M., FAHIME, P., AND SAYED, M. H. A new approach based on artificial neural networks for prediction of high pressure vapor-liquid equilibrium. *Australian Journal of Basic and Applied Sciences* 3 (2009), 1851–1862.
- [2] ABDOLREZA, M., FAHIME, P., SAYED, M. H., AND SHARIFI, A. Prediction of pvt properties of ammonia by using artificial neural network and equations of state. *ARPJ Journal of Engineering and Applied Sciences* 3 (2008), 18–27.
- [3] ADAM, O., AND LEONARD, O. Explicit thermodynamic properties using radial basis functions neural networks.
- [4] AMERICAN SOCIETY OF HEATING, R., AND ENGINEERS, A. C. *ASHRAE Handbook in compact disk*. ASHRAE, 2001.
- [5] ARMAS, J. C., LAPIDO RODRÍGUEZ, M., GÓMEZ, J. R., AND VALDIVIA NODAL, Y. A thermodynamic evaluation of chilled water central air conditioning systems using artificial intelligence tools. *Ingeniería e Investigación* 31 (2011), 134–142.
- [6] BALZHISER, S., AND ELIASSEN. *Chemical Engineering Thermodynamics: The Study of Energy, Entropy And Equilibrium*. Ed. Prentice-Hall, 1985.
- [7] BOUBLIK, T., FRIED, V., AND HALA, E. *The Vapor Presures of Pure Substances*. Ed. Elsevier, 1984.

- [8] DE ARMAS TEYRA, M., GOMEZ SARDUY, J., PEREZ TELLO, C., MERRIÑO STAND, L., AND SEPULVEDA CHAVERRA, J. *Inteligencia Artificial Aplicada Al Análisis De Sistemas Energéticos Con Matlab*. Fundación Universitaria Tecnológico Comfenalco, 2008.
- [9] ÇENGEL, Y. A., AND BOLES, M. A. *Termodinámica*. Ed. McGraw-Hill, 1997.
- [10] GANDHIDASAN, P., AND MOHANDÉS, M. A. Predictions of vapor pressures of aqueous desiccants for cooling applications by using artificial neural networks. *Applied Thermal Engineering* 28, 2-3 (feb 2008), 126–135.
- [11] GHARAGHEIZI, F., AND SALEHI, G. R. Prediction of enthalpy of fusion of pure compounds using an artificial neural network-group contribution method. *Thermochimica Acta* 521, 1-2 (jul 2011), 37–40.
- [12] ISASI V., P., AND GALVÁN L., I. M. *Redes de Neuronas Artificiales, Un enfoque práctico*. Pearson, 2004.
- [13] LAZZÚS, J. A. Prediction of solid vapor pressures for organic and inorganic compounds using a neural network. *Thermochimica Acta* 489, 1-2 (may 2009), 53–62.
- [14] MALLO, C. Predicción de la demanda eléctrica horaria mediante redes neuronales artificiales.
- [15] MANDISCHER, M., GEYER, H., AND ULBIG, P. Neural networks and evolutionary algorithms for the prediction of thermodynamic properties for chemical engineering.
- [16] PIZARRO, J., GUERRERO, E., AND GALINDO, P. L. P.: A statistical model selection strategy applied to neural networks.

- [17] REZEKI, S., SUBANAR, GURITNO, S., AND SUTIJO, B. Statistical model selection based on resampling procedure for neural networks classification. *Memories of First International Conference on Mathematics and Statistics, Bandung Islamic University*. (2006).
- [18] SANDHYA, S. *Neural Networks for Applied Sciences and Engineering*. Auerbach Publications, 2007.
- [19] SETAREH SHEIKH, M. B. Estimating the saturation thermodynamic properties of propene using a feed forward neural network. *World Applied Sciences Journal 4 (2): 169-173, 2008* (2008).
- [20] SHAUMM. *Análisis Vectorial*. Ed. Shaumm, 1997.
- [21] SMITH, J., VAN NESS, H., AND ABBOTT, M. *Introducción a la Termodinámica en Ingeniería Química*. Ed. McGraw-Hill, quinta edición, 1997.
- [22] SÖZEN, A., ÖZALP, M., AND ARCAKLIOGLU, E. Calculation for the thermodynamic properties of an alternative refrigerant (R508b) using artificial neural network. *Applied Thermal Engineering 27, 2-3 (feb 2007), 551–559*.
- [23] VASHISHTHA, M. Application of artificial neural networks in prediction of vapor-liquid equilibrium data.
- [24] XU, Q.-S., LIANG, Y.-Z., AND DU, Y.-P. Monte carlo cross-validation for selecting a model and estimating the prediction error in multivariate calibration. *Journal of Chemometrics 18, 2* (2004).
- [25] YORU, Y., KARAKOC, T. H., AND HEPBASLI, A. Application of artificial neural network (ANN) method to exergy analysis of thermodynamic systems. 715–718.