



UNIVERSIDAD SIMÓN BOLÍVAR
DECANATO DE ESTUDIOS PROFESIONALES
COORDINACIÓN DE INGENIERÍA GEOFÍSICA

**HIBRIDIZACIÓN DE LÓGICA DIFUSA Y ALGORITMOS GENÉTICOS EN
LA PREDICCIÓN DE REGISTROS DE VELOCIDAD DE ONDA S.
CAMPO GUAFITA**

Por:
Br. Edinson Leandro Medina Alfonzo

PROYECTO DE GRADO
Presentado ante la Ilustre Universidad Simón Bolívar
como requisito parcial para optar al título de
Ingeniero Geofísico

Sartenejas, Octubre de 2011



UNIVERSIDAD SIMÓN BOLÍVAR
DECANATO DE ESTUDIOS PROFESIONALES
COORDINACIÓN DE INGENIERÍA GEOFÍSICA

**HIBRIDIZACIÓN DE LÓGICA DIFUSA Y ALGORITMOS GENÉTICOS EN
LA PREDICCIÓN DE REGISTROS DE VELOCIDAD DE ONDA S.
CAMPO GUAFITA**

Por:
Br. Edinson Leandro Medina Alfonzo

Realizado con la asesoría de:
Dra. Milagrosa Aldana

PROYECTO DE GRADO
Presentado ante la Ilustre Universidad Simón Bolívar
como requisito parcial para optar al título de
Ingeniero Geofísico

Sartenejas, Octubre de 2011



UNIVERSIDAD SIMÓN BOLÍVAR
Decanato de Estudios Profesionales
Coordinación de Ingeniería Geofísica

ACTA DE EVALUACIÓN DEL PROYECTO DE GRADO

CÓDIGO DE LA ASIGNATURA: EP3312

FECHA: 17/05/2011

ESTUDIANTE: Edinson Leandro Medina Alfonzo CARNET: 05-38512

TÍTULO DEL TRABAJO: "Hibridización de Lógica Difusa y Algoritmos Genéticos en la predicción de Registros de Velocidad de Onda S. Campo Guafita".

TUTOR ACADEMICO: Prof. Milagrosa Aldana JURADO: Prof. Evert Duran

JURADO: Prof. Crelia Padrón

APROBADO: REPROBADO:

OBSERVACIONES:

El Jurado considera **por unanimidad** que el trabajo es EXCEPCIONALMENTE BUENO:

SI NO En caso positivo, justificar razonadamente:

Tutor Académico

Prof. Milagrosa Aldana

Evert Leonardo Durán
Jurado

Prof. Evert Durán

Jurado

Prof. Crelia Padrón

RESUMEN

En la presente investigación se aplican conceptos basados en Lógica Difusa-Redes Neuronales (ANFIS) y Algoritmos Genéticos para la predicción de registros de Velocidad de Onda S a partir de los registros disponibles (Densidad, Velocidad de Onda P, Porosidad, Saturación de Agua y Volumen de Arcilla) en los pozos GF-87N y GF-109N del campo Guafita (Noroeste del Estado Apure). Asimismo, se propone una metodología que hibridice ambos métodos para intentar una mejora en las predicciones. Para ello se utilizaron las herramientas del programa **Matlab versión R2009a**.

Al momento de llevar a cabo las predicciones, se utilizaron en todos los casos uno y dos registros de entrada (todas las combinaciones posibles con los registros disponibles). En la aplicación de Algoritmos Genéticos, el método consiste en estimar los parámetros de una determinada ecuación que asocie los registros de entrada con el de salida (Velocidad de Onda S). Para ello, se consideraron dos ecuaciones: una lineal y una polinómica.

En relación al uso de ANFIS, se consideraron dos opciones, Grid Partition y Subtractive Clustering, que se diferencian en la combinación de las funciones de membresía. En el primer caso, las reglas difusas son introducidas por el usuario, mientras que en el segundo, las reglas indicadas de acuerdo a la distribución de los datos de entrada. La hibridación entre ambos sistemas se llevó a cabo partiendo de la salida del ANFIS, tomando los parámetros de las funciones de salidas de las reglas difusas y optimizándolos mediante Algoritmos Genéticos.

En el caso de un registro, las tres metodologías arrojan buenas predicciones cuando se utiliza como entrada Velocidad de Onda P y Densidad. Mientras que con dos registros para los cálculos de la Velocidad de Onda S, el algoritmo híbrido difuso-genético logra optimizar la estimación realizada previamente mediante ANFIS en los casos de Densidad-Velocidad de Onda P y Velocidad de Onda P-Porosidad.

DEDICATORIA

En particular deseo dedicarle este trabajo a 4 personas:

Dos de ellas quienes siempre me han apoyado en todas mis decisiones sin juzgarme, queriendo sólo lo mejor. Nada de lo que he logrado y de lo que pueda lograr habrá sido posible sin ellos. **Mamá, Papá**, esto es por ustedes y para ustedes.

A ti **Alysson**, que eres la persona más importante de mi vida. Te Amo con todo mi corazón. Desde que llegaste a mi vida te he querido cada día más. Este trabajo sienta las bases de lo que sé que tu también en su momento y en tu área serás capaz de lograr. Sólo me queda esperar que me superes y que el tuyo sea muchísimo mejor que el mío.

Por último, a ti **abuelo nené**, que no parabas de sentirte orgulloso de mi. Recuerdo ahora con lágrimas en los ojos tu sonrisa de satisfacción al hablar de mi ante tus conocidos.

Uno no sabe lo que tiene hasta que lo pierde. Y vaya que cierto es...

AGRADECIMIENTOS

Son muchos quienes han contribuido tanto a lo largo de mi vida estudiantil (no sólo en la realización de esta tesis que constituye la primera cumbre de tantos años de trabajo y sacrificio) como cotidiana y a quienes me gustaría en su mayoría mencionar a continuación. Por lo tanto, aprovecho esta ventana para agradecer especialmente:

A Dios. Gracias Dios mío por escucharme en tantas oportunidades y por transmitirme esa sensación de paz en momentos en donde la desesperación o la frustración reinaban.

A MIS PADRES. No me alcanzarán nunca las palabras para agradecerles a quienes tienen trabajando juntos sin descanso por más de 20 años para que yo esté en donde estoy ahora. Gracias por creer incondicionalmente en mi (inclusive en momentos en que perdía la fe en mi mismo), por apoyarme en todos los sentidos, por ser un ejemplo a seguir, por enseñarme que en la vida todo es posible a base de constancia y sacrificio y vaya que sólo nosotros sabemos realmente todo lo que hemos tenido que perseverar y muchas veces sacrificar para llegar hasta aquí. Papá eres un luchador nato, incansable, vas hacia lo que quieras a pesar de lo muy duro que te lo ponga la vida. Mamá tus palabras siempre son tranquilizadoras. Los amo.

A ti hermíntita bella. Preciosa. No hay calificativo que alcance tu belleza. Gracias por existir, por tus palabras de aliento, por tus cariños, por quererme y aceptarme tal y como soy a pesar de las malas caras, los malos ratos, todas las peleas.

A mi tutora MILAGROSA ALDANA. Sin duda la persona más importante en la elaboración de este trabajo. Gracias por portarse tan bien conmigo profe, por todas las oportunidades, por su sonrisa y comprensión en momentos difíciles. Usted es una persona inigualable. Muchísimas gracias.

A mis primos: Liz por escucharme, me encanta hablar contigo siempre; Rohany, por permitirme usar tu PC todas las veces que lo necesité; Ángel, eres mi primo hermano, ahora vivimos más cerca y sin embargo, nos vemos menos, cosas de la vida ¿no?; Liseth, Jenny, Leonel, Yenderlys, Adrian, Naytsir, los quiero a todos.

A mis tíos: Adriana, Can, Carlos Juan (gracias por tu incondicionalidad man te quiero full), Nancy, Lilo, Nelly. Gracias a tod@s por su apoyo, su cariño y por confiar en mi.

A 4 mis abuelos, Armando, Carlota, Bárbara y Silvestre (a pesar que no llegué a conocerte). Sin ustedes no hubiese tenido los padres que tengo ahora.

Al resto de mis familiares que aún falta por nombrar. No por eso son menos recordados.

A los profesores del Departamento de Ciencias de la Tierra, en especial: Andrés Pilloud, Corina Campos, Oswaldo Guzmán, Crelia Padrón, por ser excelentes profesores y preocuparse por transmitir sus conocimientos a sus estudiantes. Aprendí mucho de ustedes.

La experiencia en campo: con AP es algo que sinceramente recomiendo a todos y con la dupla Oswaldo-Corina, impresiona lo mucho que puedes aprender de ellos en un sólo día.

A Raúl Alexander. Hermanazo eres como escuché por ahí hace poco: “el más cierto en las horas inciertas”. Tu apoyo ha sido importantísimo. Ya vemos como tantas noches de estudio a lo largo de estos 4 años rinden este 2011 sus frutos.

A Yareth Delgado y Marina Ramirez, ambas, a su forma, son increíbles. Las quiero mucho y nunca las olvidaré. Gracias por su amistad. Espero contar con ustedes siempre.

A Ketty. Gracias por todo, extraño ya esas noches en el 3er piso prepráandonos para los exámenes. Espero que culmines con éxito tu pasantía.

A María Rosa. Te valoro muchísimo como persona. Gracias por tu amistad desde los mismísimos vivenciales.

A Isabel. Gracias por no dejar que me retirara hace ya tres años (que rápido pasa el tiempo ¿no?). A ti, de corazón, también te deseo lo mejor, un gran abrazo.

A las demás personas maravillosas que tuve la oportunidad de conocer y compartir en la carrera, amigos de geofísica: Orianna, Miguel, Luis Castillo (gracias por tu apoyo man), Luis Villareal (por todas las colas de noche, eres un buen amigo con una visión ejemplar), Clara (te quiero muchísimo espero estar en tu defensa), Richard, Vanessa, Alberto, Angela, Keyla, Bernardo, Gabriel, Adrian, Abiel, Vicky, Mery, Carlitos, Laura, Alfredo, Liz-María Gracia (aprendí mucho de las dos en las prepas). Gracias a los que contribuyeron en aquella salida de campo de Geología del Petróleo (ese gesto no lo olvidaré jamás).

A Raidin y Rosaura por su colaboración en distintos momentos de la tesis. Carmencita por estar siempre presta a ayudar e informar a todo el que lo necesite. Y a toda la Comunidad Geofísica USB en General.

A David Hernández. Tu apoyo fue crucial para la prosecución de esta tesis en un momento clave. También a Daily. Gracias por estar pendiente de mi, éxito Dai.

A Naiyara. Te quiero mucho a pesar que nos veamos ahora tan poco. Rubén, Jesús, Viviana, María de los Ángeles, Abraham, Josmairy e Hilda, gracias por todos los momentos buenos y malos. Cada quien forja ya su propio camino, el cual se avecina exitoso. Los felicito.

A los profesores José Angel, Kafka, Odra, Marisol, Jean, Betzabeth, Tomás, Josefina, ustedes fueron los precursores de éste éxito que estoy por alcanzar.

Gracias a todos los que además han contribuido y no han sido mencionados acá. Hasta una próxima oportunidad.

ÍNDICE GENERAL

RESUMEN	iv
DEDICATORIA	v
AGRADECIMIENTOS	vi
ÍNDICE GENERAL	viii
ÍNDICE DE TABLAS	xii
ÍNDICE DE FIGURAS	xvi
INTRODUCCIÓN	1
1 MARCO TEÓRICO	4
1.1 Redes Neuronales	5
1.1.1 Funcionamiento del Cerebro Humano	5
1.1.2 Definición de Red Neuronal Artificial	6
1.1.3 Modelo de una Neurona	7
1.1.4 Ventajas	8
1.1.5 Aplicaciones a las Geociencias	9
1.2 Lógica Difusa	10
1.2.1 Fundamentos	11
1.2.1.1 Conjuntos Difusos	11
1.2.1.2 Función de Membresía	12
1.2.1.3 Operadores Lógicos	14
1.2.1.4 Reglas Difusas “Si ... Entonces”	16
1.2.2 Sistemas de Inferencia Difusa	18
1.2.2.1 Mamdani	19
1.2.2.2 Takagi-Sugeno-Khan	19
1.2.3 Ventajas	21
1.2.4 Aplicaciones a las Geociencias	22

1.3	Algoritmos Genéticos	24
1.3.1	Conceptos Básicos	24
1.3.2	Funcionamiento de un Algoritmo Genético Simple	25
1.3.2.1	Función Objetivo	25
1.3.2.2	Definición del tamaño de la población	25
1.3.2.3	Codificación de las posibles soluciones	25
1.3.2.4	Selección de individuos	26
1.3.2.5	Cruce	26
1.3.2.6	Mutación	26
1.3.2.7	Condición de parada	27
1.3.2.8	Algoritmos Genéticos en Matlab	27
1.3.3	Ventajas	28
1.4	Algoritmos Híbridos	30
1.4.1	Fusión de Lógica Difusa con Algoritmos Genéticos.	30
1.4.2	ANFIS: Lógica Difusa-Redes Neuronales	30
2	MARCO GEOLÓGICO	33
2.1	Ubicación Geográfica	33
2.2	Geología Regional	33
2.2.1	Modelo Estructural	34
2.2.2	Modelo Estratigráfico Sedimentológico	36
2.2.3	Aspectos Petrolíferos	38
2.3	Geología Local	39
2.3.1	Modelo Estructural	39
2.3.2	Modelo Estratigráfico Sedimentológico	41
2.3.3	Formación Guafita (Oligoceno-Mioceno temprano)	43
2.3.4	Producción	44
3	METODOLOGÍA	45
3.1	Estudio de los datos disponibles	46
3.1.1	Visualización de los Registros de Pozo	46
3.1.2	Arreglo de los Registros de Entrada para los Algoritmos	50
3.1.2.1	Caso 1: Cálculo de “ \log_{10} ”	51
3.1.2.2	Normalización de los Registros.	51
3.2	Criterios de Comparación	53
3.3	Técnicas de Predicción: AG-ANFIS	54
3.3.1	Algoritmos Genéticos	55
3.3.2	ANFIS	58

3.4	Pruebas Preliminares	59
3.4.1	Algoritmos Genéticos	59
3.4.2	ANFIS	60
3.5	Algoritmo Híbrido Difuso-Genético	62
3.6	Ejecución de los tres Algoritmos	66
4	RESULTADOS Y ANÁLISIS	70
4.1	Pruebas Preliminares	71
4.1.1	Algoritmos Genéticos	71
4.1.2	ANFIS	77
4.1.2.1	Pruebas #1: Un Registro de Entrada.	78
4.1.2.2	Pruebas #2: Dos Registros de Entrada.	79
4.1.2.3	Pruebas #3: Tres Registros de Entrada.	82
4.2	Un Registros de Entrada	88
4.2.1	Algoritmos Genéticos	88
4.2.2	ANFIS-Algoritmo Híbrido	92
4.2.2.1	Parte I: Grid Partition	93
4.2.2.2	Parte II: Substractive Clustering	98
4.2.3	Comentarios Adicionales	105
4.3	Dos Registros de Entrada	106
4.3.1	Algoritmos Genéticos	107
4.3.2	ANFIS-Algoritmo Híbrido	114
4.3.2.1	Parte I: Grid Partition	114
4.3.2.2	Parte II: Substractive Clustering	121
4.4	Tres Registros de Entrada	126
4.4.1	Algoritmos Genéticos	127
4.4.2	ANFIS-Algoritmo Híbrido	128
4.4.2.1	Parte I: Grid Partition	129
4.4.2.2	Parte II: Substractive Clustering	130
CONCLUSIONES Y RECOMENDACIONES	134	
REFERENCIAS BIBLIOGRÁFICAS	137	
A FUNCIONES DE MEMBRESÍA	141	
B TABLAS ADICIONALES	147	
B.1	Pruebas Preliminares: ANFIS	147
B.2	Estimaciones realizadas utilizando tres Registros de Entrada	154

B.2.1	Algortimos Genéticos	154
B.2.2	Parte I: Grid Partition	157
B.2.3	Parte II: Substractive Clustering	160
C	CÓDIGOS	163
C.1	Función “ag_lineal”	163
C.2	Función “ag_polinomial”	165
C.3	Función “carga_registro”	167
C.4	Función “eliminar”	169
C.5	Función “eval_ag_lineal”	169
C.6	Función “eval_ag_polinomio”	170
C.7	Función “graficas_registro”	171
C.8	Función “hibrido”	173
C.9	Función “indice”	179
C.10	Función “log_dif”	179
C.11	Función “normalización”	180
C.12	Función “registro_final”	181
C.13	Programa Principal	184

ÍNDICE DE TABLAS

1.1 Tabla de la verdad con las operaciones AND, OR, NOT	14
1.2 Operaciones lógicas <i>min</i> , <i>max</i> , <i>complemento</i>	15
3.1 Pruebas Realizadas modificando las condiciones iniciales de los Algoritmos Genéticos	59
4.1 Pruebas realizadas, en el Pozo de Entrenamiento, variando las condiciones iniciales de Algoritmos Genéticos. Registro de Entrada: DEN.	71
4.2 Pruebas realizadas, en el Pozo Control, variando las condiciones iniciales de Algoritmos Genéticos. Registro de Entrada: DEN.	72
4.3 Pruebas realizadas, en el Pozo de Entrenamiento, variando las condiciones iniciales de Algoritmos Genéticos. Registro de Entrada: VP.	74
4.4 Pruebas realizadas, en el Pozo Control, variando las condiciones iniciales de Algoritmos Genéticos. Registro de Entrada: VP.	74
4.5 Condiciones Iniciales pre-seleccionadas de los Algoritmos Genéticos	76
4.6 Prueba adicional realizada modificando las condiciones iniciales de los Algoritmos Genéticos	77
4.7 Valores de RMSE y R^2 en la prueba adicional realizada en ambos pozos, variando las condiciones iniciales de Algoritmos Genéticos.	77
4.8 Parámetro “ <i>a</i> ” de la Ecuación 1 AG utilizando un Registro de Entrada . . .	89
4.9 Parámetros “ <i>a</i> ” y “ <i>b</i> ” de la Ecuación 2 AG utilizando un Registro de Entrada	89
4.10 RMSE para un Registro de Entrada utilizando Algoritmos Genéticos	90
4.11 R^2 para un Registro de Entrada utilizando Algoritmos Genéticos	90
4.12 RMSE para un Registro de Entrada utilizando ANFIS-Híbrido (4 reglas difusas)	93
4.13 R^2 para un Registro de Entrada utilizando ANFIS-Híbrido (4 reglas difusas).	93
4.14 Combinación de Reglas Difusas generadas a través de la técnica “Grid Partition” para un Registro de Entrada: VP	94
4.15 Parámetros σ y c de la función de membresía <i>Gaussiana</i> generados en el SID mediante la técnica “Grid Partition”. Registro de Entrada: VP	95
4.16 Parámetros “ <i>a</i> ” y “ <i>b</i> ” de las funciones de salida, utilizando “Grid Partition”. Registro de Entrada: VP	96

4.17	Parámetros “ a ” y “ b ” optimizados de las funciones de salida, utilizando “Grid Partition”. Registro de Entrada: VP	97
4.18	Número de Reglas Difusas obtenidas en cada registro de entrada	99
4.19	RMSE para un Registro de Entrada utilizando ANFIS-Híbrido (reglas difusas variables)	100
4.20	R^2 para un Registro de Entrada utilizando ANFIS-Algoritmo Híbrido (reglas difusas variables).	100
4.21	Combinación de Reglas Difusas generadas a través de la técnica “Subtractive Clustering” para un Registro de Entrada: DEN	101
4.22	Parámetros σ y c de la función de membresía <i>Gaussian</i> a generados en el SID mediante la técnica “Subtractive Clustering”. Registro de Entrada: DEN . .	102
4.23	Parámetros “ a ” y “ b ” de las funciones de salida, usando “Subtractive Clustering”. Registro de Entrada: DEN	102
4.24	Parámetros “ a ” y “ b ” optimizados de las funciones de salida, usando “Subtractive Clustering”. Registro de Entrada: DEN	102
4.25	Parámetros “ a_1 ” y “ a_2 ”de la Ecuación 1 AG utilizando dos Registros de Entrada	108
4.26	Parámetros “ a_1 ”, “ b_1 ”, “ a_2 ” y “ b_2 ”de la Ecuación 2 AG empleando dos Registros de Entrada	109
4.27	RMSE para dos Registros de Entrada con Algoritmos Genéticos: Pozo de Entrenamiento.	110
4.28	RMSE para dos Registros de Entrada utilizando ANFIS-Híbrido (6 reglas difusas).	115
4.29	R^2 para dos Registros de Entrada utilizando ANFIS-Híbrido (6 reglas difusas).	115
4.30	Combinación de Reglas Difusas generadas a través de la técnica “Grid Partition” para dos Registros de Entrada: VP-POR	116
4.31	Parámetros σ y c de la función de membresía <i>Gaussian</i> a generados en el SID mediante la técnica “Grid Partition”. Registros de Entrada: VP-POR	116
4.32	Parámetros “ a ”, “ b ” y “ c ” de las funciones de salida, utilizando “Grid Partition”. Registros de Entrada: VP-POR	117
4.33	Parámetros “ a ”, “ b ” y “ c ” optimizados de las funciones de salida, utilizando “Grid Partition”. Registros de Entrada: VP-POR	118
4.34	Número de Reglas Difusas obtenidas en cada caso con dos registros de entrada a través de “Subtractive Clustering”	121
4.35	RMSE para dos Registros de Entrada con ANFIS-Híbrido (reglas difusas variables).	122
4.36	R^2 para dos Registros de Entrada utilizando ANFIS-Híbrido (reglas difusas variables).	123

4.37	Combinación de Reglas Difusas generadas a través de la técnica “Substractive Clustering” para dos Registros de Entrada: DEN-VP	123
4.38	Parámetros σ y c de la función de membresía <i>Gaussian</i> a generados en el SID mediante la técnica “Substractive Clustering”. Registros de Entrada: DEN-VP	124
4.39	Parámetros “ a ”, “ b ” y “ c ” de las funciones de salida, utilizando “Substractive Clustering”. Registros de Entrada: DEN-VP	124
4.40	Parámetros “ a ”, “ b ” y “ c ” optimizados de las funciones de salida, utilizando “Substractive Clustering”. Registro de Entrada: DEN-VP	124
4.41	Número de Reglas Difusas obtenidas en cada caso para tres registros de entrada a través de “Substractive Clustering”	131
B.1	RMSE utilizando un Registro de Entrada en ANFIS con 4, 6, 8 y 9 reglas difusas. Pozo de Entrenamiento	147
B.2	RMSE utilizando un Registro de Entrada en ANFIS con 4, 6, 8 y 9 reglas difusas. Pozo Control	148
B.3	RMSE utilizando dos Registros de Entrada en ANFIS con 4 y 6 reglas difusas en el Pozo de Entrenamiento	148
B.4	RMSE utilizando dos Registros de Entrada en ANFIS con 8 y 9 reglas difusas en el Pozo de Entrenamiento	149
B.5	RMSE utilizando dos Registros de Entrada en ANFIS con 4 y 6 reglas difusas en el Pozo Control	149
B.6	RMSE utilizando dos Registros de Entrada en ANFIS con 8 y 9 reglas difusas en el Pozo Control	150
B.7	RMSE utilizando tres Registros de Entrada en ANFIS con 4 reglas difusas en el Pozo de Entrenamiento	151
B.8	RMSE utilizando tres Registros de Entrada en ANFIS con 6 reglas difusas en el Pozo de Entrenamiento	151
B.9	RMSE utilizando tres Registros de Entrada en ANFIS con 8 reglas difusas en el Pozo de Entrenamiento	152
B.10	RMSE utilizando tres Registros de Entrada en ANFIS con 9 reglas difusas en el Pozo de Entrenamiento	152
B.11	RMSE utilizando tres Registros de Entrada en ANFIS con 4 reglas difusas en el Pozo Control	152
B.12	RMSE utilizando tres Registros de Entrada en ANFIS con 6 reglas difusas en el Pozo Control	153
B.13	RMSE utilizando tres Registros de Entrada en ANFIS con 8 reglas difusas en el Pozo Control	153

B.14 RMSE utilizando tres Registros de Entrada en ANFIS con 9 reglas difusas en el Pozo Control	153
B.15 Parámetros “ a_1 ”, “ a_2 ” y “ a_3 ” de la Ecuación 1 AG utilizando tres Registros de Entrada	154
B.16 Parámetros “ $a_1 - b_1$ ”, “ $a_2 - b_2$ ” y “ $a_3 - b_3$ ” de la Ecuación 2 AG empleando tres Registros de Entrada	155
B.17 RMSE para tres Registros de Entrada con Algoritmos Genéticos.	155
B.18 R^2 para tres Registros de Entrada con Algoritmos Genéticos.	156
B.19 RMSE para tres Registros de Entrada utilizando ANFIS-Híbrido (6 reglas difusas) en el Pozo de Entrenamiento.	157
B.20 R^2 para tres Registros de Entrada utilizando ANFIS-Híbrido (6 reglas difusas).	157
B.21 Combinación de Reglas Difusas generadas a través de la técnica “Grid Partition” para tres Registros de Entrada: VP-DEN-POR	158
B.22 Parámetros σ y c de la función de membresía <i>Gaussiana</i> generados en el SID mediante la técnica “Grid Partition”. Registros de Entrada: VP-DEN-POR	158
B.23 Parámetros “ a ”, “ b ”, “ c ” y “ d ” de las funciones de salida, utilizando “Grid Partition”. Registros de Entrada: VP-DEN-POR	158
B.24 Parámetros “ a ”, “ b ”, “ c ” y “ d ” óptimizados de las funciones de salida, utilizando “Grid Partition”. Registros de Entrada: VP-DEN-POR	159
B.25 RMSE para tres Registros de Entrada empleando ANFIS-Híbrido (reglas difusas variables).	160
B.26 R^2 para tres Registros de Entrada utilizando ANFIS-Híbrido (reglas difusas variables) en el Pozo de Entrenamiento.	160
B.27 Combinación de Reglas Difusas generadas a través de la técnica “Substractive Clustering” para tres Registros de Entrada: VP-DEN-POR	161
B.28 Parámetros σ y c de las funciones de membresía <i>Gaussianas</i> generadas en el SID mediante la técnica “Substractive Clustering”. Registros de Entrada: VP-DEN-POR.	161
B.29 Parámetros “ a ”, “ b ”, “ c ” y “ d ” de las funciones de salida, utilizando “Substractive Clustering”. Registros de Entrada: VP-DEN-POR	161
B.30 Parámetros “ a ”, “ b ”, “ c ” y “ d ” optimizados de las funciones de salida, utilizando “Substractive Clustering”. Registro de Entrada: VP-DEN-POR	162

ÍNDICE DE FIGURAS

1.1	Modelo de una Neurona Artificial	7
1.2	Función de pertenencia de los Días del Fin de Semana.	11
1.3	Función Gaussiana de Membresía	13
2.1	Ubicación Geográfica del área de estudio. (Fotografías tomadas utilizando el programa <i>ArcGis</i>).	34
2.2	Cuencas Petrolíferas de Venezuela, basadas en la distribución de sus Provincias Sedimentarias	35
2.3	Sección NO-SE de la Cuenca Barinas-Apure	36
2.4	Rasgos estructurales de la Cuenca Barinas-Apure	37
2.5	Corte Estructural del Campo Guafita	40
2.6	Mapa Estructural del Campo Guafita (Fm. Guafita)	41
2.7	Columna Litoestratigráfica del Campo Guafita	42
3.1	Registros del Pozo GF-87N	48
3.2	Registros del Pozo GF-109N	49
3.3	Registro Vsh original (izquierda) y Registro Vsh calculado el “ \log_{10} ” (derecha) del Pozo GF-109N.	52
3.4	Archivo de extensión .FIS generado por ANFIS para 2 Registros de Entrada: Densidad-VP	64
3.5	Primera parte del flujoograma del programa final	67
3.6	Segunda parte del flujoograma del programa final	68
4.1	Variación del RMSE en las pruebas realizadas con el registro de entrada: Densidad	72
4.2	Predicciones realizadas, en el Pozo de Entrenamiento, modificando las condiciones iniciales de los Algoritmos Genéticos. Registro de Entrada: Densidad	73
4.3	Variación del RMSE en las pruebas realizadas con el registro de entrada: Velocidad de Onda P	74

4.4	Predicciones realizadas, en el Pozo de Entrenamiento, modificando las condiciones iniciales de los Algoritmos Genéticos. Registro de Entrada: Velocidad de Onda P	75
4.5	Valores Predichos vs. Valores Reales de Velocidad de Onda S en las pruebas realizadas modificando las condiciones iniciales de Algoritmos Genéticos	76
4.6	Variación del RMSE utilizando 4, 6, 8 y 9 reglas difusas en cada entrada de un Registro de Pozo.	79
4.7	Variación del RMSE para 6 reglas difusas (combinaciones [3 2] y [2 3]) a medida que cambian las entradas de dos Registros de Pozo.	80
4.8	Variación del RMSE para 8 reglas difusas (combinaciones [4 2] y [2 4]) a medida que cambian las entradas de dos Registros de Pozo.	81
4.9	Variación del RMSE utilizando 4, 6 (combinación [2 3]), 8 (combinación [2 4]) y 9 reglas difusas en cada entrada de dos Registros de Pozo.	83
4.10	Variación del RMSE para 4 reglas difusas (combinaciones [1 2 2]-[2 1 2]-[2 2 1]) a medida que cambian las entradas de tres Registros de Pozo.	84
4.11	Variación del RMSE para 6 reglas difusas (combinaciones [1 3 2]-[2 1 3]-[3 2 1]) a medida que cambian las entradas de tres Registros de Pozo.	85
4.12	Variación del RMSE para 8 reglas difusas (combinaciones [1 4 2]-[2 1 4]-[4 2 1]) a medida que cambian las entradas de tres Registros de Pozo.	86
4.13	Variación del RMSE para 9 reglas difusas (combinaciones [1 3 3]-[3 1 3]-[3 3 1]) a medida que cambian las entradas de tres Registros de Pozo.	86
4.14	Variación del RMSE con 4 (combinación [2 2 1]), 6 (combinación [2 1 3]), 8 (combinación [2 1 4]) y 9 (combinación [3 3 1]) reglas difusas en cada entrada de tres Registros de Pozo.	87
4.15	Registros de Velocidad de Onda S: Predicho y Real, usando Algoritmos Genéticos en el Pozo Control. Registro de Entrada: VP	91
4.16	Valores Predichos vs. Valores Reales de Velocidad de Onda S usando Algoritmos Genéticos en el Pozo Control. Registro de Entrada: VP	92
4.17	Registros de Velocidad de Onda S: Predicho y Real utilizando ANFIS-Híbrido (técnica “Grid Partition”) en el Pozo Control. Entrada: Registro de Velocidad de Onda P.	97
4.18	Valores Predichos vs. Valores Reales de Velocidad de Onda S utilizando ANFIS-Híbrido (técnica “Grid Partition”) en el Pozo Control. Entrada: Registro de Velocidad de Onda P	98
4.19	Registros de Velocidad de Onda S: Predicho y Real utilizando ANFIS-Híbrido (técnica “Subtractive Clustering”) en el Pozo Control. Entrada: DEN.	103

4.20 Valores Predichos vs. Valores Reales de Velocidad de Onda S utilizando ANFIS-Híbrido (técnica “Subtractive Clustering”) en el Pozo Control. Entrada: DEN.	104
4.21 Registros Velocidad de Onda S: Predicho y Real en el Pozo Control. Registros de Entrada: SW-Vsh	111
4.22 Registros de Velocidad de Onda S: Predicho y Real utilizando Algoritmos Genéticos en el Pozo Control. Entrada: DEN-VP.	112
4.23 Valores Predichos vs. Valores Reales de Velocidad de Onda S utilizando Algoritmos Genéticos en el Pozo Control. Entrada: DEN-VP	113
4.24 Registros de Velocidad de Onda S: Predicho y Real utilizando ANFIS-Híbrido (técnica “Grid Partition”) en el Pozo Control. Entrada: VP-POR	119
4.25 Valores Predichos vs. Valores Reales de Velocidad de Onda S utilizando ANFIS-Híbrido (técnica “Grid Partition”) en el Pozo Control. Entrada: VP-POR	120
4.26 Registros de Velocidad de Onda S: Predicho y Real utilizando ANFIS-Híbrido (técnica “Subtractive Clustering”) en el Pozo Control. Entrada: DEN-VP	125
4.27 Valores Predichos vs. Valores Reales de Velocidad de Onda S utilizando ANFIS-Híbrido (técnica “Subtractive Clustering”) en el Pozo Control. Entrada: DEN-VP	126
4.28 Registros de Velocidad de Onda S: Predicho y Real utilizando Algoritmos Genéticos en el Pozo Control. Entrada: VP-DEN-POR.	128
4.29 Valores Predichos vs. Valores Reales de Velocidad de Onda S utilizando Algoritmos Genéticos en el Pozo Control. Entrada: VP-DEN-POR.	129
4.30 Registros de Velocidad de Onda S: Predicho y Real utilizando ANFIS-Híbrido (técnica “Grid Partition”) en el Pozo Control. Registros de Entrada: VP-DEN-POR	130
4.31 Valores Predichos vs. Valores Reales de Velocidad de Onda S utilizando ANFIS-Híbrido (técnica “Grid Partition”) en el Pozo Control. Registros de Entrada: VP-DEN-POR	131
4.32 Registros de Velocidad de Onda S: Predicho y Real utilizando ANFIS-Híbrido (técnica “Subtractive Clustering”) en el Pozo Control. Registros de Entrada: VP-DEN-POR	132
4.33 Valores Predichos vs. Valores Reales de Velocidad de Onda S utilizando ANFIS-Híbrido (técnica “Subtractive Clustering”) en el Pozo Control. Registros de Entrada: VP-DEN-POR	133
A.1 Funciones Lineales de Membresía	141
A.2 Funciones de Distribución Gaussiana de Membresía	142

A.3	Curvas Sigmoidales de Membresía	144
A.4	Curvas Polinomiales de Membresía	146

INTRODUCCIÓN

En la actualidad, la medición de la Velocidad de Onda S reviste especial importancia en el área de las Geociencias. El entendimiento de las propiedades de un yacimiento, el **Registro Sónico de Onda S (y su recíproco Velocidad)**, es utilizado, entre otros aspectos, en la interpretación sísmica, en el análisis de esfuerzos en las rocas y para determinar la distribución de las fracturas; en el análisis AVO, por ejemplo, relaciones como el radio de Poisson, proveen información acerca de fracturas, permeabilidad, litología, porosidad y contenido de fluido (Cuddy & Glover, 2002).

Cabe destacar que, desde los últimos 50 años, están disponibles herramientas para la medición de registros de la *Onda Compresional (P)*. Solamente es desde los últimos 10 años, aproximadamente, que se ha desarrollado una nueva generación de herramientas más sofisticadas para la medición, entre otras, de la Onda S. Sin embargo, debido a su reciente implementación y a los costos elevados en su adquisición, frecuentemente puede encontrarse zonas en donde esta información es limitada o nula (Rider, 1996). En vista de esta problemática, adquieren gran valor los distintos métodos que existan para su predicción, bien sea utilizando registros del mismo pozo o con datos de pozos cercanos.

Actualmente, técnicas como la **Inteligencia Artificial** son muy utilizadas en diversas áreas. Ésta, intentando emular la Inteligencia Humana, es un método innovador para construir sistemas inteligentes que buscan soluciones aceptables, robustas y de bajo costo tolerando lo máximo posible las imprecisiones, incertidumbres y aproximaciones sobre el problema a resolver. Sus principales campos constituyentes son: la lógica difusa, la computación neuronal (redes neuronales), la computación evolutiva (algoritmos genéticos), el aprendizaje de las máquinas y el razonamiento probabilístico, con las más recientes incorporaciones de la Teoría del Caos (Uzin, Echanobe, & Del Campo, 1999).

Introducida por primera vez por Lofti Zadeh en 1965, la “**Lógica Difusa**” es sinónima de la “Teoría de Conjuntos Difusos”, donde a diferencia de la lógica clásica, las variables están relacionados mediante límites no estrictamente establecidos (difusos) y los elementos de un conjunto pertenecen a él parcialmente mediante un grado de membresía (MathWorks,

2010a). Con base a esta flexibilidad, los sistemas difusos han sido utilizados para modelar procesos complejos, utilizando reglas “si entonces” (if-then) que representan las entradas en las salidas. Existen dos tipos de “sistemas de inferencia difusa” (Fuzzy Interference Systems): TIPO MAMDANI y TIPO TAKAGI-SUGENO-KANG (TSK).

Los **Algoritmos Genéticos**, son algoritmos de búsqueda que se basan en intentar imitar la evolución natural para buscar la mejor solución de un problema. El año 1975 se toma como punto de referencia de su origen y, desde entonces, han ido mejorando y han sido utilizados para resolver problemas de los ámbitos más diversos. Éstos operan sobre un grupo de posibles soluciones al que se le llama población y que inicialmente es creado aleatoriamente. A cada una de estas posibles soluciones se le llama individuo o cromosoma y se representa como una cadena de números, habitualmente bits (0 y 1). Lógicamente, la población que es generada al principio aleatoriamente será una población de malas soluciones, pero precisamente se trata de que la población vaya mejorando por medio de la evolución natural hasta conseguir buenas soluciones (Uzin et al., 1999).

Ambos métodos han sido aplicados para realizar predicciones de distintos datos geofísicos. Por ejemplo, en (Banchs, Jiménez & Del Pino, 2001) utilizan una función polinómica, para, mediante algoritmos genéticos, estimar registros de Potencial Espontáneo (SP) a partir de registro de Rayos Gamma. Asimismo, en (Dorrington & Link, 2007) utilizan las redes neuronales y los algoritmos genéticos para escoger el tipo de atributos sísmicos en la predicción de Porosidad.

De igual forma, en (Rezaee, Kadkhodaie & Barabadi, 2007) y (Gutiérrez, 2010) utilizan lógica difusa y sistemas neuro-difusos para la predicción de Velocidad de Onda S a partir de otros registros de pozo (como Rayos Gamma, Velocidad de Onda P y Resistividad). Finalmente, en (Saemi & Ahmadi, 2008) utilizan un algoritmo híbrido entre sistemas neuro-difusos y algoritmos genéticos en la predicción de permeabilidad.

Precisamente, es importante remarcar que la Inteligencia Artificial no es una disputa de eficacia entre todos estos campos. Al contrario, se trata de combinar las distintas metodologías para conseguir el sistema más cercano posible que imite la inteligencia humana en una tarea específica.

Es por esto que, siguiendo esta misma línea, el objetivo principal que en este trabajo se propone es, utilizando las librerías del programa **Matlab** en su versión **R2009a**, realizar la predicción de registros de Velocidad de Onda S. En primer lugar, mediante sistemas neuro-difusos y algoritmos genéticos, por separado, y luego proponer una metodología que hibridice ambos métodos en la búsqueda de una mejor estimación de este registro. Esto a partir de

los registros disponibles (Densidad, Velocidad de Onda P, Saturación de Agua, Volumen de Arcilla y Porosidad) en los pozos GF-87N y GF-109N del Campo Guafita, ubicado en la subcuenca Barinas – Apure, al Oeste del estado Apure.

Finalmente, la estructura del trabajo será de la siguiente manera: en el CAPÍTULO I se explicarán los fundamentos teóricos sobre los que se desarrolla la presente investigación; éstas son: los conceptos de la Lógica Difusa y algoritmos genéticos y su importancia en las aplicaciones en las Geociencias. En el CAPÍTULO II se establecerá el marco geológico de la zona de estudio, así como la ubicación de los pozos de estudio (GF-87N y GF-109N); también se presenta la columna estratigráfica de la Fm. Guafita. Seguidamente, en el CAPÍTULO III se presentará la metodología aplicada para la elaboración de los correspondientes algoritmos. En el CAPÍTULO IV, se analizarán los resultados obtenidos en la predicción del Registro de Velocidad de Onda S. Para culminar, se enumerarán las conclusiones a las que se llegó a partir de los análisis realizados.

CAPÍTULO 1

MARCO TEÓRICO

Existe una gran diversidad de problemas, que van desde lo más cotidiano a lo más específico, para los que la inteligencia humana es mucho más rápida y eficiente, en su resolución, que el procesamiento de la mejor computadora actual. Por lo que desde hace varias décadas, han surgido diferentes técnicas computacionales en un intento de atacar ésta deficiencia, como por ejemplo:

- Redes Neuronales Artificiales (en inglés “*Artificial Neural Networks (ANN)*”).
- Lógica Difusa (en inglés “*Fuzzy Logic (FL)*”).
- Algoritmos Genéticos (en inglés “*Genetic Algorithm (GA)*”)

Éstas y otras herramientas se suelen agrupar bajo el concepto de Inteligencia Artificial (en inglés *Artificial Intelligence (AI)*). A medida que han ido creciendo, se han caracterizado mejor las propiedades de cada una y sus áreas de aplicación. De manera que, entre ellas, **han llegado a desarrollarse arquitecturas combinadas o híbridas** para ampliar la clase de problemas que cada una puede tratar por sí misma y mejorar la solución global encontrada.

Debido a que son utilizadas en la presente investigación, a continuación se expondrán la base teórica, ventajas y aplicaciones en la Geociencias en el siguiente orden:

- En la sección 1.1 se encuentran los principales conceptos de las **Redes Neuronales Artificiales (RNA)**.
- En la sección 1.2, los fundamentos de la **Lógica Difusa**, con una breve descripción de los Sistemas de Inferencia Mamdani y Takagi-Sugeno-Khan.

- En la sección 1.3 se describirán, a manera general, los principales elementos de los **Algoritmos Genéticos**, así como su utilidad en la optimización de funciones.

Finalmente, en la sección 1.4 se describen algoritmos híbridos originados a partir de éstas tres técnicas.

1.1. Redes Neuronales

Para hablar de redes neuronales artificiales primero se debe entender el funcionamiento del cerebro humano desde las partes que conforman una neurona hasta el procedimiento que se cumple para la activación de una de ellas, ya que, en base a este proceso, se comenzaron a desarrollar las investigaciones sobre las Redes Neuronales Artificiales (RNA), desde alrededor de 1940 hasta la actualidad (Salas, 2004).

1.1.1. Funcionamiento del Cerebro Humano

El cerebro humano continuamente recibe señales de entrada de muchas fuentes y las procesa (a través de las neuronas) a manera de crear una apropiada respuesta de salida. Nuestros cerebros cuentan con millones de ellas que se interconectan para elaborar **“Redes Neuronales”**. Estas redes ejecutan los millones de instrucciones necesarias para mantener nuestra vida cotidiana (Montaño, 2002).

Las NEURONAS, al igual que las demás células del cuerpo, funcionan a través de impulsos eléctricos y reacciones químicas. Los impulsos eléctricos que utiliza para intercambiar información, viajan por el axón que hace contacto con las dendritas de una o varias neuronas diferentes un proceso denominado “sinápsis”. El contacto entre ellas no se realiza físicamente, es decir, por yuxtaposición (pues siempre hay un espacio libre entre las dentritas de dos células contiguas), sin embargo, la distancia es tan pequeña que permite el paso del impulso nervioso de una a otra mediante la libreación de sustancias transmisoras (Montaño, 2002).

En la conducción nerviosa se da el principio de **“todo o nada”**, es decir, la velocidad del impulso nervioso no depende de la intensidad del estímulo que lo produce; si el estímulo tiene energía suficiente, el impulso se transmite completo, si la energía es insuficiente, no se efectúa la transmisión del impulso. De manera que la neurona manda el impulso por su axón si la excitación excede su inhibición por un valor crítico, esto es, el umbral de excitación de la neurona.

En líneas generales, todo el sistema descrito anteriormente es el proceso que realiza el cerebro, funcionamiento complejo que las **Redes Neuronales Artificiales** tratan de imitar en la búsqueda de soluciones a problemas concretos del mundo real, mediante la integración y utilización de modelos de algoritmos matemáticos que simulan el funcionamiento de una neurona hasta la unión de todas estas en una sola red (Salas, 2004).

1.1.2. Definición de Red Neuronal Artificial

El cerebro actúa como un computador altamente complejo y no lineal; es decir, como un sistema de procesamiento de información. Tiene la capacidad de organizar sus constituyentes estructurales, conocidos como neuronas, para realizar ciertos cálculos (reconocimiento de patrones, percepción y control motor) muchas veces más rápido que la computadora digital más rápida que existe hoy en día (Haykin, 1999).

Teniendo esto en cuenta, una red neuronal, en general, es una máquina diseñada para modelar la forma en la cual el cerebro realiza una tarea particular o función de interés; la red es usualmente implementada usando componentes electrónicos o son simuladas por software en un computador digital. Para lograr un buen funcionamiento, las redes neuronales emplean una interconexión masiva de unidades de procesamiento denominadas “neuronas” (Haykin, 1999).

De manera que, vista como una máquina que se adapta, Montaño (2002) define una RNA como:

Un procesador masivo distribuido paralelamente, inspirados en sistemas biológicos naturales, los cuales son simulados en computadoras convencionales y que tienen la capacidad de almacenar el conocimiento experimental y hacerlo disponible para su uso.

Aunque las RNA sean mucho menos complejas que una red neuronal biológica, también realizan cálculos complejos para procesar información, semejándose a ésta fundamentalmente en dos aspectos (Haykin, 1999):

1. El conocimiento es adquirido por la red a través de un proceso de aprendizaje.
2. Las fuerzas de conexión entre las neuronas, conocido como pesos sinápticos, son utilizadas para almacenar el conocimiento.

El procedimiento utilizado para realizar el proceso de aprendizaje se denomina algoritmo de aprendizaje, el cual modifica los pesos sinápticos de la red para obtener el diseño del

objetivo deseado. La modificación de los pesos sinápticos proveen el método tradicional para el diseño de las redes neuronales (García, 2005).

1.1.3. Modelo de una Neurona

Salas (2004) indica que fueron W. S. McCulloch y W. Pitts en 1943, los primeros en concebir un modelo abstracto y simple de una neurona artificial, como **unidad de procesamiento de información** fundamental para la operación de una red neuronal. En la figura 1.1 se encuentra su esquema donde pueden identificarse sus principales partes:

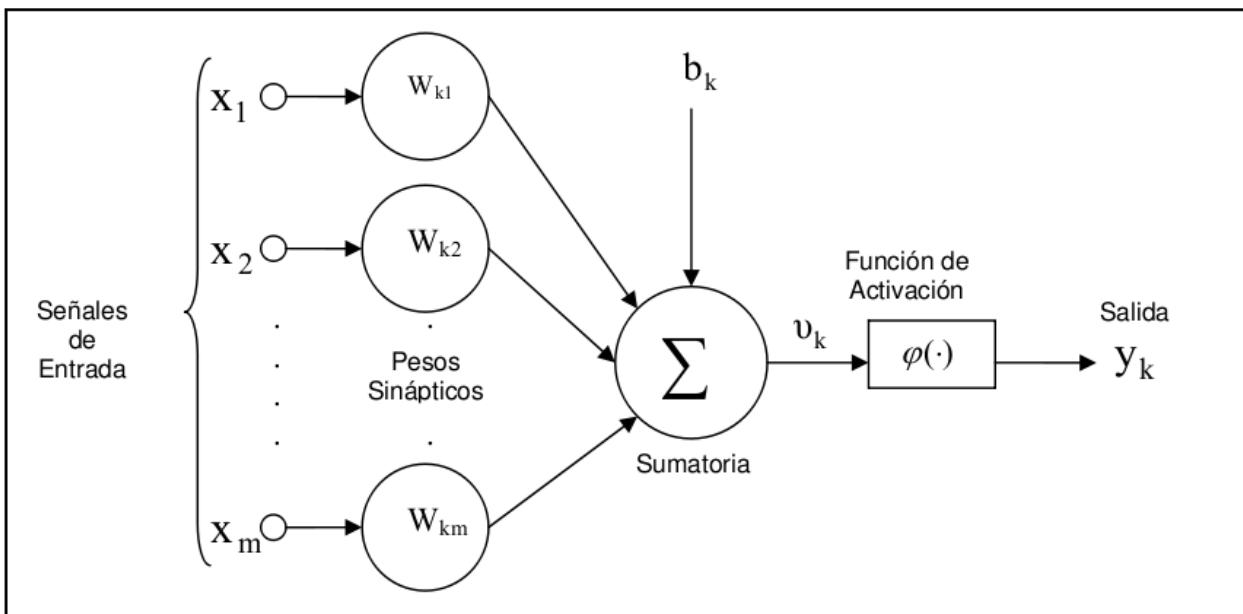


Figura 1.1: Modelo de una Neurona Artificial

- Un grupo de **sinapsis o conectores**, cada uno caracterizado por un *peso o fuerza propia*. Específicamente, una señal “ x_j ” en la entrada del conector “ j ” conectada a una neurona “ k ”, es multiplicada por el peso sináptico “ w_{kj} ”. El primer subíndice (k) está referido a la neurona en cuestión, y el segundo (j) al fin de la entrada del conector al cual se refiere el peso. A diferencia de un conector en el cerebro, el peso sináptico de una neurona artificial puede caer en un rango que incluya valores tanto positivos como negativos.
- Un **unificador** o sumador de las señales de entrada a través de los conectores, tomando en consideración los pesos de cada neurona. La operación que se describe en este punto constituye un combinador lineal.

- Una **función de activación** para limitar la amplitud de la salida de una neurona. La función de activación está también referida como la función limitante ya que lleva el rango de valores permitidos de la señal de salida a algún valor finito.

1.1.4. Ventajas

Lo importante en el desarrollo de la técnica de las RNA es su útil comportamiento al **aprender, reconocer y aplicar** relaciones entre objetos y tramas de objetos propios del mundo real. En este sentido, se utilizan las RNA como una herramienta que para resolver problemas difíciles. La posibilidad de resolverlos es debido a los principios de las redes neuronales, donde los más importantes son (Hilera & Martínez, 1995; García, 2005):

Aprendizaje adaptativo. Es quizás la característica más importante de las redes neuronales, pueden comportarse en función de un entrenamiento con una serie de ejemplos ilustrativos. De esta forma, no es necesario elaborar un modelo a priori, ni establecer funciones probabilísticas. Una red neuronal artificial es adaptativa porque puede modificarse constantemente con el fin de adaptarse a nuevas condiciones de trabajo.

No linealidad. Una neurona es básicamente un dispositivo no lineal. Consecuentemente, una red neuronal, hecha por la interconexión de neuronas, es por sí misma no lineal. Es una propiedad muy importante, particularmente si el mecanismo físico fundamental responsable de la generación de una señal de entrada es inherentemente no lineal.

Autoorganización. Mientras que el aprendizaje es un proceso donde se modifica la información interna de la red neuronal artificial, la autoorganización consiste en la modificación de la red completa con el fin de llevar a cabo un objetivo específico. De esta forma una red puede responder a datos o situaciones que no ha experimentado antes, pero que puede inferir sobre la base de su entrenamiento. Esta característica es muy útil sobre todo cuando la información de entrada es poco clara o se encuentra incompleta.

Tolerancia a fallos. En la computación tradicional la pérdida de un fragmento pequeño de información puede acarrear comúnmente la inutilización del sistema. Las redes neuronales artificiales poseen una alta capacidad de tolerancia a fallos. Se entiende por ello que las redes pueden reconocer patrones de información con ruido, distorsión o incompletos, pero que, además, pueden seguir trabajando aunque se destruya parte de la red (con cierta degradación). La explicación de este fenómeno se encuentra en

que mientras la computación tradicional almacena la información en espacios únicos, localizados y direccionables, las redes neuronales lo hacen de forma distribuida y con un alto grado de redundancia.

Operación en tiempo real. De todos los métodos existentes, las RNA son las más indicadas para el reconocimiento de patrones en tiempo real, debido a que trabajan en paralelo actualizando todas sus instancias simultáneamente.

1.1.5. Aplicaciones a las Geociencias

Como técnica computacional, tienen aplicaciones en diversos campos, incluidos las Geociencias. Sobre éstas se han realizado numerosos trabajos, entre los que se encuentran los que se muestran a continuación:

1. **Correlación de Registros de Pozo.** Alzate, Branch, Suárez & Vega (2006) presentan una herramienta computacional basada en Redes Neuronales Artificiales (RNA) para correlacionar registros entre pozos. Se hace una descripción del proceso de este proceso y sus exigencias en personal calificado y tiempo, en especial en campos de geología compleja y con un gran número de pozos. Finalmente, se presenta la herramienta de computo desarrollada y su aplicación. Los resultados obtenidos con la herramienta propuesta al aplicarla a cinco pozos, manifiestan un buen desempeño y resulta en una buena y rápida aproximación a los resultados obtenidos por un experto humano en su procedimiento visual de correlación. Esta técnica presenta buena versatilidad para correlacionar varios tipos de curvas de registros a partir de la identificación en cualquier pozo del campo y/o yacimiento en estudio de un marcador litológico previamente escogido en un pozo de referencia.
2. **Discriminación de Litologías.** García (2005) utiliza redes Neuronales Artificiales para identificar litologías como Areniscas, Lutitas, Dolomitas y Calizas en Registros Gamma Ray, Porosidad Neutrón y Porosidad por Densidad para 8 pozos del Campo Oritupano A (Venezuela).
3. **Detección de Primeras llegadas.** Para precisar las velocidades sísmicas del substituto, Bravo (2005), utiliza las Redes Neuronales como método de reconocimiento de patrones basado en las propiedades estadísticas de las muestras en consideración, en ese caso, se utilizaron los picos locales positivos como posibles indicadores de primeras llegadas.
4. **Predicción de Registros de Pozo.** Singh, Kanchan, Saigal & Verma (2004) utilizan las Redes Neuronales Artificiales en la predicción de Registros de Onda P. Silva &

Santos (2002) además comparan sus predicciones con un registro de onda P obtenido a través de una Inversión de la Traza Sísmica.

1.2. Lógica Difusa

Klir & Yuan (1995) afirman que la Lógica Difusa:

Es una metodología que proporciona una manera simple y elegante de obtener una conclusión a partir de información de entrada ambigua, intentando imitar como una persona toma decisiones basada en ciertos tipos de características.

El término “Lógica Difusa” tiene dos significados que difieren entre sí. En su sentido estricto, es un sistema lógico que constituye una extensión de la lógica clásica (*Booleana*) (MathWorks, 2010a). No obstante, en un sentido más amplio, es sinónima de la “Teoría de Conjuntos Difusos”¹, donde las variables están relacionadas mediante límites no marcados (difusos) y los elementos de un conjunto pertenecen a él parcialmente mediante un grado de membresía². Para los efectos de aplicaciones de esta teoría, generalmente se toma en cuenta el concepto más generalizado (MathWorks, 2010a).

La pertenencia o no de un elemento x en un conjunto clásico C es descrito por la función característica $\mu_C(x)$ (Rezaee et al., 2007), donde:

$$\mu_C(x) = \begin{cases} 1 & \text{si } x \in C \\ 0 & \text{en caso contrario} \end{cases} \quad (1.1)$$

Sin embargo, la teoría de conjuntos difusos redefine esta función al establecer una membresía parcial que adopta valores entre 0 y 1:

$$\mu_F(x) : X \longrightarrow [0, 1] \quad (1.2)$$

Donde X se refiere al conjunto definido por el problema y F es un conjunto difuso. Para entender claramente lo expuesto, se presenta a continuación el siguiente ejemplo, en la figura 1.2, donde las dos gráficas muestran la pertenencia de los días *Jueves, Viernes, Sábado, Domingo y Lunes* al conjunto **DÍAS DEL FIN DE SEMANA**.

¹Fue introducida por primera vez por Lofti Zadeh en 1965

²Una descripción más amplia puede encontrarse en Klir & Yuan (1995)

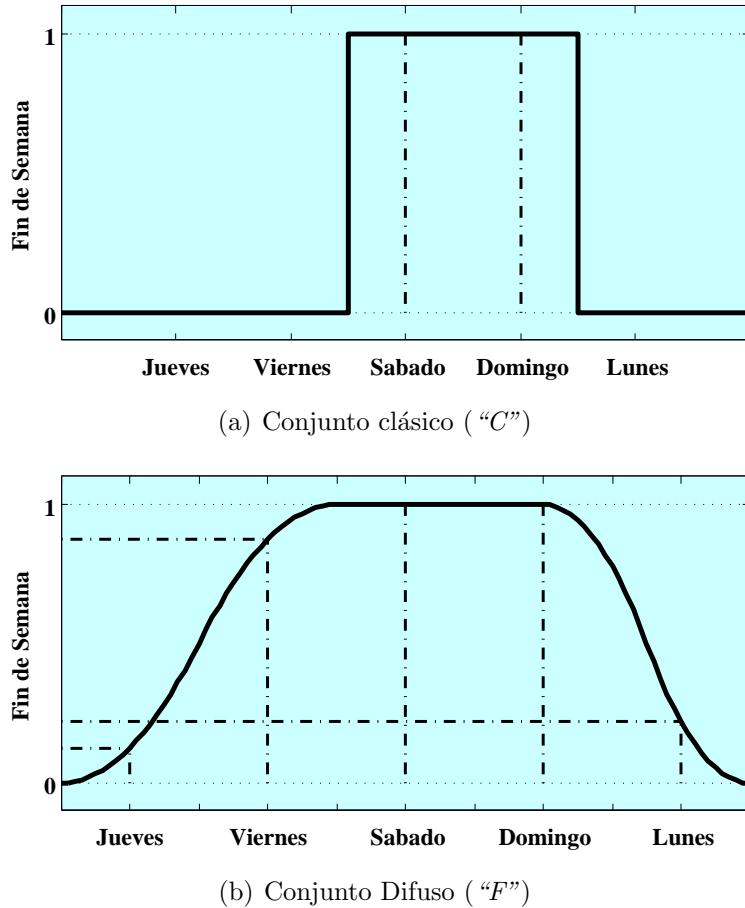


Figura 1.2: Función de pertenencia de los Días del Fin de Semana.

La **lógica clásica** nos indica que los días que pertenecen a este conjunto “ C ” serían Sábado y Domingo, siendo éstos los que alcanzan en la gráfica 1.2(a) un valor igual a uno (1), quedando los demás días *Totalmente Excluidos* de este conjunto.

Mientras que, la **lógica difusa**, muestra que el grado de membresía de cada día al conjunto difuso “ F ” DÍAS DEL FIN DE SEMANA es variable para elemento, siendo así más incluyente que en el caso anterior. Así, el *Jueves*, por ejemplo, puede ser considerado parcialmente como día de la semana con un grado de membresía de 0,1 y el *Viernes* con un grado de 0,9 (1.2(b)).

1.2.1. Fundamentos

1.2.1.1. Conjuntos Difusos

La Lógica Difusa comienza con el concepto de Conjuntos Difusos. Éste, que se define sobre un conjunto de puntos denominado “**Universo del Discurso**”, es un conjunto sin límites

claramente definidos. Puede contener elementos con sólo un parcial grado de membresía adoptando valores en el intervalo [0,1] y asigna a cada punto un valor de ese intervalo, es decir, un grado de pertenencia al conjunto difuso (MathWorks, 2010a).

Si x es un objeto en el universo y $y = C(x)$ es el valor asociado a x , se dice que el valor “ y ” es el grado de pertenencia del objeto x al conjunto difuso C . Mientras más cercano al 1 es el valor asociado a x , entonces mayor es la pertenencia (Gutiérrez, 2010). Volviendo al ejemplo dado en la sección anterior, por ejemplo el día *Viernes* pertenecería más al conjunto DÍAS DEL FIN DE SEMANA que el *Jueves*.

1.2.1.2. Función de Membresía

Es una curva que define cómo a cada punto en el espacio de entrada le es asignado un valor de membresía (o grado de membresía) entre [0,1] (MathWorks, 2010a). La única condición que realmente debe satisfacer una función de membresía es que debe variar sólo en ese rango.

Un conjunto clásico puede ser expresado de la siguiente manera:

$$A = \{x | 1 \leq x \leq 3\} \quad (1.3)$$

De donde se desprende que todos los puntos x que sean mayores o iguales a uno y menores o iguales a 3 pertenecerán al conjunto A. Por consiguiente, valores como 0,9 y 3,1 quedan totalmente excluidos de él.

El **conjunto difuso** es una ampliación de lo anterior. Si X es el universo del discurso y sus elementos denotados por x , entonces un conjunto difuso A en X es definido como un conjunto de pares ordenados:

$$A = \{x, \mu_A(x) | x \in X\} \quad (1.4)$$

donde $\mu_A(x)$ es llamada “**Función de Membresía**” de x en A .

En conclusión, para calcular entonces ese grado de “pertenencia” o de “membresía” de x en el conjunto “ A ”, basta con evaluar dicho punto en la función $\mu_A(x)$.

En otro orden de ideas, el programa **Matlab** versión **R2009a** incluye, en su librería de Lógica Difusa, 11 funciones de membresía que pueden agruparse en cuatro grandes grupos (MathWorks, 2010a). A continuación se muestran con una breve descripción ³:

³Sus expresiones matemáticas, así como las gráficas de cada función, se muestran en el Apéndice A

1. Funciones lineales. Son las más sencillas de utilizar, debido a su simplicidad en cuanto a sus formas y expresiones matemáticas:

- **trimf.** Función Triangular. Son tres puntos que forman un triángulo.
- **trapmf.** Función Trapezoidal. Consiste en cuatro puntos que le dan forma de triángulo truncado.

2. Función de Distribución Gaussiana. Debido a la suavidad de las curvas, así como su notación concisa, este tipo de funciones son las más utilizadas para representar conjuntos difusos. Tienen la ventaja de ser distintas de cero para todos los puntos⁴:

- **gaussmf.** Función Gaussiana. Ésta es la función que se utilizará en la presente investigación. Debido a esto, es a continuación se muestra tanto su gráfica característica (Ver figura 1.3) como la ecuación que la define:

$$f(x) = \exp\left(\frac{-0.5(x - c)^2}{\sigma^2}\right) \quad (1.5)$$

donde:

c = **valor medio** y σ = **varianza**. En el ejemplo de la figura 1.3 los valores de estos parámetros son: $[\sigma, c] = [2, 5]$

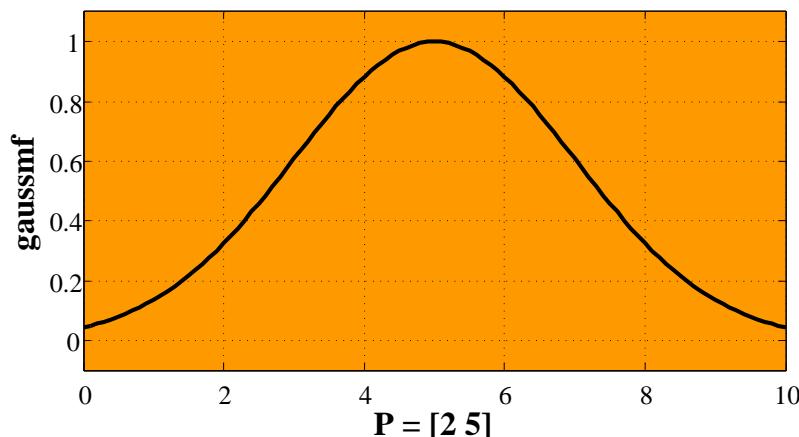


Figura 1.3: Función Gaussiana de Membresía

- **gauss2mf.** Función que consiste en una combinación de dos gaussianas.
- **gbellmf.** Función Campana General (en inglés “Generalized Bell”). Tiene un parámetro más que la función gaussiana, puede utilizarse para conjuntos no-difusos.

⁴Este hecho adquiere especial relevancia en la realización de este trabajo (Véase la sección 3.6, página 66)

3. Curvas sigmoidales. Se diferencian de las Gaussianas debido a que éstas son asimétricas:

- **sigmf.** Función sigmoidal básica, la cual abre hacia la izquierda o hacia la derecha.
- **psigmf.** Se genera por el producto de dos funciones sigmoidales.
- **dsigmf.** Función que representa la diferencia entre dos funciones sigmoidales.

4. Curvas polinomiales. Este tipo de funciones son también frecuentemente utilizadas en la lógica difusa. Son denominadas de acuerdo a la forma de su gráfica:

- **zmf.** Curva asimétrica en forma de “Z” que abre hacia la izquierda.
- **smf.** Curva asimétrica en forma de “S” que abre hacia la derecha.
- **pimf.** Curva que se hace cero hacia sus extremos con una elevación en su centro (similar a la forma de la letra griega “π”).

1.2.1.3. Operadores Lógicos

Si se utilizan sólo los extremos que puede adoptar una función de membresía, 1 (completamente verdadero) y 0 (completamente falso), se obtienen los valores estándares de la lógica Booleana y las operaciones AND-OR-NOT que para dar un ejemplo, se muestran en la siguientes tablas de la verdad:

Tabla 1.1: Tabla de la verdad con las operaciones AND, OR, NOT

A	B	C=A and B	A	B	C=A or B	A	C=not A
1	0	0	1	0	1	0	1
1	1	1	0	1	1	1	0
0	0	0	1	1	1	0	1
0	1	0	0	0	0	1	0

AND OR NOT

La primera fila en la tabla de la izquierda (operador AND) se lee de la siguiente manera: “*Si A es verdadero y B es falso, entonces C es falso*”. Mientras que, en caso de la operación OR (tabla central) es “*Si A es verdadero o B es falso, entonces C es verdadero*”. Por último, en el caso de la fila 1 del ejemplo del operador NOT: “*Si A es falso entonces C es verdadero*”.

Sin embargo, en la lógica difusa, la membresía de elemento a un conjunto determinado, como ya se ha descrito, no sólo es cero y uno, sino que varía adoptando cualquier número real

en ese rango. Por lo tanto, surge la necesidad de buscar funciones que arrojen los mismos resultados de “C” que la tabla anterior y que además extienda “A” y “B” a todos los números entre $[0,1]$ (MathWorks, 2010a).

Éstas son las operaciones **min** (mínimo), **max** (máximo) y **1-A** (complemento estándar). Por ejemplo, al aplicar estas funciones a los valores “A” y “B” de la tabla 1.1, se obtiene:

Tabla 1.2: Operaciones lógicas *min*, *max*, *complemento*

A	B	C = min(A,B)	A	B	C = max(A,B)	A	C = 1 - A
1	0	0	1	0	1	0	1
1	1	1	0	1	1	1	0
0	0	0	1	1	1	0	1
0	1	0	0	0	0	1	0

Al comparar las tablas 1.1 y 1.2, es evidente comprobar (ya que no cambia el valor de “C”) que la sentencia **A AND B** es equivalente a aplicar la función **min(A,B)** (mínimo entre “A” y “B”), mientras que **A OR B** es igual a **max(A,B)** (máximo entre “A” y “B”) y **NOT A** a **1-A**, lo que hace posible considerar otros valores reales que no sean 0 y 1 (pero que sí se encuentre en ese rango) (MathWorks, 2010a).

Cabe destacar que éstas no son las únicas relaciones con las operaciones lógicas AND, OR y NOT. En términos generales, en la lógica difusa se definen (Klir & Yuan, 1995):

Complemento Difuso (NOT). Dado “ A ” un conjunto difuso en “ X ”, por definición $\mu_A(x)$ es interpretado como el *grado en el cual “ x ” pertenece a “ A ”*. Dado ahora “ cA ” que denote el complemento difuso de “ A ” de tipo “ c ”, entonces $\mu_{cA}(x)$ puede ser interpretado no sólo como el grado en el que “ x ” pertenece a “ cA ”, sino que además el *grado en el cual “ x ” no pertenece a “ A ”*. Igualmente, $\mu_A(x)$ puede ser además interpretado como el grado en el cual “ x ” no pertenece a “ cA ”.

Como notación convencional, el complemento “ cA ” puede ser definido por la función:

$$c : [0, 1] \rightarrow [0, 1]$$

el cual asigna valores a $\mu_{cA}(x)$ de cada grado de membresía de $\mu_A(x)$ en cualquier conjunto difuso “ A ”. El valor $\mu_{cA}(x)$ es calculado como:

$$\mu_{cA}(x) = c(\mu_A(x)) \quad (1.6)$$

para todo $x \in X$. De esta ecuación se desprende que, dado un conjunto difuso “ A ”, se puede obtener “ cA ” aplicando la función “ c ” a los valores de “ $\mu_A(x)$ ”. En el programa **Matlab** éste se calcula con la sustracción $1 - \mu_A(x)$.

Intersección Difusa: T-Norms (AND). La intersección de dos conjuntos difusos “ A ” y “ B ” es especificado, en general, por una operación binaria en un intervalo unitario, ésta es una función de la forma:

$$T : [0, 1] \times [0, 1] \rightarrow [0, 1]$$

Para cada elemento “ x ” del universo del discurso, ésta función toma como argumento sus grados de las funciones de membresía en los conjuntos “ A ” y “ B ”, y produce el grado de membresía del elemento en el conjunto constituido por la intersección de “ A ” y “ B ”. Entonces queda:

$$\mu_{A \cap B}(x) = T(\mu_A(x), \mu_B(x)) \quad (1.7)$$

El operador “ T ” (usualmente llamado T-norm (*Triangular norm=Norma Triangular*)) puede representar distintas funciones, la más sencilla es la multiplicación entre $\mu_A(x)$ y $\mu_B(x)$.

Unión Difusa: S-Norms (OR). La discusión de las uniones difusas es paralelo al de la intersección. Como en éste último, la unión de dos conjuntos difusos “ A ” y “ B ” es especificado por la función:

$$S : [0, 1] \times [0, 1] \rightarrow [0, 1]$$

El argumento de esta función es el grado de membresía de algún elemento “ x ” en el conjunto difuso “ A ” y el grado de membresía del mismo elemento en “ B ”. La función devuelve el grado de membresía del elemento en el conjunto $A \cup B$ quedando:

$$\mu_{A \cup B}(x) = S(\mu_A(x), \mu_B(x)) \quad (1.8)$$

Según Klir & Yuan (1995) el operador “ S ” (también llamado T-Conorms (*Triangular conorm=Conorma Triangular*)) puede representar diversas funciones, sin embargo, la utilizada en el programa **Matlab** es la suma entre $\mu_A(x)$ y $\mu_B(x)$.

1.2.1.4. Reglas Difusas “Si ... Entonces”

López (2005) citada por Gutiérrez (2010) afirma que:

Las reglas difusas son un modo de representar estrategias o técnicas apropiadas cuando el conocimiento proviene de la experiencia o la intuición (careciendo de demostración matemática o física), es decir, las reglas utilizan variables lingüísticas como vocabulario.

Una regla difusa Si-Entonces asume la forma:

SI x_1 es A_1 “y” x_2 es A_2 ... “y” x_n es A_n **ENTONCES** v es B

donde ésta tiene varias partes en su estructura:

- “ $x-v$ ” son variables cuyos valores pertenecen a los conjuntos “ $A - B$ ”.
- “ $A - B$ ” son los valores lingüísticos definidos por los conjuntos difusos (a través de las funciones de membresía) en los rangos (Universos del Discurso) “ $X - V$ ” respectivamente.
- El término “ y ” se refiere a los operadores lógicos AND, OR, NOT.

Las reglas difusas combinan uno o más conjuntos difusos de entrada denominada **antecedente o premisa** (resaltados en color rojo en la sentencia anterior) y le asocian un conjunto difuso de salida llamado **consecuente o conclusión** (en color azul).

En relación al antecedente, las partes “ x_1 es A_1 ”, “ x_2 es A_2 ”, “ x_n es A_n ” representan números comprendidos entre [0,1] midiendo la pertenencia en que los puntos “ x_1 ”, “ x_2 ”, “ x_n ” a los conjuntos “ A_1 ”, “ A_2 ”, “ A_n ” respectivamente.

Mientras que, en cuanto al cálculo del consecuente, hay dos tipos de formato, *tipo Mamdani*, en donde al consecuente le asocian un conjunto difuso de otro universo de discurso y de *tipo Sugeno*, en donde la función de salida es una combinación lineal de las variables de entrada o una función genérica de las mismas.

Por otra parte, una de las incógnitas en la implementación de la lógica difusa es el número de reglas difusas. Existen así dos caminos para obtener el conjunto de reglas correspondiente a un conjunto de datos numéricos ⁵:

1. Dejar que los datos establezcan los conjuntos difusos que aparecen en los antecedentes y consecuentes.

⁵Se amplía un poco más de este tema en la sección 1.2.2

2. Predefinir los conjuntos difusos para antecedentes y consecuentes y luego asociar los datos a esos conjuntos.

Los sistemas borrosos descritos con varias entradas y una sola salida se conocen como **MISO (Multiple Input Single Output)**, mientras que los que también tienen varias salidas se llaman **MIMO (Multiple Input Multiple Output)** (Klir & Yuan, 1995).

Finalmente, la interpretación de las reglas difusas consta de un proceso de tres partes (MathWorks, 2010a):

Fusificación de las entradas. Resuelve todas las sentencias difusas en el antecedente a un grado de pertenencia entre 0 y 1.

Aplicación del operador difuso a las múltiples partes antecedentes. Si hay varias partes en el antecedente, se aplican los operadores lógicos difusos para convertir el antecedente en un solo número entre 0 y 1. Este es el grado de apoyo para la regla.

Aplicación del método de implicación. Se utiliza el grado de apoyo de toda la regla para modelar el grupo difuso de salida. El consecuente de una regla difusa, asigna un grupo difuso completo a la salida. Este conjunto es representado por una función de pertenencia que es escogida para indicar las características del consecuente.

De forma general, una sola regla difusa no es efectiva. Se requieren de dos o más interactúen entre sí para obtener mejores resultados. La salida de cada regla es un grupo difuso, pero en general se desea que la salida para un conjunto de reglas sea un número simple. Estos grupos difusos de salida serán luego agregados en un solo grupo y luego desfusificados.

1.2.2. Sistemas de Inferencia Difusa

El Sistema de Inferencia Difusa (SID), en inglés *Fuzzy Inference System (FIS)*, es el proceso de formular una relación entre un conjunto determinado de entrada y una salida utilizando lógica difusa; ésta provee una base en la cual pueden tomarse decisiones, o discernirse patrones e involucra todas las piezas descritas en la sección 1.2.1. Pueden implementarse dos tipos de inferencia difusa, que varían en cómo son determinadas las salidas (MathWorks, 2010a):

- Mamdani.
- Takagi-Sugeno-Khan.

1.2.2.1. Mamdani

El Sistema de Inferencia Difusa tipo Mamdani es comúnmente el más utilizado en la lógica difusa. Propuesto en 1975 por EBRAHIM MAMDANI, se convirtió entre los primeros sistemas de control construidos utilizando la teoría de conjuntos difusos, en un intento de obtener reglas a partir de la experiencia humana. Sus esfuerzos estuvieron basados en el trabajo de Lofti Zadeh de 1973: “*Algoritmos Difusos para sistemas complejos y procesos de decisión*” (MathWorks, 2010a).

La Inferencia Difusa Tipo Mamdani, que se centra en la interpretabilidad, supone las salidas de las funciones de membresía son conjuntos difusos. Después del proceso de agregación, hay un conjunto difuso para cada variable de salida que debe desfusificarse, esto es, asociar al conjunto difuso un sencillo número a través de varios métodos. Es posible, y en muchos casos es mucho más eficiente, usar una sola variable de salida. En MathWorks (2010a) se hace una descripción detallada del funcionamiento de este modelo.

1.2.2.2. Takagi-Sugeno-Khan

En ingeniería en general, el modelo de Takagi-Sugeno-Khan también se ha convertido en un método de razonamiento lógico muy utilizado. Este modelo se caracteriza por relaciones basadas en reglas difusas, donde las premisas de cada regla representan subespacios difusos y las consecuencias son una relación lineal de entrada-salida (López, 2005).

Dado que se utilizará este tipo de sistema en la presente investigación, se dará a continuación una descripción más detallada de su funcionamiento. Por ejemplo, una típica regla difusa de dos entradas tiene la forma:

SI entrada1=x y entrada2=v , **ENTONCES** salida es $z = ax + by + c$

Como puede observarse, la salida “ z ” está escrita como combinación **lineal** de las entradas “ $x-v$ ” mediante los términos “ a, b, c ”. Cuando $a=b=0$, se obtiene un modelo Sugeno de orden cero, es decir, “ z ” es **constante**. Por lo tanto, al construir un Sistema de Inferencia de este tipo, debe especificarse cuál de las dos opciones se quiere utilizar.

El proceso de Inferencia Difusa mediante este método se realiza, en general, como sigue:

- 1. Fusificar las Entradas.** Antes de que las reglas difusas puedan ser evaluadas, las entradas deben ser fusificadas de acuerdo a cada conjunto lingüístico (MathWorks,

2010a). Esto es tomar las variables numéricas de entrada “ x ”, dado por el problema, y determinar el grado al cual pertenecen a cada conjunto difuso “ F ” utilizando la función de membresía escogida $\mu(x)$.

La entrada en este caso es cualquier valor numérico limitado al universo del discurso de la variable de entrada y la salida el grado de membresía -siendo siempre un valor comprendido entre cero y uno (Ver ecuación 1.2)-. Es decir, la pertenencia de ese elemento “ x ” al conjunto difuso “ F ” depende del valor numérico que adopte al evaluarlo en la función.

- 2. Aplicar el Operador Difuso.** Luego de fusificar las entradas, debe saberse el grado en el cual cada parte del antecedente es satisfecho por cada regla. Si el antecedente de una regla dada tiene más de una parte, el operador difuso es aplicado para obtener un número que represente el resultado del antecedente para cada regla. Este número luego es aplicado a la función de salida. La entrada en este paso son dos o más valores de membresía de variables de entrada difusas y la salida un sólo valor que los representa a todos (MathWorks, 2010a).

En el programa **Matlab versión R2009a**, los métodos que se pueden utilizar son los siguientes:

$$AND = \begin{cases} min \\ prod \end{cases} \quad OR = \begin{cases} max \\ probor \end{cases}$$

Las funciones *min* y *max* fueron ya descritas en la sección 1.2.1.3. La operación “*prod*” implica el producto de las funciones de membresía y “*probobr*” (*probabilistic OR*) es calculado mediante la ecuación:

$$probobr(a, b) = a + b - ab$$

- 3. Implicación y Desfusificación final.** Antes de obtener la salida final, ha de obtenerse el peso “ w_i ” de cada regla difusa utilizando todas las entradas dadas en el antecedente. Por ejemplo, para dos entradas, se calcula así:

$$w_i = \text{MétodoAND}(\mu_1(x), \mu_2(y))$$

donde $\mu_1(x)$ y $\mu_2(y)$ son las funciones de membresía de las entradas 1 y 2 respectivamente.

Para defusificar el sistema y obtener la salida final, pueden escogerse entre dos métodos (N es el número de reglas difusas):

- Promedio Pesado o ‘wtaver’ (del inglés *weighted average*). La salida se calcula de acuerdo a la ecuación:

$$Z = \frac{\sum_{n=1}^N w_i z_i}{\sum_{n=1}^N w_i} \quad (1.9)$$

- Suma Pesada o ‘wtsum’ (del inglés *weighted sum*). La ecuación para calcular la salida es similar a la anterior.

$$Z = \sum_{n=1}^N w_i z_i \quad (1.10)$$

1.2.3. Ventajas

Hay muchas razones por las cuales es conveniente la lógica difusa como técnica de resolución de problemas (MathWorks, 2010a):

Es conceptualmente fácil de entender. Los conceptos matemáticos que hay detrás del razonamiento difuso no son complejos, el proceso es más intuitivo.

Es tolerante a datos imprecisos. En la naturaleza, todo es impreciso dependiendo del punto de observación. Por lo que el razonamiento difuso se construye tomando en cuenta este hecho.

Puede modelar funciones no lineales de variada complejidad. Se pueden crear sistemas difusos para ajustar cualquier conjunto de datos de entrada-salida. Este proceso se hace particularmente sencillo mediante la utilización de técnicas adaptativas, como por ejemplo el Sistema Adaptativo de Inferencia Neuro-Difusa (“ANFIS” por sus siglas en inglés: *Adaptive Neuro-Fuzzy Inference Systems*) ⁶.

Puede ser mezclado con técnicas de control convencionales. No necesariamente reemplaza los métodos convencionales de control. En muchos casos, los sistemas difusos simplifican su implementación.

Está basado en un lenguaje natural. Se fundamenta en la comunicación humana. Es fácil de utilizar debido a que su descripción se realiza cualitativamente en un lenguaje cotidiano.

⁶Desde el año 1995 está disponible como Librería en las versiones posteriores del programa *MATLAB*

1.2.4. Aplicaciones a las Geociencias

La diversidad de aplicaciones de la Lógica Difusa equipara o puede ser superior a la de Redes Neuronales. Hechos como datos inciertos y pobres correlaciones entre conjuntos de datos, son condiciones inherentes a las geociencias, debido en gran parte al desafío que resulta medir en formaciones complejas en medios “hostiles”. Incluso en el laboratorio es en ocasiones difícil relacionar la respuesta de un registro a un parámetro físico (Cuddy & Glover, 2002).

Muchos factores como la mineralogía, contenido de fluidos y tipos de lodos de perforación pueden influenciar una simple medida, llámese por ejemplo porosidad. Diversas técnicas convencionales tratan de minimizar o ignorar el error; sin embargo, la lógica difusa afirma que hay **información útil** en este error. Ésta puede ser utilizada para proveer una poderosa herramienta predictiva para los geocientíficos y complementar así las técnicas convencionales (Cuddy & Glover, 2002).

Para aclarar la importancia del término difuso aplicado a un ejemplo geocientífico, se puede poner un ejemplo de dos litofacies. Las facies eólicas pueden presentar una porosidad promedio de 20% y una varianza de $\pm 2\%$. Mientras que las facies fluviales, pueden tener una porosidad promedio de 10%, con una variación de $\pm 4\%$. Ahora, si al medir la porosidad de una facie desconocida y se obtiene 16%, ésta puede pertenecer a cualquiera de las dos litofacies.

Sin embargo, es menos probable que sea eólica debido a que en ésta la distribución es mucho más estricta, a pesar que su porosidad está a la misma distancia del “Más probable” o porosidad promedio esperado para cada tipo litotípico. Así como éste, son diversos los ejemplos en donde es aplicable la lógica difusa; por consiguiente, se presentan a continuación una breve descripción de diversos trabajos realizados:

1. Predicción de Registros de Pozo. Similar a las Redes Neuronales, la lógica difusa también se ha utilizado en la inferencia de registros de pozo⁷. Gutiérrez (2010) aplicó conceptos de la Lógica Difusa para estimar Velocidad de Onda de Cizalla (VS) a partir de otros registros (Porosidad, Volumen de Arcilla y Resistividad). La predicción se hizo en tres pozos, GF-109, GF-41 y GF-145, ubicados los dos primeros en el Bloque Norte del campo Guafita (Edo. Apure) y el tercero en el bloque Sur del mismo campo, obteniéndose buenos resultados cuando se combinaron las tres variables en cuestión.

Asimismo, Rezaee et al. (2007) usan sistemas de inferencia neuro-difusos para la predicción de VS en la cuenca Carnarvon, noroeste de Australia, a partir de otros

⁷Véase la sección 1.1.5

registros convencionales (Velocidad de Onda P, Rayos Gamma, Resistividad Profunda, Densidad y Porosidad Neutrón). Estos estudios son importantes, ya que, como se ha mencionado, tienen especial relevancia en la caracterización de yacimientos: determinación de litología, identificación de tipo de fluido y en interpretación geofísica.

- 2. Detección de Primeras llegadas.** Al igual que con las Redes Neuronales⁸, la lógica difusa también se ha utilizado para detectar primeras llegadas. González (2006) realiza la detección de primeras llegadas de datos de refracción sísmica tanto manual como automáticamente utilizando un algoritmo basado en lógica difusa, siendo la aplicación de este algoritmo efectiva para registros que no presentaran mucho ruido.

Asimismo, López (2005) determina las primeras llegadas en datos de reflexión sísmica, **VSP** (Vertical Seismic Profile = Perfil Sísmico Vertical) y **Check Shots** (Tiros de Verificación), utilizando Sistemas de Inferencia Difusa, de tipo Sugeno, basado en el modelo de reglas difusas Si – Entonces, donde igualmente el programa se ejecuta exitosamente salvo en los casos donde hay presencia de ruido de amplitudes mayores a las amplitudes de primera llegada, pues domina sobre éstas y la selección de los picos de entrada se realiza de manera errónea.

- 3. Predicción de Litofacies.** En Cuddy & Glover (2002) se utiliza la lógica difusa para la predicción de litofacies en diversos campos del Mar del Norte. Esto reteniendo la posibilidad que un tipo de facie en particular puede dar cualquier lectura en un registro, siendo más probables en unos que en otros.

Estos cálculos son especialmente útiles ya que sirven para la correlación de pozos y es importante para la construcción de modelos 3D del campo por geoestadística o técnicas estocásticas. Estos modelos a su vez, pueden ser utilizados para la evaluación de volúmenes de hidrocarburos en un yacimiento así como en la ubicación de nuevos pozos.

- 4. Predicción de Permeabilidad.** Cuddy & Glover (2002) utilizan la lógica difusa para actualizar el modelo de un yacimiento en el campo Ula, ubicado a 130 millas al suroeste de Noruega, y así abrir el potencial de una unidad superior utilizando nuevas técnicas de perforación. Este intervalo contenía aproximadamente el 50% de las reservas restantes que fueron inicialmente ignoradas debido a las características de las rocas.

Este estudio adquiere relevancia ya que la permeabilidad, aún siendo útil en el estudio de yacimientos de hidrocarburos, resulta costoso en la toma y estudio de muestras de núcleos y demora mucho más tiempo que utilizando técnicas computacionales para obtener información acerca de esta propiedad.

⁸Véase la sección 1.1.5

1.3. Algoritmos Genéticos

En el siglo XIX el científico británico **Charles Darwin** introdujo lo que se conoce como la **Teoría de la Evolución**, en ella afirmaba que *las especies naturales van evolucionando para adaptarse al medio que las rodea y aquellos individuos que mejor se adapten tendrán mayor probabilidad de sobrevivir.*

Esto llevó a pensar casi un siglo después, que tal y como la naturaleza es capaz de ir optimizando las características de un individuo, perfeccionando sus genes para adaptarse mejor al entorno, de igual manera se puede crear una población digital que se optimice para adaptarse mejor a una función objetivo planteada (Uzin et al., 1999).

Así fue como **John Holland**, un investigador de la Universidad de Michigan, a fines de la década de los sesenta se planteó desarrollar una técnica que fuera capaz de lograr que las computadoras aprendieran por sí mismas, denominándola “planes reproductivos”. Sin embargo, luego de la publicación de su libro “**Adaptación en Sistemas Naturales y Artificiales**” en 1975, ésta se hizo popular bajo el nombre de “**Algoritmos Genéticos**” (Castro, 2009), sentando las bases de numerosos estudios posteriores.

En base a esto, Koza (1992) propone una definición bastante completa de Algoritmos Genéticos:

Es un algoritmo matemático altamente paralelo que transforma un conjunto de objetos matemáticos individuales con respecto al tiempo usando operaciones modeladas de acuerdo al principio Darwiniano de reproducción y supervivencia del más apto, y tras haberse presentado de forma natural una serie de operaciones genéticas de entre las que destaca la recombinación sexual. Cada uno de estos objetos matemáticos suele ser una cadena de caracteres (letras o números) de longitud fija que se ajusta al modelo de las cadenas de cromosomas, y se les asocia con una cierta función matemática que refleja su aptitud.

1.3.1. Conceptos Básicos

El principio sobre el cual se desarrollan los Algoritmos Genéticos es tomar, dentro de una población de individuos dada, los individuos que, según unos parámetros y condiciones preestablecidas, tengan una mayor aptitud para sobrevivir dentro de esa población, y así crear una nueva generación mejorada de individuos (Gómez, 2000).

Los algoritmos genéticos operan sobre un grupo de posibles soluciones al que se le llama población y que inicialmente es creado aleatoriamente. A cada una de estas

posibles soluciones se les llama individuo o cromosoma y se representa como una cadena, habitualmente bits (0 y 1) (Gómez, 2000).

Los individuos tienen una longitud concreta y a cada uno de los miembros que conforman un individuo se le llama gen. Lógicamente, la población que es generada al principio aleatoriamente será una población de malas soluciones, pero precisamente se trata de que la población vaya mejorando por medio de la evolución natural hasta conseguir buenas soluciones (Uzin et al., 1999).

1.3.2. Funcionamiento de un Algoritmo Genético Simple

1.3.2.1. Función Objetivo

De acuerdo con el problema que se desea desarrollar, se debe definir la forma matemática de la función que debemos mejorar, la cual se denomina función objetivo y es el mecanismo de evaluación de las cadenas (Gómez, 2000). La evolución natural se basa en que las especies se vayan adaptando a su entorno. Del mismo modo, en los algoritmos genéticos, las soluciones han de ir adaptándose al problema para que, al final tengamos una población de buenas soluciones. Esta función mide lo adaptado que está un individuo al problema, será la que diga lo cerca que está una posible solución de la mejor solución del problema (Uzin et al., 1999).

La función fitness es la única función específica de cada problema de todas las funciones que realiza un algoritmo genético. Es en cierto modo, la que define el problema dentro del algoritmo genético (Uzin et al., 1999)

1.3.2.2. Definición del tamaño de la población

Se debe escoger la población de manera que sea manejable para el desarrollo y solución del problema. El tamaño de la población no variara durante el desarrollo del algoritmo genético, ya que se volverá inmanejable el proceso de búsqueda (Gómez, 2000).

1.3.2.3. Codificación de las posibles soluciones

Se necesita desarrollar un mecanismo de codificación para representar las variables del problema de optimización, y depende de la naturaleza de dichas variables. Para el proceso

de codificación se debe utilizar el alfabeto más pequeño posible, que por lo general se utilizan los números binarios como codificación de las variables y posibles soluciones (Gómez, 2000).

1.3.2.4. Selección de individuos

En cada generación de individuos debe hacerse una selección de los individuos más cualificados para pasar sus genes a la siguiente generación, al igual que ocurre con la selección natural. Para acercarse lo más posible a ésta última, la selección en un algoritmo genético se hará de manera aleatoria, pero teniendo cada individuo una probabilidad de ser seleccionado proporcional a su función de fitness, es decir, a la bondad de sus genes. De este modo conseguimos que en general se vayan eligiendo a los mejores individuos, pero, al mismo tiempo, los que no son tan buenos también tienen posibilidades de pasar y aportar algún gen que haga falta para formar mejores individuos en la siguiente generación (Uzin et al., 1999).

1.3.2.5. Cruce

Una vez seleccionados los individuos cuyos genes pasarán a la siguiente generación, se procede a la formación de ésta. Para ello primero se realizarán un número aleatorio de cruces (como máximo población/2). Los individuos que se vayan a cruzar también se deberán elegir aleatoriamente, teniendo en cuenta que un mismo individuo no puede cruzarse más de una vez. El cruce consiste en un intercambio de genes entre dos individuos a partir de un punto que, como casi todo, se elige aleatoriamente (Uzin et al., 1999).

1.3.2.6. Mutación

En la naturaleza se dan cambios en las especies que no se deben a los genes heredados de sus antepasados, sino a las mutaciones que se producen por azar en su cadena genética y pueden aportar valores positivos o negativos en la adaptación de la especie a su entorno. Esto mismo se realiza en los algoritmos genéticos. Se elige al azar cuántos y cuáles de los individuos van a mutar (un individuo no puede mutar más de una vez), se elige qué gen de su cadena mutará y se cambia ese gen por su complementado, como se ve a continuación (Uzin et al., 1999):

$$1|0|101 \implies 1|1|101$$

Una vez realizadas las mutaciones, ya tendremos la nueva generación, que, previsiblemente tendrá mejores características de adaptación al problema que la anterior, es decir, mejores funciones de fitness (Uzin et al., 1999)

1.3.2.7. Condición de parada

Se deben realizar numerosos ciclos de selección, cruce y mutación para mejorar suficientemente la población de individuos. El algoritmo genético debe finalizar cuando considere que la mejor solución encontrada hasta ese momento es bastante buena o que no va a poder encontrar una mejor. Para eso se define una condición de parada. La condición de parada más sencilla es hacer que el algoritmo pare siempre después de haber hecho un número de ciclos predeterminado. Otra posible condición de parada es que la función de fitness alcance un valor determinado que se considere que es suficientemente bueno para ese problema. La condición que se utilizará en este trabajo como condición de parada es una mezcla de ambas. Se guardará el individuo que tenga el valor de fitness más alto de la generación inicial. A partir de ahí, si ese valor no es superado en veinte ciclos, el algoritmo parará. Siempre que el valor de fitness del individuo sea superado, se guardará el nuevo mejor individuo y se volverá a contar hasta veinte ciclos del algoritmo. La solución al problema será el individuo que esté guardado en el momento que pare el algoritmo (Uzin et al., 1999)

1.3.2.8. Algoritmos Genéticos en Matlab

Son muy diversas las opciones de Algoritmos Genéticos que pueden variarse utilizando Matlab. Algunas de éstas son:

Tamaño de la población. Donde se especifica cuántos hay en cada generación.

Población Inicial. Permite especificar una población inicial para el algoritmo genético.

Selección. Funciones que permiten escoger los padres para la siguiente generación. Se puede escoger entre:

1. “Stochastic Uniform.”
2. “Roulette.”
3. “Shift Linear.”

Reproducción. Determina cómo el algoritmo genético crea los hijos para la nueva generación.

1. “Elite Count.” Especifica el número de individuos que están garantizados para sobrevivir a la siguiente generación. Debe ser menor o igual a la población inicial.
2. “Crossover Fraction.” Especifica la fracción de la siguiente generación que produce el cruce. La mutación produce los individuos restantes en la siguiente generación. Debe estar entre cero y uno.

Mutación. Hce pequeños cambios aleatorios en los individuos en la población, el cual proporciona una diversidad genética y permite al algoritmo buscar un espacio límite.

Cruce. Combina 2 individuos, o padres, para formar un nuevo individuo, o hijo, para la siguiente generación.

1. “Scattered.” Crea un vector aleatorio de cruce
2. “Single Point.” Coloca un punto aleatorio de cruce.
3. “Two Points.” Coloca dos puntos aleatorios de cruce.

1.3.3. Ventajas

Al igual que las Redes Neuronales y la Lógica Difusa, la aplicación de los Algoritmos Genéticos también ofrece diversas ventajas, entre las que se encuentra:

Son intrínsecamente paralelos. Mientras que otros algoritmos son en serie y sólo pueden explorar el espacio de soluciones hacia una solución en una dirección al mismo tiempo (si la solución que descubren resulta subóptima, no se puede hacer otra cosa que abandonar todo el trabajo hecho y empezar de nuevo), los AGs tienen descendencia múltiple, pueden explorar el espacio de soluciones en múltiples direcciones a la vez. Si un camino resulta ser un callejón sin salida, pueden eliminarlo fácilmente y continuar el trabajo en avenidas más prometedoras, dándoles una mayor probabilidad en cada ejecución de encontrar la solución (Castro, 2009).

Es altamente No Lineal. Funcionan particularmente bien resolviendo problemas cuyo espacio de soluciones potenciales es realmente grande -demasiado vasto para hacer una búsqueda exhaustiva en un tiempo razonable. En la vida real, se imponen los hechos no lineales, es decir, cambiar un componente puede tener efectos en cadena en todo el sistema, y donde cambios múltiples que, individualmente, son perjudiciales, en combinación pueden conducir hacia mejoras en la aptitud mucho mayores⁹ (Forrest, 1993).

⁹En un problema lineal, la aptitud de cada componente es independiente, por lo que cualquier mejora en alguna parte dará como resultado una mejora en el sistema completo

Es independiente del tipo de función utilizada. Se desenvuelven bien en problemas con un paisaje adaptativo complejo (aquellos en los que la función de aptitud es discontinua, ruidosa, cambia con el tiempo, o tiene muchos óptimos locales). La mayoría de los problemas prácticos tienen un espacio de soluciones enorme, imposible de explorar exhaustivamente; el reto se convierte entonces en cómo evitar los óptimos locales (soluciones que son mejores que todas las que son similares a ella, pero que no son mejores que otras soluciones distintas situadas en algún otro lugar del espacio).

Muchos algoritmos de búsqueda pueden quedar atrapados en los óptimos locales: si llegan a lo alto de una colina del paisaje adaptativo, descubrirán que no existen soluciones mejores en las cercanías y concluirán que han alcanzado la mejor de todas, aunque existan picos más altos en algún otro lugar del mapa (Koza, Forest, Andre & Keane, 1999).

Manipulan muchos parámetros simultáneamente. Muchos problemas de la vida real no pueden definirse en términos de un único valor que hay que minimizar o maximizar, sino que deben expresarse en términos de múltiples objetivos, a menudo involucrando contrapartidas: uno sólo puede mejorar a expensas de otro. Los AGs son muy buenos resolviendo estos problemas: en particular, su uso del paralelismo les permite producir múltiples soluciones, igualmente buenas, al mismo problema, donde posiblemente una solución candidata optimiza un parámetro y otra candidata optimiza uno distinto, y luego un supervisor humano puede seleccionar una de esas candidatas para su utilización (Forrest, 1993).

Se desconoce el problema a resolver. Una característica de los algoritmos genéticos que, a primera vista, puede parecer un desastre, resulta ser una de sus grandes virtudes: a saber, los AGs no saben nada de los problemas que deben resolver. En lugar de utilizar información específica conocida a priori para guiar cada paso y realizar cambios con un ojo puesto en el mejoramiento, como hacen los diseñadores humanos, son “relojeros ciegos”; realizan cambios aleatorios en sus soluciones candidatas y luego utilizan la función de aptitud para determinar si esos cambios producen una mejora (Koza et al., 1999).

1.4. Algoritmos Híbridos

1.4.1. Fusión de Lógica Difusa con Algoritmos Genéticos.

Como en los sistemas difusos y en las redes neuronales, los sistemas difusos y los algoritmos genéticos pueden complementarse unos con otros. Investigaciones en esta fusión pueden ser clasificadas en dos categorías:

1. Identificar Sistemas Difusos con algoritmos genéticos.
2. Controlar parámetros de algoritmos genéticos con sistemas difusos.

Como se ha mencionado, los sistemas difusos no tienen algoritmos de aprendizaje, de modo que los algoritmos genéticos pueden ser utilizados como algoritmos de aprendizaje de los sistemas difusos. Algoritmos Genéticos tienen algunos parámetros a ser escogidos, de manera que las reglas difusas para cambiar esos parámetros durante el proceso de búsqueda.

Aunque los sistemas difusos han sido utilizados en el control de un número de sistemas, la selección de aceptables funciones de membresía ha sido subjetivo y una tarea de tiempo de consumo. Cuando se construye un sistema difuso, se podría determinar el número de términos lingüísticos de las variables de entrada y salida, cuyas funciones de membresía y sus partes consecuentes de reglas difusas.

La estructura SI-ENTONCES de reglas difusas es fácil de entender y de construir con conocimiento “a priori”, pero muchos parámetros deben ser especificados. La identificación de éstos parámetros puede ser visto como un problema de optimización; encontrar los parámetros que optimicen el comportamiento del modelo. Por lo tanto, han sido muchas investigaciones en la aplicación de los algoritmos genéticos a la identificación de reglas difusas. Éstos codifican los parámetros del sistema difuso en cromosomas, y éstos cromosomas son evolucionados para encontrar los parámetros los cuales hacen del sistema difuso estimar (fit) los sistemas reales a los datos dados.

1.4.2. ANFIS: Lógica Difusa-Redes Neuronales

En general, una combinación de sistemas difusos con redes neuronales puede llevarse a cabo principalmente de dos formas (Jang, 1992):

1. Dotando una red neuronal con la capacidad de soportar información difusa (Redes Neuronales Difusas. En inglés *Fuzzy-Neural Networks*, “FNN”.)
2. Aumentando un sistema borroso con una red neuronal que le proporcione algunas de sus características, como la flexibilidad, la velocidad y la adaptabilidad (Sistemas Neuro-Difusos. En inglés *Neural-Fuzzy Systems*, “NFS”.)

En el primer caso, las entradas, los pesos y las salidas de la red son conjuntos difusos. Mientras que, en el segundo, que está diseñado para procesar razonamiento difuso, se tiene que los pesos de las conexiones de la red se corresponden con parámetros de dicho razonamiento (Jang, 1992). Se sigue entonces que, mediante algoritmos de retropropagación, los **NFS** pueden identificar reglas difusas y aprender funciones de pertenencia del razonamiento difuso, constituyendo así el fundamento principal de ANFIS (Rezaee et al., 2007)

Es oportuno destacar entonces que los componentes de un Sistema Neuro-Difuso son similares a los de un FIS, sin embargo, añadiendo una capa oculta de neuronas en cada etapa/fase para aprovechar la capacidad de aprendizaje de las Redes Neuronales y aumentar el conocimiento del sistema (MathWorks, 2010a).

De manera que **ANFIS** es una técnica computacional (basada en el Sistema de Inferencia Difusa Tipo Takagi-Sugeno-Khan) que **permite sintonizar o crear la base de las reglas de un sistema difuso, utilizando una estructura de redes neuronales** (algoritmos de retro-propagación y mínimos cuadrados) a partir de la recopilación de datos de un proceso (Rezaee et al., 2007).

Esta combinación hace de ANFIS una red adaptativa, lo cual significa que, direccionada hacia delante (“feed forward”), consiste en nodos y enlaces direccionales. Los nodos son adaptivos si sus salidas dependen, no sólo de sus entradas, sino también de parámetros modificables de la función interna del nodo. Para conseguir una relación entrada-salida, los parámetros de la red se determinan con base en datos de entrada que se denominan *datos de entrenamiento* dentro de un proceso de aprendizaje, de manera similar a como se hace en las redes neuronales (Gutiérrez, 2010).

Lo que es más, en ésta técnica, desarrollada en 1993 por J. R. Jang, se aprovechan las ventajas de los sistemas difusos de “explicar y entender” y redes neuronales las cuales pueden

ser utilizados para “aprender”¹⁰ (Saemi & Ahmadi, 2008). Esto la hace aplicable a una gran cantidad de áreas como por ejemplo (Gutiérrez, 2010):

- Modelados donde se necesita explicar datos pasados y predecir datos futuros.
- Control adaptativo automatizado.
- Procesamiento y Filtrado de señales.
- Clasificación de datos.
- Extracción de características a partir de ejemplos.

¹⁰Más específicamente, de la Lógica Difusa utiliza la representación del conocimiento previo para reducir el espacio de búsqueda de optimización, mientras que las redes neuronales se emplea la adaptación para automatizar el ajuste de los parámetros.

CAPÍTULO 2

MARCO GEOLÓGICO

2.1. Ubicación Geográfica

Los pozos GF-109N y Gf-87N a objeto de estudio se encuentran ubicados en el campo **Guafita**. Éste pertenece a la **Cuenca Barinas-Apure**, específicamente en la subcuenca de Apure (Figura 2.1), en la ribera norte del Río Arauca, 45 Km al sureste de la población de Guasdualito en el suroeste de Venezuela (Kiser, 1997). Este campo fue descubierto en 1984 y asignado a Corpoven en 1986, con una superficie de explotación de 24363 hectáreas, situadas en el Municipio El Amparo del Distrito Páez en el estado Apure, 40 Km al este del Campo La Victoria y 25 Km al Norte del Campo Caño Limón en Colombia (Bello, 2004).

2.2. Geología Regional

La Cuenca Barinas-Apure está ubicada en la parte suroccidental del país, al norte de la frontera con Colombia y se extiende sobre los estados Apure, Barinas, parte de Portuguesa y Táchira meridional. Pertenece al sistema de cuencas subandinas, que constituye un área de sedimentación pericratónica que quedó estructuralmente aislado entre el Escudo Suramericano y la Cordillera de los Andes, a raíz del levantamiento de esta última en el Plio-Pleistoceno (Gil, 2003).

La Cuenca Barinas-Apure (Figura 2.2)¹, está limitada hacia el noroeste por Los Andes venezolanos, al norte por la prolongación occidental de la Serranía del Interior Central, al este y noreste por el levantamiento de El Baúl y al sur está separada de la Cuenca de los

¹Tomada de (Yoris & Ostos, 1997)

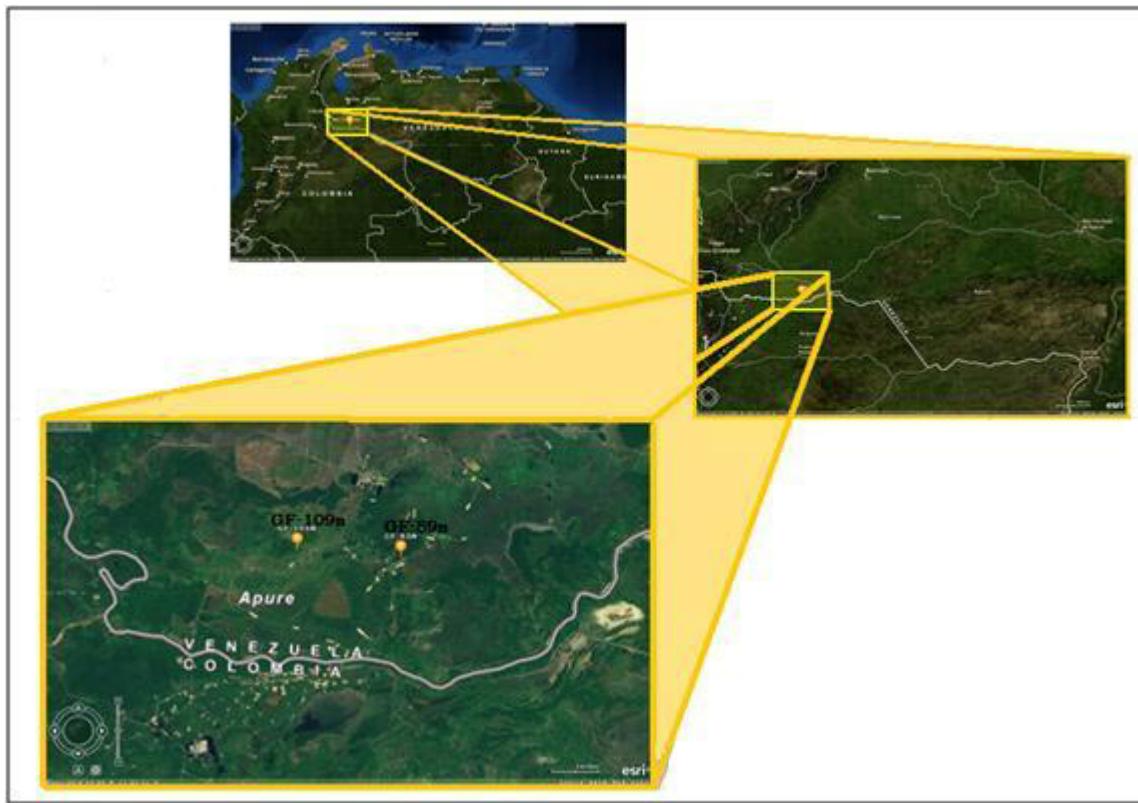


Figura 2.1: Ubicación Geográfica del área de estudio. (Fotografías tomadas utilizando el programa *ArcGis*).

llanos colombianos por un alto gravimétrico situado entre los ríos Apure y Arauca (González de Juana, Iturrealde & Picard, 1980).

Martínez (1978) estima que la Cuenca Barinas-Apure tiene una superficie de 95.000 Km^2 y un volumen de sedimentos de 167.000 km^3 (González de Juana et al., 1980). Los campos **Guafita y la Victoria**, constituyen descubrimientos recientes en la zona de Apure, en el borde sur de la Cuenca, cerca de la frontera colombo-venezolana (Gil, 2003).

2.2.1. Modelo Estructural

La depresión estructural que caracteriza la Cuenca Barinas-Apure se desarrolla desde la antefosa andina al noreste hasta las planicies situadas entre los ríos Arauca y Apure en el sureste (González de Juana et al., 1980). A lo largo del perfil noroeste-sureste (en la Figura 2.3)², desde los Andes y hacia el Escudo Guayanés, se evidencian rasgos estructurales

²Esta imagen es tomada de (Yoris & Ostos, 1997)

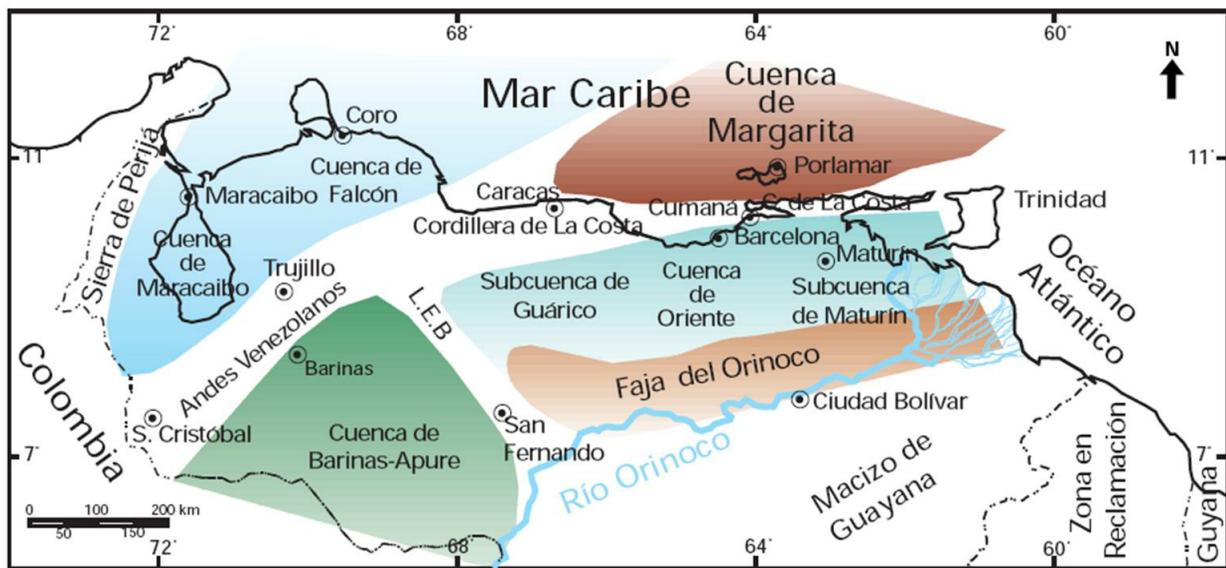


Figura 2.2: Cuencas Petrolíferas de Venezuela, basadas en la distribución de sus Provincias Sedimentarias

de rumbo noreste-suroeste que la Cuenca Barinas-Apure comparte con la mayoría de las cuencas subandinas.

En esta cuenca se puede apreciar una depresión estructural asimétrica con formaciones que se engrosan y buzan suavemente desde el Escudo hacia Los Andes. Seguidamente al oeste, los buzamientos se invierten dando lugar a una faja de pliegues con fallas inversas y sobrecorrimientos hacia el este, en el frente piemontino. En sentido noreste-suroeste se evidencian rasgos geológicos más antiguos, con dirección perpendicular a los Andes y la fossa subandina (Yoris & Ostos, 1997). En la figura 2.4 se muestra los elementos estructurales más resaltantes en dirección Sur a Norte³, en donde se señalan:

- *Arco de Arauca*. Ubicado cerca de la frontera colombo-venezolana en sentido noroeste-sureste.
- *Arco de Mérida*. Es un paleoalto paralelo al arco de Arauca, ubicado al sur de la ciudad de Barinas. La zona productora se localiza en su flanco septentrional.
- *Arco de El Baúl*. Se localiza en el extremo noreste de la Cuenca Barinas-Apure y es subparalelo a los arcos Arauca y Mérida. Este arco no tiene continuidad hacia el noreste, donde es sustituido por la Depresión Estructural de Guarumen.

³Esta figura es tomada de (Gil, 2003)

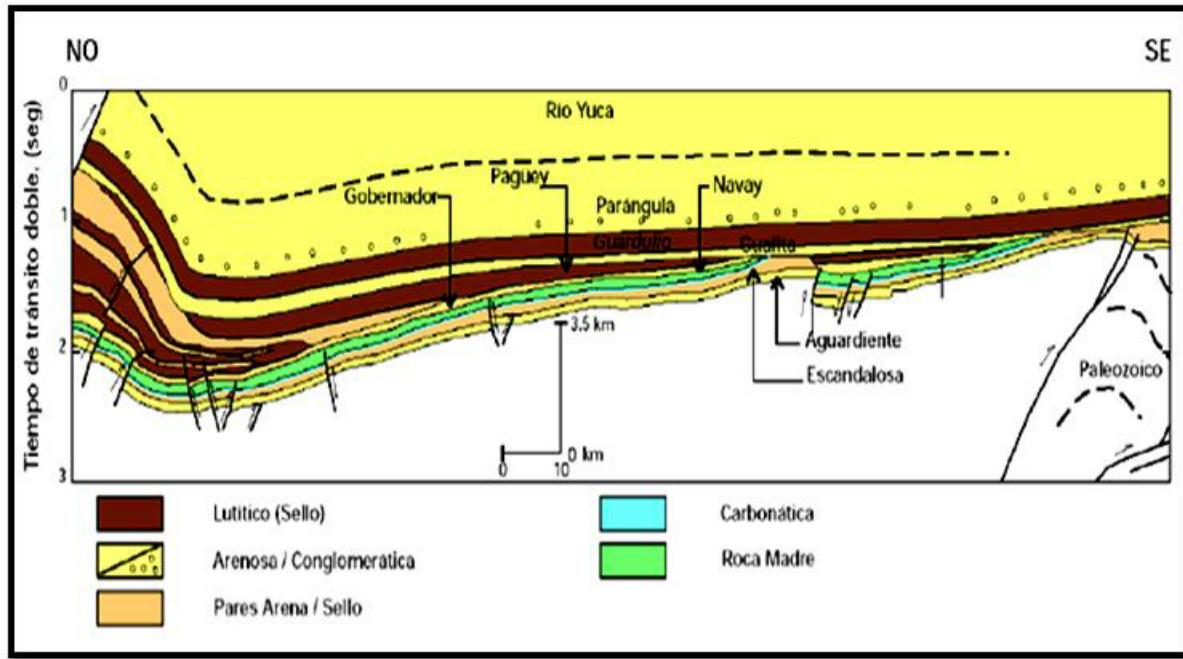


Figura 2.3: Sección NO-SE de la Cuenca Barinas-Apure

2.2.2. Modelo Estratigráfico Sedimentológico

Barberii (1989) explica la sedimentación en la cuenca Barinas – Apure con ciclos sedimentarios que inician en el Cretácico, conformado cada uno por secuencias de tendencias transgresivas y regresivas, separadas por un intervalo de máxima invasión marina, en donde son depositadas las arenas petrolíferas de esta cuenca. Estos ciclos son descritos a continuación:

Ciclo 1: Albiense. Comienza con la sedimentación de arenas y conglomerados de la **Formación Río Negro** sobre el peniplano Cretácico. Durante el Albiense la transgresión alcanza un máximo, sedimentando una capa lutítica marina denominada miembro “S” de la **Formación Escandalosa**. La fase regresiva deposita areniscas de llanura deltaica, llamadas miembro “P” de la misma formación.

Ciclo 2: Coniaciense. Al inicio de este ciclo se produce la sedimentación de facies marinas de frente deltaico, representadas por las calizas, areniscas y lutitas del miembro “O” de la **Formación Escandalosa**. Durante la máxima invasión marina se sedimentan las lutitas de la **Formación La Morita**. En la fase regresiva, primero se depositan las areniscas constituyentes de las barras del frente deltaico intercaladas con lutitas de la **Formación Quevedo**, para cerrar con arenas de llanura deltaica de la parte basal de la **Formación Burgüita**.

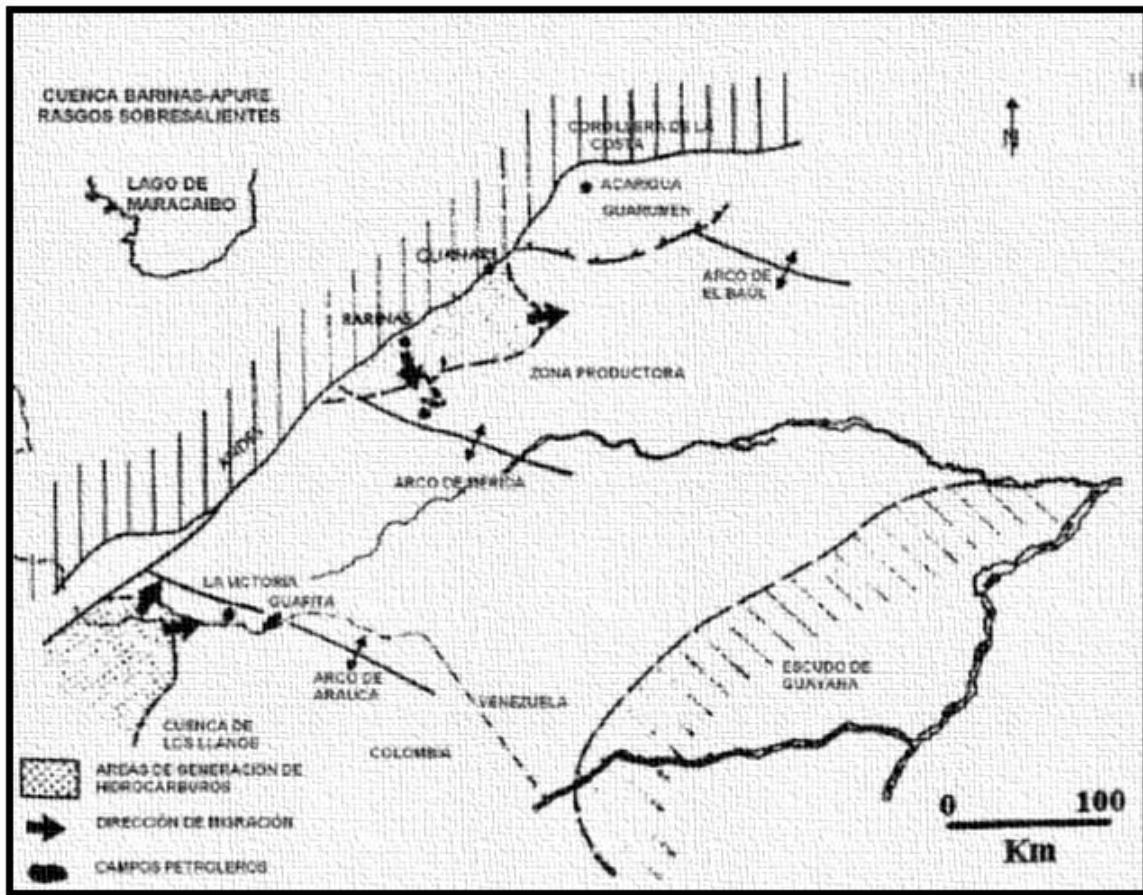


Figura 2.4: Rasgos estructurales de la Cuenca Barinas-Apure

Ciclo 3: Maestrichtiense. Los sedimentos de este ciclo se encuentran profundamente erosionados debajo de la discordancia en la base del Terciario. La parte transgresiva se denomina miembro “E” de la **Formación Burgüita** y está preservada en los ejes sinclinales pre-terciarios que rodean el Arco de Mérida. Localmente también está preservada la parte regresiva, miembro “D”, Formación Burgüita, constituida por areniscas producto de la orogénesis del final del Cretácico.

Ciclo 4: Paleoceno-Eoceno Inferior. Este ciclo está muy difundido en áreas vastas de Colombia y de la Cuenca de Maracaibo, pero en Barinas-Apure sólo se han identificado en la cuenca infrandina al norte del Alto de Brujas. Representa la evolución de un gran sistema deltaico con el cual comienza la sedimentación terciaria en el área.

Ciclo 5: Eoceno medio. Este ciclo está restringido al norte del Arco de Arauca. Su base está marcada por areniscas que representan una transgresión marina clásica con sedimentos de playa identificados en la zona productora como **Formación Gobernador**.

Ciclo 6: Eoceno superior. Encima de la **Formación Cobre**, en el Arco septentrional del Arco de Arauca, se ha identificado un intervalo de lutitas marinas, de probable edad Eoceno superior que hacia la zona productora constituyen claramente la parte alta de la **Formación Pagüey**. Este intervalo marino, representa un vestigio de un nuevo ciclo sedimentario profundamente erosionado por debajo de la base de la molasa.

Ciclo 7: Oligoceno-Mioceno Medio. La sedimentación Terciaria que fue posterior al tope erosionado del Cretácico empieza por un paquete de areniscas que representan un ambiente de llanura deltaica con canales distributarios, abanicos de rotura y sedimentos de bahía con débiles y esporádicas influencias marinas atribuidas a la **Formación Guafita** de edad Oligoceno. Este ciclo se ha identificado en el flanco sur del **Arco de Arauca** en la zona de los campos Guafita y La Victoria. Más hacia arriba, las influencias marinas se hacen más frecuentes y finalmente la máxima transgresión del ciclo deposita las lutitas de la **Formación León**, que en la actualidad se atribuye también a la Formación Guafita.

Ciclo 8: Mioceno superior-Pleistoceno. Este último ciclo, representa el relleno de la Cuenca, ocurrido durante el levantamiento de la cadena andina. Se trata de sedimentos arenoso-lutíticos de ambiente fluvio-deltaico conocidos como **Formaciones Parángula y Río Yuca** (Gil, 2003).

2.2.3. Aspectos Petrolíferos

La Cuenca Barinas-Apure tiene a la **Formación Navay** (Cretácico tardío) con facies laterales equivalentes a la Formación La Luna como roca madre por excelencia. Las principales **rocas yacimiento clásicas** son, según Kiser (1997):

- Las Formaciones Escandalosa y Burgüita (Cretácico).
- El Grupo Orocué (Paleoceno).
- Las Formaciones Mirador y Guafita (miembro Arauca) del Eoceno-Oligoceno.

Además, el **yacimiento carbonático** lo constituyen las calizas con porosidad secundaria del miembro Guayacán de la Formación Escandalosa (Caliza “O”).

Por otra parte, los **sellos regionales** más importantes son los intervalos lutíticos de las Formaciones:

- Burgüita (Cretácico tardío).
- Pagüey (Eoceno).
- Guafita (miembro Guardulio).

Se han identificado dos eventos para la generación, migración y entrampamiento (Yoris & Ostos, 1997):

1. El primero se relaciona con el sistema La Luna-Burgüita debido a la generación de petróleo en la cuenca de Maracaibo y su migración al sur-sureste. En este caso, el sello lo constituyen unidades cretácicas: el superior, la lutita basal de la Formación Burgüita y el inferior, las lutitas del Miembro La Morita (**Formación Navay**). El yacimiento principal estaría constituido por unidades cretácicas, como las **Formaciones Aguardiente, Escandalosa** (la Caliza “O” o Miembro Guayacán) y las arenas basales de la **Formación Burgüita**.
2. El segundo evento se relaciona con el depocentro del flanco surandino, el cual tiene su momento crítico hoy en día. La roca madre sigue siendo cretácica y el yacimiento llega a incluir unidades eocenas como las **Formaciones Gobernador y Pagüey**, siendo el Miembro Guardulio de la **Formación Guafita** el sello Oligoceno de mayor importancia regional.

Barberii (1989) presenta un esquema simple para la cuenca Barinas-Apure: Dos cocinas generadoras cuyas rocas madre cretácicas han generado petróleo al final de la subsidencia subandina; dos arcos receptores de las migraciones de hidrocarburo, al mismo tiempo barreras para las mismas y un área entre los dos arcos que constituye la Depresión de Capitanejo, aislada de migraciones importantes y por consiguiente de pobre expectativa petrolífera.

2.3. Geología Local

Una vez ubicado en la geología regional de la Cuenta Barinas-Apure se explicará la geología local del campo Guafita, de la misma manera que la sección anterior.

2.3.1. Modelo Estructural

El Campo Guafita se encuentra ubicado el área sur del estado Apure al Norte del Río Arauca. El mismo se halla sobre un alto estructural de rumbo suroeste-noreste y el sistema

de fallas longitudinales **Guafita-Caño Limón** lo dividen en dos bloques, tal y como se muestra en la Figura 2.5 (Limongi, 2000).

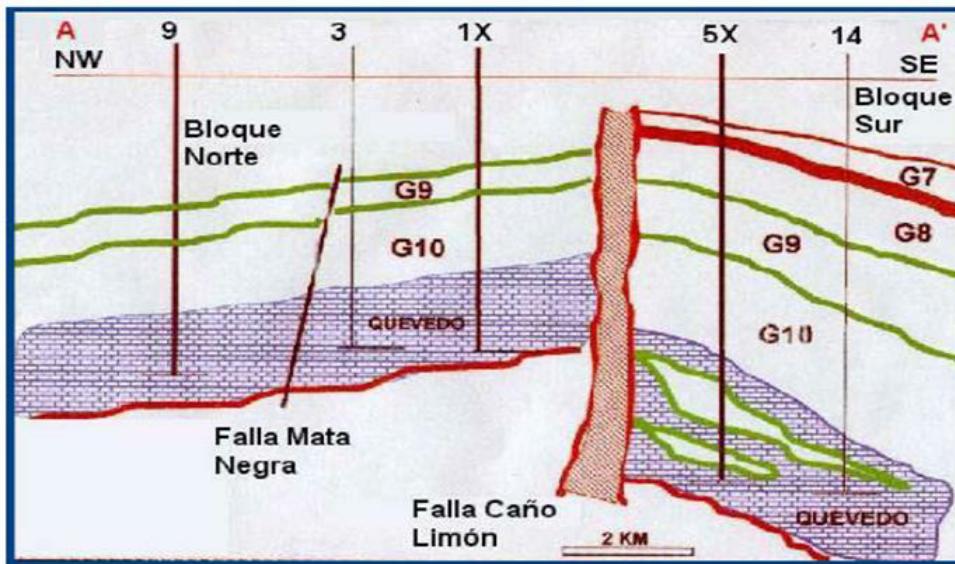


Figura 2.5: Corte Estructural del Campo Guafita

La estructura del campo Guafita es un anticlinal con eje noreste-suroeste, este fue descubierto por el Arco de Arauca y está dividido en dos sectores por una fallas longitudinales de tipo transcurrente, con un rumbo noreste y buzamiento hacia el sur, denominada Guafita-Caño Limón (ver Figura 2.6)⁴ que lo dividen en dos bloques (Limongi, 2000):

- Bloque Norte. Es un anticlinal suave.
- Bloque Sur. Sus estructuras principales son un anticlinal fallado de rumbo N45E y la falla de Yuca de rumbo N22E que es de carácter inverso y que presenta un salto de 60 m aproximadamente.

El sistema de fallas Guafita-Caño Limón es de tipo transcurrente, con desplazamiento dextral de 5 Km y componente vertical de 120 m. Éste está relacionado con esfuerzos en dirección noroeste-sureste que resultan en dos eventos de deformación (Bello, 2004):

1. Plegamiento y fallamiento inverso de dirección noreste-suroeste (**Falla La Yuca**).
2. Fallamiento transcurrente dextral noreste-suroeste (**Falla Guafita-Caño Limón**) asociado con fallamiento normal de rumbo este-oeste (**falla Arauca**).

⁴Esta figura fue tomada de (Almarza, 1995)

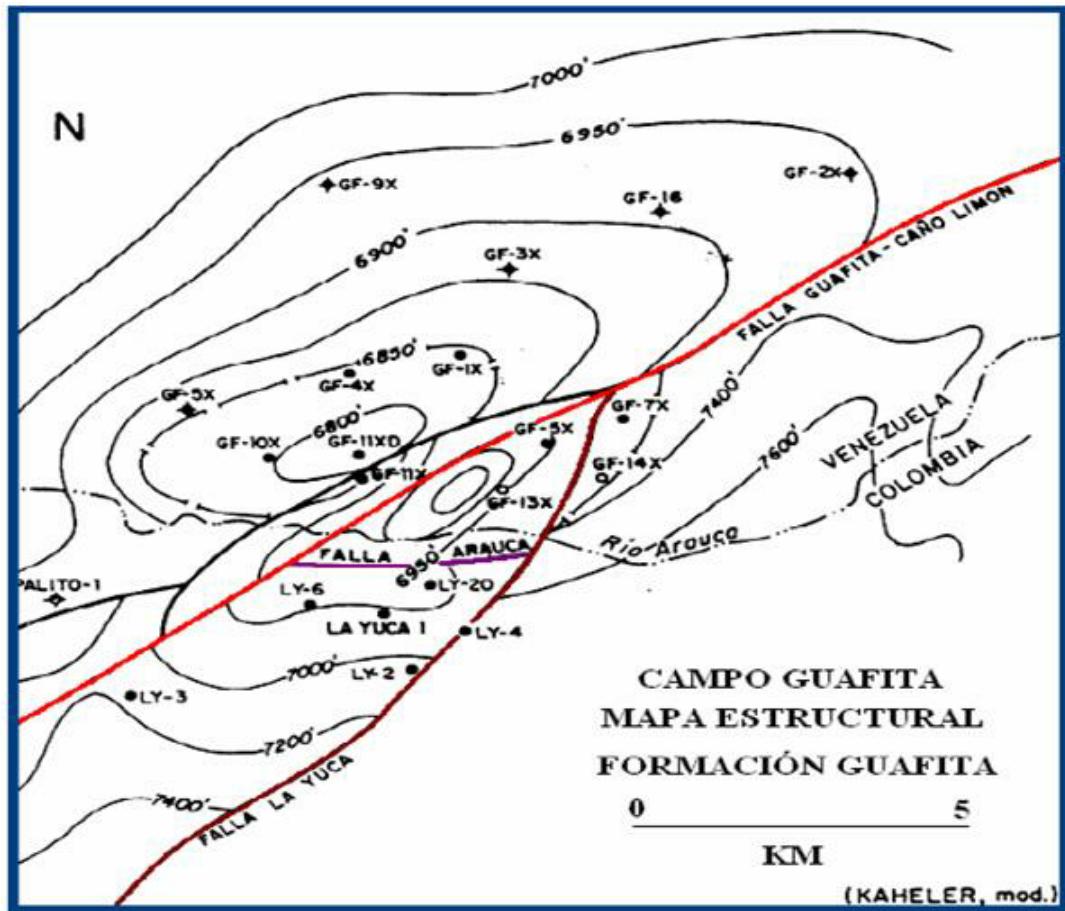


Figura 2.6: Mapa Estructural del Campo Guafita (Fm. Guafita)

La estructura de los campos petrolíferos de Apure, así como sus horizontes estratigráficos, están asociados a la estructura de la cuenca sedimentaria de los llanos colombianos (Gil, 2003).

2.3.2. Modelo Estratigráfico Sedimentológico

Según Gil (2003), el espesor de sedimentos en la subcuenca Apure-Arauca varía entre los Campos La Victoria y Guafita desde 2440 m en el oeste hasta 3500 m en el este. Los Campos Guafita y La Victoria son esencialmente arenosos; en ellos se encuentran areniscas arcillosas de grano fino a medio, poco consolidadas, piríticas y carbonáceas intercaladas con lutitas carbonosas, compactas y piríticas de hasta 10 m de espesor.

La sección perteneciente al campo Guafita está caracterizada por arenas basales de la Formación Terciaria Guafita y una sección arenosa del cretáceo, hasta alcanzar 90 pies de la **Formación La Quinta** (Jurásico) con limolitas y arcillitas rojas.

Como se muestra en la figura 2.7 (tomada de Limongi (2000)), la secuencia de Guafita contiene la formación Guafita (Oligoceno), dividida en dos secciones llamadas lutitas de Guafita de origen fluvial, un ambiente de llanura deltaica formado por las arenas basales, de canales distributarios y abanicos de rotura. El sello corresponde con la creciente arcillosidad que aumenta en sentido vertical con lutitas de bahía intercaladas dentro del conjunto arenoso.

UNIDAD CRONOLÓGICA	FORMACIÓN	MIEMBRO		LITOLOGÍA
		UNIDAD INFORMAL		
PLEISTOCENO	PARÁNGULA			Areniscas de grano fino, limonitas y iodolitas abigarradas
PLIOCENO				
MIOCENO	GUAFITA	G		40% Lutitas
OLIGOCENO		U		40% Arcilitas
EOCENO		A	G-7	20% Areniscas
PALEOCENO		R	LUTITA LIGNÍTICA	
CRETACEO		A	G-8	75% Arenas y areniscas
JURÁSICO		U	LUTITA DE GUAFITA	20% Limolitas
PALEOZOICO		C	G-9	5% Lutitas
		A	G-10	
	NAVAY	QUEVEDO LA MORITA		Lutitas, ffnitas y areniscas calcáreas a silíceas
	ESCANDALOSA			areniscas y lutitas calcáreas
	AGUARDIENTE			Areniscas calcáreas, lutitas y calizas
	LA QUINTA			Tobas, areniscas gruesas y conglomeráticas, limolitas y calizas
	METAMÓRFICAS			

Figura 2.7: Columna Litoestratigráfica del Campo Guafita

Durante el Jurásico, entre Norte y Sur América, se originó un mar abierto que durante el Cretácico invadió el norte de Venezuela por el surco de Barquisimeto, mientras que una transgresión marina avanzaba desde el Sur por los surcos de Machiques y Uribante. Estos sedimentos que recibió la subcuenca de Apure dieron lugar a la **Formación Aguardiente**.

Suprayacente a ésta continúa concordantemente el Cretácico, con calizas compactas de la **Formación Escandalosa**, lutitas y areniscas de la **Formación La Morita** (máxima transgresión en Guafita y La Victoria) y la **Formación Quevedo**, que se hace más arenosas

hacia arriba. La secuencia cretácica termina con la **Formación Burgüita**, más arenosa y menos marina.

A principios del Terciario, una elevación generalizada del borde septentrional del cratón de Guayana ocasionó una amplia retirada de los mares hacia el Norte, por lo que la subcuenca de Apure no presenta sedimentos del Paleoceno y Eoceno.

En el Oligoceno se presenta una nueva transgresión marina, por lo cual la secuencia de Guafita tiene la **Formación Guafita** dividida en dos secciones arenosas de origen fluvial (de ambiente de llanura deltaica formado por las arenas basales, de canales distributarios y abanicos de rotura), por el intervalo lutítico denominado **Lutita de Guafita**.

Seguidamente, hacia el Mioceno Medio las entradas marinas se hacen más frecuentes y la máxima transgresión del Terciario, depositó las lutitas de la Formación Guafita.

El ciclo Mioceno superior-Pleistoceno deposita, discordantemente, sedimentos arenosos lutíticos de ambiente fluvio-deltaico (**Formaciones Parángula y Río Yuca**) que representan el relleno discordante de la Cuenca Barinas-Apure después del levantamiento de los Andes (Bello, 2004).

2.3.3. Formación Guafita (Oligoceno-Mioceno temprano)

La Formación Guafita está compuesta por intercalaciones de wacas cuarzosas y arcósicas, lutitas, arcilitas, limolitas y algunas capas delgadas de lignito. En el Campos Guafita, se pueden reconocer dos miembros (Ortega, Van Evre & Monroy, 1987):

Miembro Arauca (inferior). Compuesto de un 75% de arenas, areniscas (wacas cuarcíticas y arcósicas) con matriz arcillosa-caolinítica; comúnmente con pirita, nódulos de hematita y esferulitas de siderita. Las limolitas representan el 20% del miembro. Las lutitas son laminares a masivas, bien estratificadas y representan entre el 5 y el 10% de la unidad. En la base del miembro, se observan capas delgadas y aisladas de caliza tipo packstone de oncolitos.

Miembro Guardulio (superior). Compuesto por un 40% de lutitas masivas a laminares, ocasionalmente moteadas de óxido de hierro. Las lutitas generalmente gradan a lutitas carbonosas y lignitos. Las arcilitas representan el 40%, generalmente abigarradas y manchadas de óxido de hierro; además, se presentan verdosas y blanquecinas (caoliníticas), mal compactadas, con fragmentos aislados de fítnita y wacas cuarzosas y representan entre el 10 a 20% del miembro.

El ambiente de sedimentación de la Formación Guafita, se caracteriza por un complejo de canales distributarios activos y abandonados, abanicos de rotura, bahías y llanuras interdistributarias, canales de mareas y marismas (pantanoso-costeras), propios de un sistema deltaico progradante. Ambos miembros de la Formación Guafita son transgresivos, siendo el inferior más litoral y el superior más marino.

Las arenicas y wacas cuarzosas de la Formación Guafita, constituyen excelentes rocas yacimientos de hidrocarburos livianos en el Campo Guafita de la región de Apure y los campos del complejo Caño Limón La Yuca - Mata Negra (Ortega et al., 1987).

2.3.4. Producción

El Campo Guafita forma una unidad geológica con los campos colombianos La Yuca y Caño El Limón. Se ha inferido comunicación de fluidos del yacimiento Guafita Sur con los de La Yuca a través de la falla Arauca, en el caso que la falla no tenga carácter sellante y exista yuxtaposición de las arenas.

En el Campo Guafita, el horizonte productor más importante corresponde a la sección superior de las areniscas basales de la Formación Guafita (Oligoceno) con una columna petrolífera de 72 m (Gil, 2003).

El pozo descubridor de este campo fue el GF-1X, perforado en el **Bloque Norte**. Este pozo encontró petróleo en tres intervalos de areniscas basales pertenecientes a la Formación Guafita, produciendo unos 2900 B/D de gravedad 29.5° API. Luego 8 pozos fueron perforados de los cuales 5 resultaron productores, delimitando así el área petrolífera del Bloque Norte.

En el **Bloque Sur** el pozo descubridor de petróleo fue el GF-5X, con un rendimiento de 8.400 B/D de crudo de gravedad 30° API. La delimitación del área petrolífera se llevó a cabo por medio de la perforación de 3 nuevos pozos, dos de los cuales resultaron productores.

En la estructura Guafita-Caño Limón se han determinado 15 intervalos de interés petrolífero en la Formación Guafita (Oligoceno) y cuatro en el Cretácico. El mecanismo de producción es el impulso hidráulico. La relación gas-petróleo en el Campo Guafita es baja (10pc/bbl), por lo que se estima muy escasa la reserva de gas (Bello, 2004).

CAPÍTULO 3

METODOLOGÍA

Con la finalidad de realizar la predicción de Registros de Velocidad de Onda S mediante la utilización sistemas basados tanto en Lógica Difusa como de Algoritmos Genéticos, así como para codificar un algoritmo que hibridice a ambos, se contó con el programa *Matlab* versión *R2009a*, el cual a su vez cuenta con dos librerías especializadas en ambos métodos:

Fuzzy Logic Toolbox. (*Librería de Lógica Difusa*). Contiene la herramienta ANFIS (*Artificial Neuro-Fuzzy Inference System = Sistema de Inferencia Neuro-Difuso*), para la realización de inferencias difusas utilizando el método Takagi - Sugeno - Khan.

Genetic Algorithm (GA) and Direct Search Toolbox. (*Librería de Algoritmos Genéticos (AG) y Búsqueda Directa*). En él, entre otros aspectos, pueden realizarse optimizaciones de funciones paramétricas mediante Algoritmos Genéticos.

Asimismo, para la obtención de los resultados se contó con los datos disponibles de dos pozos del Campo Guafita: GF-87N y GF-109N, los cuales se encuentran ubicados al Noroeste del Estado Apure (Ver Figura 2.1). En ambos los registros disponibles son:

- Densidad.
- Sónico de Onda S.
- Sónico de Onda P.
- Saturación de Agua.
- Porosidad.
- Volumen de Arcilla.

De manera que, teniendo en cuenta los objetivos planteados inicialmente, y con estos estos recursos disponibles, la metodología de trabajo fue dividida de la siguiente manera:

1. Estudio de los datos disponibles.
2. Escogencia de los Criterios de Comparación entre las predicciones.
3. Codificación de las técnicas de predicción:
 - (a) Algoritmos Genéticos.
 - (b) Sistemas de Inferencia Neuro-Difusa.
4. Realización de las pruebas preliminares con ambas técnicas
 - (a) Algoritmos Genéticos. Se seleccionan los parámetros que dan el mejor ajuste de Velocidad de Onda S .
 - (b) Sistemas de Inferencia Neuro-Difusa. Se busca la mejor combinación de reglas difusas utilizando la técnica “Grid Partition” de ANFIS.
5. Codificación del Algoritmo Híbrido Difuso-Genético.
6. Ejecución de los Algoritmos basándose en los resultados obtenidos en los pasos 4a y 4b.

3.1. Estudio de los datos disponibles

3.1.1. Visualización de los Registros de Pozo

Un fenómeno que ocurre usualmente en la medición de los registros de pozo es el hecho que no todos son medidos en las mismas profundidades, siendo común generalmente el intervalo que se quiere estudiar (un espesor de arenas de alta porosidad, por ejemplo). Las causas son de diversa índole; pueden deberse a costos de las mediciones, problemas con las herramientas durante las grabaciones, condiciones especiales de los pozos, entre otros.

Por esta razón, la primera fase debía consistir en un estudio detallado de los registros que tienen los pozos disponibles, los cuales no escaparon de esta realidad. En ellos había intervalos donde la medición es nula¹, siendo estos puntos de escaso interés para los efectos de realizar la predicción de un registro así como para la visualización y estudio de los mismos.

¹Todos los registros de pozo tienen un número con el cual se designan los puntos en que no hubo medición de una propiedad determinada. Estos suelen ser -999.0000; -999.2500; -999.5000 entre otros.

Por consiguiente, se hizo necesario realizar una función que graficara los registros de los pozos excluyendo aquellos puntos donde no hubo medición, además de incluir otro tipo de información, como por ejemplo, los topes de la litología de interés, que, en este estudio, se corresponden con los topes **G-9** y **G-10** de las arenas del **Miembro Arauca** de la **Formación Guafita**. Esta función se denominó "*graficas_registro*"².

Como consecuencia de aplicación de esta función, en la figuras 3.1 y 3.2 se muestran los registros de los dos pozos, donde los topes de las arenas de interés son señalados con líneas de color verde.

Posteriormente, se hizo necesario suprimir los puntos donde no hubo medición común entre todos los registros. Esto para evitar asociar intervalos de un registro medido con otro donde no existiera medición. La función "*eliminar*" (cuyo código se muestra en el Apéndice C, página 169), realiza este proceso, excluyendo además el registro de Velocidad de Onda S, el cual se desea predecir (con la finalidad de luego completarlos utilizando los algoritmos de predicción.)

Al llegar a este punto, se hizo necesario definir términos fundamentales para la generación de los resultados finales:

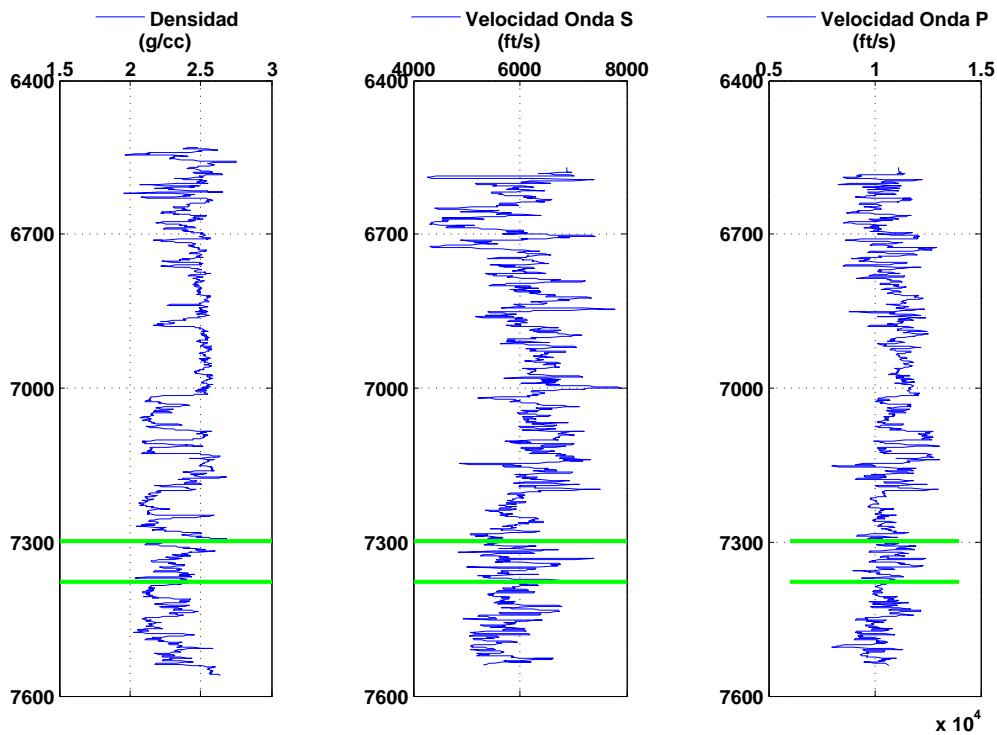
Pozo de Entrenamiento. Es aquél del cual se selecciona un porcentaje de los datos para realizar el entrenamiento de los **algoritmos de predicción** utilizados en esta investigación (**ANFIS, Algoritmos Genéticos, Algoritmo Difuso-Genético**). Esta escogencia se realizó de la siguiente manera: Primero se seleccionó el intervalo comprendido entre 7200'-7500' (aunque puede ser en cualquier intervalo de profundidad), ya que resulta superior e incluyente al intervalo litológico de interés de estudio. De estos, se obtiene aleatoriamente el 50% del total para el entrenamiento final del algoritmo.

Con esto, se tomó en cuenta que, según (MathWorks, 2010a) los datos de entrenamiento de un algoritmo deben representar, como mínimo, una quinta parte ($\frac{1}{5}$) del total³. Debido a que se seleccionó el pozo GF-87N, por ser el que mayor cantidad de datos posee en profundidad y porque contiene el registro que se desea predecir⁴, este esquema de selección implica ($\frac{1}{7}$) del total de los datos del registro.

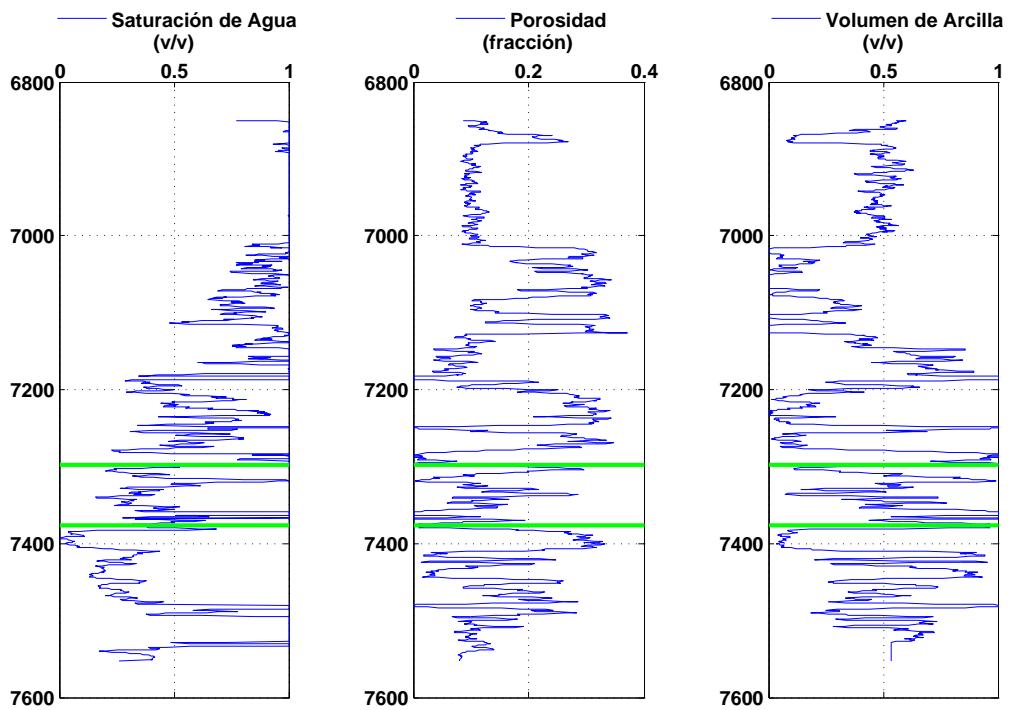
²Se presenta en la sección C.7 en la página 171

³Con la finalidad también de generar un sistema que no sea particular sólo para ese pozo, siendo así lo suficientemente general como para extrapolarlo a otro pozo distante

⁴Aunque es requisito obligatorio para un Pozo de Entrenamiento, esta particularidad no influye directamente en este estudio debido a que, como ya se ha mencionado, ambos pozos lo poseen

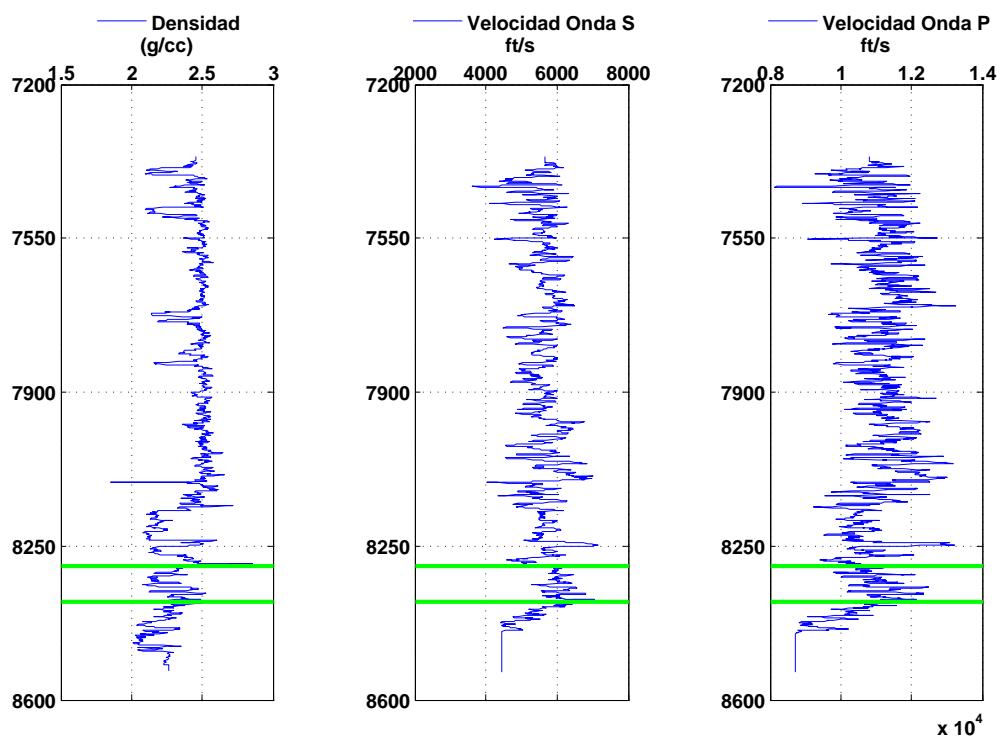


(a) Densidad, Velocidad de Onda S, Velocidad de Onda P

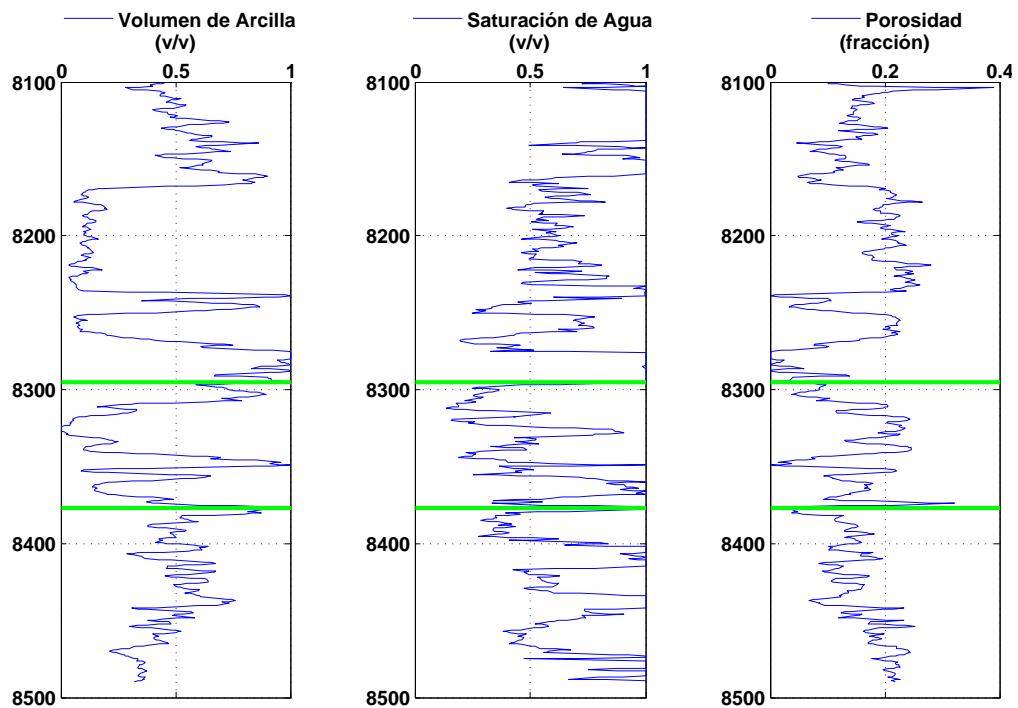


(b) Saturación de Agua, Porosidad, Volumen de Arcilla

Figura 3.1: Registros del Pozo GF-87N



(a) Densidad, Velocidad de Onda S, Velocidad de Onda P



(b) Volumen de Arcilla, Saturación de Agua, Porosidad

Figura 3.2: Registros del Pozo GF-109N

Pozo Control. Aquél que se utilizará para calibrar la efectividad de cada uno de los sistemas generados en la predicción del registro de Velocidad de Onda S. Para ello queda entonces el pozo GF-109N.

Registro(s) de Entrada. Registro(s) seleccionado(s) para el entrenamiento de los sistemas de predicción (Algoritmos Genéticos , ANFIS o Algoritmo Híbrido Difuso-Genético) y el posterior cálculo del Registro de Salida

Registro de Salida. Registro estimado (calculado).

De estas dos definiciones se desprenden dos más:

Área de Entrenamiento. Intervalo de profundidad que se ha seleccionado para el entrenamiento del algoritmo. En este estudio se correspondió con el intervalo comprendido entre 7200'-7500'.

Área de Estudio o Interés. Intervalo de profundidad donde se encuentra la litología productora. En el Pozo GF-87N es 7297'-7377' y en el Pozo GF-109N 8295'-8377'.

Finalmente, es necesario resaltar que, a lo largo del estudio, se utilizaron las siguientes abreviaciones para referirse a los registros:

DEN: Registro Densidad.

VP: Registro Velocidad de Onda P.

VS: Registro Velocidad de Onda S.

SW: Registro Saturación de Agua.

POR: Registro Porosidad.

Vsh: Registro Volumen de Arcilla.

3.1.2. Arreglo de los Registros de Entrada para los Algoritmos

Es importante determinar la manera en que se utilizarán los registros de entrada a cualquiera de los sistemas ya especificados. Para esto, se evaluó la conveniencia de calcular el logaritmo en base 10 a cada registro, y/o realizar una normalización de los mismos.

3.1.2.1. Caso 1: Cálculo de “ \log_{10} ”

Debido a que los registros disponibles presentan una desviación entre ellos supera tres órdenes de magnitud (registro de Velocidad de Onda P está por el orden de 10^4 mientras que el registro de Densidad es igual a 10^0) se consideró en primer lugar calcular el logaritmo en base 10 de cada uno de los registros para así realizar los entrenamientos respectivos. De esta manera, resultaría un conjunto de datos de entrada cuyos valores se encontrarían en el rango entre cero (0) y cinco (5) aproximadamente, más apropiado para los procesos de predicción de los registros (Banchs et al., 2001).

Sin embargo, al llevar a cabo esta operación, ha de notarse que los registros, tanto de Saturación de Agua como de Volumen de Arcilla, adquieren valores entre cero y uno (ambos inclusive), y como \log_{10} de números en ese rango es negativo y además $\log_{10}(0) = -\infty$, se tiene como consecuencia una variación en el comportamiento de dichas gráficas.

Esta alteración puede observarse por ejemplo en la figura 3.3, la cual a su izquierda muestra el registro de Volumen de Arcilla del pozo GF-109N y a su derecha la gráfica correspondiente al cálculo de su logaritmo en base 10. Es evidente la diferencia en el comportamiento entre ambos registros a medida que aumenta la profundidad (además que en un caso las magnitudes son positivas y para el otro son negativas).

Cabe destacar lo que ocurre a una profundidad aproximada de 8320' (ocho mil trescientos veinte pies); allí $Vsh = 0$ por lo que su logaritmo es $-\infty$, dando como consecuencia una breve interrupción de la gráfica en ese punto. Más aún, esta alteración en el patrón de comportamiento, puede inducir a falsas predicciones e interpretaciones.

3.1.2.2. Normalización de los Registros.

Debido a los resultados de la sección anterior, se decidió probar una normalización a los datos antes de proceder a la identificación del modelo. Realizar una normalización en este caso implica cambiar el orden de magnitud de los datos, para la debida realización de las predicciones (Banchs et al., 2001). Esto se hace con el fin de intentar obtener más información haciendo comparables magnitudes que “a priori” no lo son directamente, protegiendo siempre la integridad de los datos (Peña, Carelli & Di Sciascio, 2002).

Peña et al. (2002) recomiendan utilizar una normalización a unos datos cuando se tienen los siguientes problemas:

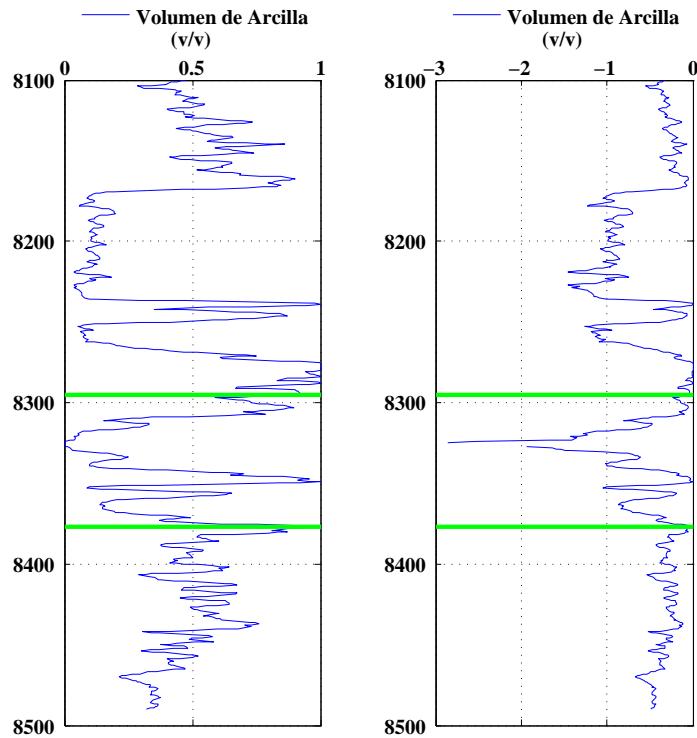


Figura 3.3: Registro Vsh original (izquierda) y Registro Vsh calculado el “ \log_{10} ” (derecha) del Pozo GF-109N.

1. **Las variables involucradas son de distinta naturaleza y magnitud.** De esto surge que los rangos de variación de las mismas pueden ser muy dispares. Esto hace que el proceso de formación de grupos dé prioridad a las variables que tienen un rango de variación mayor.
2. **Las señales involucradas pueden tener un rango de variación no lineal.** Por ejemplo para puntos de muestras donde la pendiente es cercana a 90 grados los valores de la pendiente cambian en una magnitud mayor que para valores cercanos a cero.
3. **Los datos pueden tener una distribución estadística no favorable.** Es conveniente que haya mayor densidad de datos de entrada en donde se producen mayores cambios en la función a identificar. Sucede que no siempre esto se cumple ya que los datos de identificación se obtienen a través de trayectorias donde las señales involucradas están correlacionadas unas con otras a través de fenómenos físicos.

Se sigue que los eventos 1 y 3 ocurren con estos datos. En efecto, el caso 1 ya ha sido descrito en la sección 3.1.2.1 y en el caso 3 ciertamente existen relaciones, por ejemplo, entre Velocidad de Onda S y Densidad.

Existen así distintas ecuaciones para realizar una normalización; sin embargo, se decidió utilizar la propuesta en (Banchs et al., 2001), ya que, entre otras razones, ésta ha sido utilizada en las geociencias⁵:

$$h_i = \frac{r_i - \text{mean}(r_i)}{\max(|r_i|)} + 1 \quad (3.1)$$

donde:

h_i : Registro Normalizado.

$\text{mean}(r_i)$: Es el valor promedio del registro en el área de interés.

$\max(|r_i|)$: Es el valor absoluto de la desviación más larga.

3.2. Criterios de Comparación

Para tener un criterio cuantitativo de evaluación para medir la efectividad al aplicar cada uno de los algoritmos, se seleccionaron dos ecuaciones diferentes que se presentan brevemente a continuación:

1. RAÍZ DEL ERROR CUADRÁTICO MEDIO (en inglés “Root Mean Square Error” (*RMSE*)). Se utilizará para medir la diferencia que existe entre los valores calculados y los valores medidos del registro que se desea predecir. Éste se define de la siguiente manera (Gutiérrez, 2010):

$$RMSE = \sqrt{\frac{\sum_{i=1}^{\infty} (Y'_i - Y_i)^2}{N}} \quad (3.2)$$

donde:

Y'_i : Valores obtenidos de la predicción.

Y_i : Valores Reales (medidos)

N : Número de muestras.

2. COEFICIENTE DE DETERMINACIÓN (R^2): Es un valor estadístico que proporciona información sobre la **relación lineal existente entre dos variables**. Puede

⁵(Banchs et al., 2001), garantiza que con la ecuación 3.1 el valor promedio del registro normalizado es 1.

interpretarse como el porcentaje de variación de la variable predictora explicada o debida a la recta de regresión (Lahura, 2003).

Es importante notar que su uso sólo tiene sentido si la relación bivariada a analizar es del tipo lineal. Si ésta no fuera **no lineal**, el coeficiente de correlación sólo indicaría la ausencia de una relación lineal más no la ausencia de relación alguna. Debido a esto, muchas veces se define -de manera más general- como un instrumento estadístico que mide el grado de asociación lineal entre dos variables (Lahura, 2003).

En el caso de estudio, se utilizará para evaluar el grado de asociación entre los datos reales y los datos arrojados en las predicciones. Ésta será necesariamente lineal para aquellos casos en que se haya realizado una predicción “satisfactoria” (R^2 cercano a 1).

Para mostrar su ecuación correspondiente, primero ha de observarse el cálculo del *Coeficiente de Correlación*:

$$r = \frac{\sum_{i=1}^n (x_i y_i)}{\sqrt{\sum_{i=1}^n x_i^2} \sqrt{\sum_{i=1}^n y_i^2}} \quad (3.3)$$

donde:

$x_i = X_i - \tilde{X}$: Desviación de la i-ésima observación de la variable X respecto de su valor promedio observado X_i

$y_i = Y_i - \tilde{Y}$: Análogo al anterior, se resume simplemente como la Desviación de Y_i

Elevando al cuadrado la ecuación 3.3 queda:

$$R^2 = \frac{SSE}{\sum_{i=1}^n y_i^2} + 1 \quad (3.4)$$

Cuando todos los puntos se encuentran sobre la recta de regresión estimada, es decir, “el ajuste es perfecto”, la *Suma de Cuadrados de Residuos*, “SSE”, toma el valor cero y, por tanto, $R^2 = 1$

3.3. Técnicas de Predicción: AG-ANFIS

A continuación se muestra el procedimiento que se siguió en la elaboración de los códigos necesarios para la predicción de registros utilizando:

- Algoritmos Genéticos.

- Sistema de Inferencia Neuro-Difusa (ANFIS).

3.3.1. Algoritmos Genéticos

La estimación de un registro de pozo mediante este método, consiste en el cálculo de los parámetros de una ecuación que permita asociar el(s) Registro(s) de Entrada con el *Registro de Salida (Velocidad de Onda S)* (Banchs et al., 2001). Para ello, se tomaron en cuenta dos ecuaciones para realizar la predicción con cada una de ellas:

$$VS = a_1 \times REG_1 + a_2 \times REG_2 + \dots + a_n \times REG_n \quad (3.5)$$

$$VS = a_1 \times REG_1^{b_1} \bullet a_2 \times REG_2^{b_2} \bullet \dots \bullet a_n \times REG_n^{b_n} \quad (3.6)$$

donde:

n: es el número de Registros utilizados para realizar la predicción.

REG: Registros de Entrada.

VS: Registros de Salida (Velocidad de Onda S predicho).

a y *b*: Parámetros a optimizar mediante Algoritmos Genéticos.

De ambas ecuaciones puede decirse lo siguiente:

1. La Ecuación 3.5, que en lo sucesivo se llamará ECUACIÓN 1 AG, es lineal, la cual resulta una simplificación de la ecuación propuesta en (Banchs et al., 2001).
2. La Ecuación 3.6, que en lo sucesivo se llamará ECUACIÓN 2 AG, resulta una función polinómica propuesta en (Cuddy & Glover, 2002), un poco más compleja que la anterior, donde además se determina el signo (\bullet) de los coeficientes (que puede ser positivo o negativo).

Como ya se ha mencionado, los algoritmos genéticos representan un método de optimización de funciones, que permiten encontrar soluciones potenciales donde una función *F* se maximiza o minimiza, según sea el interés de estudio. Dicho de otro modo entonces, para realizar la predicción, en cada caso es necesario encontrar los valores de los parámetros $[a]$ y $[a, b]$ de las ecuaciones 3.5 y 3.6 respectivamente, que relacionen el comportamiento de la propiedad real con los valores de los registros disponibles de entrada, de manera que la ecuación 3.2 sea mínima.

Es decir, hallar los parámetros $[a]$ o $[a, b]$ de modo que:

$$RMSE \longrightarrow 0$$

Observando detenidamente la Ecuación 3.2, lo anterior implica que:

$$\sum_{i=1}^{\infty} (Y'_i - Y_i)^2 \longrightarrow 0 \quad (3.7)$$

donde:

Y'_i es el registro de Velocidad de Onda S calculado mediante las ecuaciones 3.5 o 3.6.

Y_i es el registro de Velocidad de Onda S medido.

Teniendo claro el fundamento de los Algoritmos Genéticos en esta investigación, se procedió entonces a revisar las opciones de su utilización en **Matlab versión R2009a**. Allí se pueden usar los comandos de AG de dos formas:

1 .Utilizando la Interfaz Gráfica de Usuario (“GUI” por sus siglas en inglés *Graphical User Interfaces*).

Tecleando en la ventana de comandos de Matlab “`optimtool('ga')`”, se provee, en una ventana sencilla, de todas las herramientas necesarias para su utilización, así como los elementos del algoritmo que se pueden modificar (como por ejemplo tamaño de la población, migración, cruce, número de generaciones, entre otras).

2. Utilizando las funciones de su Librería.

Esta forma implicaría la utilización de las funciones correspondientes desde la ventana de comandos de Matlab, lo que plantea la necesidad de programar todos los aspectos que, utilizando la GUI, se simplifican.

La particularidad en la utilización de esta técnica en la investigación exige que se trabaje de la segunda forma, teniendo en cuenta las siguientes funciones:

- “**ga**”. Busca el mínimo de una función de entrenamiento “`fun_ent`” (Véase la ecuación 3.8) utilizando Algoritmos Genéticos.
- “**gamultiobj**”. Busca el mínimo de múltiples funciones utilizando Algoritmos Genéticos.
- “**gaoptimget**”. Obtiene las opciones de la estructura del algoritmo.

- “*gaoptimset*”. Crea la estructura de opciones del algoritmo.

Particularmente, tecleando en la ventana de comandos de Matlab *help ga* se obtiene la sintaxis de la función “*ga*”, que aplicada a este trabajo, es de la siguiente manera:

$$[x, fval] = ga(func_ent, nvars, A, b, Aeq, beq, LB, UB, nonlcon, opciones) \quad (3.8)$$

donde:

x: **Valores mínimos.** Son los valores mínimos adoptados por la función de entrenamiento.

Se corresponde con los parámetros de las ECUACIONES 1 o 2 AG buscados.

fval: **Mínimos de la función.** Magnitud de la función de entrenamiento cuando ésta adquiere los valores de *x*.

func_ent: **Función de Entrenamiento.** En este caso, es la ecuación 3.7. Sin embargo, como el término Y'_i depende de la utilización de las ecuaciones 3.5 o 3.6, se codificaron dos funciones, “*fun_lineal_entrenamiento*” y “*fun_polinomial_entrenamiento*”, dependiendo de si se utiliza en la predicción, la primera o la segunda función, respectivamente.

nvars: **Número de variables.** En este caso, es el número de registros de entrada utilizados en la predicción.

$(A, b) - (Aeq, beq)$. Parámetros que condicionan la búsqueda de las soluciones de la función “*ga*”. Estas opciones no se desean en el cálculo de parámetros, por lo que se escriben corchetes vacíos [].

(LB, UB): **Límites.** Superior (“*UB*” por sus siglas en inglés *Upper Bound*) e Inferior (“*LB*” por sus siglas en inglés *Lower Bound*) de las soluciones obtenidas por el algoritmo. Como en este caso inicialmente se desconoce el rango de valores que pueden adoptar los parámetros, entonces estos campos también se colocan con corchetes vacíos [].

nonlcon: **Límites no lineales.** Sujeta el proceso de minimización a los límites definidos por la función *nonlcon*. Tampoco deseado, igualmente se colocó este campo vacío.

opciones: **Estructura del Algoritmo.** Mediante la creación de la estructura “*opciones*” (a través de la función “*gaoptimset*”) se pueden modificar los elementos que tiene por defecto el algoritmo en Matlab (Véase la sección 3.4.1).

Finalmente, con todas estas opciones especificadas, se procedió a elaborar dos funciones “*ag_lineal*” y “*ag_polinomial*”, que, utilizando sus funciones de entrenamientos⁶, permitan

⁶Estas son “*fun_lineal_entrenamiento*” o “*fun_polinomial_entrenamiento*” respectivamente.

calcular los parámetros (dependiendo de la ecuación utilizada) que mejor ajustan al registro de salida (Velocidad de Onda S) en función de uno o varios Registros de Entrada⁷

3.3.2. ANFIS

Al igual que en algoritmos genéticos, ANFIS en **Matlab versión R2009a** puede utilizarse de dos formas:

- 1. Mediante la GUI.** Tecleando en la ventana de comandos de Matlab “*anfisedit*”, se obtiene una ventana en la que se puede entrenar un FIS accediendo a una variedad de opciones para visualizar los parámetros internos y los resultados obtenidos.
- 2. Utilizando las funciones de su Librería.** A diferencia de Algoritmos Genéticos, la librería de funciones de Lógica Difusa es mucho más amplia, tiene más de 400 funciones que permiten su utilización.

En este caso, en cambio, debido a la practicidad que resulta de la utilización de la “gui” de ANFIS, se decantó por utilizar ésta. Esta decisión simplifica la codificación de una función para la predicción de registros mediante este método.

Tomando en cuenta que el primer paso, al ejecutar el Editor de ANFIS, es cargar en el sistema los datos de entrenamiento⁸ (Ver sección 3.6) se hizo necesario crear una función que, antes de realizar esa ejecución, creara el archivo en el directorio para el entrenamiento del FIS dependiendo de los registros de entrada, donde cada columna de ese archivo representa un registro diferente; con la particularidad de que, además, debe contener una columna extra con el registro medido (Ver sección 3.6). Luego de finalizar el uso de dicho editor, se cargara el FIS generado para continuar con la predicción. La función que realiza esta serie de pasos se denominó “*log_dif*” (ésta se presenta en la página 179).

⁷El desarrollo de estas dos funciones se muestra en el Apéndice C en las páginas 163 y 165.

⁸Puede hacerse bien leyendo un archivo desde el directorio utilizado o leyendo una variable desde el espacio de trabajo.

3.4. Pruebas Preeliminares

3.4.1. Algoritmos Genéticos

Las soluciones finales en el entrenamiento de un Algoritmo Genético pueden ser sensibles a las condiciones iniciales de éste, ya que éstos influyen en la forma en que buscará las posibles soluciones (Saemi & Ahmadi, 2008). Por ello, antes de la generación de los resultados finales, se hizo necesario la realización de algunas pruebas (en la predicción del registro de Velocidad de Onda S) utilizando diferentes combinaciones entre algunos de esos parámetros, para buscar los que dieran los mejores resultados; éstos son:

- Tamaño de la población.
- Función de Cruce.
- Función de Selección.
- Función de Mutación.
- Número de Generaciones.
- Elite Count.

Las combinaciones probadas se muestran en la tabla 3.1

Tabla 3.1: Pruebas Realizadas modificando las condiciones iniciales de los Algoritmos Genéticos

Opción	Prueba 1	Prueba 2	Prueba 3
Tamaño de la Población	10	20	40
Función de Cruce	@crossoversinglepoint	@crossoverscattered	@crossovertwo point
Elite Count	1	2	4
Función de Selección	@selectionroulette	@selectionstochunif	@selectionstochunif
Función de Mutación	@mutationuniform	@mutationgaussian	@mutationgaussian
Generaciones	60	80	100

3.4.2. ANFIS

Análogo al apartado anterior, las condiciones inciales con las que se crea un FIS utilizando ANFIS pueden incidir en el resultado final, a saber, el número de funciones de membresía por cada variable de entrada y, por consiguiente, el número de reglas difusas total del sistema.

La Interfaz Gráfica de Usuario de ANFIS en Matlab, provee dos alternativas en la búsqueda y utilización de las de reglas difusas. Éstas son:

Grid Partition. En donde el usuario debe señalar los siguientes tres parámetros: número de reglas difusas por variable de entrada, la función de membresía a utilizar en ellas y el tipo de sistema de inferencia Takagi-Sugeno-Khan a generar: “constante” o “lineal”.

Subtractive Clustering. A diferencia del anterior, al seleccionar ésta opción el programa, de acuerdo al agrupamiento que presenten los datos de entrada al sistema, encuentra de manera autónoma estos tres parámetros.

De manera que, en primer lugar, se efectuaron múltiples predicciones del registro de Velocidad de Onda S con uno y dos registros de entrada (en función de los registros disponibles) empleando GRID PARTITION de ANFIS, lo que conlleva a variar manualmente el número de reglas difusas en cada caso con la finalidad de determinar la mejor combinación de reglas (la que arroje la menor magnitud del RMSE) en el cálculo del Registro de Velocidad de Onda S.

Para ello, en las pruebas efectuadas, los número de reglas difusas seleccionados fueron: cuatro (4), seis (6), ocho (8) y nueve (9), lo que implica combinaciones diferentes según el número de datos de entrada que tenga el sistema, como se muestra a continuación:

Pruebas #1: Un Registro de Entrada. Se seleccionaron todos los registros disponibles en el pozo de Entrenamiento GF-87N:

- Densidad.
- Velocidad de Onda P.
- Saturación de Agua.
- Porosidad.
- Volumen de Arcilla.

Debido a que hay una sola entrada, no hay combinación posible de reglas difusas, por lo que éstas fueron 4, 6, 8 y 9 para cada entrada.

Pruebas #2: Dos Registros de Entrada. De los registros disponibles en el pozo GF-87N, se utilizaron como entrada al Sistema de Inferencia Difusa todas la permutaciones posibles entre ellos, tal y como se muestra en la siguiente lista:

- Densidad - Velocidad de Onda P.
- Densidad - Saturación de Agua.
- Densidad - Porosidad.
- Densidad - Volumen de Arcilla
- Velocidad de Onda P - Saturación de Agua.
- Velocidad de Onda P - Porosidad.
- Velocidad de Onda P - Volumen de Arcilla..
- Saturación de Agua - Porosidad.
- Saturación de Agua - Volumen de Arcilla.
- Porosidad - Volumen de Arcilla.

En este caso, se utilizaron todas las combinaciones posibles de reglas difusas para dos variables de entrada:

- Para 4 reglas difusas, única combinación [2 2]. Esto significa: dos funciones de membresía para la primera entrada y dos funciones de membresía para la segunda entrada.
- Para 6 reglas difusas, combinaciones [3 2] (tres funciones de membresía para la primera entrada y dos funciones de membresía para la segunda entrada) y [2 3] (dos funciones de membresía para la primera entrada y tres funciones de membresía para la segunda entrada)
- Para 8 reglas difusas, combinaciones [2 4] y [4 2]. Análogo a los anteriores, son dos y cuatro funciones de membresía para el primer caso, y cuatro y dos funciones para el segundo caso.
- Para 9 reglas difusas, la única combinación posible [3 3]: tres funciones de membresía para cada entrada.

Pruebas #3: Tres Registros de Entrada. Como puede notarse, para este caso, el número de permutaciones posibles es mayor al anterior. Debido a ésto, se decidió

seleccionar tres registros que presentaran los valores más bajos del RMSE en los casos anteriores y en función de éstos realizar las pruebas. Para este caso, se usaron las siguientes combinaciones:

- Para 4 reglas difusas: [1 2 2], [2 1 2] y [2 2 1].
- Para 6 reglas difusas: [1 3 2], [2 1 3] y [3 2 1].
- Para 8 reglas difusas: [1 4 2], [2 1 4] y [4 2 1].
- Para 9 reglas difusas: [1 3 3], [3 1 3] y [3 3 1].

Finalmente, para todos los casos, se utilizó la función de membresía GAUSSIANA (Véase la página 1.2.1.2), así como el sistema “lineal” tipo Sugeno.

3.5. Algoritmo Híbrido Difuso-Genético

La generación de un algoritmo híbrido basado en Lógica Difusa y Algoritmos Genéticos puede realizarse, en principio, de dos maneras (Saemi & Ahmadi, 2008) :

1. Utilizar los algoritmos genéticos para optimizar los parámetros de lógica difusa. Podrían utilizarse los Algoritmos Genéticos para optimizar, por ejemplo, el número de funciones de membresía para cada entrada, así como también para calcular los parámetros asociados a dichas funciones. Efectivamente, en (Saemi & Ahmadi, 2008) utilizan este método para la predicción de permeabilidad ⁹.
2. Utilizar la lógica difusa para encontrar parámetros de algoritmos genéticos. En este caso, aspectos como tamaño de la población, número de individuos que pasa a la siguiente generación, mutación y cruce, podrían ser ajustados utilizando lógica difusa.

Ambos puntos requerirían una revisión de las funciones de ambas técnicas en Matlab, donde se observe el funcionamiento interno de los mismos. En el caso de Algoritmos Genéticos, esto puede hacerse, buscando en la librería adecuada; los códigos pueden ser fácilmente visualizados, y con la ayuda de los comentarios, entender su ejecución en la resolución de problemas.

Sin embargo, para el caso de ANFIS esto no es posible. Éste es un archivo ejecutable de Matlab, compilado en lenguaje C, que no está disponible para el usuario, esto con dos

⁹Cabe destacar que los autores de este trabajo no utilizaron Matlab para realizar la integración de los métodos

fines principalmente, proteger la confidencialidad de los códigos y para hacer más eficiente su funcionamiento.

A pesar de todo, debido a la imposibilidad de llevar a cabo alguno de los dos métodos mencionados, se decidió realizar un algoritmo híbrido un poco más sencillo. Éste consiste simplemente en optimizar (mediante algoritmos genéticos) las funciones de salida creadas en el FIS. Esto se llevó a cabo de la siguiente manera:

Paso 1. Generación de un FIS

EL fundamento del Algoritmo Híbrido propuesto es, basado en las reglas difusas obtenidas con ANFIS, optimizarlas mediante Algoritmos Genéticos. Por lo anterior, el método sólo puede llevarse a cabo si y sólo si previamente se ha generado un Sistema de Inferencia Difusa (SID).

Paso 2. Almacenamiento del FIS creado

Cuando se genera un Sistema de Inferencia Difusa en Matlab, se genera un archivo tal y como se muestra en la figura 3.4, en donde el FIS se obtiene utilizando como:

- Registros de entrada: Densidad-Vp.
- Método de partición de los datos: “Grid Partition”
- Combinación de reglas difusas [3 2]
- Función de Membresía: Gaussiana.

En el rectángulo de colores están resaltados las cuatro divisiones principales del archivo. Éstas son:

Color Azul. Muestra las opciones utilizadas durante el proceso interno del Sistema de Inferencia Difusa (SID), tales como: número de entradas, número de salidas, métodos “And” y “OR”, método de Defusificación, entre otros.

Color Rojo. Muestra los parámetros de las funciones de membresía para los datos de entrada. En este caso, son *Baja-Media-Alta* para el caso del registro Densidad y *Baja-Alta* para el registro de entrada VP.

Color Verde. Muestra los parámetros de las funciones de salida (reglas difusas).

```
Name='Prediccion'
Type='sugeno'
Version=2.0
NumInputs=2
NumOutputs=1
NumRules=6
AndMethod='prod'
OrMethod='probOr'
ImpMethod='prod'
AggMethod='sum'
DefuzzMethod='wtaver'
```

```
[Input1]
Name='Densidad'
Range=[0.490109048569786 1.93190227435229]
NumMFs=3
MF1='Baja':'gaussmf',[0.337168349318629 0.628866893661455]
MF2='Media':'gaussmf',[0.390295995340782 1.19785348207131]
MF3='Alta':'gaussmf',[0.271597073840834 1.828809106572]
```

```
[Input2]
Name='VP'
Range=[0.477729752940691 1.71734842295055]
NumMFs=2
MF1='Baja':'gaussmf',[0.252764971260955 0.401767456522791]
MF2='Alta':'gaussmf',[0.229104461978412 1.80365772084433]
```

```
[Output1]
Name='output'
Range=[0.521568077282655 1.58580043150082]
NumMFs=6
MF1='Ec.1':'linear',[0.380965358737562 -0.216222281390819 1.08071369077505]
MF2='Ec.2':'linear',[-6.11038996889616 0.935080671073026 1.39492361864975]
MF3='Ec.3':'linear',[0.903744272176375 0.932684766201113 -0.843795324462481]
MF4='Ec.4':'linear',[-9.50600518933752 1.14752761694309 11.8765449388074]
MF5='Ec.5':'linear',[1.00443686113406 0.693565280549736 -1.78578522125004]
MF6='Ec.6':'linear',[-11.0196098124521 1.65023688994539 19.0392032157344]
```

```
[Rules]
1 1, 1 (1) : 1
1 2, 2 (1) : 1
2 1, 3 (1) : 1
2 2, 4 (1) : 1
3 1, 5 (1) : 1
3 2, 6 (1) : 1
```

Figura 3.4: Archivo de extensión .FIS generado por ANFIS para 2 Registros de Entrada: Densidad-VP

Color Morado. Muestra las reglas difusas. La Columna 1 se corresponde con la entrada 1 (Densidad) y la columna 2 con la entrada 2 (VP). Por ejemplo, la fila 3 de esa matriz se leería así: “SI Densidad es **Media** y Vp es **Alta** ENTONCES Salida es Ec.3”

Sin embargo, para poder leer y modificar eficientemente los parámetros de las reglas difusas (en verde), se decidió importar este archivo al Espacio de trabajo de matlab, esto con la función “*readfis*” (Véase la página 179). La estructura es diferente, aunque contiene exactamente la misma información ¹⁰

Paso 3. Lectura de los datos apropiados

Siguiendo el mismo ejemplo dado de la tercera fila del recuadro morado de la figura 3.4 (tercera regla difusa), se tiene que hay que encontrar primero los valores de Densidad que se encuentran en el rango “Media”. Para ello se utilizan los parámetros de la función de membresía número dos (MFS): [0.390295995340782 1.19785348207131]. Como ya se mencionó, la función de membresía seleccionada para las predicciones es la Gaussiana, por lo que éstos parámetros se corresponden con los valores de $[\sigma, c]$ que se explican en la ecuación 1.5, página 13 .

Con éstos parámetros, despejando x de la ecuación 1.5 y evaluando cuando $f(x) = 0,5$, se logra calcular el rango de valores Densidad (a media altura de la función gaussiana) que pertenece al rango “Media”. Igual se hace con VP y el rango “Alta”. Y finalmente se buscan sólo los intervalos de profundidad que cumplen con ambos señalamientos.

Paso 4. Optimización de los coeficientes Continuando con el ejemplo, se tomarían los datos la Ec. 3 de Salida para optimizarlos mediante algoritmos genéticos (sólo con los valores seleccionados del paso anterior). La función de entrenamiento es la misma ecuación 3.7, con la salvedad que Y'_i se calcula de la forma:

$$Y'_i = a_1 \times REG_1 + a_2 \times REG_2 + \dots + a_{n-1} \times REG_{n-1} + a_n \quad (3.9)$$

La función de entrenamiento se denominó “*fun_lineal_fis*”. Finalmente, la función realiza toda este proceso para todas las reglas difusas, generando como salida:

- Registro de Salida optimizado. Pozo de Entrenamiento
- Registro de Salida optimizado. Pozo Control

¹⁰Se muestra el archivo .FIS ya que es un buen resumen de la estructura generada en el espacio de trabajo y, por razones prácticas, más fácil de explicar y entender.

- Las reglas difusas que no fueron optimizadas.
- Parámetros del FIS creado por ANFIS.
- Parámetros Optimizados del FIS mediante Algoritmos Genéticos.

3.6. Ejecución de los tres Algoritmos

Finalmente, con todas las funciones, se procedió a elaborar el código final. El cual está estructurado como lo indica el fluograma que se presenta en las figuras 3.5 y 3.6. De manera general:

Carga de los Registros de Pozo. El primer paso para la predicción consiste en la carga de los registros de pozo, que generalmente están en formato *.LAS* o “*Log ASCII Standard*” o *ACII*. Hacía falta una función que obtuviera del archivo sólo la matriz cuyas columnas contienen los datos de los registros, omitiendo toda la información adicional que en ellos se da.

Para llevar a cabo este propósito se tomó como base el archivo *lectlasfilex.m*, el cual fue generado por Andrés Landa en su tesis “Diseño de un algoritmo para la visualización y procesamiento automatizado de registros de pozo”. Esta función fue modificada, denominándose “*carga_registro.m*”, en la cual se lleva a cabo esta carga independientemente si el registro está en formato LAS o ASCII. La salida de esta función es una matriz donde cada columna contiene un registro de pozo.

Gráficas preliminares de los registros de pozo. Posteriormente a la carga de los pozos, se genera una gráfica preliminar de los mismos, con la finalidad de evaluar el registro a predecir, así como seleccionar el intervalo de profundidad en el que desea realizar el entrenamiento de los sistemas de predicción.

Selección de los datos para la predicción. Como ya se mencionó en la sección 3.1.1, para realizar las predicciones con cada uno de los métodos, se tomó el intervalo comprendido entre 7200’-7500’.

Escogencia del método de predicción. A continuación, se pueden escoger entre dos métodos: Algoritmos Genéticos (Ecuación 1 ó 2) y el Sistema de Inferencia Neuro-Difuso. En caso del segundo, existen dos alternativas más, utilizar “Grid Partition” o “Subtractive Clustering”.

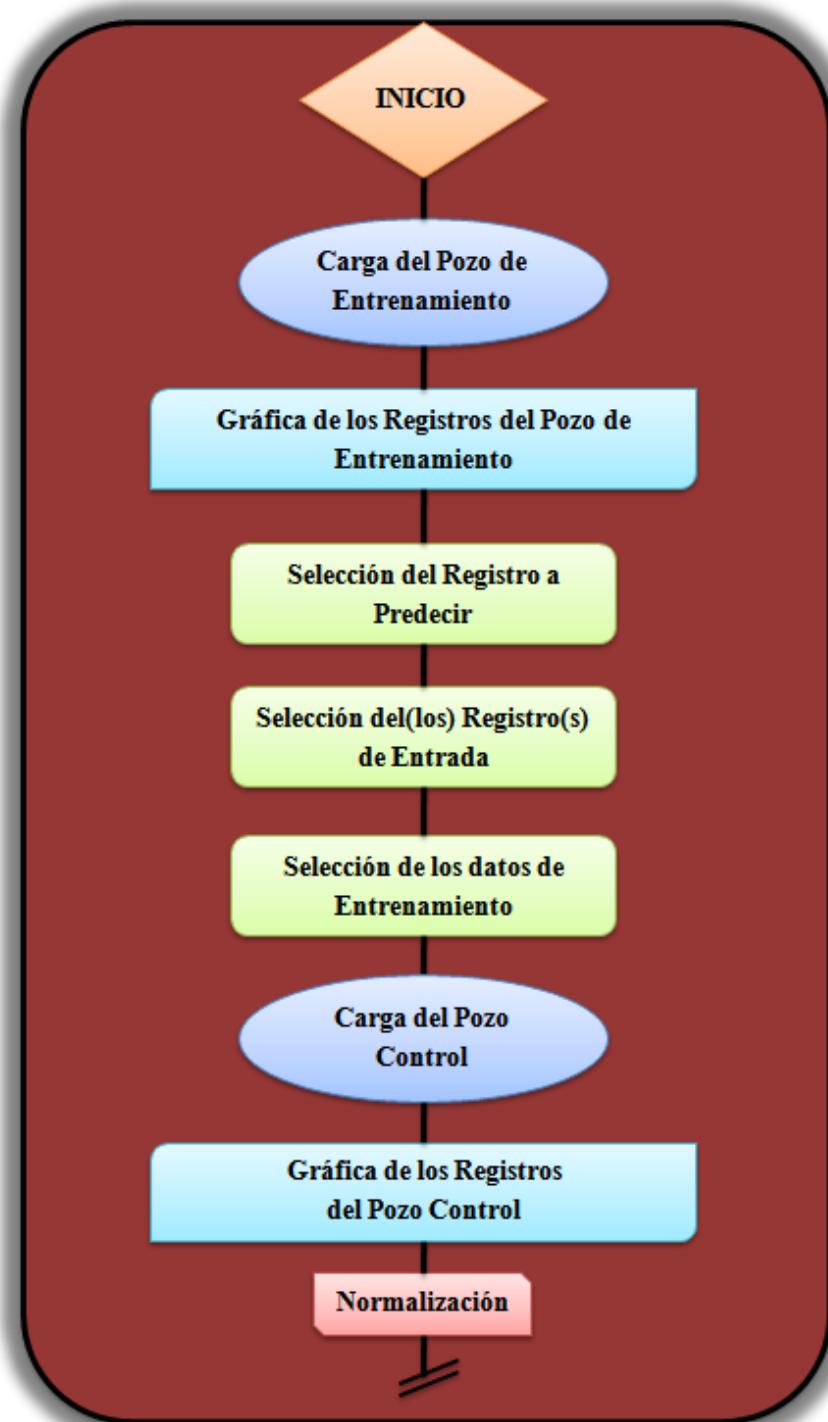


Figura 3.5: Primera parte del flujoograma del programa final

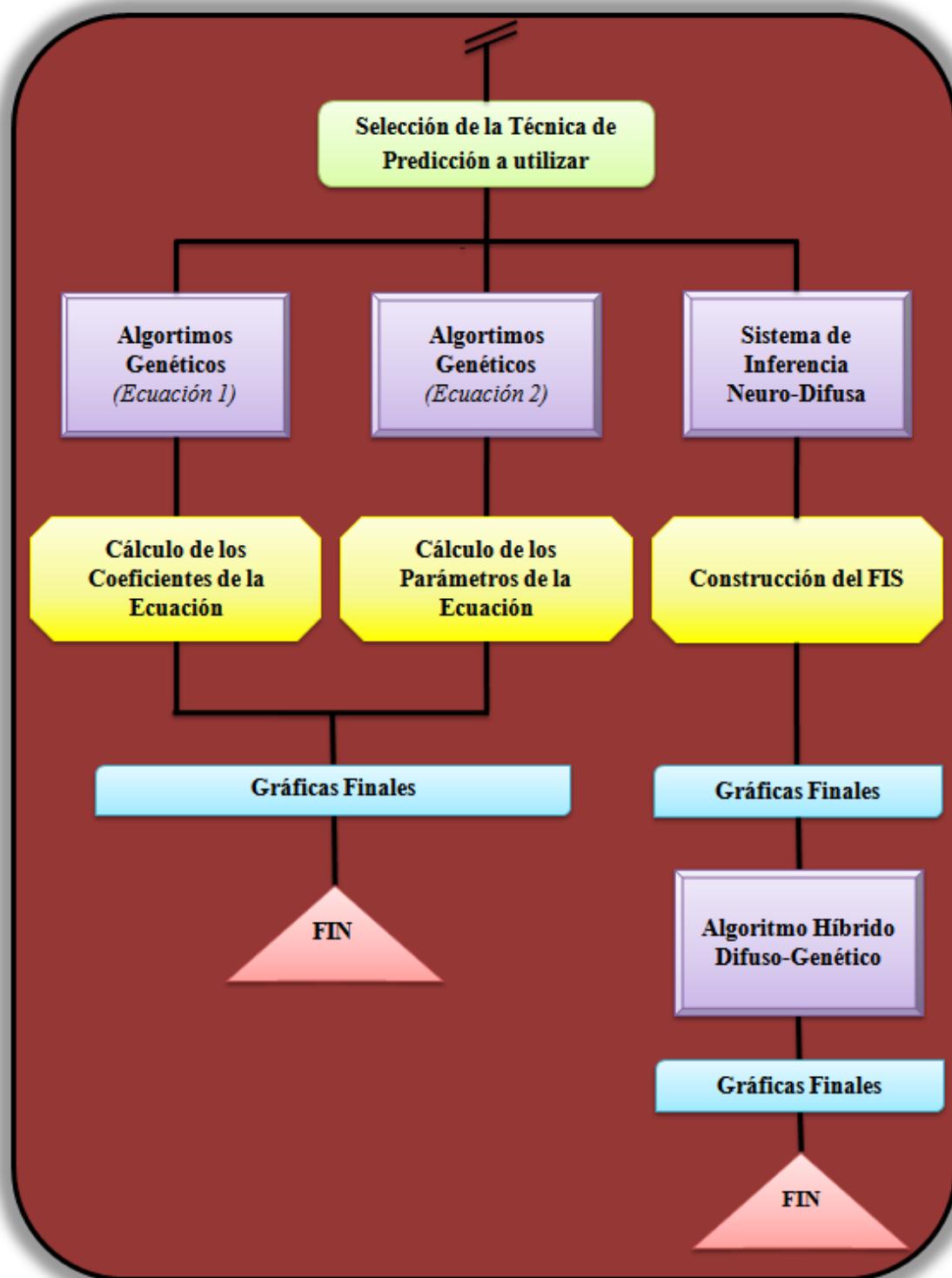


Figura 3.6: Segunda parte del flujoGRAMA del programa final

Para la generación de los cálculos finales en esta investigación, es necesario resaltar que fueron utilizadas ambas opciones. Es decir, se efectuaron primero las predicciones con “Grid Partition”, generando un SID tipo lineal, utilizando la función de membresía *Gaussiana* y con las mejores combinaciones de reglas difusas que fueron obtenidas en las pruebas efectuadas en la sección anterior.

Posteriormente se realizaron las predicciones con los mismos registros de entrada que el caso anterior salvo que se utilizó “Subtractive Clustering”, en donde éstas opciones son generadas automáticamente.

Finalmente, luego de la generación de cada Sistema de Inferencia Difusa, se hizo efectiva la Hibridización.

Graficación de los registros finales. Finalmente, utilizando la función “registro_final” se generan las gráficas correspondientes de VS predicho y VS real.

CAPÍTULO 4

RESULTADOS Y ANÁLISIS

Primeramente, antes de mostrar el producto de las predicciones del registro de Velocidad de Onda S, se presentan, en la sección 4.1, los resultados de las pruebas preeliminares señaladas con anterioridad (Algoritmos Genéticos y ANFIS):

- En la sección 4.1.1, donde se modificaron las condiciones iniciales de los Algoritmos Genéticos que fueron especificadas en la tabla 3.1, página 59.
- En la sección 4.1.2 donse se muestra la búsqueda de la mejor combinación de funciones de membresía por entrada (lo que conlleva al número de reglas difusas utilizadas en el sistema) en el Sistema de Inferencia Neuro-Difuso (ANFIS).

Luego, a fin de evaluar y visualizar la efectividad del Algoritmo Híbrido Difuso-Genético con respecto a los otros dos métodos -Algoritmos Genéticos y el Sistema de Inferencia Neuro-Difusa (ANFIS)- los resultados se presentarán de acuerdo a la cantidad de registros de entrada utilizados en los respectivos cálculos:

- En la sección 4.2 se encuentran los resultados obtenidos para un registro de entrada.
- En la sección 4.3 se encuentran los resultados obtenidos utilizando dos registros de entrada.
- En la sección 4.4 se encuentran los resultados obtenidos usando como entrada a los algoritmos tres (3) registros.

La salida de cada ejecución genera, en cada caso, lo siguiente:

Algoritmos Genéticos: Los parámetros correspondientes a las dos ecuaciones:

1. En el caso de la “**Ecuación 1 AG**” (Véase la página 55) los coeficientes que multiplican a cada registro de entrada.
2. En el caso de la “**Ecuación 2 AG**” (Véase la página 55) los parámetros que multiplican a cada registro de entrada, así como sus exponentes respectivos.

Sistema de Inferencia Neuro-Difusa (“ANFIS”): Un archivo de extensión “.FIS” (similar al de la figura 3.4) que, entre otras cosas, muestra los parámetros de las funciones de membresía, así como los coeficientes asociados a las reglas difusas.

Algoritmo Híbrido Difuso-Genético (“AH”): El mismo archivo de extensión “.FIS” y con la misma información, con la salvedad que los parámetros de las reglas difusas serán diferentes debido a que son el producto del proceso de optimización.

Además, todos tienen en común en la salida el cálculo de sus respectivos valores de R^2 y del Error Cuadrático Medio “RMSE”¹.

4.1. Pruebas Preliminares

4.1.1. Algoritmos Genéticos

Las pruebas que se muestran en la tabla 3.1 (página 59) fueron realizadas seleccionando un registro de entrada (por separado): primero Densidad y luego Velocidad de Onda P en la **ECUACIÓN 1 AG**, obteniéndose los resultados que se muestran en las tablas de la 4.1 a la 4.4 y en las figuras de la 4.1 a la 4.5.

Tabla 4.1: Pruebas realizadas, en el Pozo de Entrenamiento, variando las condiciones iniciales de Algoritmos Genéticos. Registro de Entrada: DEN.

	Prueba 1	Prueba 2	Prueba 3
RMSE	0,335	0,408	0,421
R^2	0,1722	0,1722	0,1722

De manera que, para comenzar con el análisis de la influencia de las condiciones iniciales señaladas en la sección 3.4.1 en la predicción del registro de Velocidad de Onda S, ha de observarse el comportamiento del RMSE y del R^2 en cada una de las pruebas.

¹ Véase la sección 3.2, página 53.

Tabla 4.2: Pruebas realizadas, en el Pozo Control, variando las condiciones iniciales de Algoritmos Genéticos. Registro de Entrada: DEN.

	Prueba 1	Prueba 2	Prueba 3
RMSE	0,325	0,462	0,476
R^2	0,1353	0,1353	0,1353

En las tablas 4.1 y 4.2 arriba mostradas se encuentran los valores obtenidos para ambos términos utilizando como entrada el registro DENSIDAD; en ellas puede notarse cómo el valor del R^2 no presenta variación en las pruebas 1, 2 y 3 tanto en la predicción en el Pozo de Entrenamiento (GF-87N) como en el Pozo Control (GF-109N), siendo iguales a 0,1722 y 0,1353 respectivamente.

Caso contrario ocurre con los valores del RMSE, en donde en estos casos se presenta una variación que puede verse más claramente en la figura 4.1. Bien sea para la predicción realizada en el Pozo de Entrenamiento como en el Pozo Control, el menor valor de éste término se obtiene en el caso de la prueba número 1, en donde el tamaño de la población y el número de generaciones es menor con respecto a los otros casos ².

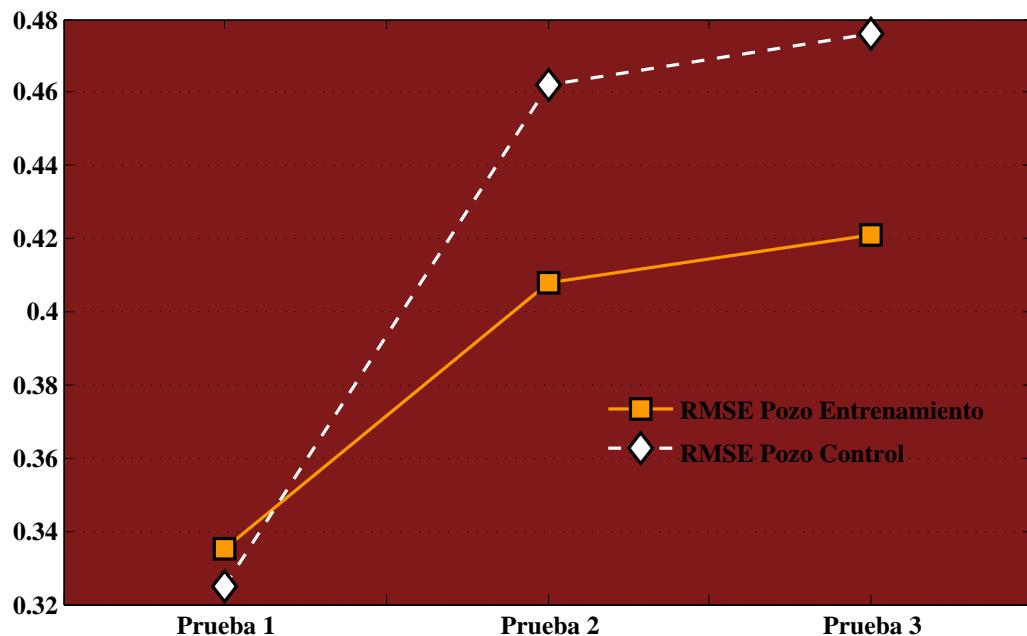


Figura 4.1: Variación del RMSE en las pruebas realizadas con el registro de entrada: Densidad

Este hecho, en donde el R^2 no varía mientras el RMSE si lo hace, indica que el comportamiento general de la curva de Onda S predicha no sufre grandes cambios en las tres

²Esto si se observa nuevamente la tabla 3.1

pruebas realizadas. En efecto, examinando la figura 4.2, donde se muestran las predicciones realizadas en el Pozo de Entrenamiento, puede advertirse como la tendencia general del registro predicho se mantiene en las tres pruebas, específicamente en las profundidades 7000'-7150', 7200'-7300', 7370'-7500' (la gráfica que muestra el comportamiento de los valores predichos en función de los valores de reales se muestra en la figura 4.5(a)).

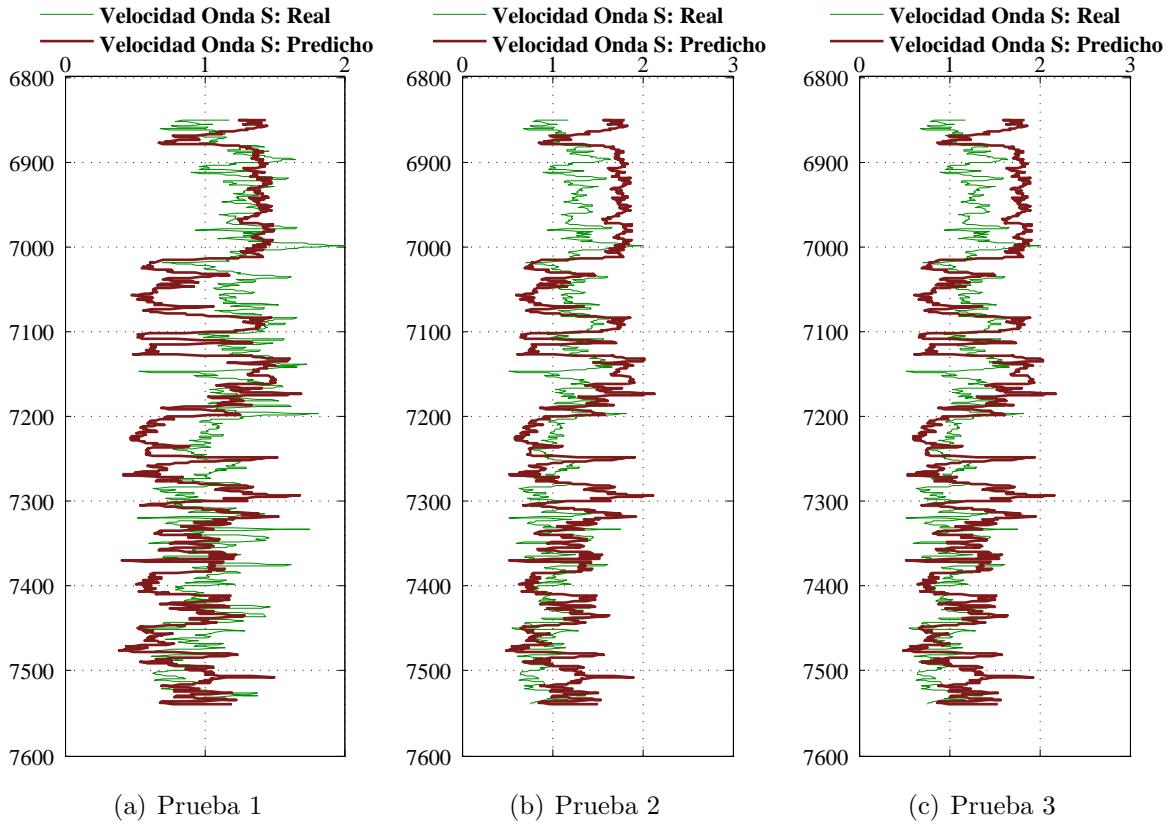


Figura 4.2: Predicciones realizadas, en el Pozo de Entrenamiento, modificando las condiciones iniciales de los Algoritmos Genéticos. Registro de Entrada: Densidad

Sin embargo, hay pequeñas diferencias, en donde la curva ajusta mejor el comportamiento del registro real, éstas son las que resultan en una disminución en el valor del RMSE en la prueba 1, como en los casos de los intervalos comprendidos entre 6900'-7000' y 7150'-7200'.

Análogo procedimiento se realizó, empero tomando ahora como entrada en el Algoritmo Genético el registro de VELOCIDAD DE ONDA P. Los valores del RMSE y del R^2 obtenidos al ejecutar la predicción se exponen en las tablas 4.3 y 4.4. Acá ocurre lo mismo que en el caso anterior, el valor del R^2 no varía en las tres pruebas, ahora siendo 0,4910 en el Pozo de Entrenamiento y 0,7741 en el Pozo Control.

El comportamiento del RMSE en las tres pruebas está bosquejado en la figura 4.3. Por ejemplo, en el Pozo Control, la prueba 1 arroja un valor igual a 0,165; sin embargo, en la

Tabla 4.3: Pruebas realizadas, en el Pozo de Entrenamiento, variando las condiciones iniciales de Algoritmos Genéticos. Registro de Entrada: VP.

	Prueba 1	Prueba 2	Prueba 3
RMSE	0,214	0,229	0,237
R^2	0,4910	0,4910	0,4910

Tabla 4.4: Pruebas realizadas, en el Pozo Control, variando las condiciones iniciales de Algoritmos Genéticos. Registro de Entrada: VP.

	Prueba 1	Prueba 2	Prueba 3
RMSE	0,165	0,178	0,187
R^2	0,7741	0,7741	0,7741

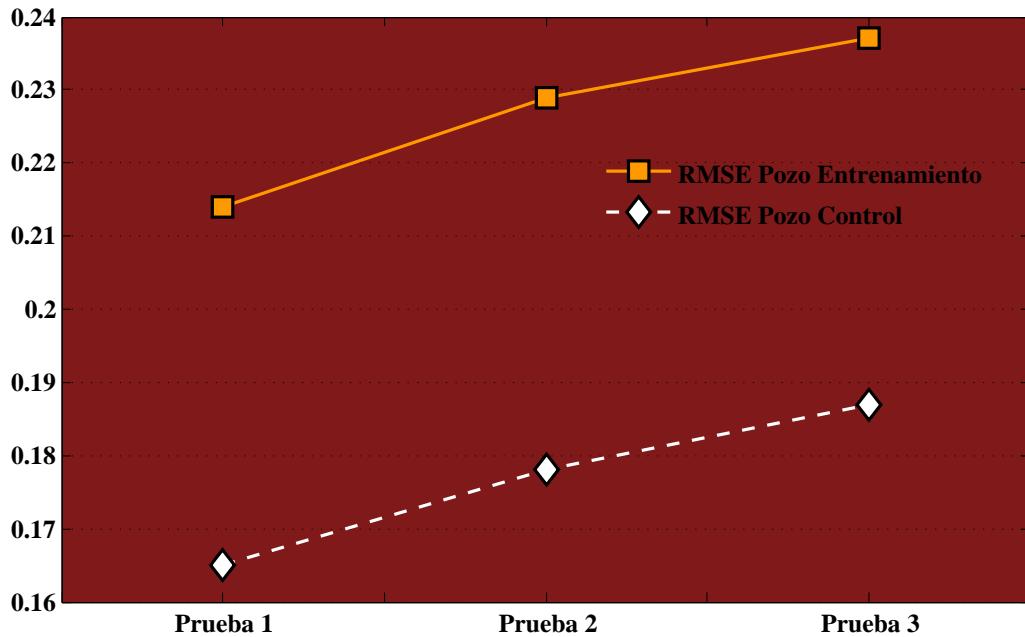


Figura 4.3: Variación del RMSE en las pruebas realizadas con el registro de entrada: Velocidad de Onda P

siguiente prueba aumenta hasta 0,178 y en la última a 0,187, incrementos en magnitud que similarmente son observados en la figura 4.1.

Asimismo, los valores de R^2 obtenidos en este ejemplo indican una gráfica predicha que ajusta mucho mejor el comportamiento del registro de Velocidad de Onda S real que el caso anterior cuando el registro de entrada era Densidad. Esto se evidencia en la gráfica

correspondiente a los valores predichos vs. reales 4.5(b), en donde, para las pruebas en el Pozo de Entrenamiento, $R^2 = 0,4910$ (en contraste con el 0,1722 del punto anterior).

De ahí que, en la figura 4.4, la tendencia del registro predicho es estimar correctamente el patrón de comportamiento del registro real en las tres pruebas. Gráficamente, ambas disminuyen y/o aumentan a lo largo de todo el registro en la misma proporción, llegándose incluso mayormente a superponerse ambos.

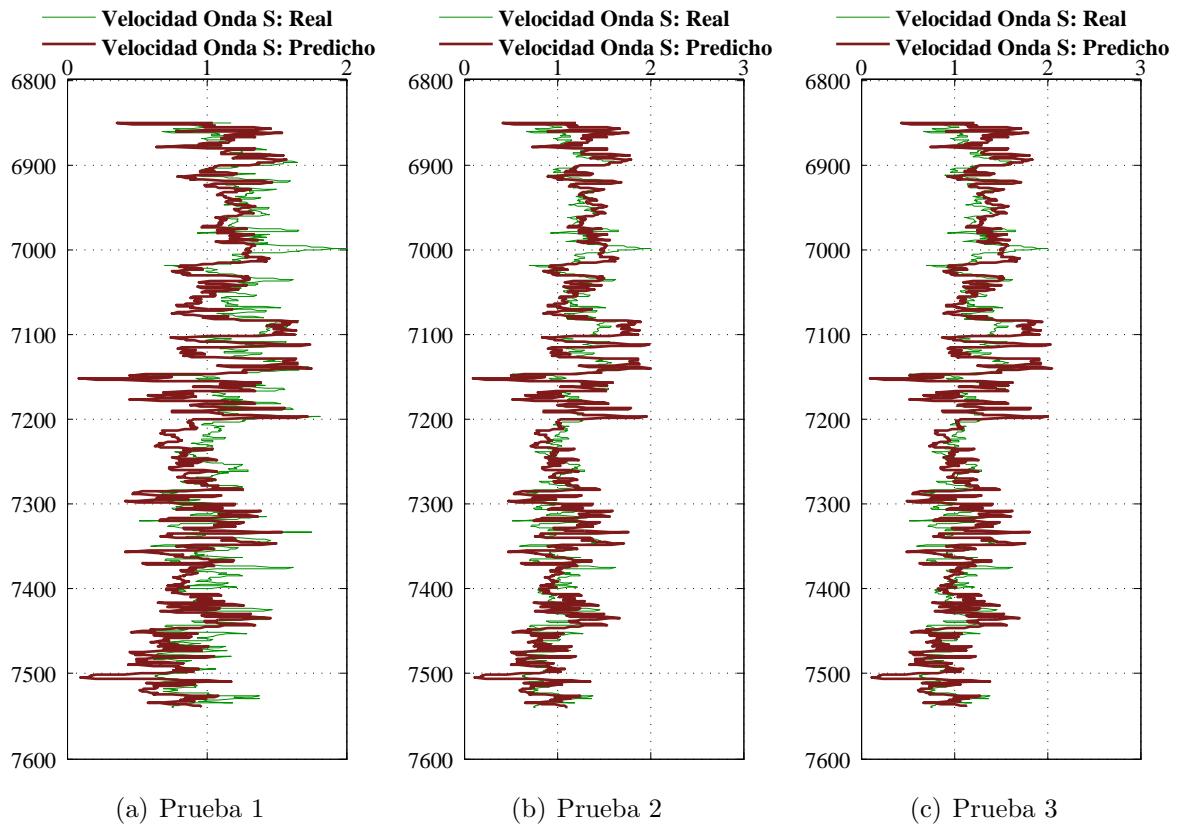


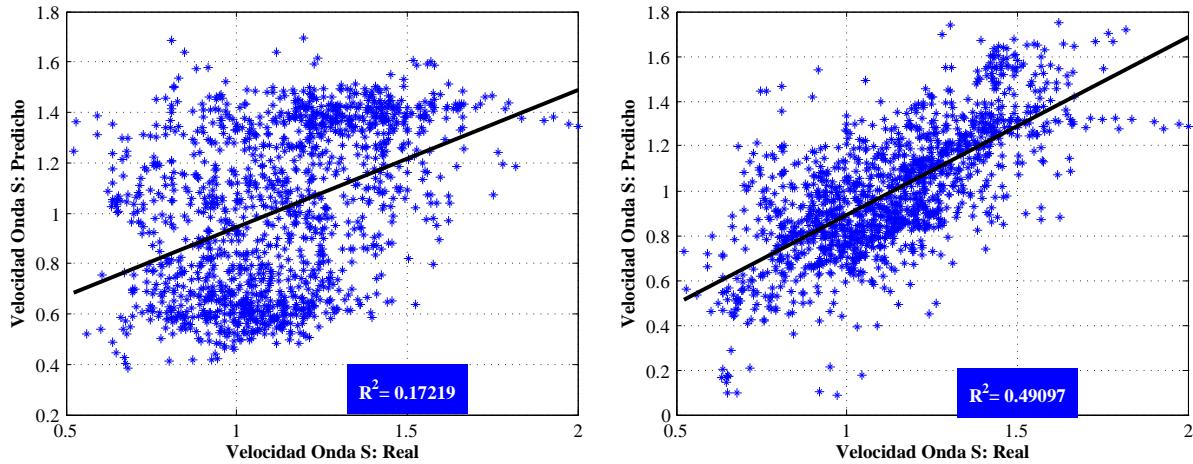
Figura 4.4: Predicciones realizadas, en el Pozo de Entrenamiento, modificando las condiciones iniciales de los Algoritmos Genéticos. Registro de Entrada: Velocidad de Onda P

Debido a que la distribución de los valores predichos y los reales de Onda S es el mismo en todos los ejemplos mostrados (esto lo señala el R^2 que no varía), se puede concluir que no se verá alterado en gran medida el comportamiento del registro final estimado al cambiar las condiciones iniciales³ de los Algoritmos Genéticos. La influencia de éstos es sobre el valor del RMSE, que presenta un valor mínimo en la Prueba 1 en los dos ejemplos mostrados.

Por tanto, para los posteriores entrenamientos de los sistemas de predicción utilizados en este trabajo (tanto Algoritmos Genéticos como el Algoritmo Híbrido Difuso-Genético) fueron

³En la tabla 3.1, página 59.

pre-seleccionados las condiciones señaladas en la Prueba 1, las cuales se muestran en la tabla 4.5 (valores extraídos de la tabla 3.1).



(a) Pozo de Entrenamiento. Registro de Entrada: DEN.

(b) Pozo Control. Registro de Entrada: VP.

Figura 4.5: Valores Predichos vs. Valores Reales de Velocidad de Onda S en las pruebas realizadas modificando las condiciones iniciales de Algoritmos Genéticos

La principal razón de escoger las menores cantidades en el tamaño de la población, generaciones y Elite Count (número de individuos que tiene asegurado su pase a la siguiente generación), son principalmente computacionales. El tiempo de ejecución del algoritmo genético es menor en estos casos (Niño, 2006; Saemi & Ahmadi, 2008).

Tabla 4.5: Condiciones Iniciales pre-seleccionadas de los Algoritmos Genéticos

Opción	Prueba 1
Tamaño de la Población	10
Función de Cruce	@crossoversinglepoint
Elite Count	1
Función de Selección	@selectionroulette
Función de Mutación	@mutationuniform
Generaciones	60

Sin embargo, de las condiciones iniciales pre-seleccionadas, vale la pena deducir las que tienen mayor influencia en el resultado final. Para ello hizo falta realizar una prueba adicional variando las funciones de Cruce, Selección y Mutación, que se exponen en la tabla 4.6.

Los resultados obtenidos al realizar esta prueba adicional se muestran en la tabla 4.7. Tanto para el Pozo de Entrenamiento como para el Pozo Control, nótese que los valores

Tabla 4.6: Prueba adicional realizada modificando las condiciones iniciales de los Algoritmos Genéticos

Opción	Prueba Adicional
Tamaño de la Población	10
Función de Cruce	@crossoverscattered
Elite Count	1
Función de Selección	@selectionroulette
Función de Mutación	@mutationuniform
Generaciones	60

Tabla 4.7: Valores de RMSE y R^2 en la prueba adicional realizada en ambos pozos, variando las condiciones iniciales de Algoritmos Genéticos.

	Pozo de Entrenamiento		Pozo de Control	
	DEN	VP	DEN	VP
RMSE	0.337	0.214	0.320	0.165
R^2	0.1722	0.4910	0.1353	0.7741

obtenidos del RMSE y de R^2 son iguales a los obtenidos en la Prueba 1 (tablas 4.1, 4.2, 4.3 y 4.4). Por ello, fueron elegidos de forma definitiva para todos los cálculos donde se involucren los algoritmos genéticos, las opciones que se muestran en la tabla 4.6.

4.1.2. ANFIS

Es importante resaltar nuevamente que la salida de los sistemas difusos está sujeta al conjunto de reglas difusas y a la cantidad de funciones de membresía que se dispongan (Gutiérrez, 2010). Es por esto que se realizaron diversas pruebas variando el número de reglas difusas (4, 6, 8 y 9) usando la función de membresía *Gaussian*, para así poder discernir cuál es la mejor combinación de funciones de membresía para uno, dos y tres registros de entrada.

Debido a la cantidad de pruebas realizadas, en este caso sólo se analizará gráficamente las variaciones del RMSE. En el Apéndice B, sección B.1 se encuentran tabulados todos los valores de RMSE obtenidos que sustentan los gráficos de las figuras 4.6 a la 4.9.

4.1.2.1. Pruebas #1: Un Registro de Entrada.

Cabe aclarar que en las figuras 4.6(a) y 4.6(b), los registros de entrada (en el eje de las abscisas) que se encuentran numerados del 1 al 5, se corresponden a los que se utilizaron en estas pruebas, los cuales, tal y como se mostró en la sección 3.4.2, fueron los siguientes:

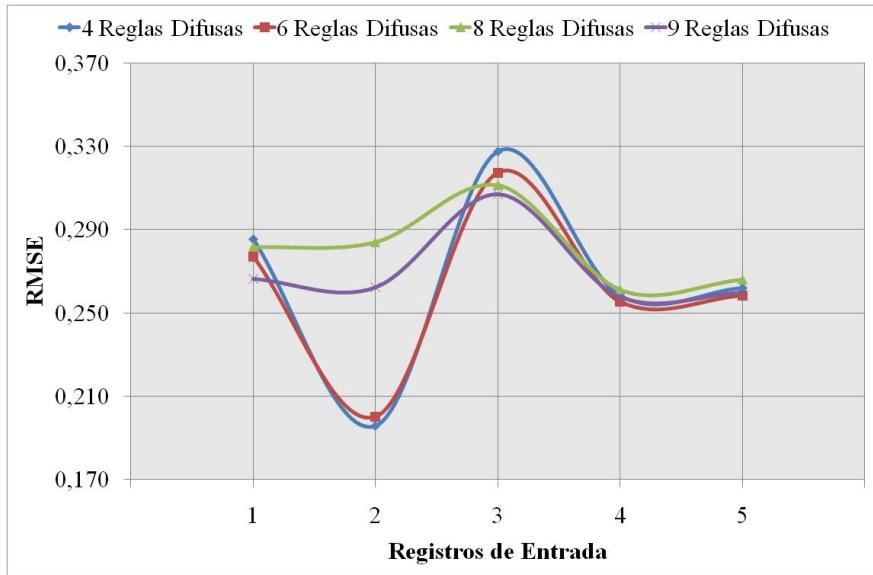
1. Densidad.
2. Velocidad de Onda P.
3. Saturación de Agua.
4. Porosidad.
5. Volumen de Arcilla.

En estas figuras puede verse como el registro de Velocidad de Onda S predicho alcanza un valor de RMSE bastante alto en el Pozo Control, cuando el registro de entrada es Velocidad de Onda P y se usan 8 reglas difusas. Caso contrario ocurre en el mismo punto pero ahora en el Pozo de Entrenamiento, donde para 4 y 6 reglas difusas se obtiene el mínimo valor del RMSE.

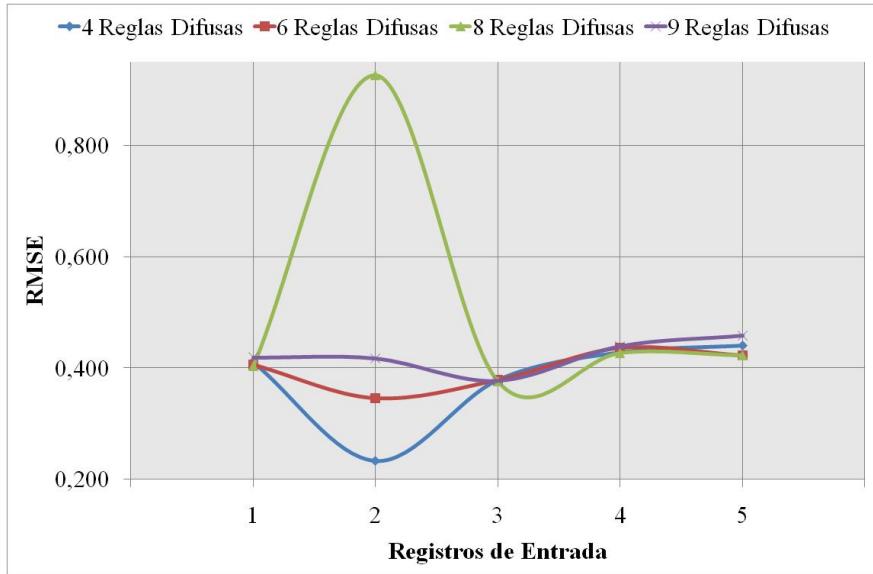
Por otra parte, cuando los registros de entrada son Saturación de Agua, Porosidad o Volumen de Arcilla, el RMSE es similar para ambos pozos independientemente del número de reglas difusas utilizadas. Esto difiere al comportamiento observado cuando la entrada es **Densidad o Velocidad de Onda P**, donde el RMSE es menor para cuatro y seis reglas difusas.

Este hecho es cónsono con los resultados obtenidos en las pruebas anteriores de Algoritmos Genéticos, en donde los valores más bajos del RMSE se obtuvieron cuando el registro de que se utilizó para la predicción fue Velocidad de Onda P.

Finalmente, de ambas graficas se desprende que, en base al RMSE, para los distintos registros de entrada se realizó una mejor predicción cuando se emplearon cuatro y seis reglas difusas.



(a) Pozo de Entrenamiento



(b) Pozo Control

Figura 4.6: Variación del RMSE utilizando 4, 6, 8 y 9 reglas difusas en cada entrada de un Registro de Pozo.

4.1.2.2. Pruebas #2: Dos Registros de Entrada.

Ahora veamos los resultados obtenidos cuando se efectaron las predicciones preliminares utilizando dos registros de entrada en ANFIS. Tal y como se explica en la sección 3.4.2, para este caso fueron seleccionadas las siguientes combinaciones:

1. Densidad - Velocidad de Onda P.

2. Densidad - Saturación de Agua.
3. Densidad - Porosidad.
4. Densidad - Volumen de Arcilla
5. Velocidad de Onda P - Saturación de Agua.
6. Velocidad de Onda P - Porosidad.
7. Velocidad de Onda P - Volumen de Arcilla..
8. Saturación de Agua - Porosidad.
9. Saturación de Agua - Volumen de Arcilla.
10. Porosidad - Volumen de Arcilla.

Análogo a lo ocurrido en las figuras 4.6(a) y 4.6(b), la numeración contenida en el eje de las abscisas de las imágenes 4.7 a la 4.9, se corresponde con la anterior. Resulta poco estético (por cuestiones de espacio) incluir las etiquetas de los registros de entrada utilizados dentro de las figuras.

Igualmente, como se explicó en la sección 3.4.2, cuando se utilizan 6 u 8 reglas difusas, pueden darse dos combinaciones en cada una, [2 3]-[3 2] y [2 4]-[4 2] respectivamente. Es por esto que se decidió graficar primero el comportamiento del RMSE a medida que cambian los registros de entrada en estos casos (las imágenes obtenidas se encuentran en las figuras 4.7 y 4.8).

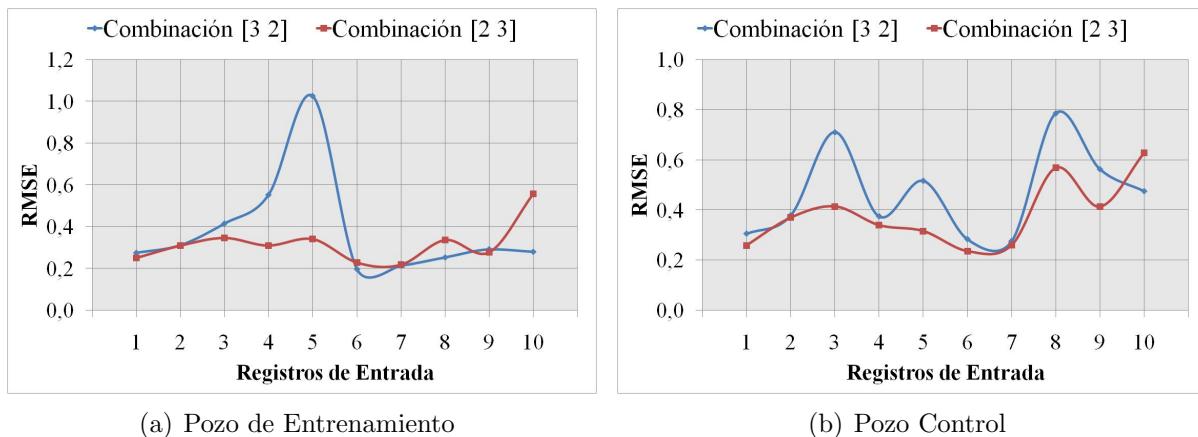


Figura 4.7: Variación del RMSE para 6 reglas difusas (combinaciones [3 2] y [2 3]) a medida que cambian las entradas de dos Registros de Pozo.

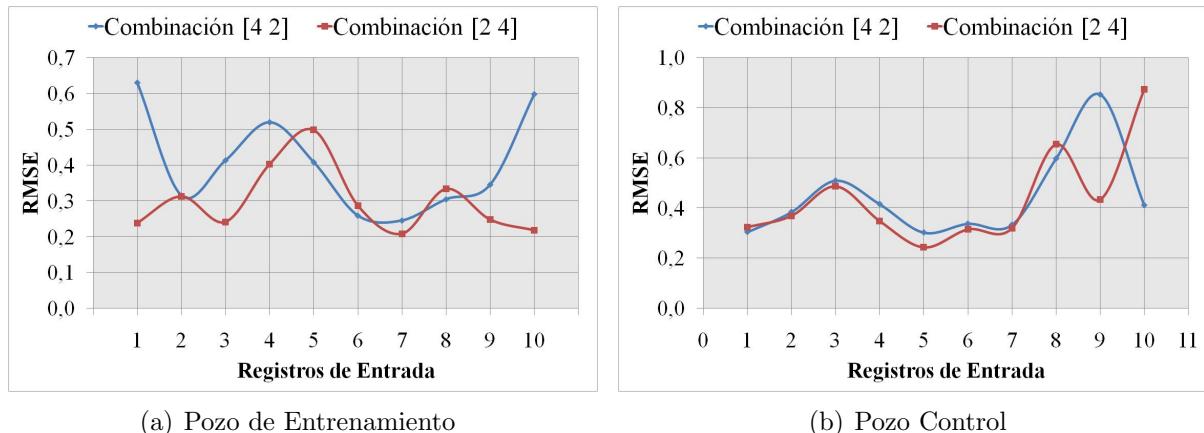


Figura 4.8: Variación del RMSE para 8 reglas difusas (combinaciones [4 2] y [2 4]) a medida que cambian las entradas de dos Registros de Pozo.

En la figura 4.7 vemos como, en el Pozo de Entrenamiento, la combinación [2 3] ofrece una variación menor del RMSE a medida que cambian los registros de Entrada, en efecto, varían sólo entre 0,2 y 0,4 exceptuando el caso en que el registro de entrada es Porosidad - Volumen de Arcilla ⁴. Esto habla del buen comportamiento de ésta combinación en la predicción del registro de Velocidad de Onda S.

En el Pozo Control, la curva de la combinación de reglas difusas [2 3] tiene un rango de variación mayor con respecto a las obtenidas en el pozo de entrenamiento, sin embargo, sigue siendo menor que la que ocurre en todos los casos de registros de entrada en la [3 2].

Por otra parte, en la figura 4.8 la diferencia entre ambas curvas no es tan notoria. En ambos pozos, las curvas de combinación [2 4] y [4 2] muestran la misma tendencia de aumentar y disminuir el RMSE en cada par de registro de entrada, salvo en el caso en que éstos son Saturación de Agua - Volumen de Arcilla (punto 9), en donde en la combinación [2 4] el RMSE es menor, criterio que termina siendo en decisivo en cuanto a la selección de esta opción como la que da una mejor predicción.

Ahora con la finalidad de observar cuál número de reglas difusas permite, en general, obtener una mejor estimación (de Velocidad de Onda S) en base a sus valores de RMSE, se decidió entonces graficar (Véase la figura 4.9) para ambos pozos, los cuatro casos que se encuentran a continuación:

- 4 reglas difusas, cuya única combinación posible es [2 2].

⁴Se sabe que son estos los registros de entrada al comparar el punto número 10 con la lista dada.

- 6 reglas difusas, la mejor combinación, que arroja RMSE menores, es [2 3].
- 8 reglas difusas, la mejor combinación encontrada [2 4].
- 9 reglas difusas, única combinación posible es [3 3].

Estudiando esta figura, vemos cómo en ambos pozos (GF-87N y GF-109N) los valores más altos se obtienen cuando se usan 9 reglas difusas en las predicciones, específicamente en los puntos 3, 9 y 10, es decir, cuando los registros con los que se hicieron los cálculos fueron Densidad-Porosidad y Saturación de Agua-Volumen de Arcilla.

Cabe destacar el comportamiento de la mayoría de éstas curvas de las reglas difusas en los puntos 1, 6 y 7 en ambos pozos, en donde tienden a disminuir el RMSE y presentar así sus valores bajos, éstos son los casos en que las entradas son **Densidad - Velocidad de Onda P, Velocidad de Onda P - Porosidad y Velocidad de Onda P - Volumen de Arcilla.**

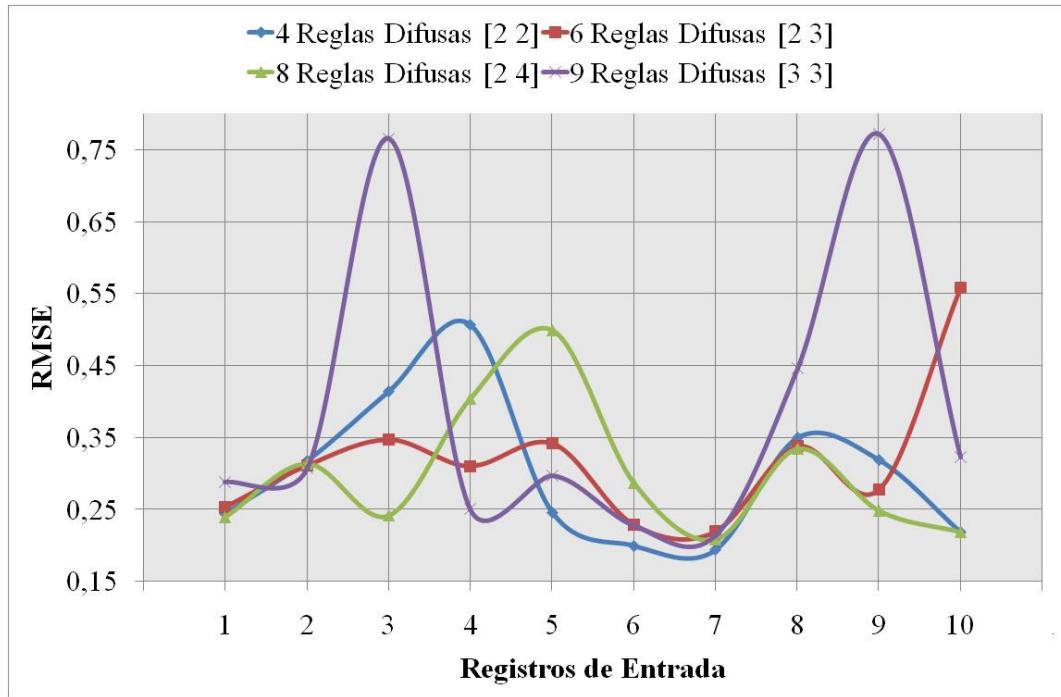
Observando punto a punto, la curva que tiene los valores más bajos del RMSE en **ambos pozos** y/o cuyo rango de variación es menor en comparación con las otras tres, es el caso de **6 reglas difusas en combinación [2 3]**. Esto se ve más claramente si se descartan los puntos 9 y 10, ya que presentan tendencias dispares en los dos pozos de estudio.

4.1.2.3. Pruebas #3: Tres Registros de Entrada.

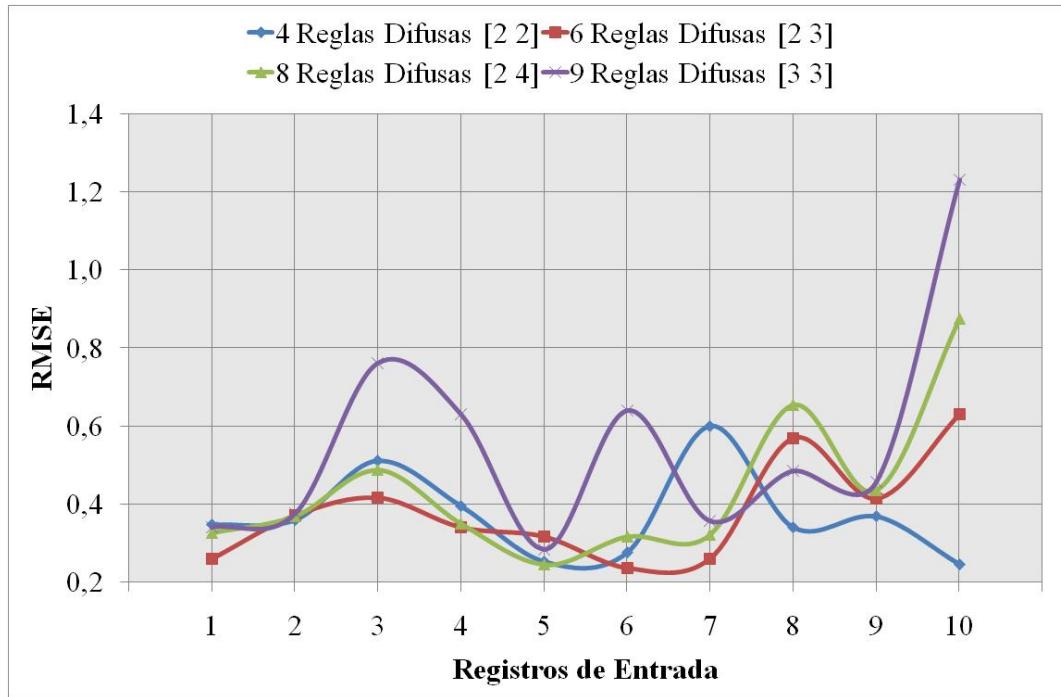
En las pruebas #1, los registros con los que se efectuaron las mejores predicciones (RMSE más bajos) fueron en este orden (con 4 y 6 reglas difusas): Velocidad de Onda P, Densidad y Porosidad. Igualmente, cuando fueron dos registros de entrada, los pares que dieron los mejores resultados fueron:

- Densidad - Velocidad de Onda P.
- Velocidad de Onda P - Porosidad.
- Velocidad de Onda P - Volumen de Arcilla.

Es claro que el registro de Velocidad de Onda P se repite en todos los casos, y en segundo lugar Densidad y Porosidad. De manera que, siguiendo lo que se señaló en la sección 3.4.2, para éstas últimas pruebas se escogieron combinaciones que los incluyeran a éstos.



(a) Pozo de Entrenamiento



(b) Pozo Control

Figura 4.9: Variación del RMSE utilizando 4, 6 (combinación [2 3]), 8 (combinación [2 4]) y 9 reglas difusas en cada entrada de dos Registros de Pozo.

De modo similar a la forma en que se han tratado las imágenes que en esta sección, los números [1,2,3,4,5,6,7,8] en el eje de las abscisas de las figuras 4.10 a la 4.14 se corresponden con las combinaciones de registros de entrada que se muestran en la siguiente lista:

1. Densidad - Velocidad de Onda P - Saturación de Agua.
2. Densidad - Saturación de Agua - Porosidad.
3. Densidad - Porosidad - Volumen de Arcilla.
4. Velocidad de Onda P - Saturación de Agua - Porosidad.
5. Velocidad de Onda P - Porosidad - Volumen de Arcilla.
6. Velocidad de Onda P - Densidad - Volumen de Arcilla.
7. Velocidad de Onda P - Densidad - Porosidad.
8. Velocidad de Onda P - Saturación de Agua - Volumen de Arcilla.

Dado que para 4-6-8-9 reglas difusas con tres entradas han se seleccionarse una combinación específica, se decidió estudiar cada una por separado y luego comparar sólo los mejores resultados.

En la figura 4.10 se encuentra el comportamiento del RMSE en las predicciones realizadas con 4 reglas difusas, con las combinaciones de funciones de membresía [1 2 2], [2 1 2] y [2 2 1]. En el Pozo Control, las dos últimas (líneas verde y roja) son las que ofrecen los menores valores de RMSE; sin embargo, en la figura de la izquierda (Pozo de Entrenamiento) vemos una marcada diferencia entre ámbas, sobre todo en el caso en que la entrada es Densidad - Porosidad - Volumen de Arcilla (punto número 3), lo que conlleva a seleccionar en este caso la **combinación [2 2 1]**.

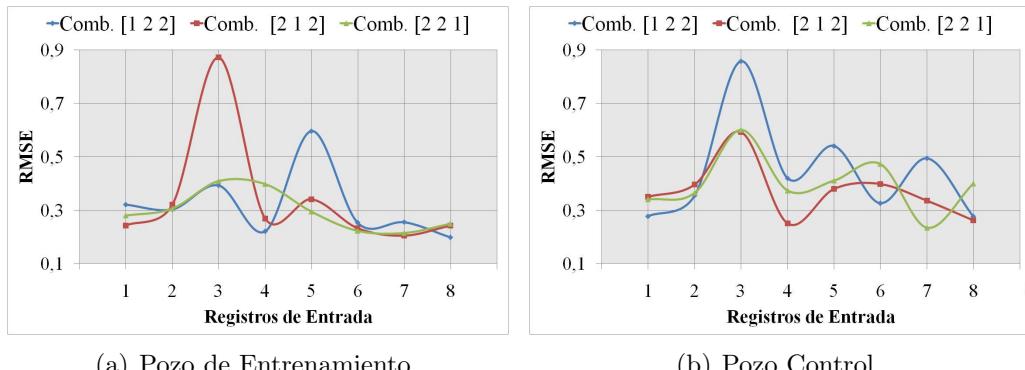


Figura 4.10: Variación del RMSE para 4 reglas difusas (combinaciones [1 2 2]-[2 1 2]-[2 2 1]) a medida que cambian las entradas de tres Registros de Pozo.

Igualmente se hizo en el caso de seis reglas difusas. Se graficaron las combinaciones [1 3 2], [2 1 3] y [3 2 1] en ambos pozos para todos los registros de entrada ya señalados (figura

4.11). En este caso es bien claro observar que es la **estructura [2 1 3]** (línea roja), la que presenta en todos los puntos los mínimos valores de RMSE.

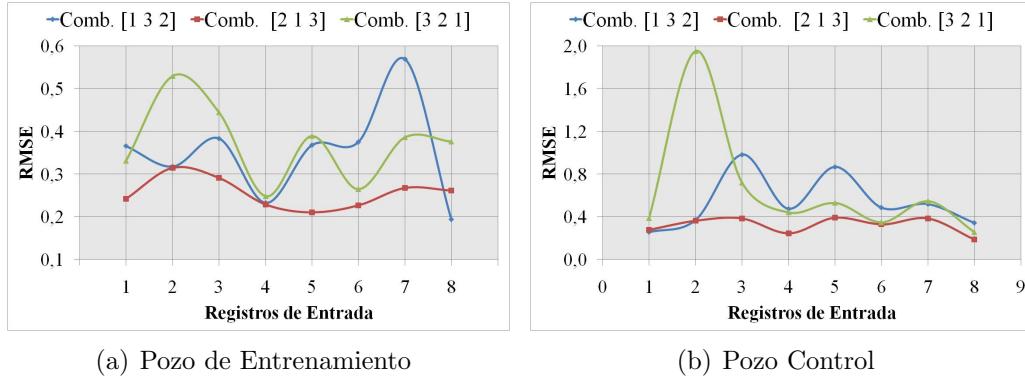


Figura 4.11: Variación del RMSE para 6 reglas difusas (combinaciones [1 3 2]-[2 1 3]-[3 2 1]) a medida que cambian las entradas de tres Registros de Pozo.

En el caso de 8 reglas difusas, las combinaciones escogidas para el estudio fueron [1 4 2], [2 1 4] y [4 2 1] (figura 4.12). En general es la **combinación [2 1 4]** la que tiende a mostrar la menor diferencia entre el máximo y el mínimo valor de RMSE en ambos pozos.

Por último, en la figura 4.13 se muestran las tres combinaciones utilizadas en el caso de nueve reglas difusas. Se puede descartar primeramente la combinación [1 3 3], ya que es la que muestra valores bastante altos de RMSE en relación a las demás en ambos pozos, específicamente en los puntos 6-8⁵ en el Pozo de Entrenamiento y en 3-4-8⁶ en el Pozo Control. En este caso, es la estructura [3 3 1] (línea verde) la que presenta en promedio las menores magnitudes de RMSE.

De manera que, los mejores resultados obtenidos para cada cantidad de reglas difusas, fueron agrupadas en un sólo gráfico (figura 4.14). El caso de **6 reglas (combinación de funciones de membresía [2 1 3])** tiene el menor máximo (0,4 en el Pozo Control) de entre todas las curvas que se encuentran en la gráfica y además presenta el mínimo global de RMSE (0,22 aproximadamente en ambos pozos). Por consiguiente, es ésta la seleccionada para realizar en los análisis posteriores.

⁵Las entradas son en este caso Velocidad de Onda P-Densidad-Volumen de Arcilla y Velocidad de Onda P-Saturación de Agua-Volumen de Arcilla respectivamente.

⁶Las entradas son en este caso Densidad-Porosidad-Volumen de Arcilla, Velocidad de Onda P-Saturación de Agua-Porosidad y Velocidad de Onda P-Saturación de Agua-Volumen de Arcilla respectivamente.

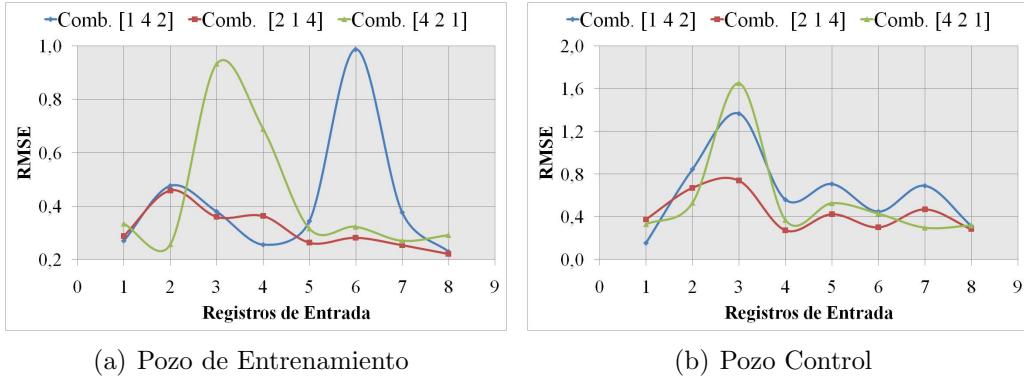


Figura 4.12: Variación del RMSE para 8 reglas difusas (combinaciones [1 4 2]-[2 1 4]-[4 2 1]) a medida que cambian las entradas de tres Registros de Pozo.

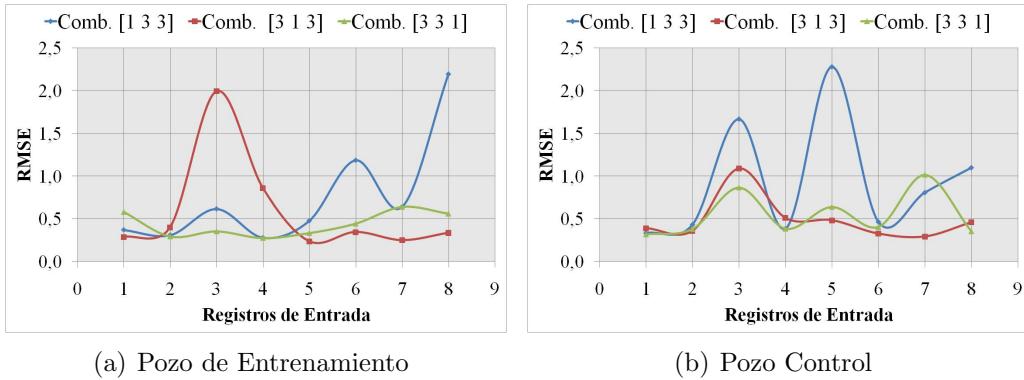
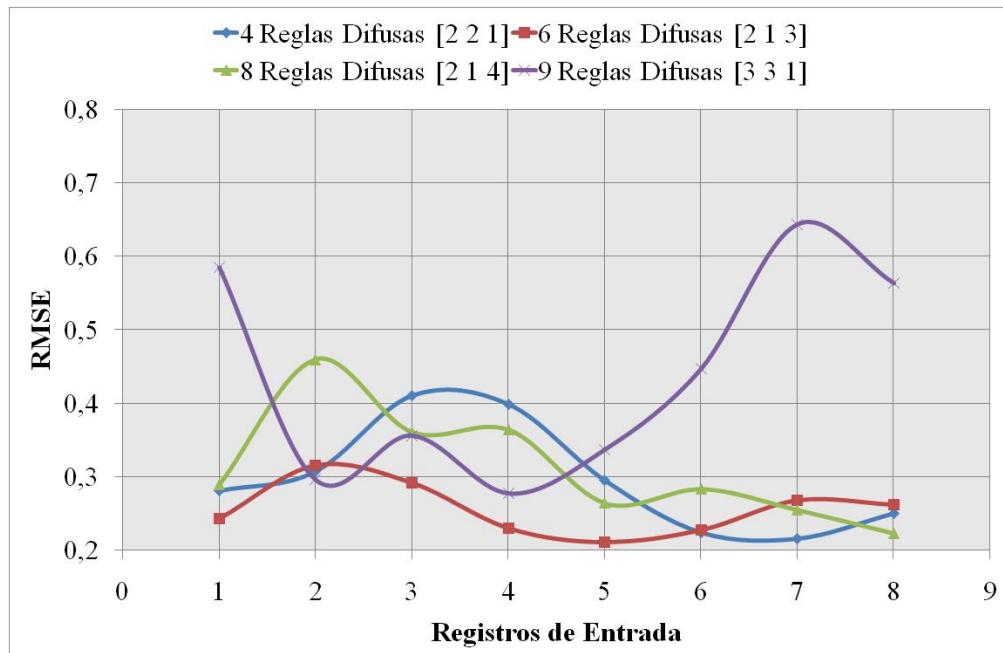


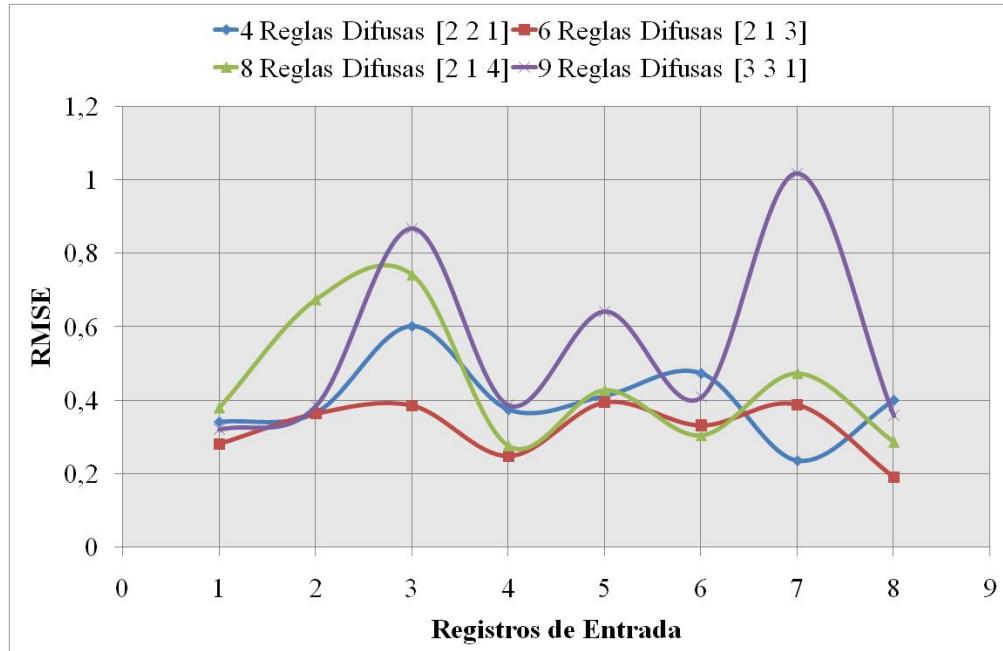
Figura 4.13: Variación del RMSE para 9 reglas difusas (combinaciones [1 3 3]-[3 1 3]-[3 3 1]) a medida que cambian las entradas de tres Registros de Pozo.

Finalmente, a continuación se mostrarán las **predicciones de Velocidad de Onda S realizadas con uno, dos y tres registros de entrada** con los mejores parámetros que en estas pruebas se ha determinado mediante:

1. Algoritmos Genéticos.
2. Sistemas de Inferencia Neuro-Difusa (ANFIS).
3. Algoritmo Híbrido Difuso-Genético.



(a) Pozo de Entrenamiento



(b) Pozo Control

Figura 4.14: Variación del RMSE con 4 (combinación [2 2 1]), 6 (combinación [2 1 3]), 8 (combinación [2 1 4]) y 9 (combinación [3 3 1]) reglas difusas en cada entrada de tres Registros de Pozo.

4.2. Un Registro de Entrada

Para este primer caso se utilizaron como entrada a los algoritmos los siguientes registros:

- DEN (Densidad).
- VP (Velocidad de Onda P).
- SW (Saturación de Agua).
- POR (Porosidad).
- Vsh (Volumen de Arcilla).

4.2.1. Algoritmos Genéticos

Tal y como se explicó en la sección 3.3.1, en el cálculo de Velocidad de Onda S mediante ésta técnica, se calcularon los parámetros (“ a ” y “ $a - b$ ”) que mejor lo relacionan con el registro de entrada. Para ello se tomaron dos ecuaciones diferentes (3.5 y 3.6), las cuales aplicadas a ésta sección quedan:

Ecuación 1 AG:

$$VS = a \times REG$$

Ecuación 2 AG:

$$VS = a \times REG^b$$

donde:

a y b : son los parámetros que asocian al registro de entrada con el de salida.

REG : Registro de Entrada.

Teniendo en cuenta que los registros están normalizados (Véase la sección 3.1.2.2), los parámetros obtenidos para ambas ecuaciones se presentan en las tablas 4.8 y 4.9.

De éstas tablas se desprende, por ejemplo, que para calcular el registro de Velocidad de Onda S a través de la Densidad con la Ecuación 1 AG, se utilizó:

$$VS = 1,076 \times DEN \quad (4.1)$$

Tabla 4.8: Parámetro “a” de la Ecuación 1 AG utilizando un Registro de Entrada

Registro de Entrada	Parámetro “a”
DEN	1,076
VP	0,996
SW	1,570
POR	1,645
Vsh	1,828

Tabla 4.9: Parámetros “a” y “b” de la Ecuación 2 AG utilizando un Registro de Entrada

Registro de Entrada	Parámetros “a”	“b”
DEN	1,078	0,477
VP	1,013	0,407
SW	1,153	0,329
POR	1,109	0,341
Vsh	1,132	0,337

Y con Velocidad de Onda P, por ejemplo, a través de la Ecuación 2 AG:

$$VS = 1,013 \times VP^{0,407} \quad (4.2)$$

Para cada cálculo (bien sea con la Ecuación 1 AG o con la Ecuación 2 AG) se obtuvo un valor de la **Raíz del Error Cuadrático Medio (RMSE)** y del **Coeficiente de Determinación (R^2)** tanto en el Pozo de Entrenamiento como en el Pozo Control, que se presentan en las tablas 4.10 y 4.11.

Como se puede observar en ambos casos, aunque en distintas magnitudes, hay una clara disminución del RMSE para cualquier Registro de Entrada, cuando se utiliza la **ECUACIÓN 2 AG** en comparación de la **ECUACIÓN 1 AG**, donde resalta el caso de *Saturación de Agua*, donde éste va de 1,479 a 0,324 en el Pozo de Entrenamiento y de 1,676 a 0,643 en el Pozo Control.

Tabla 4.10: RMSE para un Registro de Entrada utilizando Algoritmos Genéticos

Reg. Entrada	RMSE			
	Pozo de Entrenamiento		Pozo Control	
	Ec. 1 AG	Ec. 2 AG	Ec. 1 AG	Ec. 2 AG
Densidad	0,413	0,241	0,468	0,424
VP	0,227	0,198	0,175	0,215
SW	1,479	0,324	1,676	0,643
POR	0,937	0,333	1,092	0,492
Vsh	1,139	0,344	1,514	0,538

Tabla 4.11: R^2 para un Registro de Entrada utilizando Algoritmos Genéticos

Reg. Entrada	R^2			
	Pozo de Entrenamiento		Pozo Control	
	Ec. 1 AG	Ec. 2 AG	Ec. 1 AG	Ec. 2 AG
Densidad	0,1722	0,1674	0,1353	0,1567
VP	0,4910	0,4616	0,7741	0,7640
SW	0,0405	0,0423	0,1436	0,1483
POR	0,0226	0,0107	0,0249	0,0194
Vsh	0,0033	0,0001	0,0223	0,0310

Asimismo, en cuanto al R^2 , vemos en la tabla 4.11, cómo entre los casos de entrada Saturación de Agua, Porosidad o Volumen de Arcilla, éste llega a superar el valor 0,1 solamente en el primero de estos tres casos en el Pozo Control (GF-109N); indicativo que la relación existente entre el registro calculado y real es nula, lo que conlleva una pobre predicción del registro de Velocidad de Onda S.

No obstante, las mejores estimaciones (menor valor de RMSE) se obtuvieron utilizando como entrada (en ambas ecuaciones) los registro DENSIDAD y VELOCIDAD DE ONDA P. En el Pozo Control (tabla 4.10), en el caso del segundo, es el único entre todos los resultados obtenidos donde éste parámetro aumenta, va de 0,175 a 0,215, lo cual hace evidenciar que la relación entre éste registro y la Velocidad de Onda S es del tipo lineal.

En la figura 4.15, se encuentra el registro predicho de Onda S cuando la entrada es Velocidad de Onda P. Puede advertirse cómo éste fenómeno modifica la calidad de la predicción.

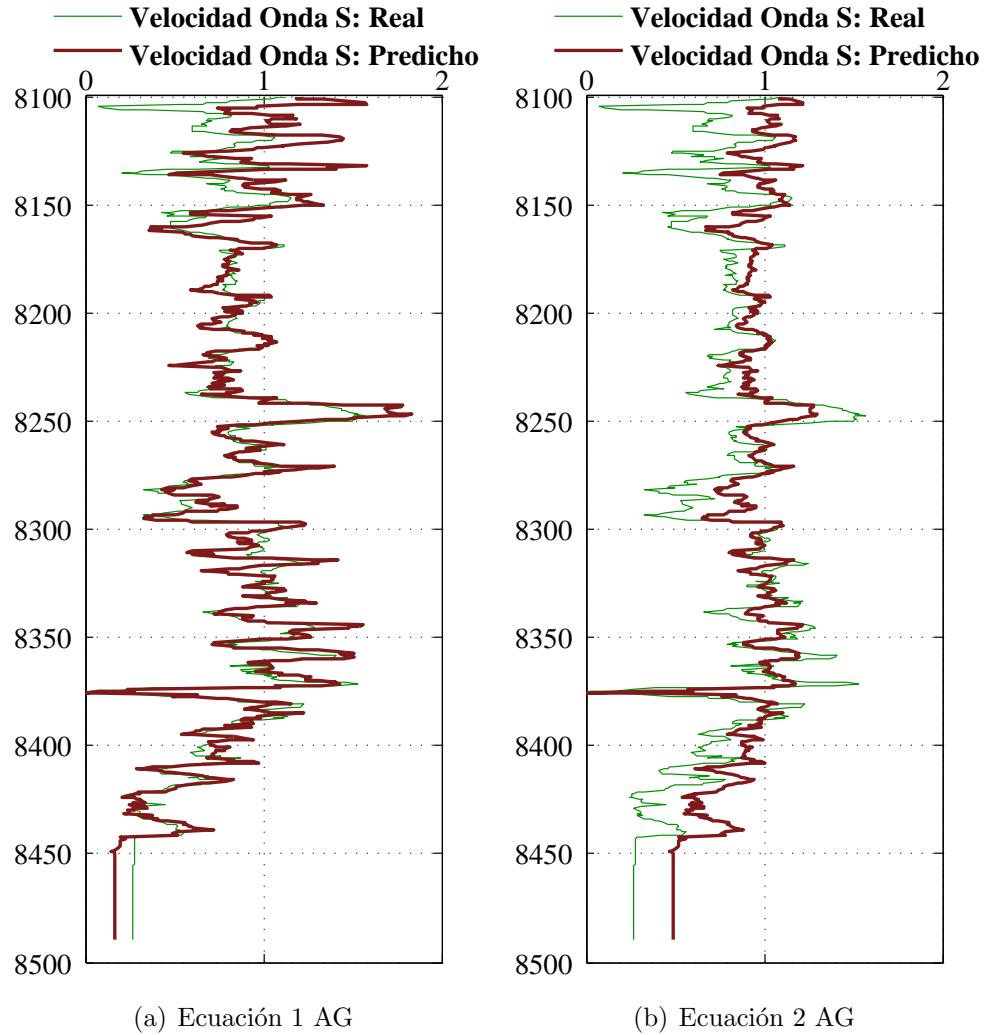


Figura 4.15: Registros de Velocidad de Onda S: Predicho y Real, usando Algoritmos Genéticos en el Pozo Control. Registro de Entrada: VP

Por ejemplo, nótese la diferencia en ambas gráficas en los primeros doscientos pies. A pesar del cambio litológico existente, donde se encuentran una arena de 50' de espesor con una porosidad promedio del 20% y una capa de lutita de 20' de espesor aproximadamente con una porosidad que tiende a disminuir⁷, la Ecuación 1 AG al reconocer cualitativamente mejor el comportamiento del registro en esa zona.

Otro intervalo que ejemplifica esta mejora, es el intervalo lutítico comprendido entre 8400' y 8450'. En la figura 4.15(b), el comportamiento del registro predicho es similar al registro

⁷Esto se sigue luego de interpretar los registros de este Pozo (GF-109N) en la figura 3.2, página 49

real de Onda S, aunque sus valores son mayores, aspecto que cambia en la figura 4.15(a) a su derecha, donde ambas son “iguales”, llegando incluso a superponerse con el registro real.

Al graficar los valores predichos vs. reales en los registros ya mostrados, se obtienen los graficos de la figura 4.16. La imagen 4.16(a) se corresponde con los registros de la 4.15(a) y el gráfico 4.23(b) con la 4.15(b). En ambos casos, el R^2 supera el valor 0,7 específicamente 0,77413 cuando el registro de entrada es VP y la ecuación utilizada la número uno, y 0,76402 cuando se utiliza la segunda ecuación.

La distribución de los puntos entre los valores predichos y los valores reales para ambos casos, exhibe un comportamiento lineal que enfatiza la relación existente entre ambos. Esto habla de la buena predicción realizada cuando la entrada es Velocidad de Onda P, lo que cualitativamente es notorio en los registros mostrados en la figura 4.15.

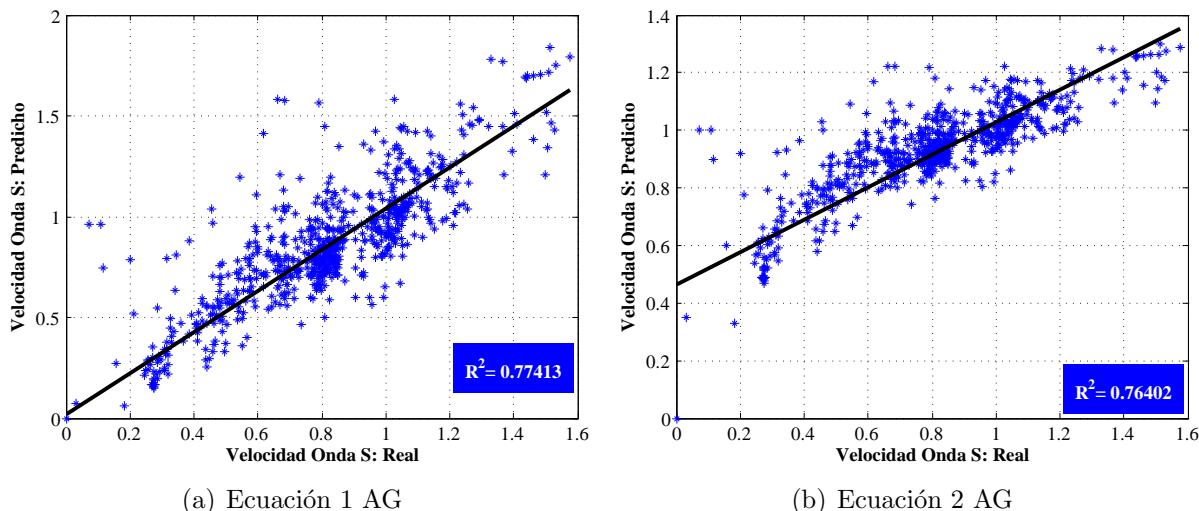


Figura 4.16: Valores Predichos vs. Valores Reales de Velocidad de Onda S usando Algoritmos Genéticos en el Pozo Control. Registro de Entrada: VP

4.2.2. ANFIS-Algoritmo Híbrido

Como fue señalado en la sección 3.6, en la predicciones del registro de Velocidad de Onda S utilizando ANFIS se usaron separadamente las dos alternativas que el programa ofrece cuando se manipula el programa, la opción “Grid Partition” o “Subtractive Clustering”. En cada caso se aplicó el Algoritmo Híbrido Difuso-Genético propuesto en ésta investigación.

De manera que, los resultados presentados en esta sección, se dividen en dos partes: cuando se utilizó “Grid Partition” (con el número de reglas difusas obtenidas en las

pruebas preeliminares) y luego “**Subtractive Clustering**” (las reglas son determinadas por el programa) para todos los registros de entrada seleccionados.

4.2.2.1. Parte I: Grid Partition

En esta sección se efectuó la predicción del Registro de Velocidad de Onda S utilizando la función de membresía *Gaussiana* con 4 reglas difusas, ya que ésta fue la mejor combinación obtenida en las pruebas preliminares realizadas (Véase la sección 4.1.2.1). A continuación, en la tabla 4.12 se muestran los valores de RMSE en cada una de ellas en el Pozo de Entrenamiento y en el Pozo Control y en la tabla 4.13 se muestran los de R^2 en los mismos pozos.

Tabla 4.12: RMSE para un Registro de Entrada utilizando ANFIS-Híbrido (4 reglas difusas)

Reg. Entrada	RMSE			
	Pozo de Entrenamiento		Pozo Control	
	ANFIS	Híbrido	ANFIS	Híbrido
Densidad	0,277	0,247	0,407	0,392
VP	0,194	0,188	0,271	0,211
SW	0,319	0,319	0,372	0,603
POR	0,243	0,366	0,454	0,458
Vsh	0,250	0,396	0,429	0,515

Tabla 4.13: R^2 para un Registro de Entrada utilizando ANFIS-Híbrido (4 reglas difusas).

Reg. Entrada	R^2			
	Pozo de Entrenamiento		Pozo Control	
	ANFIS	Híbrido	ANFIS	Híbrido
Densidad	0,0002	0,1424	0,0762	0,1391
VP	0,4465	0,4940	0,6927	0,7348
SW	0,0064	0,0416	0,1187	0,1515
POR	0,0644	0,0176	0,0309	0,0211
Vsh	0,1012	0,0011	0,0240	0,0203

Cuando se optimizan mediante algoritmos genéticos los parámetros de las reglas difusas generados en el SID y se calcula el registro de Velocidad de Onda S, el RMSE es disminuido sólo en los casos en que los Registros de Entrada son DENSIDAD y VELOCIDAD DE ONDA P en ambos pozos. Empero, en el resto de los casos, éste aumenta al efectuar el Algoritmo Híbrido Difuso-Genético, lo que resultaría en un empobrecimiento de la predicción.

Mientras que el R^2 , no obstante, se espera que tenga una tendencia a aumentar entre la ejecución de ANFIS y el Algoritmo Híbrido. Esto indicaría una mejora en la predicción utilizando el segundo método. Esto ocurre, nuevamente en las entradas Densidad y Velocidad de Onda P (como consecuencia de la disminución del RMSE). Por ejemplo, en el pozo de Entrenamiento éste va en el primer caso de 0,0002 a 0,1424 (si bien el R^2 resultante de la optimización sigue siendo bajo, es clara variación positiva que existe entre uno y otro).

Específicamente, en el resto de los casos, no hay un aumento significativo, que indica la incapacidad que tiene el método de reconocer patrones de comportamiento del registro de Velocidad de Onda S real cuando las entradas son Porosidad, Saturación o Volumen de Arcilla.

Con todo, el valor de RMSE más bajo se obtiene cuando se realiza la predicción utilizando como entrada el registro de **Velocidad de Onda P**. Éste va en el Pozo de Entrenamiento (GF-87N) de 0,194 a 0,188 y en el Pozo Control (GF-109N) de 0,257 a 0,211. Es por esto que se presentarán a continuación, sólo para este caso, las reglas difusas utilizadas, así como los parámetros generados en ANFIS y los optimizados por el Algoritmo Híbrido y finalmente las gráficas con el registro predicho para ambos métodos.

En primer lugar, las reglas difusas generadas en ANFIS se indican en la tabla 4.14; esto siguiendo el ejemplo mostrado en el capítulo anterior, recuadro morado de la figura 3.4 (*fm=función de membresía y FS=Función de Salida*):

Tabla 4.14: Combinación de Reglas Difusas generadas a través de la técnica “Grid Partition” para un Registro de Entrada: VP

Entrada: VP Número de fm	Salida: VS Número de FS
1	1
2	2
3	3
4	4

Las reglas difusas, presentadas en ésta tabla, pueden interpretarse de la siguiente manera:

- Si** VP es $fm1$ **entonces** VS es $FS1$
- Si** VP es $fm2$ **entonces** VS es $FS2$
- Si** VP es $fm3$ **entonces** VS es $FS3$
- Si** VP es $fm4$ **entonces** VS es $FS4$

Por ejemplo, el primer caso se leería (análogamente se hace con las otras):

Si *Velocidad de Onda P* pertenece a la *función de membresía (fm) Nº 1 entonces* la *Velocidad de Onda S* es la *Función de Salida (FS) Nº 1.*

Seguidamente, en la tabla 4.15 se presentan los parámetros σ y c (ancho y centro de la curva) asociados a las cuatro funciones de membresía Gaussianas ($fm1$, $fm2$, $fm3$ y $fm4$).

Tabla 4.15: Parámetros σ y c de la función de membresía *Gaussiana* generados en el SID mediante la técnica “Grid Partition”. Registro de Entrada: VP

Entrada: VP		
Número de fm	sigma	c
1	0,2286	0,5356
2	0,0322	0,9526
3	0,1727	1,3183
4	0,2144	1,6909

Además, debido a que se trata de un sólo dato de entrada en ANFIS, las Funciones de Salida son de la siguiente manera (los valores de “ a ” y “ b ” en cada caso se muestran en la tabla 4.16):

$$\begin{aligned} FS1 : VS_1 &= a_1 * VP + b_1 \\ FS2 : VS_2 &= a_2 * VP + b_2 \\ FS3 : VS_3 &= a_3 * VP + b_3 \\ FS4 : VS_4 &= a_4 * VP + b_4 \end{aligned}$$

Luego, tomando en cuenta los datos de las tablas 4.15 y 4.16, las reglas difusas mencionadas, con las que finalmente se calculó el registro de Velocidad de Onda S mostrado en

Tabla 4.16: Parámetros “a” y “b” de las funciones de salida, utilizando “Grid Partition”. Registro de Entrada: VP

Función de Salida	Parámetros	
	a	b
Nº 1	0,5218	0,5394
Nº 2	2,4974	-1,3709
Nº 3	-0,5224	1,5104
Nº 4	-1,5203	4,1269

la figura 4.17(a), pueden reescribirse de la siguiente manera (*gaussmf=función de membresía gaussiana*):

- Si** VP es gaussmf[0,2286 0,5356] **entonces** VS=0,5218 * VP + 0,5394
- Si** VP es gaussmf[0,0322 0,9526] **entonces** VS=2,4974 * VP - 1,3709
- Si** VP es gaussmf[0,1727 1,3183] **entonces** VS=-0,5224 * VP + 1,5104
- Si** VP es gaussmf[0,2144 1,6909] **entonces** VS=-1,5203 * VP + 4,1269

Seguidamente, con los datos de la tabla 4.15 se obtienen sólo los datos de entrada que se encuentran a media altura de cada gaussiana para llevar a cabo el proceso de optimización. Los nuevos parámetros “a” y “b” calculados mediante algoritmos genéticos, con los que se realiza la predicción mostrada en la figura 4.17(b), se encuentran en la tabla 4.17. Obsérvese cómo cambian en relación con los de la tabla anterior, en este caso ahora todos son positivos y tanto “a” y “b” tienen un rango de variación menor. Por ejemplo, en la regla difusa nº 4, los valores obtenidos con ANFIS son -1,5203 y 4,1269 mientras que optimizados cambian a 0,4524 y 0,4990 respectivamente.

Finalmente, ya presentados todos los datos obtenidos, es importante prestar atención a la figura 4.17. En general, ambos registros predichos describen el comportamiento del registro real de Onda S. En todo el rango, se observa que es el sistema híbrido que describe mejor las variaciones del registro original.

Esto se puede observar en varios intervalos específicos: en los primeros 50 pies, por ejemplo, hay una fluctuación mayor de la velocidad de Onda S predicha por el Algoritmo Híbrido que por el registro calculado en ANFIS; ésta alta variación puede deberse a que éste es un cuerpo lutítico que tiende a aumentar progresivamente el tamaño de grano hacia arriba, donde también aumenta la porosidad en la misma dirección.

Tabla 4.17: Parámetros “*a*” y “*b*” optimizados de las funciones de salida, utilizando “Grid Partition”. Registro de Entrada: VP

Función de Salida	Parámetros	
	<i>a</i>	<i>b</i>
Nº 1	0,5218	0,5394
Nº 1	0,4613	0,4995
Nº 2	0,4661	0,5561
Nº 3	0,5264	0,6018
Nº 4	0,4524	0,4990

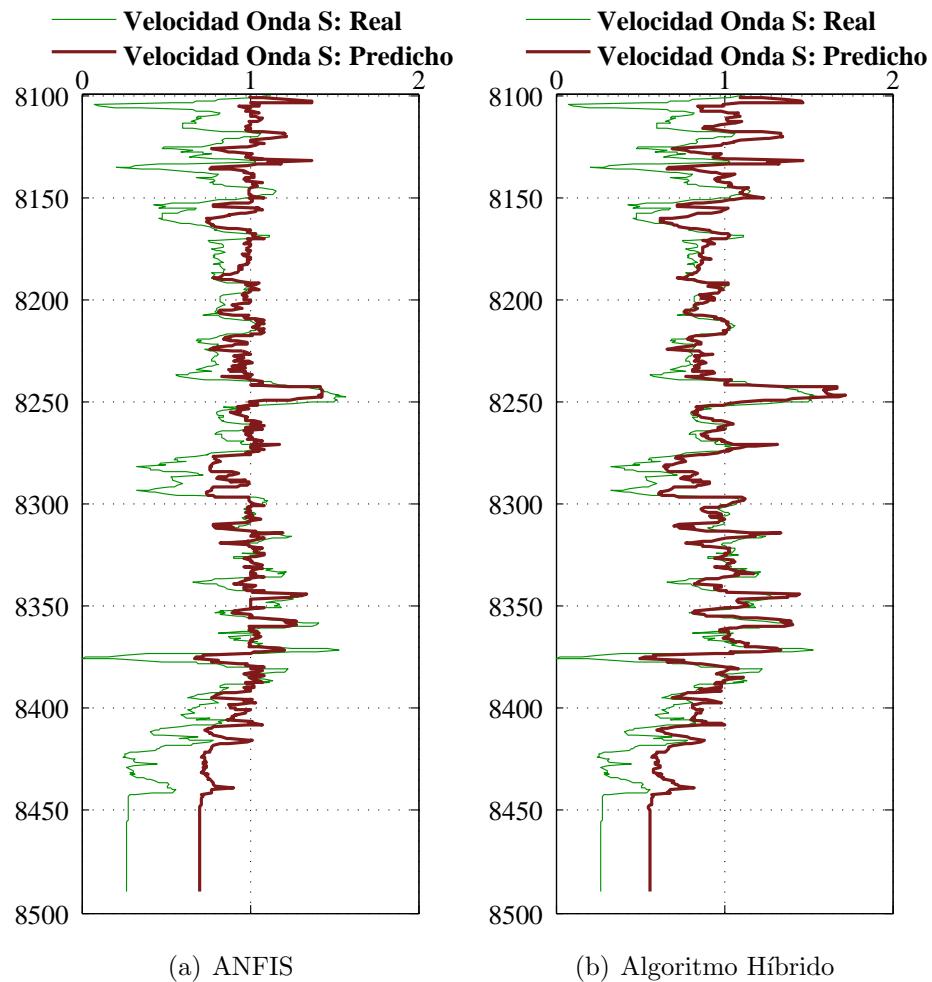


Figura 4.17: Registros de Velocidad de Onda S: Predicho y Real utilizando ANFIS-Híbrido (técnica “Grid Partition”) en el Pozo Control. Entrada: Registro de Velocidad de Onda P.

Asimismo, en el intervalo comprendido entre 8150'-8250' se observa un acortamiento de la distancia entre ambas curvas, indicativo de una disminución del RMSE y un aumento del R^2 (figura 4.18). Estudiando nuevamente los registros de este pozo (GF-109N) mostrados el capítulo anterior (figura 3.2), vemos cómo en ésta zona la litología es variable, encontrándose primero tanto una estrato de arena “ limpia” aproximadamente de 70 pies de espesor con una porosidad alrededor del 20%, seguido de una capa delgada de lutita.

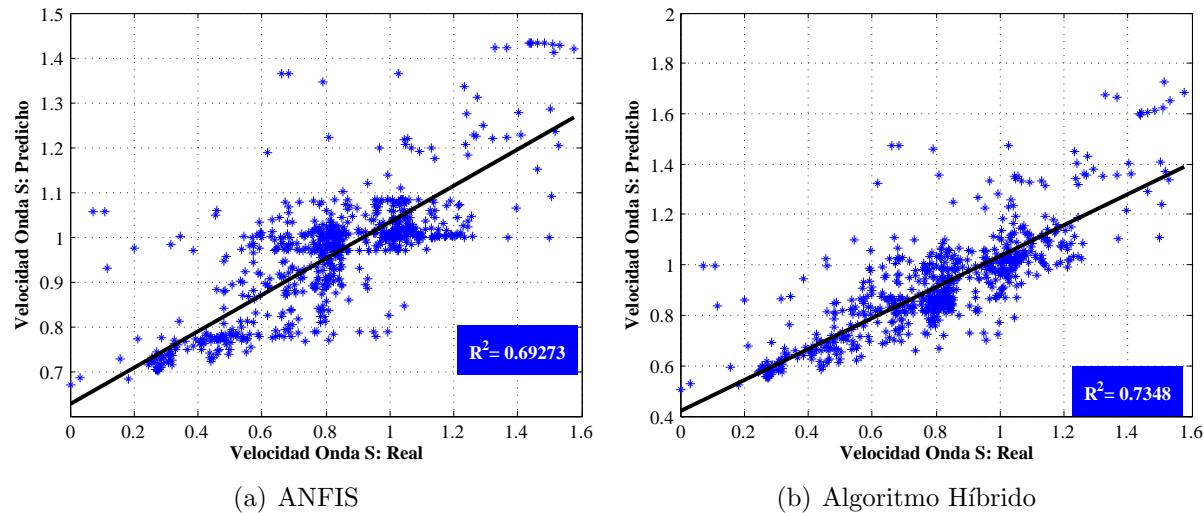


Figura 4.18: Valores Predichos vs. Valores Reales de Velocidad de Onda S utilizando ANFIS-Híbrido (técnica “Grid Partition”) en el Pozo Control. Entrada: Registro de Velocidad de Onda P

Similar ocurre entre 8250'-8300', donde no se encuentra un cuerpo litológico bien definido, más sí se observa una transición granulométrica que tiende a afinar el tamaño de grano hacia arriba (de una arena que le infrayace a una lutita superior). Aquí el algoritmo Híbrido acerca mucho más la curva de velocidad de Onda S predicha.

Sin embargo, el comportamiento del registro predicho en ambos casos en el intervalo entre 8400'-8500' es muy similar; a diferencia de los intervalos superiores, acá ambos métodos modelan al registro VS original de la misma forma. Este intervalo se corresponde con una capa de arena con alto contenido de sedimentos arcillosos o un estrato lutítico con alto contenido de arena (la variación de la porosidad es muy pequeña).

4.2.2.2. Parte II: Subtractive Clustering

Esta sección es similar a la anterior, con la salvedad que, al ejecutar el Sistema de Inferencia Neuro-Difusa en la predicción del Registro de Velocidad de Onda S, el número

de reglas difusas son generadas automáticamente por el programa. También acá la función de membresía utilizada fue la gaussiana.

De manera que, en la tabla 4.18 se muestra la variación del número de reglas difusas por registro de Entrada. El mayor número de reglas difusas se obtiene en los casos de Saturación de Agua, Porosidad y Volumen de Arcilla. Puede interpretarse que, en estos casos, el programa aumenta el número de reglas difusas para intentar reconocer la relación entre el registro de entrada y el de Velocidad de Onda S real.

Esto no ocurre cuando la Entrada es Densidad o Velocidad de Onda P; el número de reglas es menor, lo que puede indicar el grado de relación de estos registros con el de Velocidad de Onda S.

Tabla 4.18: Número de Reglas Difusas obtenidas en cada registro de entrada

Reg. de Entrada	Nº Reglas Difusas
DEN	3
VP	2
SW	4
POR	4
Vsh	4

A continuación, en las tablas 4.19 y 4.20 se muestran los valores RMSE y R^2 en las predicciones realizadas utilizando los registros ya mencionados. Nuevamente, cuando se aplica el Algoritmo Híbrido a la Inferencia Difusa generada en ANFIS, vemos que los registros predichos que presentan una disminución de RMSE y un aumento de R^2 en ambos pozos son los casos en que los Registros de entrada son DENSIDAD y VELOCIDAD DE ONDA P (3 y 2 reglas difusas respectivamente).

Cabe detenerse un poco en este punto para percatarse de lo siguiente. Los valores de RMSE luego de las predicciones obtenidas con ANFIS y posteriormente con el proceso de optimización, cuando el registro de Entrada es **Velocidad de Onda P**, son 0,206-0,190 y 0,218-0,218 en los Pozos de Entrenamiento y Control respectivamente (tabla 4.19). Estos, no obstante, son de magnitud similar a los obtenidos cuando se utilizó “Grid Partition” (4 reglas difusas) en la sección anterior (tabla 4.12) con la misma entrada, los cuales fueron 0,194-0,188 y 0,257-0,211 para ambos pozos.

Tabla 4.19: RMSE para un Registro de Entrada utilizando ANFIS-Híbrido (reglas difusas variables)

Reg. Entrada	RMSE			
	Pozo de Entrenamiento		Pozo Control	
	ANFIS	Híbrido	ANFIS	Híbrido
Densidad	0,278	0,242	0,404	0,386
VP	0,206	0,190	0,218	0,218
SW	0,314	0,335	0,376	0,603
POR	0,259	0,347	0,422	0,441
Vsh	0,256	0,497	0,433	0,312

Tabla 4.20: R^2 para un Registro de Entrada utilizando ANFIS-Algoritmo Híbrido (reglas difusas variables).

Reg. Entrada	R^2			
	Pozo de Entrenamiento		Pozo Control	
	ANFIS	Híbrido	ANFIS	Híbrido
Densidad	0,0004	0,1548	0,0544	0,1379
VP	0,3789	0,4950	0,6820	0,7067
SW	0,0155	0,0380	0,1158	0,1351
POR	0,0399	0,0143	0,0117	0,0228
Vsh	0,0637	0,0105	0,0124	0,0018

Mientras tanto, ocurre algo similar con los valores de R^2 , que se usa para medir la relación lineal entre el registro estimado y el registro real. En la sección anterior, éstos son 0,7198 y 0,7438 en el Pozo Control con la entrada VP (tabla 4.13). Mientras que, en este caso son 0,6820 y 0,7067 en el mismo pozo (ver tabla 4.20). En estos casos, el sistema híbrido nuevamente mejora la calidad de la predicción (al aumentar el R^2).

Esta semejanza entre los valores de RMSE y R^2 utilizando dos métodos diferentes de ANFIS, lleva a afirmar que la estimación en ambos casos es equivalente, es decir, el registro de Onda S predicho utilizando “Subtractive Clustering” (que genera dos reglas difusas) tendrá un comportamiento similar al observado en la sección anterior en la figura 4.17.

En base a esto, se hace interesante analizar la predicción realizada con otro registro de entrada, para evaluar el efecto de la aplicación del Algoritmo Híbrido Difuso-Genético. Así que a continuación se mostrarán los resultados cuando se tomó como entrada el registro DENSIDAD.

En primer lugar, en la tabla 4.21 se presenta la combinación de las tres reglas difusas generadas en este caso. Las abreviaciones utilizadas allí son: fm=función de membresía y FS=Función de Salida.

Tabla 4.21: Combinación de Reglas Difusas generadas a través de la técnica “Substractive Clustering” para un Registro de Entrada: DEN

Entrada: DEN Número de fm	Salida: VS Número de FS
1	1
2	2
3	3

De ésta se sigue el siguiente ejemplo:

Si DEN pertenece al valor lingüístico definido por la función de membresía 2 **entonces** se utilizará en el cálculo la función de salida número 2

Esta regla difusa hace alusión a la función de salida 2, que por su parte se escribe como en la sección anterior, ya que igualmente se trata de un registro de entrada:

$$\text{FS2} : VS_2 = a_2 * DEN + b_2$$

Los parámetros σ y c (ancho y centro de la función Gaussiana) de las tres funciones de membresía así como de las tres Funciones de Salida “a” y “b” están en las tablas 4.22 y 4.23.

De manera que, la regla difusa dada como ejemplo anteriormente puede reescribirse como:

Si DEN pertenece al valor lingüístico definido por gaussmf[0,1803 1,3490] **entonces** se utilizará en el cálculo $VS_2 = -0,4110 * DEN + 1,6921$

Cabe destacar nuevamente que el fragmento “**Si** DEN pertenece al valor lingüístico definido por gaussmf[0,1803 1,3490]” se realiza evauando el punto que se quiere calcular (con estos

parámetros) en la función de membresía gaussiana 1.5 (dada en la sección 1.2.1.2 en la página 13).

Tabla 4.22: Parámetros σ y c de la función de membresía *Gaussiana* generados en el SID mediante la técnica “Substractive Clustering”. Registro de Entrada: DEN

Entrada: DEN		
Número de fm	sigma	c
Nº 1	0,2169	0,8290
Nº 2	0,1803	1,3490
Nº 3	0,0786	1,3366

Tabla 4.23: Parámetros “ a ” y “ b ” de las funciones de salida, usando “Substractive Clustering”. Registro de Entrada: DEN

Función de Salida	Parámetros	
	a	b
Nº 1	0,0735	0,9245
Nº 2	-0,4110	1,6921
Nº 3	-2,2293	3,8083

Asimismo, en la tabla 4.24 se encuentran los parámetros “ a ” y “ b ” de las reglas difusas optimizados por el programa propuesto mediante algoritmos genéticos. Nótese cómo ahora todos los valores son positivos y con un rango de variación menor.

Tabla 4.24: Parámetros “ a ” y “ b ” optimizados de las funciones de salida, usando “Substractive Clustering”. Registro de Entrada: DEN

Función de Salida	Parámetros	
	a	b
Nº 1	0,5014	0,5042
Nº 2	0,4476	0,5283
Nº 3	0,5800	0,5363

Ahora bien, recordando que la ejecución del Algoritmo Híbrido consiste en cambiar los valores de la tabla 4.23 por los optimizados de la tabla 4.24 en las reglas difusas y reevaluar

el Sistema de Inferencia, los registros mostrados en la figura 4.19 son obtenidos antes y después de proceso en el Pozo Control. En el lado izquierdo se encuentra el registro estimado a través de ANFIS y en el lado derecho el que resulta luego de la optimización.

En el caso de ANFIS, a lo largo de todo el registro, la velocidad predicha se ve como un valor promedio de la velocidad real. Es decir, no modela las variaciones de velocidad ocurridas en el pozo. Cosa contraria ocurre en el registro obtenido mediante el Algoritmo Híbrido, donde el registro predicho se acerca al real en los intervalos comprendidos entre 8170'-8240' y 8300'-8380'. Esta diferencia se puede notar en primer lugar en el valor del RMSE, donde éste disminuye de 0,404 a 0,386.

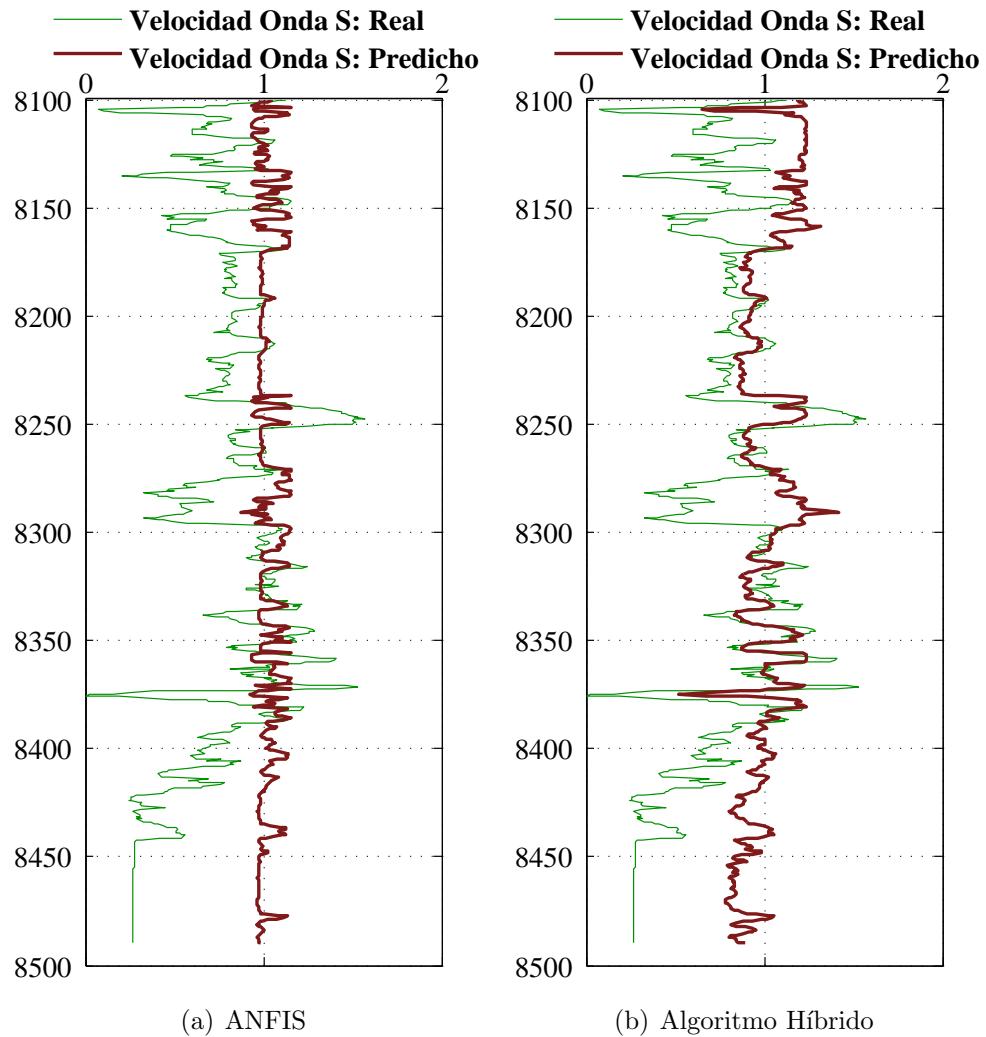


Figura 4.19: Registros de Velocidad de Onda S: Predicho y Real utilizando ANFIS-Híbrido (técnica “Subtractive Clustering”) en el Pozo Control. Entrada: DEN.

Observando los registros del Pozo GF-109 (Pozo Control en la figura 3.2), vemos cómo ambos intervalos corresponden a estratos arenosos con una porosidad que flutúa alrededor

del 20%, de hecho el segundo es efectivamente la arena de interés de estudio. Es decir, es esta litología en particular en donde se logra mejorar el ajuste con la ejecución de la optimización.

Esto se puede ver también en otros intervalos. Por ejemplo, en los primeros 70 pies de profundidad, la variación entre el registro real y predicho es elevada en relación al estrato subyacente. Su respuesta en los registros indica que puede tratarse de una capa de arena aunque con un contenido de sedimentos arcillosos que van en aumento conforme aumenta la profundidad, teniendo una porosidad que disminuye hacia la base (de 20% a 5% aproximadamente) con alto contenido de agua.

Similar en el intervalo entre 8250'-8300', en donde el registro real no estima correctamente el comportamiento de la curva de Velocidad de Onda S medida (en verde), no posee una litología bien definida, con una granulometría que disminuye hacia la base.

Finalmente, la mejora en la predicción utilizando el algoritmo propuesto en la presente investigación, puede verse en la figura 4.20. En la figura superior, vemos que no hay relación lineal entre el registro de Velocidad de Onda S real y el predicho mediante ANFIS. Cosa que cambia ligeramente en la figura inferior, donde si bien es cierto que la relación no es absolutamente lineal (el valor del R^2 está muy lejos de 1), hay un aumento significativo.

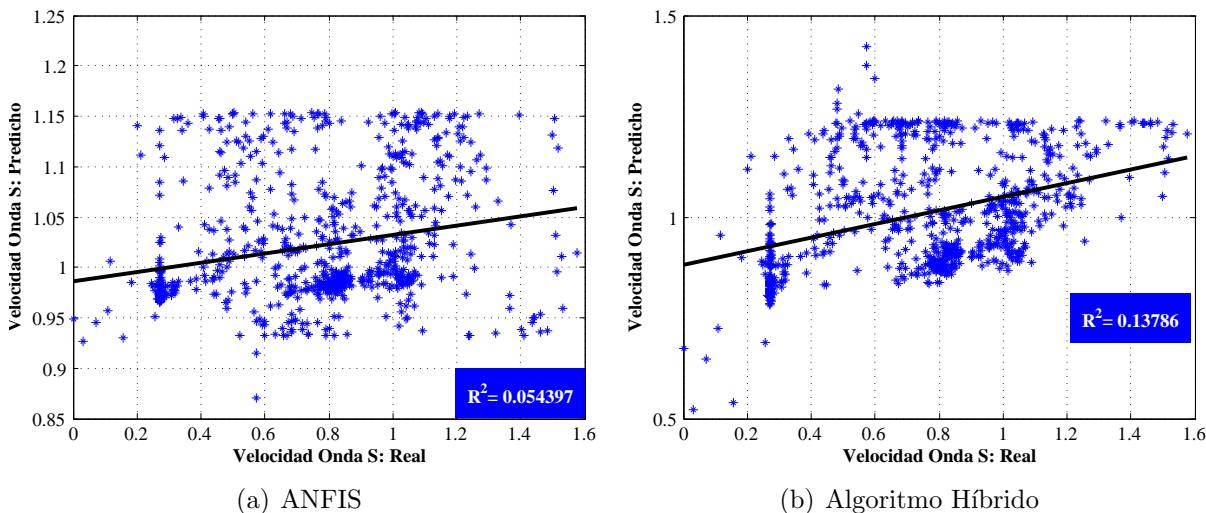


Figura 4.20: Valores Predichos vs. Valores Reales de Velocidad de Onda S utilizando ANFIS-Híbrido (técnica “Subtractive Clustering”) en el Pozo Control. Entrada: DEN.

4.2.3. Comentarios Adicionales

En esta primera sección, se realizó la predicción del Registro de Velocidad de Onda S con un registro de Entrada mediante dos ecuaciones de Algoritmos Genéticos y con ANFIS, al que luego se optimizan sus parámetros de las funciones de salidas de las reglas difusas en la búsqueda de obtener mejores resultados. En los cuatro casos se obtienen buenas estimaciones, como lo indican sus valores de R^2 los cuales no son inferiores a 0,7.

En todos los casos, los valores de RMSE más bajos y los valores de R^2 más altos se obtuvieron cuando el registro de entrada fue Velocidad de Onda P y Densidad tanto en el Pozo de Entrenamiento como en el Pozo Control en ese orden⁸.

El hecho que sea la **Velocidad de Onda P** el que ofrece la mejor predicción de Onda S se debe a la íntima relación que tienen éstas propiedades. En realidad, ambas miden la capacidad que tiene una formación determinada en transmitir ondas sonoras, dependiendo de las mismas variables como litología, textura y porosidad de las rocas presentes (Rider, 1996). Se diferencian sólo en la dirección que tienen éstas en su propagación.

Desde hace varias décadas, la relación entre ambas se ha estudiado tanto a nivel de sísmica como de pozo, donde se trabaja, bajo ciertas condiciones, incluso con la relación (Donati & Regueiro, 1988):

$$VP = 2 \times VS \quad (4.3)$$

O también de forma más general (Canning, n.d.), donde A y B dependerían de cada caso de estudio:

$$VP = A \times VS + B \quad (4.4)$$

Como se puede observar, en estos ejemplos, la relación entre VP y VS es “marcadamente lineal” como se mencionó en secciones anteriores. Es por esta razón que la Ecuación 1 AG modela incluso mejor el comportamiento de la Velocidad de Onda S que la Ecuación 2 AG.

En cuanto a la utilización de ANFIS y luego el Algoritmo Híbrido Difuso-Genético, se aprecia claramente como el segundo caso mejora una la predicción realizada en los Pozos de Entrenamiento y Control (GF-87 y GF-109) con el primer método. No obstante, no se distinguen diferencias notables de RMSE cuando se utilizan la técnica “Grid Partition” o “Subtractive Clustering”.

⁸Cabe destacar que todas las imágenes que se presentaron se corresponden con el pozo GF-109N, ya que (además que por tener menos datos de medición, hacen una visualización “más sencilla”) la finalidad es evaluar los sistemas generados en un pozo diferente a donde se toman los datos para el entrenamiento.

Por otra parte, la Velocidad de Onda S, como ya se ha dicho, depende intrínsecamente del tipo de roca por la que se efectúe su propagación; y uno de los parámetros que ayuda en la descripción e identificación de litologías es la **Densidad**. Canning (n.d.) indica que la **Relación de Gardner** es de la forma:

$$\log(\rho) = A \times \log(VP) + B \quad (4.5)$$

Por lo que pudiera existir una relación igualmente entre la Densidad y la Velocidad de Onda S, razón que explicaría el hecho que también se obtienen resultados satisfactorios cuando se usa como entrada a los algoritmos dicho registro.

4.3. Dos Registros de Entrada

En base a los registros disponibles en ambos pozos, para esta segunda sección se tomaron en cuenta como Registros de Entrada a las predicciones, todas las combinaciones posibles entre ellos. De manera que resultaron los siguientes:

- DEN-VP (Densidad-Velocidad de Onda P).
- DEN-SW (Densidad-Saturación de Agua).
- DEN-POR (Densidad-Porosidad).
- DEN-Vsh (Densidad-Volumen de Arcilla).
- VP-SW (Velocidad de Onda P-Saturación de Agua).
- VP-POR (Velocidad de Onda P-Porosidad).
- VP-Vsh (Velocidad de Onda P-Volumen de Arcilla).
- SW-POR (Saturación de Agua-Porosidad).
- SW-Vsh (Saturación de Agua-Volumen de Arcilla).
- POR-Vsh (Porosidad-Volumen de Arcilla).

4.3.1. Algoritmos Genéticos

Las ecuaciones 1 y 2 AG con las que se realizaron las predicciones quedan, para este caso de dos registros de entrada, tal y como están a continuación (basándose en las ecuaciones 3.5 y 3.6 respectivamente):

Ecuación 1 AG:

$$VS = a_1 \times REG_1 \bullet a_2 \times REG_2$$

Ecuación 2 AG:

$$VS = a_1 \times REG_1^{b_1} \bullet a_2 \times REG_2^{b_2}$$

donde:

a y b : son los parámetros que asocian al registro de entrada con el de salida.

REG_1 y REG_2 : Registros de Entrada.

Al efectuar los entrenamientos respectivos en cada ecuación, utilizando los registros de entrada ya mencionados, se obtuvieron los parámetros a y b : en las tablas 4.25 y 4.26 se muestran los valores a_1 y a_2 correspondientes a la Ecuación 1 AG y de a_1 , b_1 , a_2 , b_2 de la Ecuación 2 AG respectivamente en todos los casos.

Como se puede ver, la variación de la magnitud de los parámetros es similar a pesar del cambio de registros de entrada, es decir, el valor máximo no supera a 0,7 mientras que el mínimo 0,4 en todos los casos. Esto es consecuencia directa de la normalización que se realiza a los registros previo al entrenamiento del algoritmo.

Análogamente a lo mostrado en la sección 4.2 cuando se utilizó un registro de entrada a las ecuaciones, acá por ejemplo, el registro de Velocidad de Onda S en función de la Densidad y Porosidad con la Ecuación 1 AG se tiene efectuando la siguiente operación:

$$VS = 0,558 \times DEN + 0,495 \times POR \quad (4.6)$$

y con la Ecuación 2 AG:

$$VS = 0,569 \times DEN^{0,409} + 0,506 \times POR^{0,388} \quad (4.7)$$

Tabla 4.25: Parámetros “ a_1 ” y “ a_2 ” de la Ecuación 1 AG utilizando dos Registros de Entrada

Ecuación 1 AG		
Registros de Entrada	Parámetros “ a_1 ” “ a_2 ”	
DEN-VP	0,452	0,557
DEN-SW	0,558	0,530
DEN-POR	0,558	0,495
DEN-Vsh	0,655	0,506
VP-SW	0,616	0,463
VP-POR	0,663	0,443
VP-Vsh	0,577	0,570
SW-POR	0,620	0,614
SW-Vsh	0,633	0,587
POR-Vsh	0,537	0,520

De manera que, empleando los datos obtenidos en las tablas 4.25 y 4.26, se plantearon ecuaciones como las exemplificadas arriba para realizar las predicciones del registro de Velocidad de Onda S. Por consiguiente, para cada par de registros de entrada, se generaron los siguientes valores del RMSE en cada uno de los cálculos en los pozos GF-87N y GF-109N:

En estos casos, igualmente es la ecuación polinómica (número 2) la que describe mejor el comportamiento del registro de Velocidad de Onda S en función de cualesquiera dos registros de entrada que se encuentran en la tabla anterior (salvo algunas excepciones que se comentarán más adelante). Cabe destacar la mayor variación del RMSE ocurre cuando los registros de Entrada son **Saturación de Agua y Volumen de Arcilla (SW-Vsh)**.

Para ilustrar esta situación, se muestra en la figura 4.21 el resultado del cálculo de la Velocidad de Onda S con estos dos registros de entrada.

A la izquierda se encuentra el registro de Velocidad de Onda S calculado con la Ecuación 1 AG, mientras que a la derecha se encuentra la estimación a partir de la Ecuación 2 AG. Vemos como en ambos casos hay tres intervalos que se hace necesario destacar. Éstos son⁹:

⁹La interpretación litológica se obtuvo al estudiar los registros de la figura 3.2

Tabla 4.26: Parámetros “ a_1 ”, “ b_1 ”, “ a_2 ” y “ b_2 ” de la Ecuación 2 AG empleando dos Registros de Entrada

Ecuación 2 AG				
Registros de Entrada	Parámetros			
	“ a_1 ”	“ b_1 ”	“ a_2 ”	“ b_2 ”
DEN-VP	0,530	0,419	0,538	0,557
DEN-SW	0,567	0,508	0,506	0,465
DEN-POR	0,569	0,409	0,506	0,388
DEN-Vsh	0,645	0,458	0,488	0,576
VP-SW	0,525	0,513	0,539	0,487
VP-POR	0,642	0,441	0,465	0,416
VP-Vsh	0,601	0,464	0,551	0,479
SW-POR	0,620	0,516	0,556	0,436
SW-Vsh	0,540	0,492	0,545	0,571
POR-Vsh	0,555	0,451	0,577	0,489

- Entre 8000'-8170'. Hay grandes variaciones de la Velocidad de Onda S, lo cual puede deberse a su contenido litológico lutítico en la base y más arenoso hacia el tope, con intervalos donde la saturación de agua es máxima (igual a 1).
- Entre 8270'-8300'. Cambio granulométrico grano-creciente hacia arriba, con una porosidad que aumenta de 1% a 20% en la misma dirección.
- Entre 8400'-8500'. Donde pueden interpretarse un espesor de lutita arenosa de 50 pies y una arena “sucia” (con alto contenido de arcilla) del mismo espesor que le infrayace. La porosidad en este caso varía entre 10% y 30%.

A pesar que ninguna de las dos ecuaciones estima el comportamiento real del registro de Velocidad de Onda S (véase que el contenido litológico dominante en los 3 casos es lutita), con la Ecuación 2 AG se disminuye notoriamente la distancia entre ambas curvas, en efecto, muchos picos que se calculan con la Ecuación 1 AG son mucho menores que con la otra, de ahí que el RMSE en este pozo (Control) baje de 0,968 a 0,593.

Sin embargo, observando de nuevo la tabla 4.27 vemos cómo la mejor predicción se realizó utilizando como entrada los registros DENSIDAD - VELOCIDAD DE ONDA P¹⁰. En este caso,

¹⁰Esto tambien ocurre en el caso en que los registros de entrada son VP-POR

Tabla 4.27: RMSE para dos Registros de Entrada con Algoritmos Genéticos: Pozo de Entrenamiento.

Reg. Entrada	RMSE			
	Pozo de Entrenamiento		Pozo Control	
	Ec. 1 AG	Ec. 2 AG	Ec. 1 AG	Ec. 2 AG
DEN-VP	0,243	0,187	0,247	0,302
DEN-SW	0,538	0,267	0,725	0,520
DEN-POR	0,234	0,234	0,428	0,432
DEN-Vsh	0,488	0,313	0,635	0,518
VP-SW	0,392	0,227	0,501	0,423
VP-POR	0,264	0,224	0,293	0,345
VP-Vsh	0,356	0,252	0,460	0,414
SW-POR	0,598	0,326	0,901	0,651
SW-Vsh	0,622	0,317	0,968	0,593
POR-Vsh	0,335	0,277	0,521	0,519

el valor del RMSE en el Pozo de Entrenamiento disminuye de 0,243 a 0,187 al cambiar de ecuación. Mientras que en el Pozo Control hay un pequeño aumento de 0,055 en éste parámetro. Éstos aumentos puede deberse a lo que se explicó en la sección anterior, la dependencia entre la Velocidad de Onda S y Onda P, en este caso, es marcadamente lineal.

Asimismo, en la figura 4.22 se muestran los registros predichos utilizando las ecuaciones 1 y 2 con estos registros de entrada. El pequeño aumento del RMSE entre una y otra puede observarse entre las profundidades comprendidas entre 8270'-8300' y 8400'-8500', donde en ambos hay una disminución de la magnitud de las variaciones registro de Velocidad de Onda S predicho.

Sin embargo, para este caso en particular, debido al ligero aumento que sufren los valores de RMSE (un empobrecimiento de la estimación), se esperaría una disminución del R^2 , cosa que no ocurre. Éste parámetro aumenta entre la Ecuación 1 AG y la Ecuación 2 AG de 0,5933 a 0,6477 (figura 4.23); esto indica una mejor relación lineal entre los valores predichos y los valores reales en el segundo caso, lo que conlleva a afirmar que, a pesar de las discrepancias entre ambas gráficas en los intervalos mencionados, con la Ecuación 2 AG se realiza una mejor predicción de VS.

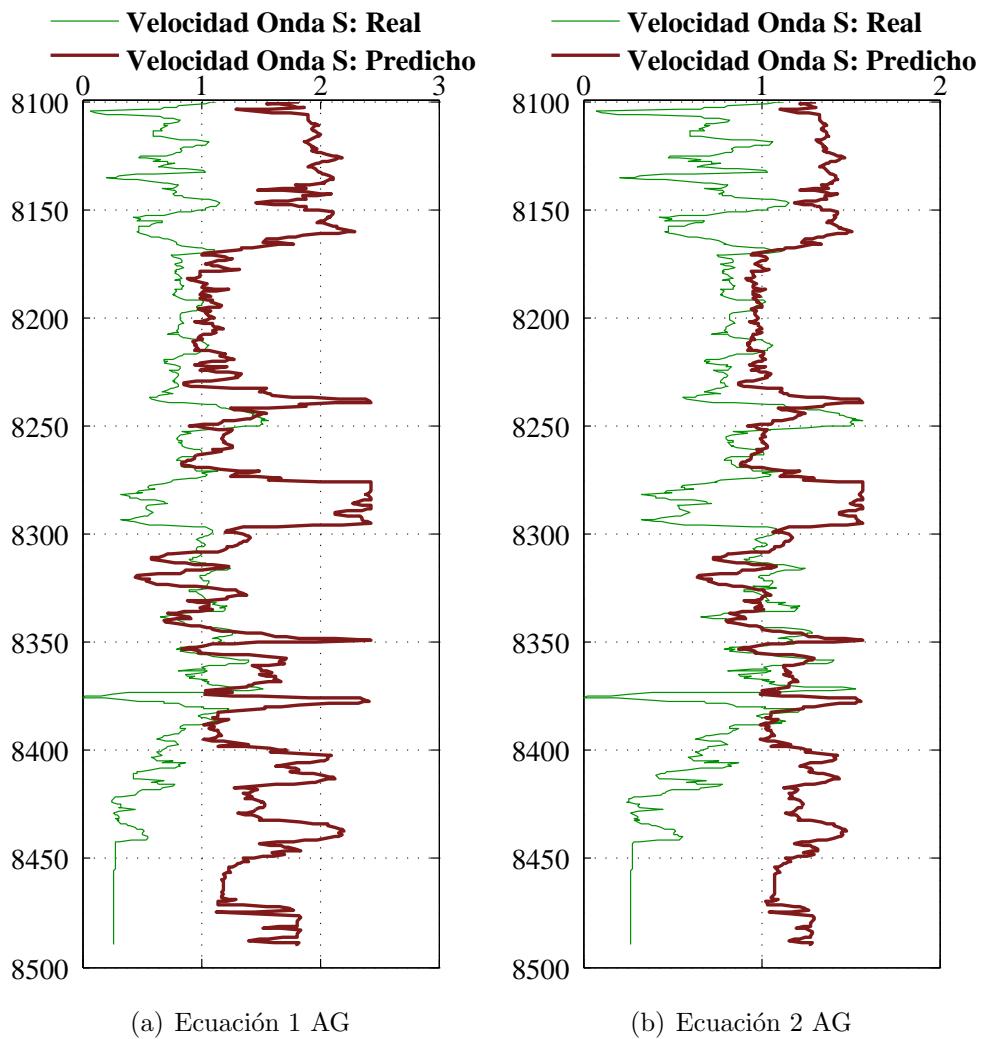


Figura 4.21: Registros Velocidad de Onda S: Predicho y Real en el Pozo Control. Registros de Entrada: SW-Vsh

Este fenómeno se puede ver en las profundidades 8100'-8170', y 8300'-8350'. A pesar que la litología en ambos casos es diferente (Véase la figura 3.2), las variaciones del registro estimado son menores en el caso de la Ecuación 2 AG con respecto al registro real.

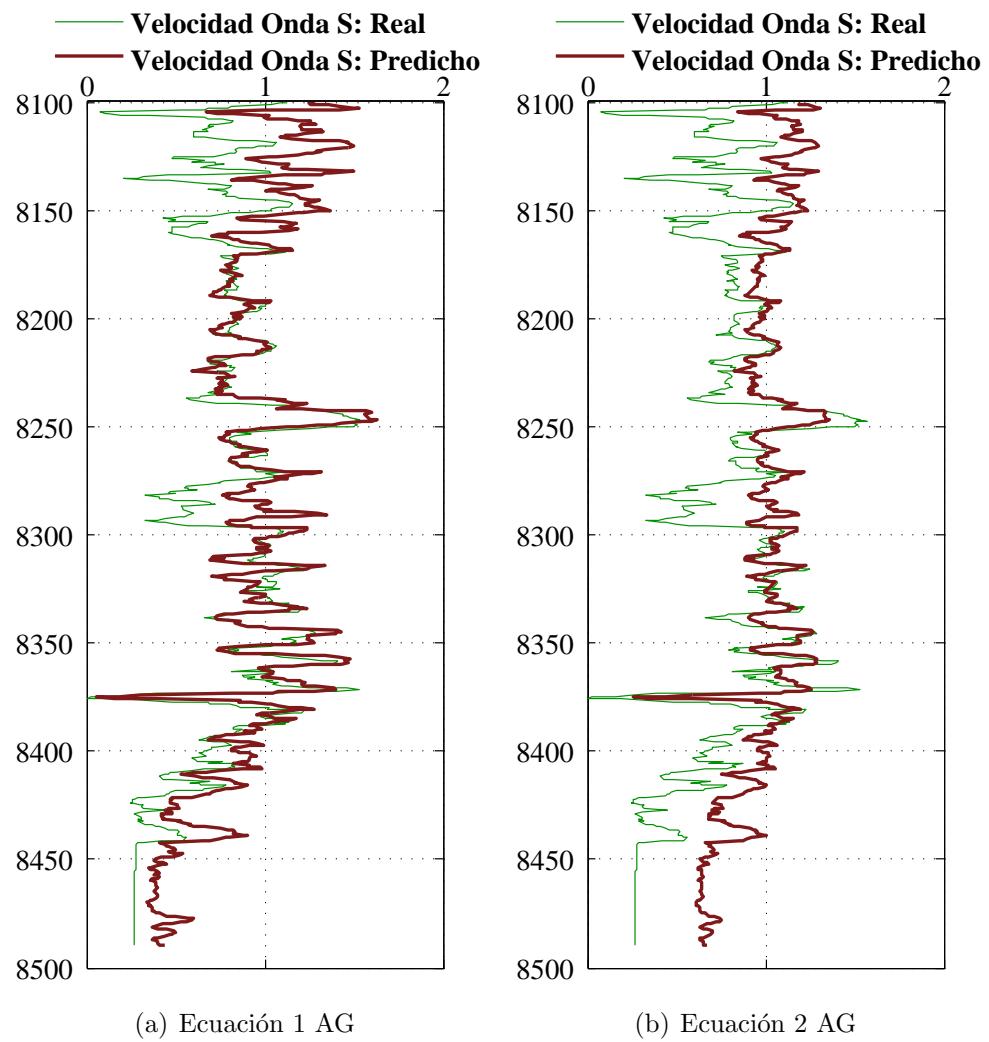
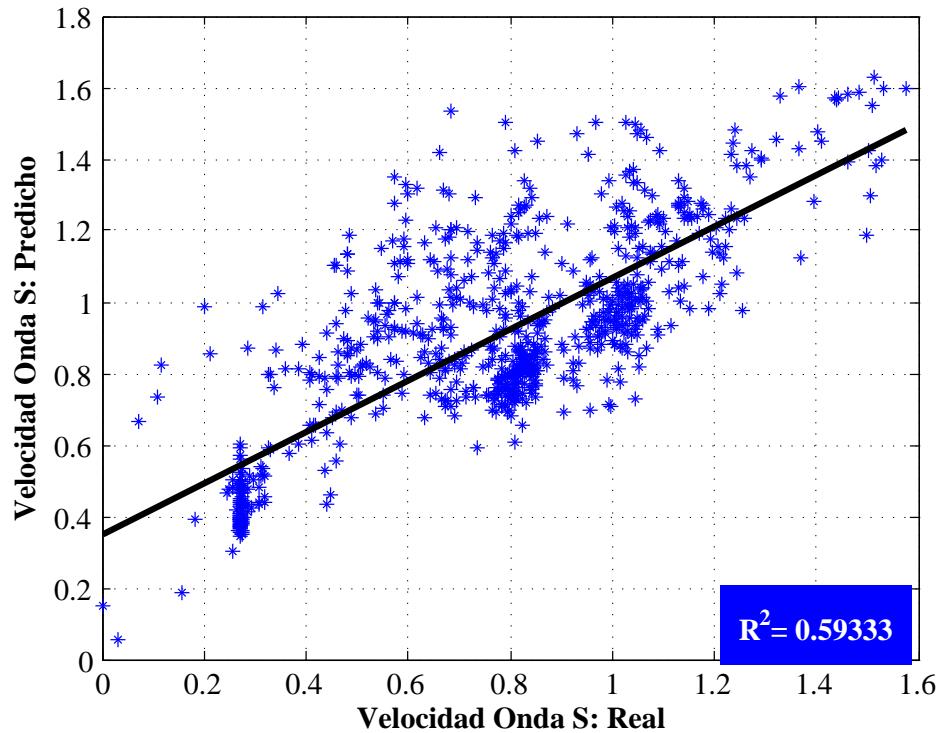
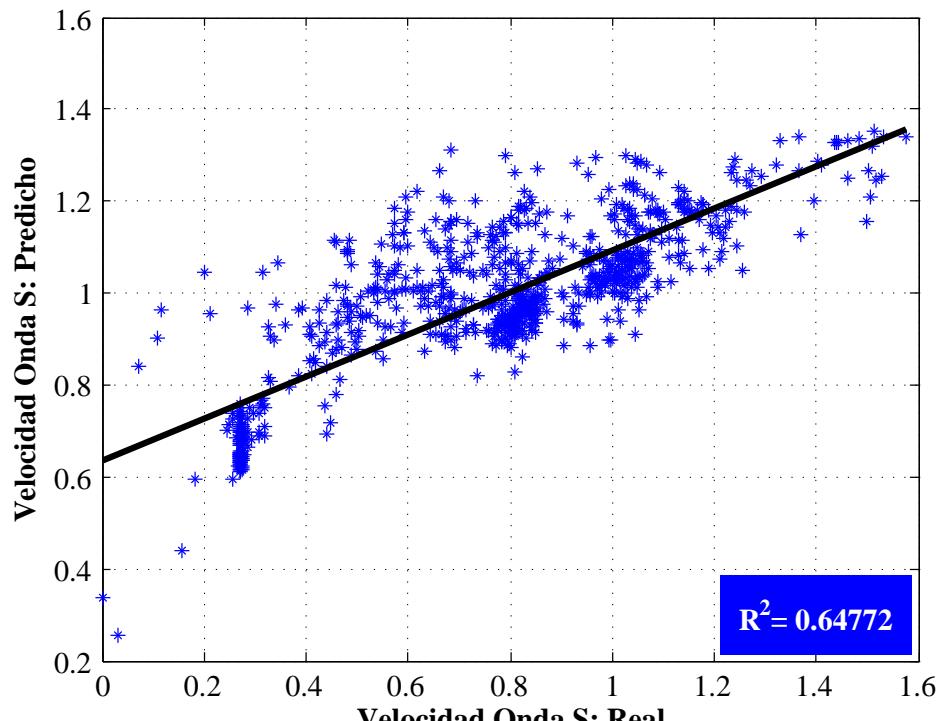


Figura 4.22: Registros de Velocidad de Onda S: Predicho y Real utilizando Algoritmos Genéticos en el Pozo Control. Entrada: DEN-VP.



(a) Ecuación 1 AG



(b) Ecuación 2 AG

Figura 4.23: Valores Predichos vs. Valores Reales de Velocidad de Onda S utilizando Algoritmos Genéticos en el Pozo Control. Entrada: DEN-VP

4.3.2. ANFIS-Algoritmo Híbrido

Al igual que en la sección 4.2.2 cuando se utilizó un registro de entrada en las predicciones del registro de Velocidad de Onda S, en este caso también se emplearon separadamente las dos alternativas que el ofrece el programa **Matlab versión R2009a** con ANFIS, la opción “Grid Partition” y “Subtractive Clustering”. Vale recordar que a cada caso se aplica posteriormente el Algoritmo Híbrido Difuso-Genético diseñado en esta investigación.

Por consiguiente, los resultados presentados en esta sección también se dividen en dos partes: primero “**Grid Partition**” y luego “**Subtractive Clustering**” para todos los registros de entrada seleccionados.

4.3.2.1. Parte I: Grid Partition

Según las pruebas preliminares realizadas, tal y como se obtuvo en la sección 4.1.2.2, la mejor combinación de reglas difusas a utilizar en este caso de dos registros de entrada fue [2 3], es decir, dos funciones de membresía para la primera entrada y tres funciones de membresía para la segunda entrada¹¹.

En la tabla 4.28 se encuentran los valores de RMSE en cada una de las predicciones realizadas (con ANFIS y luego con el Algoritmo Híbrido) tanto en el Pozo de Entrenamiento (GF-87N) como en el Pozo Control (GF-109N) y en la tabla 4.29 sus valores de R^2 correspondientes para ambos pozos respectivamente.

Vemos como solamente en casos muy puntuales el algoritmo híbrido logra disminuir el valor de RMSE respecto a la predicción previa realizada con ANFIS. Éstos son los casos en que los registros de entrada son: DEN-VP, SW-POR y SW-Vsh. Sin embargo, al observar los valores de R^2 en estos mismos casos, aunque se esperaría un aumento del mismo¹², esto no ocurre; en efecto, éste disminuye en el Pozo de Entrenamiento así: 0,3046 a 0,2499, 0,0199 a 0,0336 (aumento poco significativo) y 0,1645 a 0,0011 respectivamente, en el Pozo Control ocurre un hecho similar.

Empero, los valores de R^2 más altos se obtienen cuando se usó como variables de entrada los registros VELOCIDAD DE ONDA P Y POROSIDAD.

¹¹Es necesario recordar que la función de membresía seleccionada fue la *Gaussian*.

¹²Una baja de la Raíz del Error Medio Cuadrático entre ambos métodos indicaría una mejor predicción, por consiguiente, una mejor relación entre la variable predicha y la variable real (un R^2 más cercano a 1)

Tabla 4.28: RMSE para dos Registros de Entrada utilizando ANFIS-Híbrido (6 reglas difusas).

Reg. Entrada	RMSE			
	Pozo de Entrenamiento		Pozo Control	
	ANFIS	Híbrido	ANFIS	Híbrido
DEN-VP	0,255	0,328	0,707	0,415
DEN-SW	0,307	0,427	0,369	0,728
DEN-POR	0,670	1,169	0,427	1,149
DEN-Vsh	0,450	0,558	0,351	0,429
VP-SW	0,415	0,483	0,256	0,575
VP-POR	0,259	0,402	0,340	0,493
VP-Vsh	0,199	0,332	0,237	0,535
SW-POR	0,702	0,304	1,216	0,413
SW-Vsh	1,060	0,327	1,024	0,558
POR-Vsh	0,223	2,467	0,340	3,171

Tabla 4.29: R^2 para dos Registros de Entrada utilizando ANFIS-Híbrido (6 reglas difusas).

Reg. Entrada	R^2			
	Pozo de Entrenamiento		Pozo Control	
	ANFIS	Híbrido	ANFIS	Híbrido
DEN-VP	0,3046	0,2499	0,4186	0,0065
DEN-SW	0,0014	0,0528	0,1698	0,1164
DEN-POR	0,1579	0,1084	0,0429	0,0288
DEN-Vsh	0,1555	0,1151	0,3151	0,0202
VP-SW	0,2183	0,1214	0,5992	0,0553
VP-POR	0,2124	0,2493	0,6212	0,5209
VP-Vsh	0,4110	0,0242	0,6932	0,3500
SW-POR	0,0199	0,0336	0,0058	0,0459
SW-Vsh	0,1645	0,0011	0,0391	0,1247
POR-Vsh	0,1048	0,0343	0,0340	0,0117

Por esta razón, los resultados que se analizarán a continuación son los obtenidos en este caso. La combinación de reglas se muestran en la tabla 4.30.

Tabla 4.30: Combinación de Reglas Difusas generadas a través de la técnica “Grid Partition” para dos Registros de Entrada: VP-POR

Número de fm		Número de FS
Entrada		Salida: VS
VP	POR	
1	1	1
1	2	2
1	3	3
2	1	4
2	2	5
2	3	6

En conjunto, éstos números se interpretan de la siguiente manera para obtener las reglas difusas:

Si VP es fm_1 y POR es fm_1 **entonces** VS es FS_1

Si VP es fm_1 y POR es fm_2 **entonces** VS es FS_2

Si VP es fm_1 y POR es fm_3 **entonces** VS es FS_3

Si VP es fm_2 y POR es fm_1 **entonces** VS es FS_4

Si VP es fm_2 y POR es fm_2 **entonces** VS es FS_5

Si VP es fm_2 y POR es fm_3 **entonces** VS es FS_6

Tabla 4.31: Parámetros σ y c de la función de membresía *Gaussiana* generados en el SID mediante la técnica “Grid Partition”. Registros de Entrada: VP-POR

Número de fm	Registros de Entrada			
	VP		POR	
	sigma	c	sigma	c
Nº 1	0,5287	0,5001	0,1041	0,0599
Nº 2	0,5198	1,8037	0,4287	1,0476
Nº 3			0,4743	1,8236

Luego, en la tabla 4.31 se encuentran los valores correspondientes al ancho y centro de la función de membresía utilizada. Detállese como el campo Nº 3 de la entrada VP esta vacío (la combinación de reglas indica que son sólo dos funciones de membresía en cada entrada).

Tabla 4.32: Parámetros “*a*”, “*b*” y “*c*” de las funciones de salida, utilizando “Grid Partition”. Registros de Entrada: VP-POR

Función de Salida	Parámetros		
	a	b	c
Nº 1	-3,9001	-2,2223	2,4389
Nº 2	0,9645	0,3318	0,3209
Nº 3	-0,3862	0,3959	0,2858
Nº 4	-18,4552	23,5700	21,2323
Nº 5	2,2273	0,1256	-2,2308
Nº 6	-2,2095	-0,2602	4,1903

Además en la tabla 4.32 se encuentran los valores calculados mediante ANFIS de cada regla difusa. Se encuentran especificados “*a*”, “*b*” y “*c*” debido a que en cada función de salida (FS) se escribe ahora de la siguiente forma:

$$\text{FS1} : VS_1 = a_1 \times VP + b_1 \times POR + c_1$$

$$\text{FS2} : VS_1 = a_2 \times VP + b_2 \times POR + c_2$$

$$\text{FS3} : VS_1 = a_3 \times VP + b_3 \times POR + c_3$$

$$\text{FS4} : VS_1 = a_4 \times VP + b_4 \times POR + c_4$$

$$\text{FS5} : VS_1 = a_5 \times VP + b_5 \times POR + c_5$$

$$\text{FS6} : VS_1 = a_6 \times VP + b_6 \times POR + c_6$$

Tomando en consideración los datos de las tablas de la 4.30 a la 4.32, las reglas difusas generadas en ANFIS se escriben de la siguiente forma (*gaussmf=función de membresía gaussiana*):

Si *VP* es *gaussmf*[0,5287 0,5001] y *POR* es *gaussmf*[0,1041 0,0599] **entonces**
VS=-3,9001 * *VP* - 2,2223 * *POR* + 2,4389

Si *VP* es *gaussmf*[0,5287 0,5001] y *POR* es *gaussmf*[0,4287 1,0476] **entonces**
VS=0,9645 * *VP* + 0,3318 * *POR* + 0,3209

Si *VP* es *gaussmf*[0,5287 0,5001] y *POR* es *gaussmf*[0,4743 1,8236] **entonces**
VS=-0,3862 * *VP* + 0,3959 * *POR* + 0,2858

Si *VP* es *gaussmf*[0,5198 1,8037] y *POR* es *gaussmf*[0,1041 0,0599] **entonces**

$$VS = -18,4552 * VP + 23,5700 * POR + 21,2323$$

Si VP es $\text{gaussmf}[0,5198 \quad 1,8037]$ y POR es $\text{gaussmf}[0,4287 \quad 1,0476]$ **entonces**
 $VS = 2,2273 * VP + 0,1256 * POR - 2,2308$

Si VP es $\text{gaussmf}[0,5198 \quad 1,8037]$ y POR es $\text{gaussmf}[0,4743 \quad 1,8236]$ **entonces**
 $VS = -2,2095 * VP - 0,2602 * POR + 4,1903$

Los parámetros optimizados se encuentran en la tabla 4.33. Nótese, al compararla con la tabla 4.32 cómo los valores de las funciones de salida número 1 y 4 siguen siendo los mismos. Esto es como resultado que no existieron valores dentro de los datos de entrenamiento del sistema que cumplieran con las condiciones de esas reglas difusas.

Tabla 4.33: Parámetros “ a ”, “ b ” y “ c ” optimizados de las funciones de salida, utilizando “Grid Partition”. Registros de Entrada: VP-POR

Función de Salida	Parámetros		
	a	b	c
Nº 1	-3,9001	-2,2223	2,4389
Nº 2	0,3334	0,3881	0,3629
Nº 3	0,4903	0,5506	0,5379
Nº 4	-18,4552	23,5700	21,2327
Nº 5	0,6830	0,1478	0,7589
Nº 6	0,2456	0,1257	0,2686

En otras palabras, para llevar a cabo la optimización de los datos de la función de salida número 1, por ejemplo, se han de tomar los puntos que cumplan con las condiciones impuestas por el antecedente de la correspondiente regla difusa; es decir buscar aquellas profundidades dentro de los datos de entrenamiento que tengan datos de Velocidad de Onda P que se encuentren a media altura de la gaussiana de parámetros $\text{gaussmf}[0,5287 \quad 0,5001]$ y que a su vez datos de Porosidad que se encuentren en $\text{gaussmf}[0,1041 \quad 0,0599]$.

Con los datos de entrenamientos seleccionados, esto no fue posible en las reglas nº 1 y 4. Nótese además como esos parámetros que no fueron optimizados son incongruentes con respecto a los otros que si lo fueron. Es decir, el parámetro “ b ” de la FS4 es 23,57 muy superior al 0,5506 de la FS3 (es evidente la diferencia del orden de magnitud en todos los demás casos).

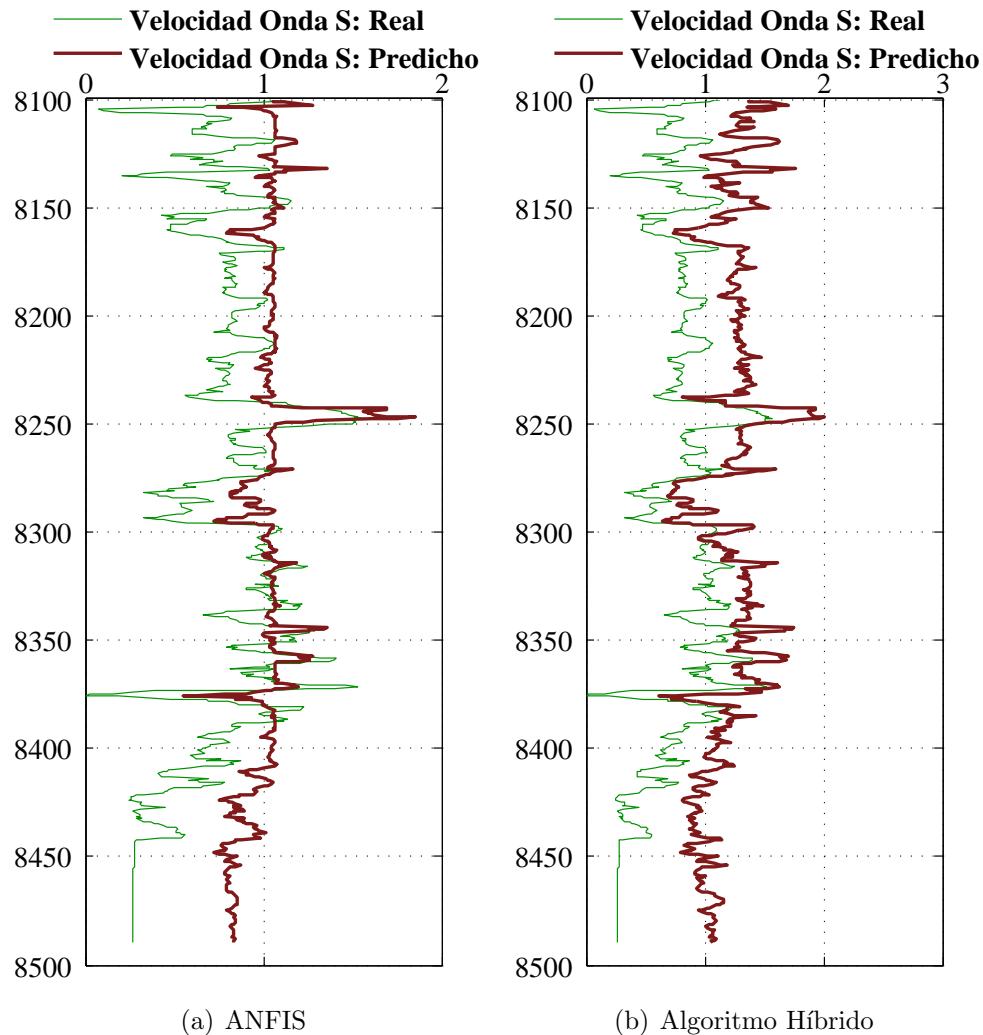


Figura 4.24: Registros de Velocidad de Onda S: Predicho y Real utilizando ANFIS-Híbrido (técnica “Grid Partition”) en el Pozo Control. Entrada: VP-POR

Este hecho ocurre en las otras predicciones realizadas con los demás registros de entrada. Por lo que también puede explicar porqué el Algoritmo Híbrido Difuso Genético, NO logra reconocer en ningún caso nuevos patrones de comportamiento aparte de los que pudieron haber sido calculados previamente utilizando ANFIS. Esto trae como consecuencia el fenómeno que se explicó anteriormente, el R^2 no aumenta (como se espera que ocurra) al aplicar la hibridización.

Finalmente, en la figura 4.24 se muestran los resultados obtenidos en la estimación del registro de Velocidad de Onda S mediante ANFIS (a la izquierda) y luego de la hibridización (a la derecha).

En cuanto a la figura 4.24(a) se observa que no hay un intervalo en particular en donde el registro estimado modele las variaciones que presenta el registro real. En donde se puede ver una semejanza del comportamiento (aunque también los valores están sobreestimados) es entre 8240'-8300', que se corresponde con una variación litológica que presenta un cambio que tiende a aumentar el tamaño de grano y luego finalizar con una capa lutítica hacia el tope de esta sección.

Mientras que, en la figura 4.24(b) se reconocen mejor patrones de comportamiento que el caso anterior, aunque la distancia entre la curva predicha y la real es mucho mayor¹³. Esto explica el hecho que el R^2 disminuya en entre un método y otro (figura 4.25) y que el RMSE también lo haga 0,340 a 0,493.

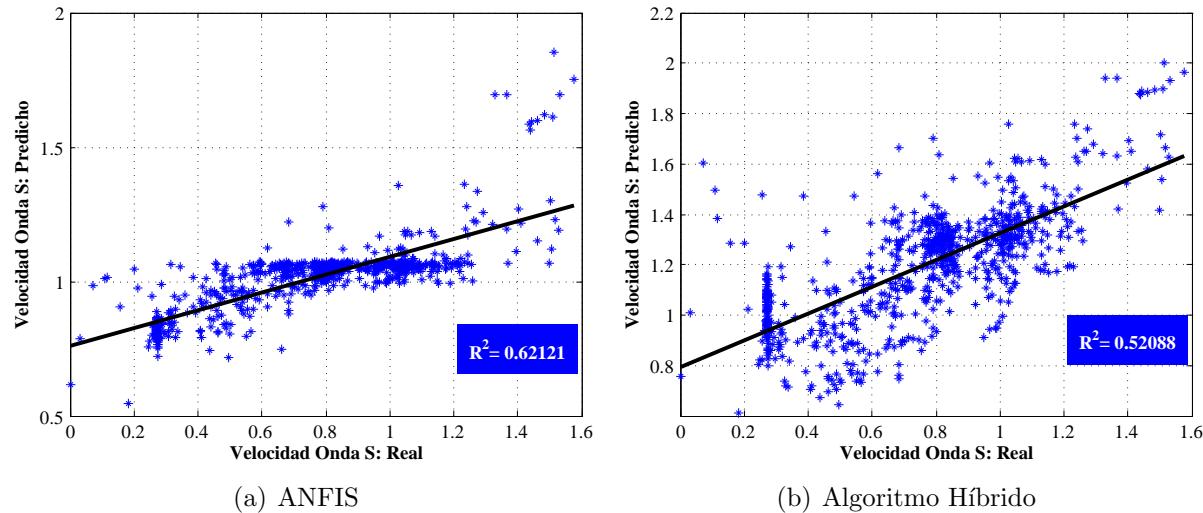


Figura 4.25: Valores Predichos vs. Valores Reales de Velocidad de Onda S utilizando ANFIS-Híbrido (técnica “Grid Partition”) en el Pozo Control. Entrada: VP-POR

Este hecho mencionado ocurre específicamente entre 8100'-8170', 8300'-8400', que cabe destacar que presentan continuidades litológicas diferentes, el primero se corresponde con una lutita cuyo contenido de arena va disminuyendo a medida que aumenta la profundidad y el segundo con intercalaciones de arena y lutita (es acá donde se encuentra el intervalo de las arenas productoras del pozo).

¹³Puede discernirse mucho mejor si se detalla el cambio que hay en la escala entre ambas gráficas

4.3.2.2. Parte II: Substractive Clustering

Como consecuencia de la aplicación de ésta técnica, la cantidad de reglas difusas varía por entrada, tal y como se muestra en la tabla 4.34. Además, en las tablas de la 4.35 a la 4.36 se muestran sus valores equivalentes de RMSE y R^2 en las predicciones.

Tabla 4.34: Número de Reglas Difusas obtenidas en cada caso con dos registros de entrada a través de “Substractive Clustering”

Reg. de Entrada	Nº Reglas Difusas
DEN-VP	2
DEN-SW	7
DEN-POR	5
DEN-Vsh	4
VP-SW	4
VP-POR	5
VP-Vsh	3
SW-POR	8
SW-Vsh	8
POR-Vsh	6

Cuando se trabajó en la sección anterior con un solo registro de entrada, vimos cómo no se obtenían buenos resultados cuando se realizaba la predicción a partir de la Porosidad, Saturación de Agua o Volumen de Arcilla. En este caso, son justamente en las predicciones en donde están involucrados estos registros donde el número de reglas difusas es mayor, lo que explica un intento del programa de reconocer modelos de comportamiento del registro real de Onda S.

Aún así, el objetivo no se logra en estos casos, basta con observar por ejemplo el RMSE entre la predicción realizada con ANFIS y luego con el Algoritmo Híbrido cuando los registros de entrada son SW-Vsh o POR-Vsh, en donde son 0,542-1,033 y 0,542-0,775 respectivamente. El valor de R^2 en ambos casos es poco significativo (0,0002-0,0079 y 0,0057-0,0247), evidenciando la nula relación lineal entre el registro real y el predicho.

No obstante, la mejor predicción (RMSE más bajos y R^2 más cercanos a uno) se realizó con los registros DENSIDAD-VELOCIDAD DE ONDA P, afirmando los buenos resultados obtenidos

Tabla 4.35: RMSE para dos Registros de Entrada con ANFIS-Híbrido (reglas difusas variables).

Reg. Entrada	RMSE			
	Pozo de Entrenamiento		Pozo Control	
	ANFIS	Híbrido	ANFIS	Híbrido
DEN-VP	0,195	0,212	0,376	0,314
DEN-SW	0,323	0,703	0,377	0,875
DEN-POR	0,362	0,266	0,415	0,512
DEN-Vsh	0,428	0,404	0,361	0,461
VP-SW	0,252	0,318	0,318	0,342
VP-POR	0,206	0,332	0,299	0,311
VP-Vsh	0,208	0,334	0,320	0,380
SW-POR	0,291	0,338	0,501	0,449
SW-Vsh	0,273	0,781	0,542	1,033
POR-Vsh	0,302	0,592	0,542	0,775

en el caso de un registro de entrada. En este caso, entonces se muestran a continuación la combinación de reglas difusas:

Estas se interpretan de la siguiente forma:

Si VP es $fm1$ y DEN es $fm1$ **entonces** VS es $FS1$
Si VP es $fm2$ y DEN es $fm2$ **entonces** VS es $FS2$

Nótese la variación sustancial que hay en el tratamiento de estas reglas difusas con las que se utilizan en la sección anterior “Grid Partition” para dos registros de entrada. En ese caso, las funciones de membresía se combinan entre sí creando todas las permutaciones posibles entre ellas. Esto no ocurre en este caso.

De manera que, en las tablas de la 4.38 a la 4.40, se encuentran primero los valores asociados a las funciones de membresía “ σ y c ” y luego los parámetros de las funciones de salida generados en ANFIS y luego los optimizados mediante algoritmos genéticos:

Tabla 4.36: R^2 para dos Registros de Entrada utilizando ANFIS-Híbrido (reglas difusas variables).

Reg. Entrada	R^2			
	Pozo de Entrenamiento		Pozo Control	
	ANFIS	Híbrido	ANFIS	Híbrido
DEN-VP	0,4063	0,4458	0,2775	0,6266
DEN-SW	0,0007	0,1138	0,0623	0,0112
DEN-POR	0,0931	0,0694	0,0514	0,0176
DEN-Vsh	0,1019	0,0113	0,2036	0,0223
VP-SW	0,2253	0,2151	0,3650	0,5377
VP-POR	0,3725	0,4356	0,5937	0,4830
VP-Vsh	0,3660	0,0069	0,4891	0,0550
SW-POR	0,0099	0,0356	0,0003	0,0002
SW-Vsh	0,1985	0,0886	0,0002	0,0079
POR-Vsh	0,0848	0,0539	0,0057	0,0247

Tabla 4.37: Combinación de Reglas Difusas generadas a través de la técnica “Substractive Clustering” para dos Registros de Entrada: DEN-VP

		Número de fm	Número de FS
		Entrada	Salida: VS
VP	POR		
1	1		1
2	2		2

Con estos parámetros se construyen las gráficas que se muestran en las figuras 4.26 y 4.27. En la primera de ellas se encuentran los registros predichos y reales en el pozo control y en la segunda la gráfica donde se muestran su valores de R^2 correspondientes.

Vemos cómo el valor de R^2 aumenta (de 0,2747 a 0,62662) cuando se hace la estimación mediante el algoritmo híbrido difuso-genético, lo cual evidencia una mejor predicción del registro de Velocidad de Onda S. A lo largo de todo el pozo hay ejemplos de ello.

Tabla 4.38: Parámetros σ y c de la función de membresía *Gaussiana* generados en el SID mediante la técnica “Substractive Clustering”. Registros de Entrada: DEN-VP

Número de fm	Registros de Entrada			
	VP		POR	
	sigma	c	sigma	c
Nº 1	0,2365	0,8491	0,1254	0,9864
Nº 2	0,1165	1,5410	0,2608	1,2216

Tabla 4.39: Parámetros “ a ”, “ b ” y “ c ” de las funciones de salida, utilizando “Substractive Clustering”. Registros de Entrada: DEN-VP

Función de Salida	Parámetros		
	a	b	c
Nº 1	0,2987	-0,0299	0,7950
Nº 2	0,1232	0,6861	0,0756

Tabla 4.40: Parámetros “ a ”, “ b ” y “ c ” optimizados de las funciones de salida, utilizando “Substractive Clustering”. Registro de Entrada: DEN-VP

Función de Salida	Parámetros		
	a	b	c
Nº 1	0,2926	0,4039	0,3838
Nº 2	0,0627	0,5648	0,5568

Por ejemplo, en los intervalos comprendidos entre 8170'-8240' y entre 8300'-8400' (arena productora del pozo). Ambos casos se corresponden con arenas que tienen una porosidad que varía entre el 10% y el 25%. La variación de la curva predicha con respecto a la real de Onda S es mucho menor con el sistema híbrido.

Similar ocurre en las altas fluctuaciones de los primeros 70 pies del registro y en los últimos 100 pies. A pesar que la litología es diferente en este caso (son cuerpos dominante mente lutíticos), es evidente una mejoría en la predicción con el algoritmo híbrido.

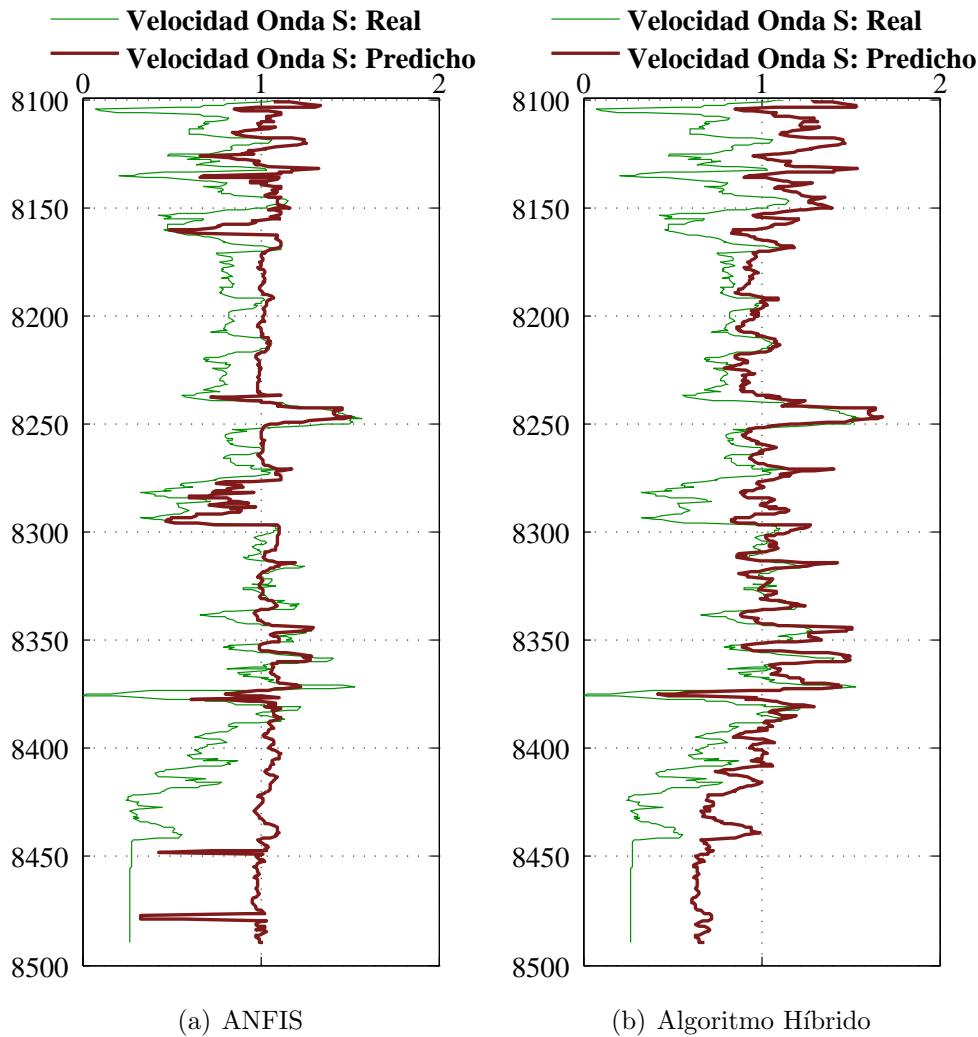


Figura 4.26: Registros de Velocidad de Onda S: Predicho y Real utilizando ANFIS-Híbrido (técnica “Subtractive Clustering”) en el Pozo Control. Entrada: DEN-VP

Cuando se realizaron las predicciones utilizando un registro de entrada, los mejores resultados se obtuvieron con Velocidad de Onda P y Densidad, por lo que es de esperarse que una combinación entre ambos registros sea el que arroje una mejor predicción.

En efecto, utilizando tanto Algoritmos Genéticos como el Sistema de Inferencia Neuro-Difuso, esta combinación arroja mejores resultados entre todas las combinaciones posibles de dos registros de entrada, aunque los valores de R^2 tanto en el Pozo de Entrenamiento como en el Pozo Control, no son tan altos en comparación con los que se obtuvieron en un registro de entrada (Velocidad de Onda P).

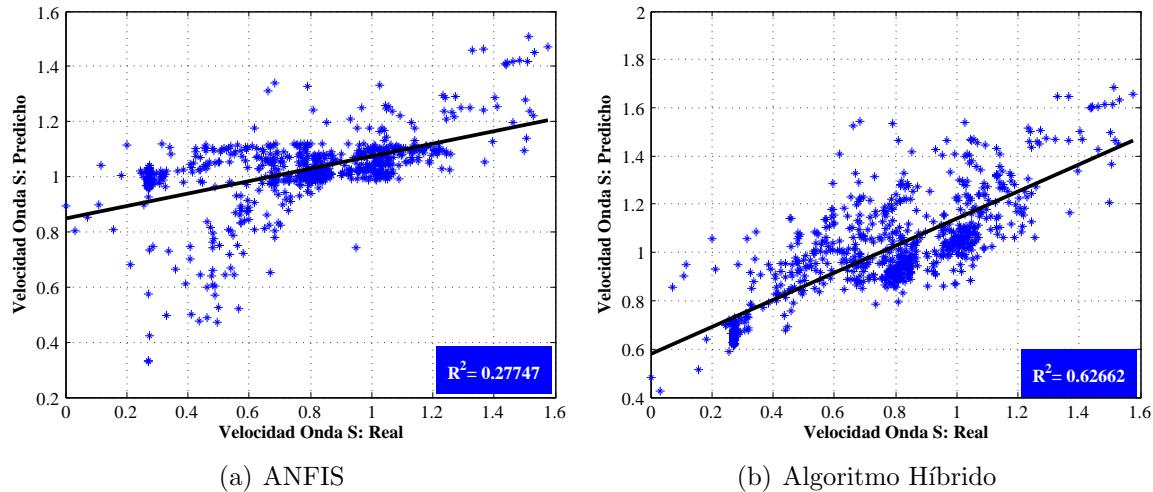


Figura 4.27: Valores Predichos vs. Valores Reales de Velocidad de Onda S utilizando ANFIS-Híbrido (técnica “Subtractive Clustering”) en el Pozo Control. Entrada: DEN-VP

4.4. Tres Registros de Entrada

Debido a que en las secciones anteriores los mejores resultados se obtuvieron con los registros de entrada Velocidad de Onda P, Densidad y Porosidad, se procuró entonces realizar estimaciones en donde se incluyeran éstos en su mayoría, quedando así las combinaciones siguientes:

- DEN-VP-SW (Densidad - Velocidad de Onda P - Saturación de Agua).
 - DEN-SW-POR (Densidad - Saturación de Agua - Porosidad).
 - DEN-POR-Vsh (Densidad - Porosidad - Volumen de Arcilla).
 - VP-SW-POR (Velocidad de Onda P - Saturación de Agua - Porosidad).
 - VP-POR-Vsh (Velocidad de Onda P - Porosidad - Volumen de Arcilla).
 - VP-DEN-Vsh (Velocidad de Onda P - Densidad - Volumen de Arcilla).
 - VP-DEN-POR (Velocidad de Onda P - Densidad - Porosidad).
 - VP-SW-Vsh (Velocidad de Onda P - Saturación de Agua - Volumen de Arcilla).

4.4.1. Algoritmos Genéticos

Las ecuaciones 1 y 2 AG con las que se realizan las predicciones de Velocidad de Onda S quedan, para este caso tal y como están a continuación (basándose en las ecuaciones 3.5 y 3.6 respectivamente):

Ecuación 1 AG:

$$VS = a_1 \times REG_1 \bullet a_2 \times REG_2 \bullet a_3 \times REG_3$$

Ecuación 2 AG:

$$VS = a_1 \times REG_1^{b_1} \bullet a_2 \times REG_2^{b_2} \bullet a_3 \times REG_3^{b_3}$$

donde: a y b : son los parámetros que asocian al registro de entrada con el de salida.

REG_1 y REG_2 : Registros de Entrada.

De manera similar a secciones anteriores, aquí se generaron los parámetros “ a_1 , a_2 y a_3 ” de la Ecuación 1 y “ a_1 , b_1 , a_2 , b_2 , a_3 , b_3 ” de la Ecuación 2, que se muestran en las tablas B.15 y B.16, en el Apéndice B.2.1.

Igualmente, los valores de RMSE y R^2 se encuentran en las tablas B.17 y B.18 en el Apéndice B.2.1. De éstas es posible observar que la mejor estimación (los valores de RMSE más bajos y de R^2 más altos en los pozos GF-87N y GF-109N) se realizó con la Entrada: Velocidad de Onda P - Densidad - Porosidad. A continuación se muestra las dos ecuaciones características obtenidas en este caso:

Ecuación 1 AG:

$$VS = 0,349 \times VP + 0,363 \times DEN + 0,360 \times POR$$

Ecuación 2 AG:

$$VS = 0,396 \times VP^{0,544} + 0,336 \times DEN^{0,501} + 0,375 \times POR^{0,314}$$

En la figura 4.28 se muestran el registro estimado de Onda S usando VP-DEN-POR y en la figura 4.29 sus valores de R^2 . A lo largo de todo el registro se observa cómo ambas ecuaciones existe diferencia entre el predicho y el real. En efecto, sólo en el intervalo comprendido entre

8300' - 8350', que es donde se encuentra la arena productora, las dos gráficas realizan una estimación satisfactoria.

Cabe agregar que la estimación del comportamiento de la Velocidad de Onda S no mejora en relación a las gráficas obtenida con dos o un registro de entrada.

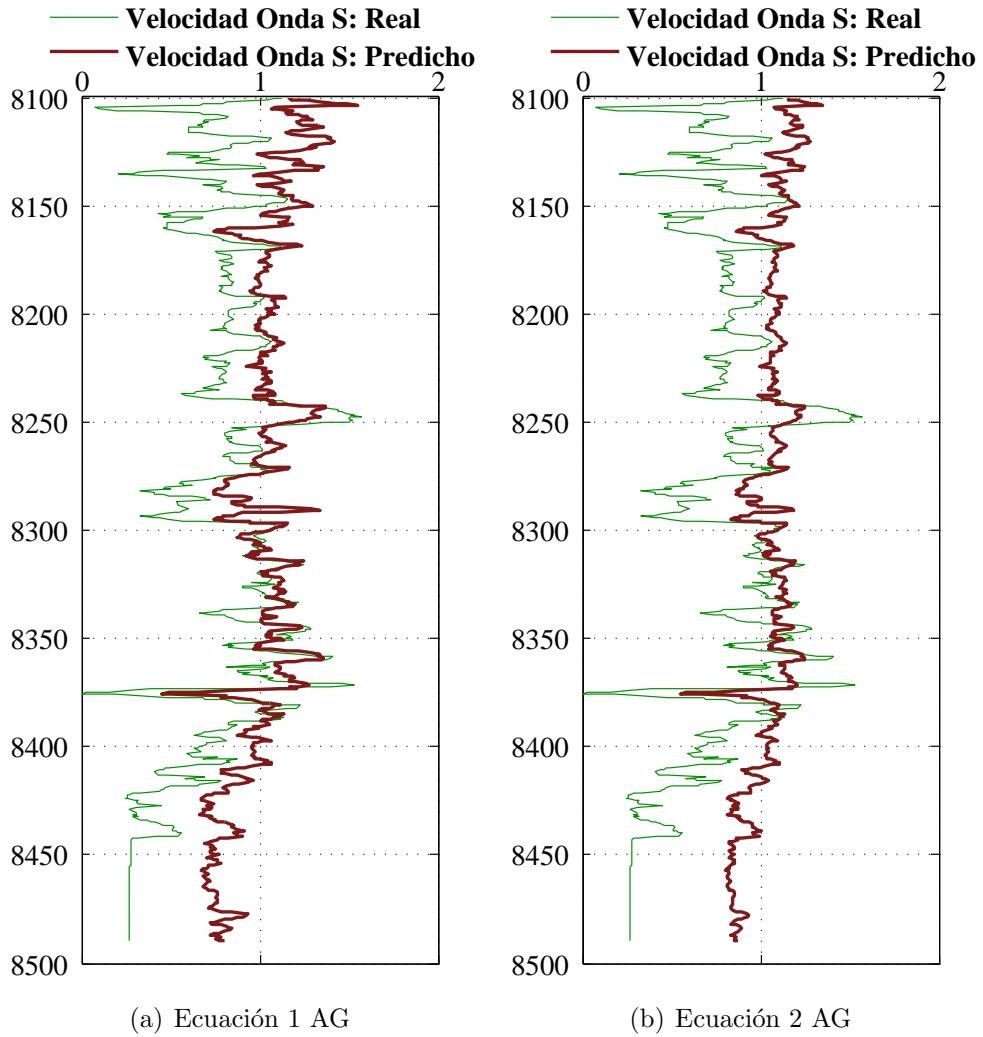


Figura 4.28: Registros de Velocidad de Onda S: Predicho y Real utilizando Algoritmos Genéticos en el Pozo Control. Entrada: VP-DEN-POR.

4.4.2. ANFIS-Algoritmo Híbrido

Tanto en la “Parte I: Grid Partition” como en la “Parte II: Subtractive Clustering” se generaron resultados que son análogos a las secciones anteriores, es decir, valores de RMSE y R^2 , combinación de funciones de membresía para obtener reglas difusas, parámetros “ σ y c ” de las funciones de membresía, parámetros “ a ” - “ b ” - “ c ” - “ d ” de las funciones de salida

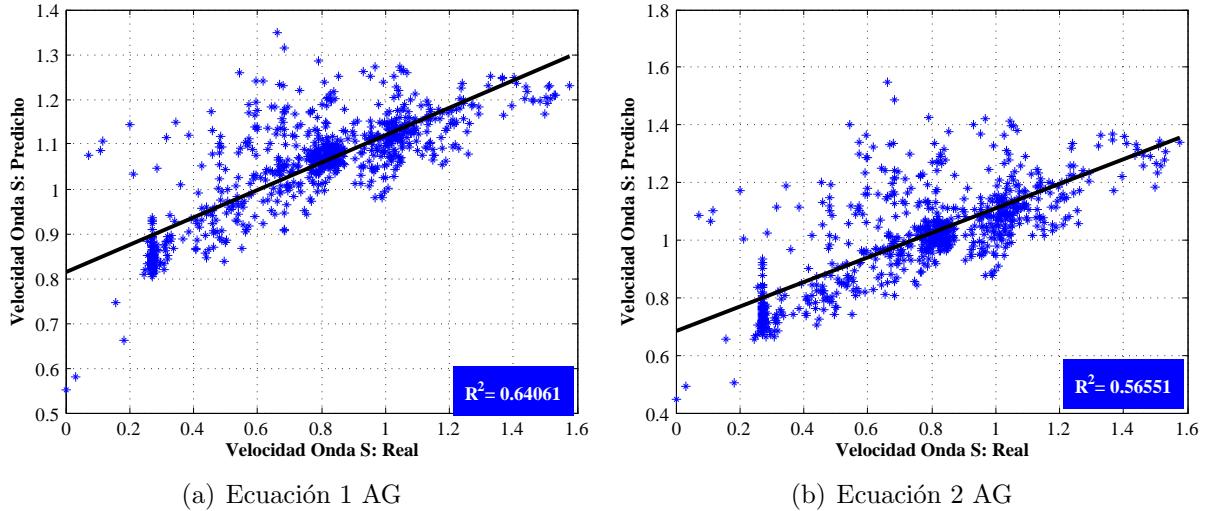


Figura 4.29: Valores Predichos vs. Valores Reales de Velocidad de Onda S utilizando Algoritmos Genéticos en el Pozo Control. Entrada: VP-DEN-POR

generadas en el ANFIS y los optimizados en el sistema híbrido; es por esto que, para ambos casos, se encuentran en el Apéndice B.

4.4.2.1. Parte I: Grid Partition

A continuación se muestran el registro de Velocidad de Onda S estimado con 6 reglas difusas y la función de membresía Gaussiana (debido a que esta combinación fue la que mejor resultados arrojó en las pruebas preliminares realizadas). A partir de la página 157 se muestran los resultados obtenidos en esta sección.

La mejor estimación se realizó con la entrada Velocidad de Onda P - Densidad - Porosidad, donde el valor de R^2 en el caso de ANFIS es 0,562 y con el algoritmo híbrido de 0,293 (Véase la figura 4.31).

Esta disminución se puede observar igualmente la figura 4.30, donde el sistema propuesto en este trabajo aumenta los picos de Velocidades de Onda S estimados a través del Sistema de Inferencia Difusa, específicamente en los intervalos 8100'-8150', 8240'-8250', 8270'-8300', 8330'-8400'.

Tanto la porosidad como el contenido litológico en éstos intervalos es variado, por lo que este empobrecimiento en la predicción se debe a la incapacidad que tiene el algoritmo en la optimización de todas las reglas difusas.

Por ejemplo, para este caso, es la regla difusa número 6 la que permanece invariable (esto se observa al comparar las tablas B.23 y B.24)

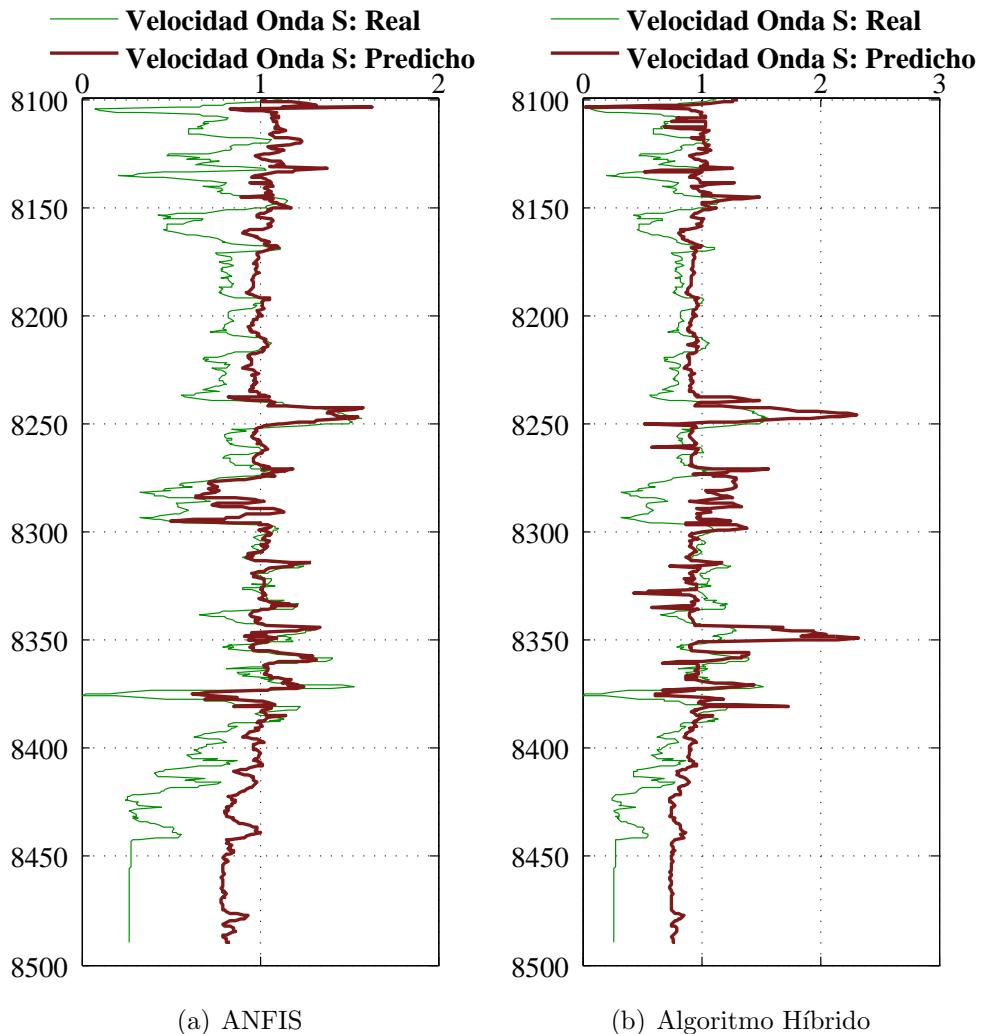


Figura 4.30: Registros de Velocidad de Onda S: Predicho y Real utilizando ANFIS-Híbrido (técnica “Grid Partition”) en el Pozo Control. Registros de Entrada: VP-DEN-POR

4.4.2.2. Parte II: Subtractive Clustering

Como ya se ha mencionado, con la técnica “Subtractive Clustering” se utilizaron reglas difusas dependiendo de las entradas en las predicciones, mostrándose en la tabla 4.41. Véase que el número es menor en el caso en que la entrada es VELOCIDAD DE ONDA P-DENSIDAD-POROSIDAD, reflejando así que hay una mayor relación entre éstos y Velocidad de Onda S.

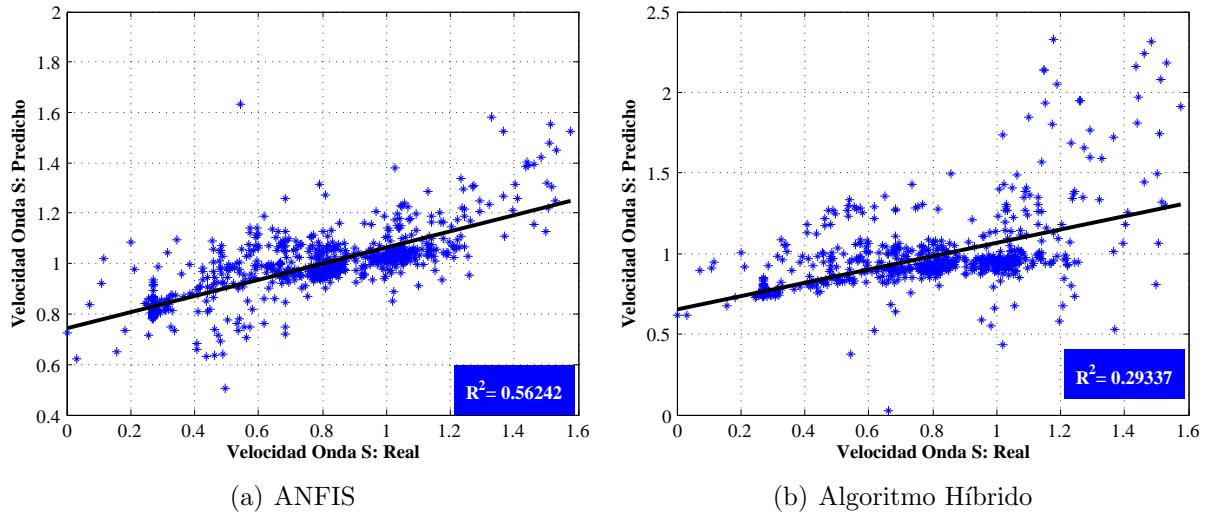


Figura 4.31: Valores Predichos vs. Valores Reales de Velocidad de Onda S utilizando ANFIS-Híbrido (técnica “Grid Partition”) en el Pozo Control. Registros de Entrada: VP-DEN-POR

Tabla 4.41: Número de Reglas Difusas obtenidas en cada caso para tres registros de entrada a través de “Subtractive Clustering”

Reg. de Entrada	Nº Reglas Difusas
DEN-VP-SW	8
DEN-SW-POR	9
DEN-POR-Vsh	5
VP-SW-POR	8
VP-POR-Vsh	5
VP-DEN-Vsh	5
VP-DEN-POR	4
VP-SW-Vsh	8

Es efectivamente con esta combinación (VP-DEN-POR) con la que se realiza una mejor estimación, mostrando valores de R^2 : 0,248 en el caso de ANFIS y 0,431 con el sistema híbrido (Véase la figura 4.33).

A diferencia cuando se utilizó “Grid Partition”, en este caso si se aprecia una clara diferencia entre ambos sistemas en los registros estimados (figura 4.32). Véase por ejemplo los intervalos entre:

1. 8100'-8170'.
2. 8170'-8240'.
3. 8300'-8400'.
4. 8400'-8450'.

En el caso de ANFIS, las variaciones del registro calculado es mínima, es decir, no hay un reconocimiento de patrones de comportamiento del registro de Onda S real. Sin embargo, cuando se aplica el algoritmo híbrido difuso-genético, hay un reconocimiento mucho mayor, sobre todo en los intervalos 2 y 4 que se corresponde con capas arenosas de porosidades entre 10 y 20 %.

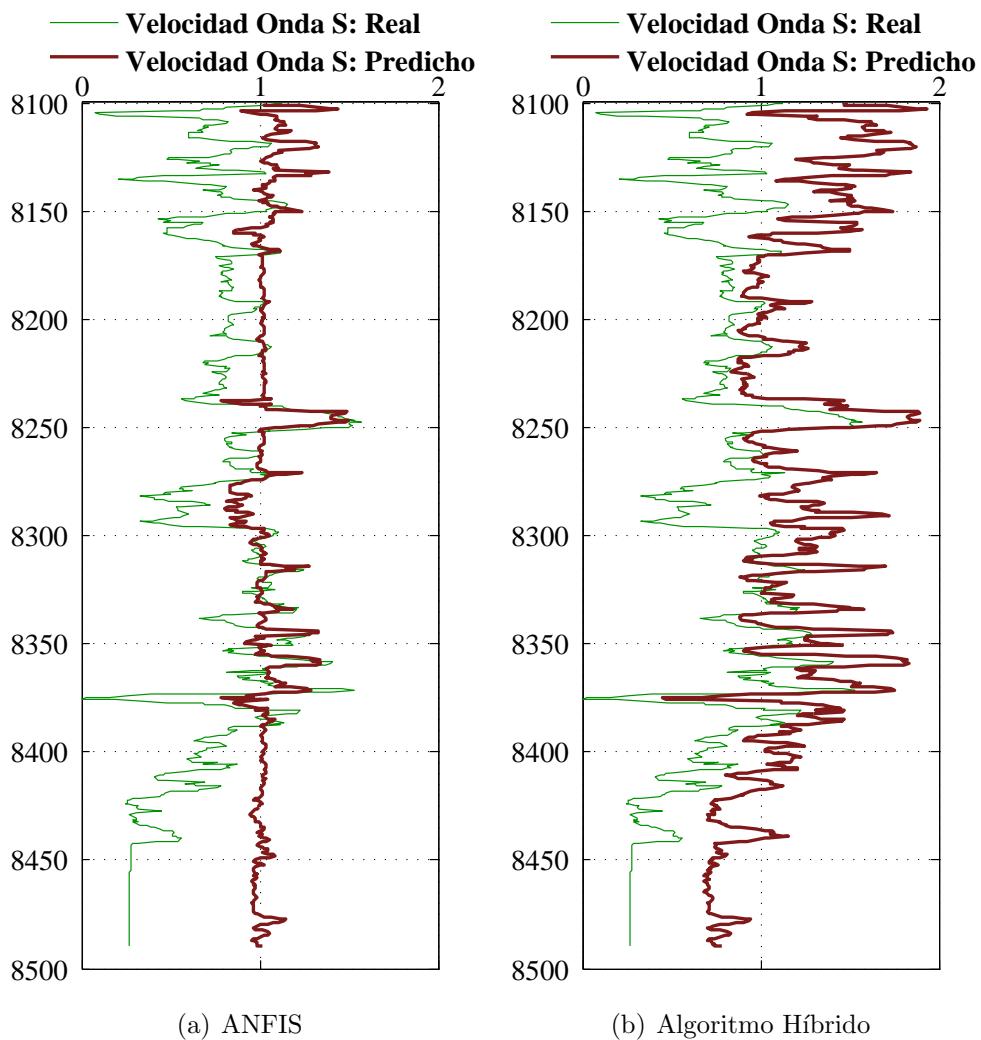


Figura 4.32: Registros de Velocidad de Onda S: Predicho y Real utilizando ANFIS-Híbrido (técnica “Substractive Clustering”) en el Pozo Control. Registros de Entrada: VP-DEN-POR

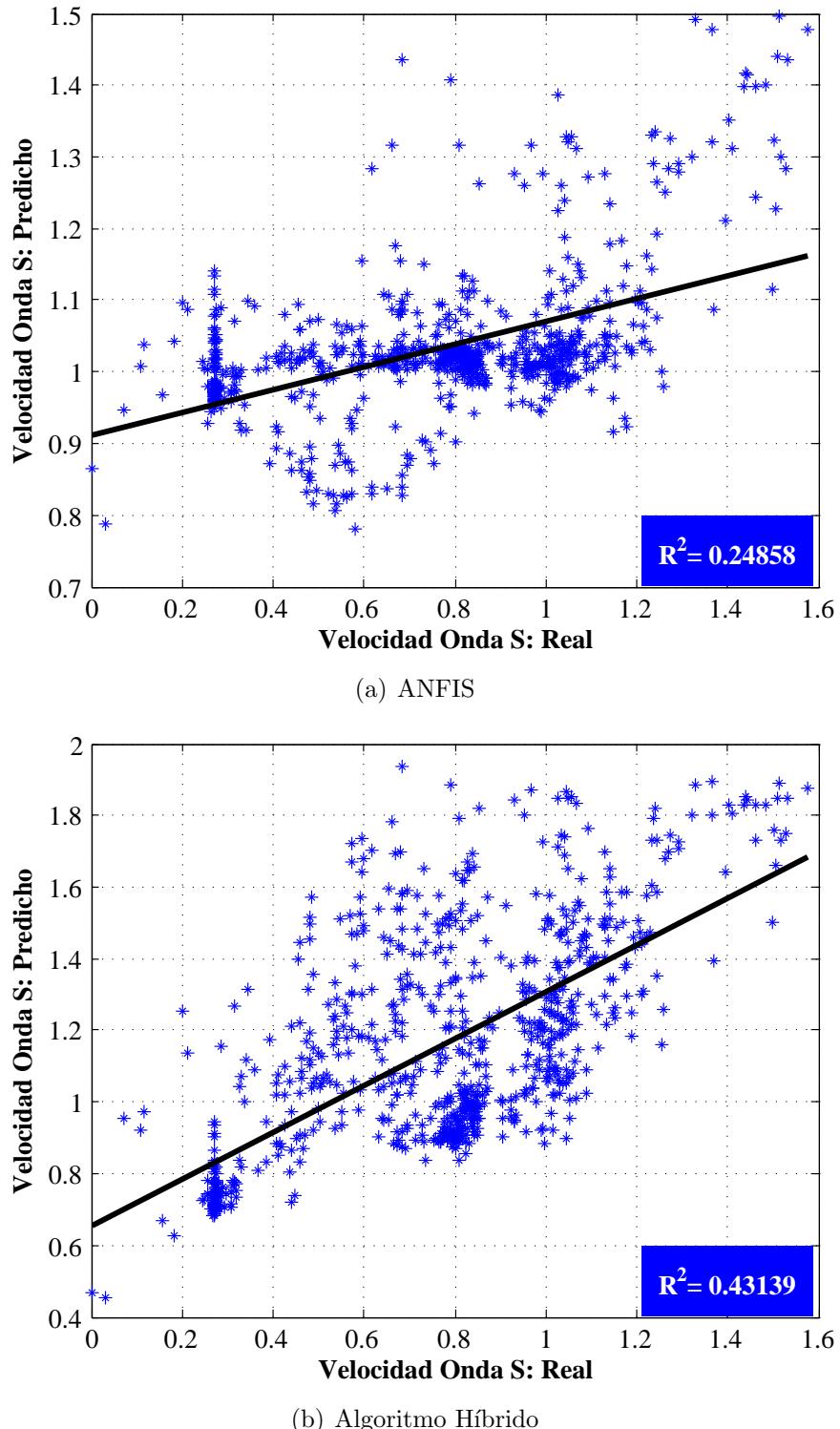


Figura 4.33: Valores Predichos vs. Valores Reales de Velocidad de Onda S utilizando ANFIS-Híbrido (técnica “Subtractive Clustering”) en el Pozo Control. Registros de Entrada: VP-DEN-POR

CONCLUSIONES Y RECOMENDACIONES

En la presente investigación, se utilizaron dos sistemas computacionales basados en conceptos de Lógica Difusa-Redes Neuronales (Sistema de Inferencia Neuro-Difusa) y Algoritmos Genéticos para la estimación de Registro de Velocidad de Onda S, a partir de otros registros disponibles en dos pozos del campo Guafita (GF-87N y GF-109N). Estos registros son: Densidad, Velocidad de Onda P, Saturación de Agua, Porosidad y Volumen de Arcilla.

En el caso de Algoritmos Genéticos, se utilizaron dos ecuaciones para estimar el registro de velocidad, una lineal y una polinómica. Mientras que, en el caso de Lógica Difusa, se usó un algoritmo adaptativo neuro-difuso, capaz de, por ejemplo, reconocer patrones de comportamiento de un registro de pozo. Una vez realizada la predicción del registro mediante esta técnica, se propuso una metodología que permite optimizar sus reglas difusas mediante algoritmos genéticos, es decir, combinando ambos sistemas. Los resultados obtenidos permiten, entonces, establecer lo siguiente:

- En general, entre las dos ecuaciones utilizadas para la estimación del registro de pozo mediante algoritmos genéticos, se obtuvieron mejores resultados cuando su utilizó la función polinómica.
- Cuando se utiliza un registro de entrada, tanto los Algoritmos Genéticos como el Sistema de Inferencia Neuro Difusa, presentan buenas estimaciones del registro de Velocidad de Onda S cuando la entrada es Velocidad de Onda P y Densidad. En el primer caso, la relación entre los valores predichos y los reales (R^2) es superior a 0,7 y en el segundo a 0,13.
- Igualmente, cuando se usa un registro de entrada, el Algoritmo Híbrido propuesto en esta investigación logra mejorar la estimación de Velocidad de Onda S con respecto a la predicción realizada con el Sistema de Inferencia Difusa, siempre y cuando exista una relación previa con los registros de entrada. Por ejemplo, el RMSE disminuye y el R^2 aumenta en el caso cuando el registro de entrada es Densidad (van de 0,271 a 0,211 y

de 0,0762 a 0,1391 en el Pozo Control) y Velocidad de Onda P. Sin embargo, esto no ocurre cuando son Saturación de Agua o Volumen de Arcilla.

- Con dos registros de entrada, mediante Algoritmos Genéticos (usando la ecuación polinómica) se obtienen RMSE menores que al aplicar el Sistema de Inferencia Difusa (SID). El caso más representativo es la combinación de entrada Velocidad de Onda P-Densidad, que arrojan RMSE más bajos, lo que indica la relación entre estas propiedades con la Velocidad de Onda S.
- En términos del error cuadrático medio (RMSE), se demotró que, utilizando la técnica “Subtractive Clustering”, se obtienen mejores predicciones que al usar “Grid Partition” usando uno y dos registros de entrada. Sin embargo, esto puede deberse a que, debido a la forma en cómo son combinadas las funciones de membresía, el Sistema de Inferencia se está efectuando de forma particular a estos pozos.
- Cuando se utilizan dos y tres registros de entrada, el Algoritmo Híbrido Difuso-Genético propuesto tiene sus limitaciones. Es decir, no se reconocen nuevos patrones de comportamiento del registro de Velocidad de Onda S real con respecto a la predicción realizada mediante el Sistema de Inferencia Difusa. Esto se observa cuando el RMSE no disminuye ni el R^2 aumenta.

Esto ocurre porque, al seleccionarse los parámetros de las reglas difusas previamente generadas en ANFIS para el proceso de optimización, no se encuentran los datos suficientes (al usar conceptos de la lógica clásica en cuanto a la pertenencia de un elemento a un conjunto) para realizar el entrenamiento, por lo que no se logran aplicar los Algoritmos Genéticos en todas las reglas y, por consiguiente, no se mejora la estimación de Velocidad de Onda S.

- Con tres registros de entrada, la mejor combinación de registros de entrada en todas las técnicas es Velocidad de Onda P-Densidad-Porosidad, lo que indica la relación que tienen éstos con la Velocidad de Onda S.

El Algoritmo Híbrido propuesto en esta investigación se hizo partiendo de un resultado generado por el Sistema de Inferencia Neuro-Difusa. En vista de los resultados obtenidos, se recomienda entonces:

- Realizar el proceso de hibridización antes y no después de generar la Inferencia Difusa, es decir, utilizar los Algoritmos Genéticos en la búsqueda del mejor número de funciones de membresía y las Redes Neuronales para ajustar los parámetros de las funciones de salida del programa difuso. Para lograrlo, se sugiere la utilización de otro programa

diferente a **Matlab**, que sus códigos correspondientes a Lógica Difusa están protegidos por derechos de autor por lo que no pueden ser vistos ni modificados por cualquier usuario.

- Buscar una función para obtener, en las unidades correctas, los valores de Velocidad de Onda S estimados por cada una de las técnicas de predicción.

REFERENCIAS BIBLIOGRÁFICAS

- Almarza, R. (1995). Geología del Campo Guafita. *Revista Petroleum*, 82, 45-47.
- Alzate, G., Branch, J.W., Suárez, O. & Vega, C.A. (2006). Correlación de Registros de Pozo con Redes Neuronales Artificiales utilizando múltiples curvas o variables registradas. *Boletín de Ciencias de la Tierra*, 19.
- Banchs, R., Jiménez, J. & Del Pino, E., (2001). Nonlinear estimation of missing logs from existing well log data. *SEG/San Antonio 2001 Expanded Abstracts*.
- Barberii, E.E. (1989). *La Industria Venezolana de los Hidrocarburos*. CEPET, Caracas.
- Bello, L. (2004). *Estudio de factibilidad del uso de atributos tipo Impedancia y AVO para discriminación litológica y de fluidos en el Campo Guafita*. TRABAJO DE GRADO. Caracas: Universidad Simón Bolívar, Decanato de Estudios Profesionales, Coordinación de Ingeniería Geofísica.
- Bravo, J.E. (2005). *Diseño de un programa para la detección de primeras llegadas en trazas sísmicas y datos VSP utilizando Redes Neuronales*. TRABAJO DE GRADO. Caracas: Universidad Simón Bolívar, Decanato de Estudios Profesionales, Coordinación de Ingeniería Geofísica.
- Canning, A. (n.d.). *Introduction to AVO Theory*. Paradigm, the geoscience knowledge company.
- Castro L., M.A. (2009). *Un Algoritmo Genético distribuido con aplicación en la identificación difusa en un proceso fermentativo*. TESIS DOCTORAL. Universidad Central “Marta Abreu” de Las Villas, Facultad de Ingeniería Eléctrica.
- Cuddy, S.J. and Glover, P.W.J. (2002). *The Application of Fuzzy Logic and Genetic Algorithms to Reservoir Characterization and Modeling*. Department of Geology and Petroleum Geology, University of Aberdeen, UK.

- Donati, M. S. & Regueiro, J. (1988). *Generación de Secciones de Incidencia Normal y/o de Contraste de Poisson, utilizando el fenómeno de cambios de amplitud en función del ángulo de incidencia.* En: IV Congreso Venezolano de Geofísica.
- Dorrington, K.P. & Link, C.A. (2007). Genetic-algorithm/neural-network approach to seismic attribute selection for well-log prediction. *Geophysics*, 69(1), 212-221.
- Forrest, S. (1993). Genetic algorithms: principles of natural selection applied to computation. *Science*, 872-878.
- Freeman, J.A. & Skapura, D.M. (1993). *Redes Neuronales. Algoritmos, aplicaciones y técnicas de propagación.* México: Addison-Wesley.
- García, M. (2005). *Diseño de un programa para discriminar litologías en registros de pozo usando Redes Neuronales.* TRABAJO DE GRADO. Caracas: Universidad Simón Bolívar, Decanato de Estudios Profesionales, Coordinación de Ingeniería Geofísica.
- Gil, C. (2003). *Selección de los mejores atributos sísmicos para la estimación de propiedades de yacimientos: Aplicación de la Teoría de Información.* TRABAJO DE GRADO. Caracas: Universidad Simón Bolívar, Decanato de Estudios Profesionales, Coordinación de Ingeniería Geofísica.
- Gómez, P. (2000). Estudio de los Algoritmos Genéticos para el desarrollo de los sistemas de control. *Ingeniería & Desarrollo*, 130-138
- González F., M.A. (2006). *Utilización de JaTS y Lógica Difusa en el análisis de dos perfiles de refracción sísmica en el área 75 de Bauxilum, Puerto Ordaz, estado Bolívar.* TRABAJO DE GRADO. Caracas: Universidad Simón Bolívar, Decanato de Estudios Profesionales, Coordinación de Ingeniería Geofísica.
- González de Juana, C., Iturrealde, J.M. & Picard, X. (1980). *Geología de Venezuela y de sus Cuencas Petrolíferas.* Caracas: Ediciones Foninves.
- Gutiérrez, V. (2010). *Sistemas Difusos para la predicción de Velocidades de Onda S, Campo Guafita.* TRABAJO DE GRADO. Caracas: Universidad Simón Bolívar, Decanato de Estudios Profesionales, Coordinación de Ingeniería Geofísica.
- Haykin, S. (1993). *Neural Networks: A comprehensive foundation.* New Jersey: Prentice Hall.
- Hilera, J. & Martínez, V. (1995). *Redes neuronales artificiales: fundamentos, modelos y aplicaciones.* Madrid: RA-MA.
- Klir, G.J. & Yuan, B. (1995). *Fuzzy Sets and Fuzzy Logic. Theory and Applications.* New Jersey: Prentice Hall PTR

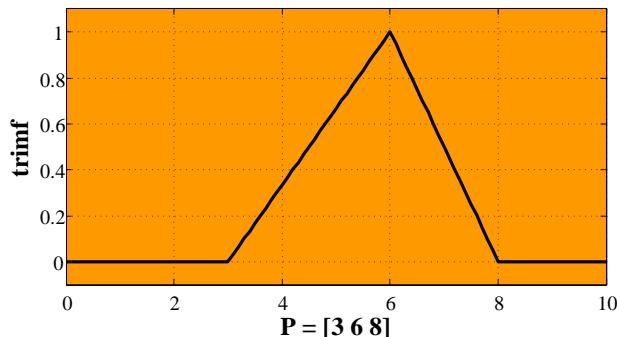
- Kiser, G.D. (1997). *GUAFITA, Formación*. Consultado en Marzo de 2011, página web del 3er Léxico Estratigráfico de Venezuela: <http://www.pdvsa.com/lexico/g24w.htm>
- Koza, J. (1992). *Genetic Programming II : Automatic Discovery of Reusable Programs*. The MIT Press.
- Koza, J., Forest, B., Andre, D. & Keane, M. (1999). *Genetic Programming III: Darwinian Invention and Problem Solving*. Morgan Kaufmann Publishers.
- Lahura, E. (2003). *El Coeficiente de Correlación y Correlaciones Espúreas*. “Disponible en <http://http://www.pucp.edu.pe/economia/pdf/DDD218.pdf>”.
- Limongi, F. (2000). *Caracterización estática de los yacimientos G-73/4 y G-8 del campo Guafita, Bloque Sur. Terciario de la Cuenca Apure-Llanos. Estado Apure*. TRABAJO DE GRADO. Caracas: Universidad Central de Venezuela .
- López, C. (2005). *Determinación de primeras llegadas en datos VSP y Check Shots usando Lógica Difusa*. TRABAJO DE GRADO. Caracas: Universidad Simón Bolívar, Decanato de Estudios Profesionales, Coordinación de Ingeniería Geofísica.
- Jang, R. (1992). *Neuro-Fuzzy Modeling: Architectures, Analyses, and Applications*. Berkeley: Department of Electrical Engineering and Computer Science, University of California at Berkeley.
- Jang, R. (1993). ANFIS: Adaptive-Network-Based Fuzzy Inference Systems. *IEEE Transactions on System, Man, and Cybernetics*, 23(5), 665-685
- MathWorks (2010). *Fuzzy Logic Toolbox 2. User's Guide*. En *Matlab User's Guide*. The MathWorks Inc: 3 Apple Hill Drive Natick, MA 01760-2098
- MathWorks (2010). *Global Optimization Toolbox 3. User's Guide*. En *Matlab User's Guide*. The MathWorks Inc: 3 Apple Hill Drive Natick, MA 01760-2098
- Martínez, A. (1976). *Cronología del Petróleo Venezolano*. Caracas: Foninves, Ediciones Ciencia y Petróleo.
- Montaño M., J.J. (2002). *Redes Neuronales Artificiales aplicadas al Análisis de Datos*. TRABAJO DE GRADO. Palma de Mallorca: Universidad de las Islas Baleares.
- Niño, M.Y. (2006). *Programa que realiza la Estimación de Registros de Pozo por medio de Algoritmos Genéticos*. TRABAJO DE GRADO. Caracas: Universidad Simón Bolívar, Decanato de Estudios Profesionales, Coordinación de Ingeniería Geofísica.

- Ortega, J.F., Van Erve, A., & Monroy, Z. (1987). Formación Guafita: Nueva unidad litoestratigráfica del Terciario en el subsuelo de la Cuenca Barinas-Apure, Venezuela Suroccidental. *Boletín de la Sociedad Venezolana de Geólogos*, 31, 9-35.
- Peña, M., Carelli, R. & Di Sciascio, F. (2002). *Identificación de Sistemas con un modelo borroso del tipo Takagi-Sugeno*. San Juan: Instituto de Automática, Facultad de Ingeniería, Universidad Nacional de San Juan, Argentina.
- Rezaee, M.R., Kadkhodaie, A., & Barabadi, A. (2007). Prediction of shear wave velocity from petrophysical data utilizing intelligent systems: An example from a sandstone reservoir of Carnarvon Basin, Australia. *Journal of Petroleum Science and Engineering*, 201-212.
- Rider, M. (1996). *The Geological Interpretation of Well Logs*. Gulf Publishing Company. Rider French Consulting, P.O. Box 1, Sutherland, IV28 3XL, Scotland
- Saemi, M. & Ahmadi, M. (2008). Integration of genetic algorithm and a coactive neuro-fuzzy inference system for permeability prediction from well logs data. *Tansp Porous Med*, 273-288
- Salas, R. (2004). *Redes Neuronales Artificiales*. Universidad de Valparaíso. Departamento de Computación.
- Silva, M.B.C., Santos, R.V. (2002). Prediction of P-Wave Sonic Logs Via Neural Network and Seismic Trace Inversion: A Comparison. *Offshore Technology Conference*. Houston, Texas U.S.A
- Singh, T.N., Kanchan, R., Saigal, K. & Verma, A.K. (2004). Prediction of p-wave velocity and anisotropic property of rock using artificial neural network technique. *Journal of Scientific & Industrial Research*, 63, 32-38.
- Uzin, I. , Echanobe, J. & Del Campo, I. (1999). *Diseño e implementación de un algoritmo genético sobre dispositivos de lógica programable*.
- YORIS, F. & OSTOS, M. (1997). *Capítulo 1: Geología de Venezuela*. En Well Evaluation Conference (WEC) Venezuela - Evaluación de pozos. Schlumberger.

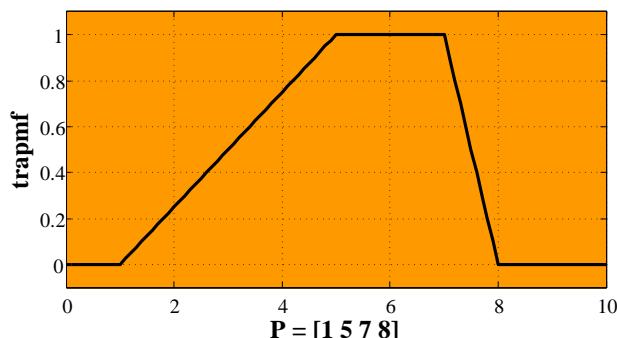
APÉNDICE A

FUNCIONES DE MEMBRESÍA

A continuación se muestran las funciones de membresía para lógica difusa que contiene el programa **Matlab** versión **R2009a**¹:



(a) Función Triangular



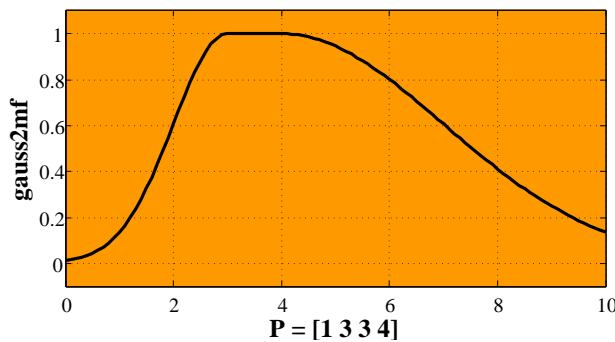
(b) Función Trapezoidal

Figura A.1: Funciones Lineales de Membresía

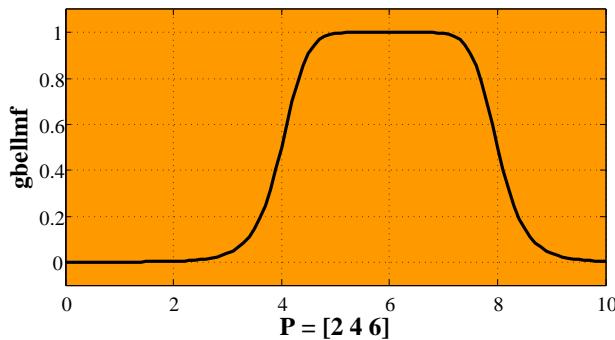
¹Por otra parte, es necesario señalar que toda la información mostrada en este apéndice, es obtenida de (MathWorks, 2010a)

1. Funciones lineales. Se muestran en la gráfica A.1

- **trimf.** Función Triangular. En la figura A.1(a) se muestra un ejemplo de la gráfica esta función. Está definida por tres parámetros $[a, b, c]$, donde a y c son los puntos izquierdo y derecho de la base de la curva y b es la ubicación del punto donde la función alcanza su valor máximo (uno). En esta figura, éstos son: $[a, b, c] = [3, 6, 8]$
- **trapmf.** Función Trapezoidal. Análoga a la anterior, los parámetros a y d son los puntos izquierdo y derecho de la base del trapecio, b y c son los que definen su tope. En la figura A.1(b) se muestra un ejemplo, donde estos puntos tienen los siguientes valores: $[a, b, c, d] = [1, 5, 7, 8]$



(a) Función Gaussiana Combinada



(b) Función Campana Generalizada

Figura A.2: Funciones de Distribución Gaussiana de Membresía

2. Funciones de Distribución Gaussiana.

- **gaussmf.** Función Gaussiana Básica. Tanto la ecuación que la define así como el bosquejo de su gráfica se encuentran en la sección 1.2.1.2

- **gauss2mf.** Combinación de dos Funciones Gaussianas. Está dada por la siguiente ecuación:

$$f_k(x) = \exp\left(\frac{-0.5(x - c_k)^2}{\sigma_k^2}\right) \quad (\text{A.1})$$

donde:

$k = 1, 2$.

c_1 y σ_1 son el valor medio y la varianza que definen la curva de la izquierda.

c_2 y σ_2 son de la curva de la derecha.

En la figura A.2(a) se presenta igualmente un ejemplo, en donde $[\sigma_1, c_1, \sigma_2, c_2] = [1, 3, 3, 4]$

- **gbellmf.** Función Campana Generalizada, está determinada mediante la siguiente ecuación:

$$f(x) = \frac{1}{1 + |\frac{x-c}{a}|^{2b}} \quad (\text{A.2})$$

donde los parámetros:

a y b varían el ancho de la curva.

c es el centro de la curva.

La figura A.2(b) se deja ver un ejemplo de graficar de esta función, en donde $[a, b, c] = [2, 4, 6]$

3. Curvas sigmoidales. Ejemplos de cada una se muestran en la figura A.3

- **sigmf.** Función sigmoidal. Está determinada mediante la siguiente ecuación:

$$f(x) = \frac{1}{1 + \exp(-a(x - c))} \quad (\text{A.3})$$

cuando el signo de a es positivo la curva incrementa de izquierda a derecha. En caso contrario, disminuye en el mismo sentido. Además el parámetro c es el punto de inflexión de la curva.

En la figura A.3(a) se expone un ejemplo en donde $[a, c] = [2, 4]$

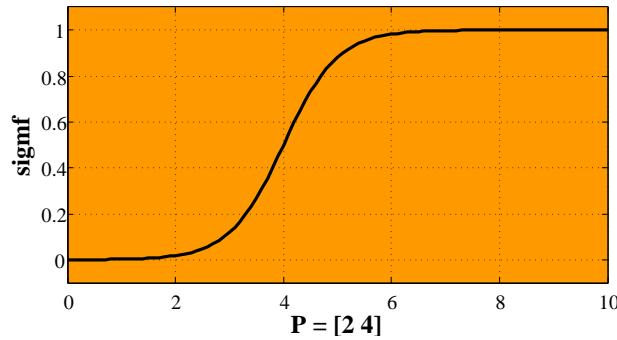
- **psigmf.** Producto de dos funciones sigmoidales. La ecuación característica de esta función es como sigue:

$$f_k(x) = \frac{1}{1 + \exp(-a_k(x - c_k))} \quad (\text{A.4})$$

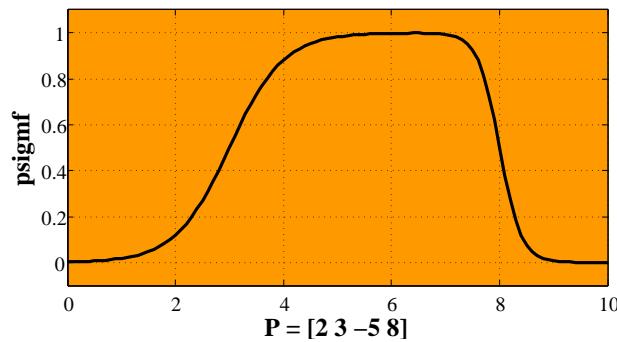
donde:

$k = 1, 2$.

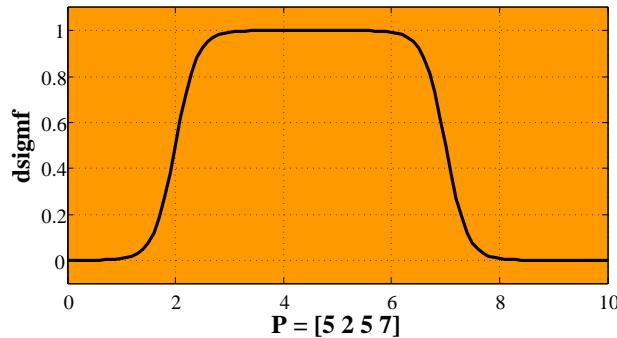
Los parámetros a_1 y a_2 (uno positivo y el otro negativo respectivamente) controlan la pendiente de las curvas de la izquierda y la derecha. Mientras que los parámetros



(a) Función Sigmoidal



(b) Función Producto Sigmoidal



(c) Función Diferencia Sigmoidal

Figura A.3: Curvas Sigmoidales de Membresía

c_1 y c_2 son los puntos de inflexión de ambas curvas.

En la figura A.3(b) se señala un ejemplo en donde $[a_1, c_1, a_2, c_2] = [2, 3, -5, 8]$

- **dsigmf.** Basada en la diferencia de dos funciones sigmoidales, a continuación se presenta la ecuación que describe esta ecuación:

$$f_k(x) = \frac{1}{1 + \exp(-a_k(x - c_k))} \quad (\text{A.5})$$

donde:

$$k = 1, 2.$$

Los parámetros a_1 y a_2 (que deben ser positivos) controlan la pendiente de las curvas de la izquierda y de la derecha. Mientras que los parámetros c_1 y c_2 son los puntos de inflexión de ambas curvas.

En la figura A.3(c) se señala un ejemplo en donde $[a_1, c_1, a_2, c_2] = [5, 2, 5, 7]$.

Vale percatarse que las ecuaciones A.4 y A.5 son iguales, sin embargo, el signo de a_2 es diferente en ambos casos.

4. Curvas polinomiales. Las tres graficas de la figura A.4 exhiben ejemplos de cada una de ellas.

- **smf.** Curva en forma de “S”. De izquierda a derecha la función incrementa de cero a uno. Los parámetros a y b ubican los extremos izquierdo y derecho de la pendiente de la curva. En la figura A.4(a) estos puntos tienen los siguientes valores: $[a, b] = [1, 8]$
- **pimf.** Curva en forma de la letra griega “Pi” (Π). Los parámetros a y d son los puntos de la base izquierda y derecha o “pies” de la curva. Mientras que b y c son los puntos de los topes de izquierdo y derecho. En la figura A.4(b) éstos son: $[a, b, c, d] = [1, 4, 5, 10]$
- **zmf.** Curva en forma de “Z”. De izquierda a derecha la función disminuye de uno a cero. Los parámetros a y b ubican los extremos izquierdo y derecho de la pendiente de la curva. En el ejemplo que se observa en la figura A.4(c) estos tienen las siguientes magnitudes: $[a, b] = [3, 7]$

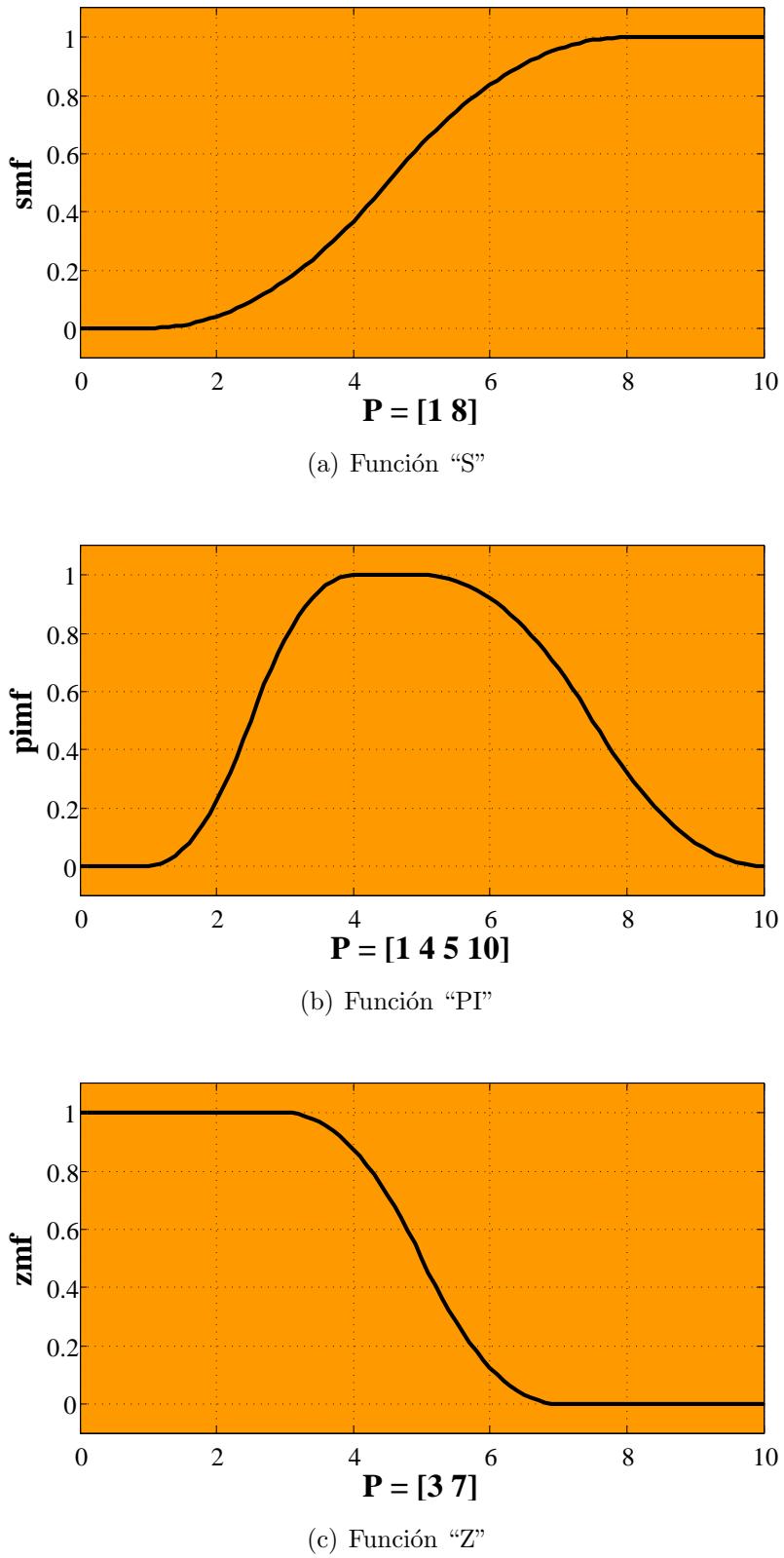


Figura A.4: Curvas Polinomiales de Membresía

APÉNDICE B

TABLAS ADICIONALES

B.1. Pruebas Preeliminares: ANFIS

A continuación, de la tabla B.1 a la B.14, se muestran los valores obtenidos del RMSE en las predicciones obtenidas al variar el número de reglas difusas (4, 6, 8, y 9), como pruebas para obtener la mejor combinación de reglas difusas en uno, dos y tres registros de entrada al Sistema de Inferencia Neuro-Difusa (ANFIS).

1 Registro de Entrada

Tabla B.1: RMSE utilizando un Registro de Entrada en ANFIS con 4, 6, 8 y 9 reglas difusas. Pozo de Entrenamiento

Registros de Entrada	Número de reglas difusas			
	4	6	8	9
DEN	0,285	0,277	0,281	0,266
VP	0,196	0,200	0,284	0,262
SW	0,328	0,317	0,311	0,307
POR	0,258	0,255	0,261	0,258
Vsh	0,262	0,258	0,266	0,260

Tabla B.2: RMSE utilizando un Registro de Entrada en ANFIS con 4, 6, 8 y 9 reglas difusas. Pozo Control

Registros de Entrada	Número de reglas difusas			
	4	6	8	9
DEN	0,410	0,405	0,403	0,419
VP	0,233	0,345	0,926	0,417
SW	0,378	0,378	0,376	0,377
POR	0,428	0,436	0,427	0,439
Vsh	0,440	0,422	0,422	0,459

2 Registros de Entrada

En este caso, tal y como se mostró en la sección 3.4.2, las combinaciones utilizadas fueron las siguientes:

- Para 4 reglas difusas, única combinación [2 2].
- Para 6 reglas difusas, combinaciones [3 2] y [2 3].
- Para 8 reglas difusas, combinaciones [2 4] y [4 2].
- Para 9 reglas difusas, la única combinación posible [3 3].

Tabla B.3: RMSE utilizando dos Registros de Entrada en ANFIS con 4 y 6 reglas difusas en el Pozo de Entrenamiento

Registros de Entrada	Número de reglas difusas		
	4	6	6
	Comb. [2 2]	Comb. [3 2]	Comb. [3 2]
DEN-VP	0,244	0,275	0,252
DEN-SW	0,317	0,311	0,311
DEN-POR	0,414	0,417	0,346
DEN-Vsh	0,506	0,552	0,310
VP-SW	0,245	1,027	0,341
VP-POR	0,199	0,196	0,228
VP-Vsh	0,193	0,214	0,219
SW-POR	0,350	0,255	0,337
SW-Vsh	0,318	0,292	0,278
POR-Vsh	0,219	0,281	0,558

Tabla B.4: RMSE utilizando dos Registros de Entrada en ANFIS con 8 y 9 reglas difusas en el Pozo de Entrenamiento

Registros de Entrada	Número de reglas difusas		
	8	9	
	Comb. [4 2]	Comb. [2 4]	Comb. [3 3]
DEN-VP	0,630	0,239	0,288
DEN-SW	0,315	0,313	0,307
DEN-POR	0,413	0,242	0,765
DEN-Vsh	0,519	0,403	0,250
VP-SW	0,408	0,498	0,297
VP-POR	0,259	0,287	0,227
VP-Vsh	0,245	0,208	0,213
SW-POR	0,305	0,334	0,446
SW-Vsh	0,345	0,248	0,772
POR-Vsh	0,598	0,218	0,322

Tabla B.5: RMSE utilizando dos Registros de Entrada en ANFIS con 4 y 6 reglas difusas en el Pozo Control

Registros de Entrada	Número de reglas difusas		
	4	6	
	Comb. [2 2]	Comb. [3 2]	Comb. [3 2]
DEN-VP	0,347	0,306	0,260
DEN-SW	0,360	0,379	0,371
DEN-POR	0,510	0,711	0,415
DEN-Vsh	0,395	0,375	0,340
VP-SW	0,252	0,517	0,317
VP-POR	0,276	0,284	0,237
VP-Vsh	0,600	0,276	0,260
SW-POR	0,341	0,786	0,569
SW-Vsh	0,368	0,563	0,414
POR-Vsh	0,245	0,476	0,629

Tabla B.6: RMSE utilizando dos Registros de Entrada en ANFIS con 8 y 9 reglas difusas en el Pozo Control

Registros de Entrada	Número de reglas difusas		
	8	9	
	Comb. [4 2]	Comb. [2 4]	Comb. [3 3]
DEN-VP	0,305	0,325	0,340
DEN-SW	0,385	0,370	0,375
DEN-POR	0,510	0,487	0,760
DEN-Vsh	0,417	0,349	0,630
VP-SW	0,303	0,244	0,284
VP-POR	0,338	0,316	0,640
VP-Vsh	0,334	0,320	0,356
SW-POR	0,598	0,654	0,484
SW-Vsh	0,854	0,435	0,456
POR-Vsh	0,412	0,873	1,230

3 Registros de Entrada

En la sección 3.4.2, se especificó que las combinaciones utilizadas para tres registros de entrada fueron las siguientes:

- 4 reglas difusas. Combinaciones: [1 2 2], [2 1 2] y [2 2 1].
- 6 reglas difusas. Combinaciones:[1 3 2], [2 1 3] y [3 2 1].
- 8 reglas difusas. Combinaciones:[1 4 2], [2 1 4] y [4 2 1].
- 9 reglas difusas: Combinaciones:[1 3 3], [3 1 3] y [3 3 1].

Tabla B.7: RMSE utilizando tres Registros de Entrada en ANFIS con 4 reglas difusas en el Pozo de Entrenamiento

Combinaciones de 4 Reglas Difusas			
Registros de Entrada	[1 2 2]	[2 1 2]	[2 2 1]
DEN-VP-SW	0,321	0,243	0,281
DEN-SW-POR	0,303	0,321	0,306
DEN-POR-Vsh	0,393	0,873	0,410
VP-SW-POR	0,222	0,269	0,398
VP-POR-Vsh	0,597	0,342	0,295
VP-DEN-Vsh	0,254	0,234	0,223
VP-DEN-POR	0,256	0,205	0,215
VP-SW-Vsh	0,199	0,243	0,250

Tabla B.8: RMSE utilizando tres Registros de Entrada en ANFIS con 6 reglas difusas en el Pozo de Entrenamiento

Combinaciones de 6 Reglas Difusas			
Registros de Entrada	[1 3 2]	[2 1 3]	[3 2 1]
DEN-VP-SW	0,366	0,242	0,331
DEN-SW-POR	0,318	0,315	0,530
DEN-POR-Vsh	0,384	0,292	0,445
VP-SW-POR	0,233	0,230	0,249
VP-POR-Vsh	0,368	0,211	0,389
VP-DEN-Vsh	0,375	0,227	0,265
VP-DEN-POR	0,569	0,268	0,387
VP-SW-Vsh	0,194	0,262	0,377

Tabla B.9: RMSE utilizando tres Registros de Entrada en ANFIS con 8 reglas difusas en el Pozo de Entrenamiento

Combinaciones de 8 Reglas Difusas			
Registros de Entrada	[1 4 2]	[2 1 4]	[4 2 1]
DEN-VP-SW	0,271	0,289	0,334
DEN-SW-POR	0,477	0,459	0,258
DEN-POR-Vsh	0,380	0,360	0,933
VP-SW-POR	0,257	0,363	0,690
VP-POR-Vsh	0,344	0,264	0,316
VP-DEN-Vsh	0,988	0,283	0,324
VP-DEN-POR	0,378	0,254	0,271
VP-SW-Vsh	0,229	0,222	0,292

Tabla B.10: RMSE utilizando tres Registros de Entrada en ANFIS con 9 reglas difusas en el Pozo de Entrenamiento

Combinaciones de 9 Reglas Difusas			
Registros de Entrada	[1 3 3]	[3 1 3]	[3 3 1]
DEN-VP-SW	0,373	0,290	0,584
DEN-SW-POR	0,312	0,405	0,296
DEN-POR-Vsh	0,618	1,995	0,356
VP-SW-POR	0,282	0,862	0,277
VP-POR-Vsh	0,480	0,240	0,336
VP-DEN-Vsh	1,188	0,349	0,447
VP-DEN-POR	0,643	0,253	0,643
VP-SW-Vsh	2,196	0,339	0,564

Tabla B.11: RMSE utilizando tres Registros de Entrada en ANFIS con 4 reglas difusas en el Pozo Control

Combinaciones de 4 Reglas Difusas			
Registros de Entrada	[1 2 2]	[2 1 2]	[2 2 1]
DEN-VP-SW	0,277	0,351	0,341
DEN-SW-POR	0,355	0,396	0,366
DEN-POR-Vsh	0,858	0,594	0,602
VP-SW-POR	0,420	0,251	0,374
VP-POR-Vsh	0,541	0,380	0,412
VP-DEN-Vsh	0,327	0,398	0,473
VP-DEN-POR	0,495	0,336	0,236
VP-SW-Vsh	0,277	0,262	0,400

Tabla B.12: RMSE utilizando tres Registros de Entrada en ANFIS con 6 reglas difusas en el Pozo Control

Combinaciones de 6 Reglas Difusas			
Registros de Entrada	[1 3 2]	[2 1 3]	[3 2 1]
DEN-VP-SW	0,260	0,282	0,392
DEN-SW-POR	0,374	0,364	1,953
DEN-POR-Vsh	0,985	0,386	0,722
VP-SW-POR	0,477	0,248	0,442
VP-POR-Vsh	0,870	0,394	0,532
VP-DEN-Vsh	0,489	0,332	0,350
VP-DEN-POR	0,523	0,388	0,548
VP-SW-Vsh	0,346	0,191	0,261

Tabla B.13: RMSE utilizando tres Registros de Entrada en ANFIS con 8 reglas difusas en el Pozo Control

Combinaciones de 8 Reglas Difusas			
Registros de Entrada	[1 4 2]	[2 1 4]	[4 2 1]
DEN-VP-SW	0,157	0,379	0,330
DEN-SW-POR	0,847	0,673	0,532
DEN-POR-Vsh	1,368	0,741	1,652
VP-SW-POR	0,562	0,276	0,371
VP-POR-Vsh	0,709	0,427	0,528
VP-DEN-Vsh	0,450	0,304	0,430
VP-DEN-POR	0,692	0,473	0,299
VP-SW-Vsh	0,316	0,286	0,320

Tabla B.14: RMSE utilizando tres Registros de Entrada en ANFIS con 9 reglas difusas en el Pozo Control

Combinaciones de 9 Reglas Difusas			
Registros de Entrada	[1 3 3]	[3 1 3]	[3 3 1]
DEN-VP-SW	0,335	0,393	0,320
DEN-SW-POR	0,435	0,362	0,383
DEN-POR-Vsh	1,670	1,093	0,867
VP-SW-POR	0,388	0,514	0,386
VP-POR-Vsh	2,281	0,487	0,641
VP-DEN-Vsh	0,467	0,329	0,407
VP-DEN-POR	0,810	0,295	1,017
VP-SW-Vsh	1,101	0,463	0,357

B.2. Estimaciones realizadas utilizando tres Registros de Entrada

B.2.1. Algoritmos Genéticos

A continuación se muestran los resultados obtenidos en las predicciones realizadas con tres registros de Entrada

Tabla B.15: Parámetros “ a_1 ”, “ a_2 ” y “ a_3 ” de la Ecuación 1 AG utilizando tres Registros de Entrada

Ecuación 1 AG			
Registros de Entrada	Parámetros		
	“ a_1 ”	“ a_2 ”	“ a_3 ”
DEN-VP-SW	0,380	0,355	0,364
DEN-SW-POR	0,391	0,312	0,405
DEN-POR-Vsh	0,409	0,306	0,346
VP-SW-POR	0,355	0,366	0,362
VP-POR-Vsh	0,413	0,319	0,359
VP-DEN-Vsh	0,331	0,449	0,392
VP-DEN-POR	0,349	0,363	0,360
VP-SW-Vsh	0,422	0,298	0,435

Tabla B.16: Parámetros “ $a_1 - b_1$ ”, “ $a_2 - b_2$ ” y “ $a_3 - b_3$ ” de la Ecuación 2 AG empleando tres Registros de Entrada

Ecuación 2 AG						
Registros de Entrada	“ a_1 ”	“ b_1 ”	“ a_2 ”	“ b_2 ”	“ a_3 ”	“ b_3 ”
DEN-VP-SW	0,359	0,478	0,406	0,482	0,329	0,595
DEN-SW-POR	0,342	0,522	0,349	0,554	0,456	0,554
DEN-POR-Vsh	0,325	0,523	0,401	0,424	0,378	0,379
VP-SW-POR	0,369	0,522	0,460	0,476	0,335	0,506
VP-POR-Vsh	0,402	0,453	0,392	0,504	0,326	0,489
VP-DEN-Vsh	0,339	0,464	0,410	0,528	0,369	0,430
VP-DEN-POR	0,396	0,544	0,336	0,501	0,375	0,514
VP-SW-Vsh	0,384	0,481	0,393	0,439	0,338	0,486

Tabla B.17: RMSE para tres Registros de Entrada con Algoritmos Genéticos.

Reg. Entrada	RMSE			
	Pozo de Entrenamiento		Pozo Control	
	Ec. 1 AG	Ec. 2 AG	Ec. 1 AG	Ec. 2 AG
DEN-VP-SW	0,414	0,234	0,539	0,443
DEN-SW-POR	0,346	0,270	0,621	0,569
DEN-POR-Vsh	0,266	0,242	0,486	0,471
VP-SW-POR	0,330	0,260	0,528	0,516
VP-POR-Vsh	0,209	0,227	0,379	0,411
VP-DEN-Vsh	0,397	0,244	0,513	0,417
VP-DEN-POR	0,194	0,202	0,330	0,369
VP-SW-Vsh	0,384	0,239	0,598	0,474

Tabla B.18: R^2 para tres Registros de Entrada con Algoritmos Genéticos.

Reg. Entrada	R^2			
	Pozo de Entrenamiento		Pozo Control	
	Ec. 1 AG	Ec. 2 AG	Ec. 1 AG	Ec. 2 AG
DEN-VP-SW	0,2541	0,2646	0,0723	0,1543
DEN-SW-POR	0,0813	0,0556	0,0378	0,0538
DEN-POR-Vsh	0,0431	0,0630	0,0009	0,0002
VP-SW-POR	0,1310	0,1467	0,0132	0,0546
VP-POR-Vsh	0,2924	0,2780	0,3673	0,5052
VP-DEN-Vsh	0,1286	0,1362	0,1558	0,2539
VP-DEN-POR	0,4138	0,3679	0,5655	0,6406
VP-SW-Vsh	0,1298	0,1438	0,0251	0,0549

B.2.2. Parte I: Grid Partition

Tabla B.19: RMSE para tres Registros de Entrada utilizando ANFIS-Híbrido (6 reglas difusas) en el Pozo de Entrenamiento.

Reg. Entrada	RMSE			
	Pozo de Entrenamiento		Pozo Control	
	ANFIS	Híbrido	ANFIS	Híbrido
DEN-VP-SW	0,263	0,324	0,390	0,496
DEN-SW-POR	0,371	2,760	0,409	1,758
DEN-POR-Vsh	0,281	0,346	0,546	0,673
VP-SW-POR	0,411	0,947	0,486	0,607
VP-POR-Vsh	0,250	0,381	0,390	0,724
VP-DEN-Vsh	0,199	0,693	0,266	1,081
VP-DEN-POR	0,237	0,540	0,326	0,343
VP-SW-Vsh	0,224	0,556	0,269	0,801

Tabla B.20: R^2 para tres Registros de Entrada utilizando ANFIS-Híbrido (6 reglas difusas).

Reg. Entrada	R^2			
	Pozo de Entrenamiento		Pozo Control	
	ANFIS	Híbrido	ANFIS	Híbrido
DEN-VP-SW	0,1770	0,2149	0,0387	0,0783
DEN-SW-POR	0,0059	0,0031	0,0382	0,0088
DEN-POR-Vsh	0,0420	0,0364	0,0321	0,0471
VP-SW-POR	0,0202	0,1066	0,0011	0,1230
VP-POR-Vsh	0,3485	0,1083	0,0470	0,0024
VP-DEN-Vsh	0,4307	0,0342	0,6790	0,0105
VP-DEN-POR	0,2235	0,1280	0,5624	0,2934
VP-SW-Vsh	0,3485	0,0178	0,5632	0,0011

Tabla B.21: Combinación de Reglas Difusas generadas a través de la técnica “Grid Partition” para tres Registros de Entrada: VP-DEN-POR

Número de fm			Número de FS
Entrada			Salida: VS
VP	DEN	POR	
1	1	1	1
1	1	2	2
1	1	3	3
2	1	1	4
2	1	2	5
2	1	3	6

Tabla B.22: Parámetros σ y c de la función de membresía *Gaussiana* generados en el SID mediante la técnica “Grid Partition”. Registros de Entrada: VP-DEN-POR

Número de fm	Registros de Entrada					
	VP		DEN		POR	
	sigma	c	sigma	c	sigma	c
Nº 1	0,1967	0,3927	Inf	1,9319	0,1317	0,3014
Nº 2	0,1864	1,8034			0,5753	1,0822
Nº 3					0,4369	1,585

Tabla B.23: Parámetros “ a ”, “ b ”, “ c ” y “ d ” de las funciones de salida, utilizando “Grid Partition”. Registros de Entrada: VP-DEN-POR

Función de Salida	Parámetros			
	a	b	c	d
Nº 1	1,1343	-0,3003	-0,4504	0,3149
Nº 2	0,1759	0,01818	-1,03572	1,3890
Nº 3	0,2639	0,5720	-0,4321	1,7817
Nº 4	0,3550	0,6878	2,3507	-0,6309
Nº 5	1,4055	-0,6701	4,2221	-1,9505
Nº 6	-1,0588	1,1206	3,4175	-7,3704

Tabla B.24: Parámetros “ a ”, “ b ”, “ c ” y “ d ” optimizados de las funciones de salida, utilizando “Grid Partition”. Registros de Entrada: VP-DEN-POR

Función de Salida	Parámetros			
	a	b	c	d
Nº 1	0,2634	0,6627	0,2401	0,5131
Nº 2	0,2140	0,1475	0,4387	0,2556
Nº 3	0,2380	0,2437	0,13412	0,1554
Nº 4	0,3468	0,7829	0,6970	0,9023
Nº 5	0,6841	0,21000	0,7926	0,8488
Nº 6	-1,0588	1,1206	3,4175	-7,3704

B.2.3. Parte II: Subtractive Clustering

Tabla B.25: RMSE para tres Registros de Entrada empleando ANFIS-Híbrido (reglas difusas variables).

Reg. Entrada	RMSE			
	Pozo de Entrenamiento		Pozo Control	
	ANFIS	Híbrido	ANFIS	Híbrido
DEN-VP-SW	0,290	1,003	0,330	1,043
DEN-SW-POR	0,349	0,517	0,422	0,899
DEN-POR-Vsh	0,375	0,486	0,578	1,002
VP-SW-POR	0,262	0,494	0,719	0,56
VP-POR-Vsh	0,292	0,840	0,452	1,090
VP-DEN-Vsh	0,251	0,872	0,367	0,776
VP-DEN-POR	0,196	0,378	0,391	0,470
VP-SW-Vsh	0,427	0,888	0,401	0,878

Tabla B.26: R^2 para tres Registros de Entrada utilizando ANFIS-Híbrido (reglas difusas variables) en el Pozo de Entrenamiento.

Reg. Entrada	R^2			
	Pozo de Entrenamiento		Pozo Control	
	ANFIS	Híbrido	ANFIS	Híbrido
DEN-VP-SW	0,2795	0,1052	0,2769	0,0006
DEN-SW-POR	0,0166	0,0733	0,0418	0,0001
DEN-POR-Vsh	0,0187	0,0254	0,0160	0,0096
VP-SW-POR	0,1380	0,0575	0,2202	0,1666
VP-POR-Vsh	0,3218	0,0233	0,0320	0,0015
VP-DEN-Vsh	0,2975	0,2088	0,2104	0,2507
VP-DEN-POR	0,4048	0,3795	0,2486	0,4314
VP-SW-Vsh	0,1992	0,2376	0,1263	0,1459

Tabla B.27: Combinación de Reglas Difusas generadas a través de la técnica “Substractive Clustering” para tres Registros de Entrada: VP-DEN-POR

Número de fm			Número de FS
Entrada			Salida: VS
VP	DEN	POR	
1	1	1	1
2	2	2	2
3	3	3	3
4	4	4	4

Tabla B.28: Parámetros σ y c de las funciones de membresía *Gaussianas* generadas en el SID mediante la técnica “Substractive Clustering”. Registros de Entrada: VP-DEN-POR.

Número de fm	Registros de Entrada					
	VP		DEN		POR	
	sigma	c	sigma	c	sigma	c
Nº 1	0,1656	0,9102	0,2050	0,7216	0,2242	1,6155
Nº 2	0,0582	1,1955	0,1793	1,2487	0,2535	0,4417
Nº 3	0,2081	1,0290	0,2266	0,9168	0,2691	1,1428
Nº 4	0,2082	1,1983	0,2243	1,1837	0,2506	0,9069

Tabla B.29: Parámetros “ a ”, “ b ”, “ c ” y “ d ” de las funciones de salida, utilizando “Substractive Clustering”. Registros de Entrada: VP-DEN-POR

Función de Salida	Parámetros			
	a	b	c	d
Nº 1	0,0572	0,0170	-0,3407	1,5001
Nº 2	-1,5987	-0,0150	-0,6323	3,1687
Nº 3	-0,0966	0,4017	-0,0396	0,7637
Nº 4	0,6200	0,0048	0,1998	0,2276

Tabla B.30: Parámetros “ a ”, “ b ”, “ c ” y “ d ” optimizados de las funciones de salida, utilizando “Substractive Clustering”. Registro de Entrada: VP-DEN-POR

Función de Salida	Parámetros			
	a	b	c	d
Nº 1	0,2160	0,1351	0,2906	0,1877
Nº 2	0,5076	0,1804	-0,2305	0,4639
Nº 3	0,3530	0,4391	0,1839	0,1528
Nº 4	0,4461	0,3316	0,2543	0,4725

APÉNDICE C

CÓDIGOS

A continuación se presentan los códigos realizados en *Matlab* para la generación de todos los resultados anteriormente expuestos ¹.

C.1. Función “ag_lineal”

```
function coef=ag_lineal(PE_AEN,seleccion,opciones)
% Función que determina los coeficientes, mediante ALGORITMOS GENÉTICOS, de
% la ecuación lineal:
% f(x)=x(1)*y(:,1) + x(2)*y(:,2) + x(n)*y(:,n)
%
% Donde: x(1), x(2) ... x(n)      coeficientes a optimizar
%          y(:,1), y(:,2) ... y(:,n)  registros de pozo para la predicción
%
% coef=ag_lineal(PRED,PE_AEN,obj,seleccion,opciones) Calcula los
% coeficientes "coef" que mejor relacionan a los elementos del(los)
% registro(s) de entrada con el de salida.
%
% Donde:
%     PE_AEN: Registros normalizados en el area de entrenamiento.
%     seleccion: vector que indica las posiciones (escogidas aleatoriamente)
%                 del 50% de los datos seleccionados para el entrenamiento
%     opciones: estructura que contiene las opciones a utilizar en el
%               entrenamiento del algoritmo genético
%
```

¹Cada función es un archivo de extensión “.m” donde, tanto los comentarios de todas las funciones, así como los caracteres del Programa, van sin acentos

164

global obj pred num

% Matriz ENT: contiene los datos para el entrenamiento del algoritmo.
ENT=PE_AEN(seleccion,pred);

% FIN: Vector que contiene el registro a calcular mediante el algoritmo,
% en los mismos puntos que la matriz ENT.
FIN=PE_AEN(seleccion,obj);

% coef: coeficientes que brindan el mejor ajuste de la función de
% entrenamiento
coef=zeros(size(ENT))

% Cálculo del coeficiente (minim) para cada elemento.
h=waitbar(0,'Ajustando mediante Algoritmos Genéticos...');
for i=1:(size(ENT,1))
 minim=ga(@(x) fun_lineal_entrenamiento(x,i,ENT,FIN),num,...
 [],[],[],[],[],[],[],opciones);
 coef(i,:)=minim;
 waitbar(i/(size(ENT,1)),h)
end
close(h)

function Z=fun_lineal_entrenamiento(x,i,y,w)
% Función de Entrenamiento del algoritmo genético de la forma (relacionada
% con el Error Medio Cuadrático):
% f=[(x(1)*y(:,1) + x(2)*y(:,2) + x(n)*y(:,n))-w]^2
%
% Parámetros de entrada:
% i: Corresponde al numero de elemento dentro del ciclo
% y: Matriz en donde cada columna contiene los datos seleccionados para la
% predicción.
% w: Corresponde al registro medido cuya predicción se desea realizar
% mediante algoritmos genéticos

% x: Coeficientes a optimizar
z=zeros(size(y,2),1);
for j=1:(size(y,2))

```

z(j)=x(j).*y(i,j);
end

if (size(y,2))==1
    zz=z;
else
    zz=sum(z);
end
Z=(zz-w(i)).^2;

```

C.2. Función “ag_polinomial”

```

function coef=ag_polinomial(PE_AEN,seleccion,opciones)
% Función que determina los parámetros, mediante ALGORITMOS GENÉTICOS, de
% la función polinomica:
% f(x)=x(1)*y(:,1)^x(2) + x(3)*y(:,2)^x(4) + ... + x(n+1)*y(:,n)^x(n+2)
%
% Donde: x(1), x(2) ... x(n)           parámetros a optimizar. Los numeros
%                               impares corresponden a los coeficientes
%                               y los pares a los exponentes.
%           y(:,1), y(:,2) ... y(:,n)   registros de pozo para la predicción
%
% coef=ag_polinomica(PRED,PE_AEN,obj,seleccion,opciones) Calcula los
% coeficientes "coef" que mejor relacionan a los elementos del(los)
% registro(s) de entrada con el de salida.
% Donde:
%     PE_AEN: Registros normalizados en el area de entrenamiento.
%     seleccion: vector que indica las posiciones (escogidas aleatoriamente)
%                 del 50% de los datos seleccionados para el entrenamiento
%     opciones: estructura que contiene las opciones a utilizar en el
%               entrenamiento del algoritmo genético

global obj pred num

% Matriz ENT: contiene los datos de entrada para el entrenamiento
%               del algoritmo.
ENT=PE_AEN(seleccion,pred);

```

```

% FIN: Vector que contiene el registro a calcular mediante el algoritmo,
%       en los mismos puntos que la matriz ENT.
FIN=PE_AEN(seleccion,obj);

% coef: coeficientes que brindan el mejor ajuste de la función de
%       entrenamiento
coef=zeros(size(ENT,1),2*size(ENT,2));

% Cálculo del coeficiente (minim) para cada elemento.
h=waitbar(0,'Ajustando mediante Algoritmos Genéticos...');
for i=1:(size(ENT,1))
    minim=ga(@(x) fun_polinomial_entrenamiento(x,i,ENT,FIN),...
        2*num,[],[],[],[],[],[],[],opciones);
    coef(i,:)=minim;
    waitbar(i/(size(ENT,1)),h)
end
close(h)

function Z=fun_polinomial_entrenamiento(x,i,y,w)
% Función de Entrenamiento (a optimizar) de la forma (relacionada
% con el Error Medio Cuadrático):
% f=[(x(1)*y(:,1)^x(2) + x(3)*y(:,2)^x(4) + ... + x(n+1)*y(:,n)^x(n+2))-w]^2
%
% Parámetros de entrada:
% i: Corresponde al número de elemento dentro del ciclo
% y: Matriz en donde cada columna contiene los datos seleccionados para la
%     predicción.
% w: Corresponde al registro medido cuya predicción se desea realizar
%     mediante algoritmos genéticos

% x: Coeficientes a optimizar
z=zeros(size(y,1),1);
k=1;
for j=1:size(y,2)
    z(j)=x(k).*(y(i,j)^x(k+1));
    k=k+2;
end

```

```
zz=sum(z);
Z=(zz-w(i)).^2;
```

C.3. Función “carga_registro”

```
function REG=carga_registro(archivo)
% Función que lee un registro de pozo.
% X = lectura_registro(ARCHIVO) arroja una matriz X cuyas columnas se
% corresponden con los registros de pozo provenientes de ARCHIVO (cuyo
% nombre debe escribirse entre comillas '' y con su propia extensión).
%
% Modificada de la función "lectlasfilex.m".
% Obtenida de la Tesis (Landa, 2004). (Ver Referencias)

assy=fopen(archivo);
switch assy %
    case -1 %
        warndlg('El archivo no se encuentra en el Directorio adecuado',...
            'Error!');
    case 2 %
        warndlg('Error Estandar al abrir el archivo','Error!');
    otherwise % El archivo puede abrirse
        A=0; %Indicara el momento en que se consiga la línea adecuada
        while A==0
            % Se lee una línea del archivo
            linlec=fgetl(assy);
            % Si es un archivo LAS, estos comienzan bien con un comentario
            % (#) o con un título (~)
            if strncmp(linlec,'~',1) || strncmp(linlec,'#',1)
                % Solo se toman en cuenta las líneas que no corresponden
                % a un comentario
                if ~strcmp(linlec(1),'#')
                    if strncmp(linlec,'^A',2)
                        reg=fscanf(assy,'%f');
                        while isempty(reg);
                            fgets(assy);
                            reg=fscanf(assy,'%f');
                    end
                end
            end
        end
    end
end
```

```
    end
    A=1;
    % Detección del número de registros
    for i=2:size(reg);
        R=reg(1)-reg(i);
        if R<=2;
            break
        end
    end
    % Orden final de los registros
    REG=reshape(reg,i-1,[])';
end
else
    %El archivo no es de formato LAS
    reg=fscanf(assy,'%f');
    while isempty(reg);
        fgets(assy);
        reg=fscanf(assy,'%f');
    end
    A=1;
    % Detección del número de registros
    for i=2:size(reg);
        R=reg(1)-reg(i);
        if R<=2;
            break
        else
            continue
        end
    end
    % Orden final de los registros
    REG=reshape(reg,i-1,[])';
end
end
end
```

C.4. Función “eliminar”

```

function r=indice(X,p)
% r = indice(X,p) busca el número de fila (posición) en que se encuentra
% el número "p" dentro del vector X (siendo un vector fila o columna)

r=0;
for i=1:size(X)
    if X(i)==p
        r=i;
    end
end

if r==0
    disp('Este elemento no se encuentra en el vector')
end

```

C.5. Función “eval_ag_lineal”

```

function Z=eval_ag_lineal(X,c,P)
% Z=eval_ag_lineal(X,c,num) Realiza el cálculo del registro de salida "Z",
% utilizando coeficientes de una ecuación lineal [c(1),c(2),...,c(n)] de
% la forma:
%
% Z=c(1)*X(:,1) + c(2)*X(:,2) + ... + c(n)*X(:,n)
%
% Donde "X" son los registros de entrada del algoritmo.

global pred num pred_control

if P==1
    pos=pred;
elseif P==2
    pos=pred_control;
end

```

170

```
% W: Vector que contiene el registro final predicho  
%       del Pozo de entrenamiento.  
W=X(:,pos);  
Z=zeros(size(W,1),num);  
for i=1:num  
    Z(:,i)=c(i).*(W(:,i));  
end  
Z=sum(Z,2);
```

C.6. Función “eval_ag_polinomio”

```
function Z=eval_ag_polinomio(X,c,P)  
% Z=evalag_polinomio(X,c,num) Realiza el cálculo del registro de salida "Z",  
% utilizando los parámetros de la función de la forma:  
%  
% Z=c(1)*X(:,1)^c(2) + c(3)*X(:,2)^c(4) + ... + c(n-1)*X(:,n)^c(n+1)  
%  
% Donde "X" son los registros de entrada del algoritmo.
```

```
global pred num pred_control
```

```
if P==1  
    pos=pred;  
elseif P==2  
    pos=pred_control;  
end
```

```
% W: Vector que contiene el registro final predicho del Pozo de  
%       Entrenamiento.  
W=X(:,pos);  
Z=zeros(size(W,1),num);  
j=1;  
for i=1:num  
    Z(:,i)=c(j).*((W(:,i)).^c(j+1));  
    j=j+2;  
end  
Z=sum(Z,2);
```

C.7. Función “graficas_registro”

```

function graficas_registro(X,N,UD,T,tsup,tinf)
% graficas_registro(X,N,obj) Realiza graficas de registros de pozo. Muestra
% tres gráficos por ventana hasta la totalidad de los registros que
% contenga el pozo. Donde:
%
% X: Matriz que contiene los registros de pozo a graficar
% N: Vector de celdas que contiene los nombres de los registros.
%
% graficas_registro(X,N,obj,UD,T,tsup,tinf) Adicionalmente, agrega la
% siguiente información a los registros:
%
% UD: Unidades en que fueron medidos los registros.
%
% T: Titulo del registro que se esta graficando.
%
% tsup: Tope superior de la arena de interes
%
% tinf: Tope inferior de la arena de interes

% Condiciones

errormsg = nargchk(4,6,nargin);
if isempty(errormsg)
    error('No hay argumentos suficientes para ejecutar la función');
end

%
% col: Número de columnas presentes en la matriz X
col=size(X,2);

%
if ~iscell(N)
    error('Los nombres deben ser introducidos como un vector de celdas')
end

%
if ~ischar(T)
    error('El título debe ser un caracter')
end

%
if size(N,2) ~= col
    error('El numero de nombres y de registros deben ser iguales')
end

```

```

if nargin > 4 && nargin > 5
    if ~isnumeric(tsup) || ~isnumeric(tinf)
        error('Los topes deben ser números')
    end

    if tsup ~= 0
        if ~indice(X(:,1),round(tsup)) || ~indice(X(:,1),round(tinf));
            msg='El(los) Tope(s) especificados(s) no se encuentra(n) en este pozo';
            error(msg)
        end
    end
end

% topes: matriz que contiene los topes de la arena de interes
topes=[round(tsup) round(tinf);round(tsup) round(tinf)];
end

% Gráfica de los registros. Cada ventana generada tendra un maximo de tres
% registros del pozo.
i=2;
while i <= col
    % La posición de la ventana dependera del monitor en que se trabaje.
    % Para este caso, el vector [1 27 1024 639] se corresponde con una
    % ventana maximizada

    figure('Name',T,'NumberTitle','off','Position',[1 27 1024 669],...
        'Color','w','Visible','off','Toolbar','none');
    for j=1:3
        try
            subplot(1,3,j), plot(X(X(:,i)>=0,i),X(X(:,i)>=0,1))

            % Propiedades de los ejes
            ylim=get(gca,'YLim'); M=ylim(2); m=ylim(1); r=M-m;
            ejey=[m , m+r/4 , m+r/2 , M-r/4, M];
            set(gca,'FontWeight','bold','XGrid','on','YGrid','on',...
                'GridLineStyle',':','YTick',ejey,'YDir','reverse',...
                'XAxisLocation','top','FontSize',13)

            if nargin > 4 && nargin > 5

```

```

% linea: Fila 1 corresponde con el punto minimo del registro a
%         graficar. Fila 2 el punto maximo. En conjunto con la
%         matriz "topes", permitira graficar una linea
%         indicando el(los) tope(s) de la arena de interes.
linea(1,[1 2])=min(get(gca,'XLim'));
linea(2,[1 2])=max(get(gca,'XLim'));

% Linea que representa los topes de la arena de interes.
line(linea([1 2],:),topes,'color','g','LineWidth',2.5)

end

% Titulo del Registro
[legend_h] = legend(N(i));
set(legend_h,'Location','NorthOutside','Box','off')
xlabel(UD(i))

catch zerror
    disp(zerror)
    break
end
i=i+1;
end
set(gcf,'Visible','on')
end

```

C.8. Función “hibrido”

```

function [PE,PC,n_r,p_dif,p_tb,zerror]=hibrido(FIS,ENT,pe,pc,op)
% [PE,PC,n_r,p_dif,p_tb,zerror]=hibrido(FIS,ENT,pe,pc,op)
% Realiza una combinación entre Lógica Difusa y Algoritmos Genéticos para
% la predicción de Registros de Pozo.
% Donde:
%     FIS: Estructura generada en ANFIS que contiene todos los parámetros
%           asociados a el.
%     ENT: Contiene los datos para el entrenamiento. Su ultima columna debe
%           corresponderse con los valores de salida.
%           (Registro que se desea predecir).

```

```

% pe: Registros Normalizados de entrada del Pozo de Entrenamiento.
% pc: Registros Normalizados de entrada del Pozo Control.
% op: Opciones para la función "ga.m" de Algoritmos Genéticos
%
% PE: Registro de Salida. Pozo de Entrenamiento
% PC: Registro de Salida. Pozo Control
% n_r: Vector que indica con ceros las reglas difusas que no
%       fueron optimizadas.
% p_dif: Parámetros del FIS creado por ANFIS.
% p_hb: Parámetros Optimizados del FIS mediante Algoritmos Genéticos.
%       En las reglas difusas que no fueron optimizadas, tendra sus
%       correspondientes parámetros de ANFIS.

global num

% mf_max= Número de funciones de membresía maximo que alcanza entrada
%         alguna dentro del FIS creado
mf_max=zeros(1,num);
for i=1:num
    mf_max(i)=size(FIS.input(i).mf,2);
end
mf_max=max(mf_max);

% Ha de tomarse en cuenta que, en dicha combinación, no necesariamente el
% número de reglas difusas para la(s) variable(s) de entrada son iguales.
sigma=zeros(mf_max,num); c=zeros(mf_max,num);
rango_min=zeros(mf_max,num); rango_max=zeros(mf_max,num);
for i=1:num
    for j=1:mf_max
        try
            % Parámetros de todas las funciones de membresía. Denotadas con
            % "sigma" y "c".
            sigma(j,i)=FIS.input(i).mf(j).params(1);
            c(j,i)=FIS.input(i).mf(j).params(2);

            % Valores topes de entrada (a media altura), de cada función de
            % membresía. Cada fila va a contener [minimo y maximo]
            rango_min(j,i)=c(j,i)-sqrt(2*log(1/0.5)*sigma(j,i)^2);
        end
    end
end

```

```

        rango_max(j,i)=c(j,i)+sqrt(2*log(1/0.5)*sigma(j,i)^2);
    catch error
        continue
    end
end

% H_ENT: Matriz que contiene los registros seleccionados para la
%         entrenamiento de la "Hibridización".
%         Su ultima columna se corresponde con el registro de salida

regla=zeros(size(FIS.output.mf,2),num);
n_r=zeros(1,size(FIS.output.mf,2));
p_dif=zeros(size(FIS.output.mf,2),num+1);
p_hb=zeros(size(FIS.output.mf,2),num+1);
for k=1:size(FIS.output.mf,2)
% Hibridización para cada Regla Difusa tipo Takagi - Sugeno - Khan
% k=1 Regla Difusa Número 1
% k=2 Regla Difusa Número 2
% k=3 Regla Difusa Número 3 ...
% Selección, desde la matriz de Entrenamiento del SID "ENT", sólo los
% datos que se encuentren dentro de cada rango calculado, siguiendo las
% correspondientes reglas difusas.
% El prefijo "H_" antes de cada nombre indica que se ha obtenido
% en el proceso de Hibridización

% Obtención de la regla difusa.
regla(k,:)=FIS.rule(k).antecedent;

% Bloque de sentencias. Devuelve un vector de unos sólo en el caso en
% que los elementos en una profundidad determinada pertenezcan al rango
% correspondiente de su función de membresía.

H_ENT=zeros(size(ENT)); sentencia=zeros(1,num);
for i=1:size(ENT,1)
    for h=1:num
        % Comparación de cada elemento con el rango calculado.

```

```

% sentencia(1)= Verifica la entrada 1 (input 1)
% sentencia(2)= Verifica la entrada 2 (input 2) ...
sentencia(h)=ENT(i,h) > rango_min(regla(k,h),h) ...
&& ENT(i,h) < rango_max(regla(k,h),h);

end

if sentencia
    H_ENT(i,:)=ENT(i,:);
else
    continue
end

end

puntos=find(H_ENT(:,1));
H_ENT=H_ENT(puntos,:);

% Verificación. Si no se encuentran puntos para una regla difusa
% determinada, no se podra realizar la optimización de esta regla. Esto
% constituye una limitante de este metodo de Hibridización. En caso que
% eso ocurra, habra que continuar hacia la siguiente regla.

% parametro: Vector que contiene los Parámetros de la k-esima ecuación
%             lineal obtenido a partir del FIS creado por ANFIS
parametro=FIS.output.mf(1,k).params;

% p_dif: Matriz cuyas filas contienen los parámetros de
%         la k-esima regla difusa
p_dif(k,:)=parametro;

if ~isempty(H_ENT)
    % Cálculo del Registro Teorico utilizando los parámetros obtenidos
    % mediante las reglas difusas.

    REGTE0=zeros(size(H_ENT,1),size(parametro,2)-1);
    for i=1:size(parametro,2)-1
        REGTE0(:,i) = parametro(i)*H_ENT(:,i);
    end
    REGTE0=REGTE0 + parametro(end);

```

```

% Ajuste, mediante algortimos genéticos, del registro calculado.
% Se utilizará la función "fun_lineal_fis"

% H_coefs: nuevos coeficientes que brindaran un nuevo ajuste
%           entre los datos de entrenamiento
H_coefs=zeros(size(H_ENT));

% Ciclo. Se calcula el coeficiente (minim) para cada elemento
h=waitbar(0,'Ajustando Reglas Difusas mediante AG...');

for i=1:(size(H_ENT,1))
    minim=ga(@(x) fun_lineal_fis(x,i,REGTEO,H_ENT(:,end)),num+1,...%
              [],[],[],[],[],[],[],op);
    H_coefs(i,:)=minim;
    waitbar(i/(size(H_ENT,1)),h)
end
close(h)

H_coefs=mean(H_coefs);

% Sustitución de los coeficientes obtenidos en la primera regla
% difusa.
FIS.output.mf(1,k).params=H_coefs;

% Matriz que guardara todos los coeficientes optimizados para
% cada regla difusa
p_hb(k,:)=H_coefs;

clear sentencia H_ENT REGTEO

else
    p_hb(k,:)=p_dif(k,:);

    % n_r: Vector que indicará la regla difusa que no ha sido
    %       optimizada (dicha regla sera el número de fila
    %       marcado con el número 1) al no encontrarse datos que
    %       cumplieran con las condiciones
    n_r(k)=1;

```

178

```
    end  
end
```

% Finalmente, se reevalua el FIS ahora modificado:

```
%-----%  
% Pozo Entrenamiento %  
%-----%  
PE=evalfis(pe,FIS);  
  
%-----%  
% Pozo Control %  
%-----%  
PC=evalfis(pc,FIS);  
  
function Z=fun_lineal_fis(x,i,y,w)  
% Función de Entrenamiento (a optimizar) de la forma:  
% f=[(x(1)*y(:,1) + x(2)*y(:,2) + x(n)*y(:,n) + x(n+1))-w]^2  
%  
% Esta función es utilizada dentro del proceso de Hibridización  
%  
% Parámetros de entrada:  
% i: Corresponde al número de elemento dentro del ciclo  
% y: Matriz en donde cada columna contiene los datos seleccionados para la  
%     predicción.  
% w: Corresponde al registro medido cuya predicción se desea realizar  
%     mediante algortimos genéticos  
  
% x: Coeficientes a optimizar  
z=zeros(size(y,2),1);  
for j=1:(size(y,2))  
    z(j)=x(j).*y(i,j);  
end  
zz=sum(z);  
  
% Ecuación final a ajustar.  
Z=(zz+x(j+1)-w(i)).^2;
```

C.9. Función “indice”

```
function r=indice(X,p)
% r = indice(X,p) busca el número de fila (posición) en que se encuentra
% el número "p" dentro del vector X (siendo un vector fila o columna)

r=0;
for i=1:size(X)
    if X(i)==p
        r=i;
    end
end

if r==0
    disp('Este elemento no se encuentra en el vector')
end
```

C.10. Función “log_dif”

```
function [FIS,ENT]=log_dif(PE_AEN,seleccion)
% Función que genera un FIS mediante la utilización de ANFIS, que mejor
% ajuste el(los) registro(s) de entrada con el de salida.
global pred obj

% Matriz ENT: contiene los datos para el entrenamiento. Su ultima columna
% debe corresponderse con los valores de salida
% (Registro que se desea predecir)
ent=PE_AEN(:,pred);
ent(:,end+1)=PE_AEN(:,obj);
ENT=ent(seleccion,:);

% Crea el archivo a ser leido por ANFIS, denominado "Entrenamiento ANFIS"
DS=dataset(ENT);
export(DS,'File','Entrenamiento ANFIS','Delimiter','tab',...
'WriteVarNames',false);
```

180

```
% Llamada al editor de anfis  
anfisedit  
  
adv=warndlg('Presione OK o cierre esta ventana al finalizar ANFIS',...  
    'Advertencia');  
waitfor(adv)  
  
% Al finalizar el editor, se ha de guardar un archivo .fis, el cual tiene  
% los datos del entrenamiento  
% A continuación, se desplegará una ventana en la que ha de escogerse el  
% archivo .fis creado  
  
FIS=readfis;  
  
% Se procederá en cada caso a evaluar los datos de entrada dentro del  
% Sistema de Inferencia Difusa para de esta manera realizar la predicción
```

C.11. Función “normalización”

```
function X=normalizacion(V,W)  
% X = normalizacion(V,W) realiza la normalización de registros de pozo, de  
% acuerdo a la función propuesta por Rafael Banchs, 2001.  
%  
% Parametros de Entrada:  
% V: Matriz que contiene los registros a normalizar (incluyendo  
% profundidad).  
% W: Matriz que contiene los registros en el area de estudio  
%  
% Parámetro de Salida:  
% X: Matriz que contiene cada registro normalizado  
  
% desv: Desviación de cada registro. Es la diferencia de este y su promedio  
%       en el area de estudio  
desv=zeros(size(V));  
for i=2:size(V,2)  
    desv(:,i)=V(:,i)-mean(W(:,i));  
end
```

```

maxdesv=max(abs(desv));

% X: Matriz que contiene cada Registro Normalizado
X=zeros(size(V));
for i=2:size(V,2)
    X(:,i)=((V(:,i)-mean(W(:,i)))/maxdesv(i))+1;
end

```

C.12. Función “registro_final”

```

function RR=registro_final(RegMed,RegPre,Prof,T,nombrePre)
% Función que realiza las siguientes graficas de comparación entre el
% registro medido (real) y el predicho.
%     RegMed: Registro Medido (Real)
%     RegPre: Registro Predicho
%     Prof: Profundidades en las que se desea graficar
%         T: Titulo que identifica al pozo
% nombrePre: Nombre del Registro que se ha predicho
%
%             R: coeficiente de determinación de la grafica

% -----
%% Registros de Pozo %%
% ----- %

figure('Name',T,'NumberTitle','off','Position',[1 31 1024 669],...
    'Color','w','Visible','off','Toolbar','none');

% Para una mejor visualización e interpretación, el mismo registro se ha
% separado en dos. De manera que, de izquierda a derecha, aparecerá así:
% El Registro Completo - La Mitad Superior - La Mitad Inferior.

% Registro Completo
pos=1;
registros_aux(RegMed,RegPre,Prof,nombrePre,pos)

% Mitad Superior
pos=2; filas=round(size(RegPre,1)/2);

```

```

registros_aux(RegMed(1:filas,:),RegPre(1:filas,:),Prof(1:filas,:),...
    nombrePre,pos)

% Mitad Inferior
pos=3;
registros_aux(RegMed(filas:end,:),RegPre(filas:end,:),Prof(filas:end,:),...
    nombrePre,pos)

% La Figura es ahora visible
set(gcf,'Visible','on')

% ----- %
%% Gráfica Comparativa: Predicho vs. Real %%
% ----- %

% Puntos que ayudan a mostrar el grado de certidumbre con el que se ha
% realizado la predicción.
% Se mostrara el valor de R^2 que indica el grado de relación entre ambos.

% Cálculo de R^2
r=corrcoef(RegMed,RegPre);
RR=r(1,2)^2;
R=num2str(RR);

figure('Name','Resultados: Pozo Entrenamiento','NumberTitle','off',...
    'Color','w','Toolbar','none')
plot(RegMed,RegPre,'*','MarkerEdgeColor',[0.5 0.1 0.1])

% Propiedades de los ejes
set(gca,'FontWeight','bold',...
    'Box','on','GridLineStyle',':','Color',get(gcf,'Color'),'FontSize',12)

% Cálculo de la línea de tendencia
[m,b]=min_cuad(RegMed,RegPre);
X=[min(RegMed) max(RegMed)]; Y=zeros(1,2);
for i=1:2
    Y(i)=m*X(i)+b;
end

```

```

% Dibujo de la línea en el gráfico
LL=line(X,Y);
set(LL,'LineWidth',2.5,'Color','k')

% Valor de R^2 en el gráfico
x=get(gca,'XTick');
y=get(gca,'YTick');
TT=text(x(end-1),(y(2)+y(1))/2,['{R}^{2}={ }' R]);
set(TT,'BackgroundColor',[0.5 0.1 0.1],'FontSize',12,'FontWeight','bold',...
    'HorizontalAlignment','center','Margin',8,'Color','w','FontName','Times')
xlabel([nombrePre ': Real'],'FontSize',13)
ylabel([nombrePre ': Predicho'],'FontSize',13)
grid('on')

end

function [m,b]=min_cuad(x,y)
% Función que calcula, mediante mínimos cuadrados, los parámetros "pendiente
% e intersección" ("m" y "b") de la ecuación lineal y = mx + b.
% Tanto "x" como "y" deben ser introducidos como columnas
m=((size(x,1)*sum(x.*y))-(sum(x)*sum(y)))/(size(x,1)*sum(x.^2)-(sum(x))^2);
b=(1/size(x,1))*(sum(y)-m*sum(x));
end

function registros_aux(RegMed,RegPre,Prof,nombrePre,pos)
% pos: Posición, dentro de la ventana, en que se encontraran los registros
%      a graficar.

% Se han de graficar dos registros en un mismo eje.

%----- 1er Registro: Real-----
ax = subplot(1,3,pos);
h=plot(RegMed,Prof,RegPre,Prof); grid on

% Opciones de los ejes
set(gca,'FontWeight','bold','GridLineStyle',':','Box','on',...
    'XAxisLocation','top','YDir','reverse','FontSize',12);
% xlabel({' '});

```

```
% Legenda
[legend_h] = legend(findobj(h,'Type','line'),...
    [nombrePre ': Real'],[nombrePre ': Predicho']);
set(legend_h,'Location','NorthOutside','Box','off')

end
```

C.13. Programa Principal

```
clc
clear all

% Constantes que seran utilizadas a lo largo del algoritmo. Por lo que se
% declararan una sola vez
global obj pred num

% Selección del registro para la predicción. Este debe encontrarse entre
% los seleccionados anteriormente.
% obj: número que indica la columna en que se encuentra el
%       registro que se ha seleccionado para la predicción.
obj=3;

% pred: Vector que contiene la posición del(los) registro(s) para el
%       entrenamiento del algoritmo y posterior predicción.
pred=[6 7];

% num: Número de variables independientes para realizar el entrenamiento
%       del algoritmo. Se corresponde con el número de registros de entrada
%       para la predicción
num=size(pred,2);

% Selección de la tecnica de predicción a utilizar
PREDICCIÓN=3;

%% Carga de los Pozos
%-----%
```

```

% Carga del Pozo de Entrenamiento %
%-----%
% PE: Matriz que contiene los registros del Pozo de Entrenamiento
PE=carga_registro('valoresGF-87N');

% Definición de los topes de la arena de interés del pozo de entrenamiento
% (en pies).
% tsupPE: Tope Superior
% tinfPE: Tope Inferior
tsupPE=7297.9702; tinfPE=7377.0601;

% Escogencia de los registros con los que cuenta el pozo.
% Arreglo de celdas que contiene los Nombres de los Registros
nombresPE={'Profundidad','Densidad de Bulk','DTs','DTp',...
    'Saturación de Agua','Porosidad','Volumen de Arcilla'};

% Arreglo de celdas que contiene las unidades en que fueron medidos cada
% registros
unidadesPE={'(ft)', '(g/cc)', '(us/ft)', '(us/ft)', '(v/v)',...
    '(fracción)', '(v/v)'};

% Titulo que identifica al pozo
tituloPE='Registros: Pozo GF-87N';

% Si entre estos registros seleccionados se encuentra el Sonico de Onda P
% y/o el de Onda S puede convertirse a Velocidad.
for i=2:size(PE,2)
    if strcmp(nombresPE{i}, 'DTs')
        % Posición del Registro DTs en el archivo
        vs=i;
        % Conversión a Registro de Velocidad. El factor 10^6 implica una
        % modificación en las unidades de ft/us a ft/s
        PE(:,vs)=10^6*(1./PE(:,vs));
        nombresPE{vs}=('Velocidad Onda S');
        unidadesPE{vs}=('(ft/s)');
    elseif strcmp(nombresPE{i}, 'DTp')
        % Posición del Registro DTp en el archivo

```

```

vp=i;
% Conversión a Registro de Velocidad. El factor 10^6 implica una
% modificación en las unidades de ft/us a ft/s
PE(:,vp)=10^6*(1./PE(:,vp));
nombresPE{vp}=('Velocidad Onda P');
unidadesPE{vp}=('(ft/s)');

else
    continue
end

% Gráficas de los registros del pozo. Solo se desean graficar las
% profundidades en las que se haya registrado.
graficas_registro(PE,nombresPE,unidadesPE,tituloPE,tsupPE,tinfPE);

% En ocasiones, los registros estan incompletos. Por esta razon:
% PEmed: Matriz que contiene todos los registros del Pozo de Entrenamiento
%         sólo en las puntos donde hubo medición
PEmed=eliminar(PE);

% Selección de los datos para el entrenamiento. Primero ha de seleccionar
% las lutitas que suprayacen la arena, luego la que le infrayace
g=ginput(2);
prof1=indice(PEmed(:,1),round(abs(g(1,2)))); 
prof2=indice(PEmed(:,1),round(abs(g(2,2))));

% Primero ha de seleccionarse el tope superior y luego el tope inferior y
% no al reves
if prof1 > prof2
    error('Se debe seleccionar primero el tope superior y luego el inferior')
end

% Matriz PE_AE: contiene cada registro del Pozo de Entrenamiento en el
%                 area de Entrenamiento
PE_AE=PEmed(prof1:prof2,:);

%-----%
% Carga del Pozo Control%

```

```
%-----%
% PE: Matriz que contiene los registros del Pozo Control
PC=carga_registro('valoresGF-109N');

% Definición de los topes de la arena de interés (en pies).
% tsupPC: Tope Superior
% tinfPC: Tope Inferior
tsupPC=8295.0703; tinfPC=8377.0703;

% Arreglo de celdas que contiene los Nombres de los Registros del Pozo
% Control.
nombresPC={'Profundidad','Densidad de Bulk','DTs','DTp',...
    'Volumen de Arcilla','Saturación de Agua','Porosidad'};

% Arreglo de celdas que contiene las unidades en que fueron medidos cada
% registros
unidadesPC={'(ft)', '(g/cc)', '(us/ft)', '(us/ft)', '(v/v)', '(v/v)', '(fracción)'};

% Titulo que identifica al Pozo
tituloPC='Registros: Pozo GF-109N';

% Si entre estos registros seleccionados se encuentra el Sonico de Onda P
% y/o el de Onda S puede convertirse a Velocidad.
for i=2:size(PC,2)
    if strcmp(nombresPC{i}, 'DTs')
        % Posición del Registro DTs en el archivo
        vs=i;
        % Conversión a Registro de Velocidad. El factor 10^6 implica una
        % modificación en las unidades de ft/us a ft/s
        PC(:,vs)=10^6*(1./PC(:,vs));
        nombresPC{vs}=('Velocidad Onda S');
        unidadesPC{vs}=('ft/s');
    elseif strcmp(nombresPC{i}, 'DTp')
        % Posición del Registro DTp en el archivo
        vp=i;
        % Conversión a Registro de Velocidad. El factor 10^6 implica una
        % modificación en las unidades de ft/us a ft/s
    end
end
```

```

PC(:,vp)=10^6*(1./PC(:,vp));
nombresPC{vp}=('Velocidad Onda P');
unidadesPC{vp}=('ft/s');

else
    continue
end

end

% Gráficas de los registros del pozo. Solo se desean graficar las
% profundidades en las que se haya registrado.
graficas_registro(PC,nombresPC,unidadesPC,tituloPC,tsupPC,tinfPC);

% El Pozo Control no necesariamente tiene el registro seleccionado para la
% predicción.

% Matriz PCmed: contiene todos los registros del Pozo Control en aquellos
% puntos donde hay medición.
PCmed=eliminar(PC);

% Matriz PC_AI: cada registro del Pozo Control en el area de Interes
PC_AI=PCmed((indice(PCmed(:,1),round(abs(tsupPC)))):...
(indice(PCmed(:,1),round(abs(tinfPC)))),:);

% ##### %
%% Entrenamiento del Sistema de Predicción %%
% ##### %

% Normalización de los registros de pozo
% RN_PE: Registros Normalizados del Pozo de Entrenamiento
RN_PE=normalizacion(PEmed,PE_AE);

% RN_PC: Registros Normalizados del Pozo Control
RN_PC=normalizacion(PCmed,PC_AI);

% Registros normalizados en el area de entrenamiento y area de interés
% respectivamente.
PE_AEN=RN_PE(prof1:prof2,:);

```

```

% Para el entrenamiento sólo ha de seleccionarse aleatoriamente el
% 50% del total de los datos. El término 0.4 asegura que la selección
% sea dentro de ese rango.
seleccion=find((rand(1,size(PE_AE,1)))>0.4,round(size(PE_AE,1)*0.5));

% Opciones de los algoritmos genéticos
opciones=gaoptimset('PopulationSize',10,'EliteCount',1, ...
    'CrossoverFcn',@crossoverscattered, ...
    'SelectionFcn',@selectionstochunif, ...
    'Mutation',@mutationgaussian, ...
    'Generations',60, ...
    'Display','off');

if PREDICCIÓN==1
    % Ajuste de los coeficientes utilizando una ecuación lineal
    coef=ag_lineal(PE_AEN,seleccion,opciones);

elseif PREDICCIÓN==2
    % Ajuste de los coeficientes utilizando una ecuación polinomial
    coef=ag_polinomial(PE_AEN,seleccion,opciones);

else
    % Se utilizará la Lógica Difusa como metodo de predicción
    [FIS,ENT]=log_dif(PE_AEN,seleccion);
end

% ##### %
%% Predicción del Registro de Pozo %%
% ##### %

% Cálculo del Registro final. Dependera del metodo seleccionado.
global pred_control

% Selección de los Registros del Pozo Control para la predicción. Deben
% ser los mismos que se utilizaron para el entrenamiento
pred_control=zeros(1,size(pred,2));
for i=1:size(pred,2)
    for j=1:size(nombresPC,2)

```

190

```
if isequal(nombresPE{pred(i)},nombresPC{j})
    pred_control(i)=j;
    break
end
end

if PREDICCIÓN==1
    % Cálculo del(los) coeficiente(s)
    coeficientes=mean(coef);

    %-----%
    % Pozo de Entrenamiento %
    %-----%
P=1; RF_PE=eval_ag_lineal(RN_PE,coeficientes,P);

    %-----%
    % Pozo Control %
    %-----%
P=2; RF_PC=eval_ag_lineal(RN_PC,coeficientes,P);

elseif PREDICCIÓN==2
    % Cálculo de los coeficientes
    coeficientes=mean(coef);

    %-----%
    % Pozo de Entrenamiento %
    %-----%
P=1; RF_PE=eval_ag_polinomio(RN_PE,coeficientes,P);

    %-----%
    % Pozo Control %
    %-----%
P=2; RF_PC=eval_ag_polinomio(RN_PC,coeficientes,P);

else
    %-----%
    % Pozo de Entrenamiento %
```

```

%-----%
RF_PE=evalfis(RN_PE(:,pred),FIS);

%-----%
% Pozo Control %
%-----%
RF_PC=evalfis(RN_PC(:,pred_control),FIS);

end

% ##### %
%% Gráficas Finales %%
% ##### %

%-----%
% Pozo de Entrenamiento %
%-----%

R(1)=registro_final(RN_PE(:,obj),RF_PE,PEmed(:,1),tituloPE,nombresPE{obj});

%-----%
% Pozo Control %
%-----%
R(2)=registro_final(RN_PC(:,obj),RF_PC,PCmed(:,1),tituloPC,nombresPC{obj});

% ##### %
%% Cálculo del Error %%
% ##### %

% En el Pozo de Entrenamiento
Error_PE=sqrt(sum((RF_PE-RN_PE(:,3)).^2)/size(RF_PE,1));

% En el Pozo Control
Error_PC=sqrt(sum((RF_PC-RN_PC(:,3)).^2)/size(RF_PC,1));

Error=[Error_PE Error_PC];

% ##### %
%% Hibridización %%
% ##### %

```

```

if PREDICION==3
    clear RF_PE RF_PC

    % pdif: Parámetros de las reglas difusas antes de ser optimizadas.
    % phib: Parámetros de las reglas difusas optimizadas.

    [RF_PE,RF_PC,reglas,pdif,phib]=hibrido(FIS,ENT,RN_PE(:,pred),...
        RN_PC(:,pred_control),opciones);

    % ##### %
    %% Nuevas Gráficas %%
    % ##### %

    %-----%
    % Pozo de Entrenamiento %
    %-----%
    H_R(1)=registro_final(RN_PE(:,obj),RF_PE,PEmed(:,1),tituloPE, ...
        nombresPE{obj});

    %-----%
    % Pozo Control %
    %-----%
    H_R(2)=registro_final(RN_PC(:,obj),RF_PC,PCmed(:,1),tituloPC, ...
        nombresPC{obj});

    % ##### %
    %% Cálculo del Error %%
    % ##### %

    % En el Pozo de Entrenamiento
    H_Error_PE=sqrt(sum((RF_PE-RN_PE(:,3)).^2)/size(RF_PE,1));

    % En el Pozo Control
    H_Error_PC=sqrt(sum((RF_PC-RN_PC(:,3)).^2)/size(RF_PC,1));

    H_Error=[H_Error_PE H_Error_PC];
end

```