MEZCLA DE EXPERTOS SUPERPUESTOS CON PENALIZACION ENTROPICA

Billy Peralta
Escuela de Informática
Universidad Católica de Temuco
Temuco, Chile
Email: bperalta@uct.cl

Ariel Saavedra
Departamento de Ciencias de Computación
Pontificia Universidad Católica de Chile
Santiago, Chile
Email: adosaa@gmail.com

Luis Caro
Escuela de Informática
Universidad Católica de Temuco
Temuco, Chile
Email: lacaro@uct.cl

***Abstract*—In these days, there are a growing interest in pattern recognition for tasks as predicting weather events, recommending best routes, intrusion detection or face detection. These tasks can be modelled as a classification problem, where a common alternative is using an ensemble model of classification. An usual ensemble model is given by Mixture of Experts model, which belongs to modular artificial neural networks consisting of two subcomponents type: networks of experts and Gating network, and whose combination creates an environment of competition among experts seeking to obtain patterns of the data source, in order to specialize in that particular task, all this supervised in the Gating network, which is the mediator agent and ponders the quality delivered by each expert model solution. We observe that this architecture assume that one gate influence one data point, consequently the training can be misleading to real datasets where the data is better explained by multiple experts. In this work, we present a variant of traditional MoE model, which consists of maximizing the entropy of the evaluation function in the Gating network in conjunction with standard error minimization. The results show the advantage of our approach in multiple datasets in terms of accuracy metric. As a future work, we plan to apply this idea to the Mixture-of-Experts with embedded feature selection.**

I. INTRODUCCION

El aprendizaje automático es una de las áreas con mayor crecimiento de la computación y se refiere al estudio de métodos computacionales para el descubrimiento de nuevos conocimientos. Los métodos de aprendizaje de la máquina se han aplicado a diversos dominios de aplicación tales como microbiología, web mining, sistemas de recomendación y detección de spam [1]. En términos de aplicaciones, una tarea relevante está dada por la clasificación de datos para lo cual existen múltiples técnicas donde destacan los modelos basados en ensambles.

Una modelo de clasificación basado en ensamble típico está dado por la Mezcla de Expertos (conocida como MoE por *mixture of experts*) fue propuesta por [2] y tiene como foco la idea de presentar un modelo de aprendizaje supervisado, compuesto de varias subredes que emulan el comportamiento modular y a la vez, estratificar el espacio de entrada llevando a cabo la estrategia de "dividir y vencer". Se puede considerar que la Mezcla de expertos pertenece a a las llamadas redes neuronales artificiales modulares (MANN por Modular Artificial neural networks). Los modelos de Mezcla de Expertos tienen dos componentes básicos: redes de expertos y redes de agregación (o mejor conocida como *Gating*).

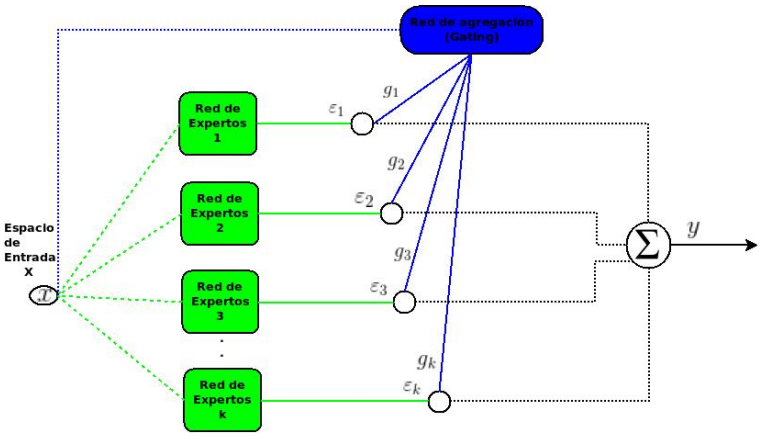

Fig. 1. Arquitectura de la red Mezcla de Expertos

En la arquitectura de la red de la Figura I, los cuadrados ovalados de color verde simbolizan a las redes de expertos, y son aquellas subredes que compiten por aprender sobre la fuente de datos y entregar la mejor predicción sobre los patrones de ella, que a su vez, dicho ambiente de competición es arbitrado por la red Gating simbolizada por el cuadrado ovalado de color azul. La arquitectura MoE presenta ciertas características que son detalladas a continuación:

- Las salidas de la red Gating están definidas por la cantidad de redes de expertos que hay en la arquitectura, es decir, que el numero de salidas del Gate debe ser igual al numero de expertos en total.
- Cada una de las salidas de la red Gating g_i tiene un valor entero no negativo, y la suma de todas las salidas g_k es igual a uno.
- La salida final de la arquitectura de la red es denotada por y y es igual a la sumatoria de las salidas de las redes de expertos, ponderada por cada una de las salidas de la red Gating, de tal forma que:

$$y = \sum_{i=0}^{k} g_i \varepsilon_i \tag{1}$$

Las redes de expertos tratan de especializarse en regiones particulares de la fuente de datos, y su resultado es evaluado por medio de las salidas de la red Gating, que es interpretado como la relevancia que tiene dicho experto ε_i para la clasificación, y esto se refleja directamente en la configuración de la red, por medio de sus parámetros o pesos, dando de paso a dos posibles entornos: competitivo y cooperativo.

En entornos competitivos, los parámetros de la red Gating se ajustaran de modo que solo una de las salidas de dicha red tendrá el valor 1 y el resto en 0, y en entorno cooperativo se basa en que los parámetros de la red Gating están definidos para que las salidas presenten valores más uniformes entre ellos, esto indica que múltiples expertos pueden ser relevantes en la clasificaron de la fuente de datos.

El presente trabajo propone una variante del modelo clásico de Mezcla de Expertos donde se introduce la idea de maximizar la entropía en el entrenamiento de la red Gating para la obtención de sus parámetros, con el fin de modificar el entorno de competitivo a uno más cooperativo.

II. Mezcla de expertos

La arquitectura se compone de un conjunto de redes de Expertos en el cual cada uno de ellos resuelve una parte del problema usando una función de aproximación sobre el espacio de entrada. La idea principal de la mezcla de expertos es obtener modelos locales, donde cada uno se especialice en una parte particular de los datos. Asumiendo N ejemplos etiquetados de entrenamiento, donde el n-ésimo dato está dado por (x_n, y_n). Tenemos que $x_n \in \mathbb{R}^D$ e $y_n \in C$ que es igual al conjunto de clases etiquetas con cardinalidad Q y está conformado por $\{c_1, c_2....c_Q\}$. Estableciendo un modelo probabilístico en el i-ésimo experto que relacion la salida y con la entrada x mediante un parámetro ω_i se tiene:

$$p(y|x, \omega_i) = p(y|x, \omega_{l,i}), i = 1, 2, ..., k \tag{2}$$

Donde ω_j es el vector de parámetros del experto j-ésimo y l es una variable que indica la clase evaluada. Siguiendo a Moerland [3], para la clasificación vamos a usar una función de densidad multinomia. Luego la función de los expertos es definida como:

$$p(y = c_l|x, \omega_i) = \frac{exp(\omega_{li}^T x)}{\sum_{j=1}^{Q} exp(\omega_{ji}^T x)} \tag{3}$$

Específicamente ω_{li} denota el vector de parámetros del modelo, que depende de la clase c_l y del experto i que indexa en $\{1, 2, ..., k\}$. Tomando en cuenta la dimensión D de la instancia de entrada x, entonces existe un total de QxK vectores ω_{li} por cada x_j. Similar a los expertos, la arquitectura MoE presenta un esquema de función para la red Gating, cuyo trabajo es el de dividir y evaluar el espacio de entrada x que corresponden a las distintas redes expertos. Las probabilidades en la salida de la función de compuerta de red están dadas por g_i con valores dados por $\{g_1, g_2, ...g_k\}$ y su formulación está dada por:

$$g_i(x, \nu) = p(x, \nu_i) = \frac{exp(\nu_i^T x)}{\sum_{j=1}^{K} exp(\nu_j^T x)} \tag{4}$$

Donde ν_i denota el vector de parámetros del modelo donde $i = \{1, 2, ..., k\}$. De esta forma, tomando en cuenta la dimensión D de la instancia de entrada x, podemos ver que hay j vectores ν_i.

Con las funciones de Expertos y Compuerta ya definidos, la salida final de la red corresponde a la suma ponderada de los expertos con las salidas de la red Gating, pero esto se explica desde el lado probabilístico, asumiendo desde ahora el siguiente enfoque según [4]. Se asume que para un cierto conjunto de entrenamiento x y un experto que denotaremos como ε_i es seleccionado con la probabilidad $p(\varepsilon_i|x, \nu)$ que es igual a la función del Gate $g_i(x, \nu)$. Además de una salida y, correspondiente a la clase, escogida con probabilidad $p(y|x, \omega_i)$. Por ende, la probabilidad total para generar y con respecto a x se puede calcular a través de densidades de mezcla de la siguiente manera:

$$p(y|x) = \sum_{i=1}^{k} p(\varepsilon_i|x, \nu)p(y|x, \omega_i) = \sum_{i=1}^{k} g_i(x, \nu)p(y|x, \omega_i) \tag{5}$$

Podemos concluir y resumir el funcionamiento de la arquitectura MoE de la siguiente manera: Las redes de expertos modelan los distintos procesos que generan los datos x, y la red Gating modela la decisión de utilizar uno de esos diferentes procesos. Ya sea en entornos competitivos y cooperativos , la salida y puede ser generada de k maneras diferentes, correspondientes a las k salidas de los expertos. El Algoritmo 2 nos indica el cálculo de las probabilidad de salidas para un dato de entrada.

Algorithm 1 Algoritmo de la arquitectura de Mezcla de Expertos (MoE)

- Para cada experto:
  Calcular la probabilidad de la entrada con respecto a cada posible clase por computar $p(y = c_l|x, \omega_i) = \frac{exp(\omega_{li}^T x)}{\sum_{j=1}^{Q} exp(\omega_{ji}^T x)}$
- Para cada salida de la red Gating:
  Computar la probabilidad de cada una de la salida de compuerta $g_i(x, \nu) = p(x, \nu_i) = \frac{exp(\nu_i^T x)}{\sum_{j=1}^{K} exp(\nu_j^T x)}$
- Computar la mezcla de densidades de mezcla de las redes de Expertos y las salidas de la red compuerta: $p(y|x) = \sum_{i=1}^{k} g_i(x, \nu)p(y|x, \omega_i)$

La forma de estimar los parámetros ν_i de la red compuerta y ω_i de las redes de Expertos se vera la siguiente sección.

A. Máxima Expectación para Mezcla de expertos

Primero observamos la función de log-verosimilitud sobre la base de la probabilidad total $p(y|x)$ [5] se tiene que :

$$\hat{\ell}(\nu,\omega) = ln(\mathcal{L}(\nu,\omega)) = \sum_{n=1}^{N} ln \sum_{i=1}^{K} g_i(x_n,\nu) p(y_n|x_n,\omega_i) \quad (6)$$

Siguiendo a [4] a considerar la asignación de los expertos como conocida mediante variables ocultas z, se tiene que el logaritmo de la función de verosimilitud completa está dada por:

$$\hat{\ell}(\nu,\omega|x,z) = \sum_{n=1}^{N} \sum_{i=1}^{K} z_i^n \ ln(g_i(x_n,\nu) p(y_n|x_n,\omega_i)) \quad (7)$$

Esta ecuación ya no implica el logaritmo de una suma, pero aun seguimos desconociendo z, por ende $\hat{\ell}(\nu,\omega|x,z)$ no es directamente aplicable. Hasta este punto, los requisitos mínimos para aplicar los pasos E y M del algoritmo EM ya han sido adaptados. Empezaremos definiendo la expectación de la función de log-verosimilitud para computar el paso E:

- **Paso E:**

  El valor esperado de la variable oculta, puede ser inferido a través del teorema de Bayes dado la probabilidad:

  $$\mathbb{E}(z_i^n) = p(z_i^n = 1|y_n,x_n) = \frac{p(y_n|z_i^n=1,x_n)p(z_i^n=1|x_n)}{p(y_n,x_n)} \quad (8)$$

  Reemplazando las probabilidades del numerador de la ecuación 2.72 obtenemos que $p(y_n|z_i^n = 1,x_n)$ es equivalente a la densidad del expertoε_i $p(y_n|x_n,\omega_i)$ y $p(z_i^n = 1|x_n)$ es equivalente a la salida de la red compuerta $g_i(x_n,\nu)$. Mientras que el denominador es igual a la salida total ponderada de la red. Dicho valor es interpretado como la variable de responsabilidad π_j^n que es definida como la probabilidad **a posteriori** del experto $i-esimo$ para la muestra o ejemplo de entrenamiento n.
- **Paso M:**
  Ya definido el paso E, aplicando la propiedad del producto del logaritmo obtenemos el valor esperado de la función de log-verosimilitud para los datos completos, que serán usados por el algoritmo EM en cada iteración:

  $$Q^m = \sum_{n=1}^{N} \sum_{i=1}^{K} \pi_i^n \ ln(g_i(x_n,\nu)) + ln(p(y_n|x_n,\omega_i)) \quad (9)$$

  Aplicando cálculo, tenemos que respecto a la función de compuerta obtenemos:

  $$\frac{\partial Q}{\partial \nu_i} = \sum_{n=1}^{N} \sum_{i=1}^{K} \pi_i^n \frac{g_i(x_n,\nu)'}{g_i(x_n,\nu)} \quad (10)$$

  Y respecto a los parámetros de la densidad de los expertos se tiene:

  $$\frac{\partial Q}{\partial \omega_i} = \sum_{n=1}^{N} \pi_i^n \ \frac{p(y_n|x_n,\omega_i))'}{p(y_n|x_n,\omega_i)} \quad (11)$$

Operando la optimización se generan el siguiente para de ecuaciones:

$$\sum_{n=1}^{N} (g_i(x_n,\nu) - \pi_i^n) x_n = 0 \quad (12)$$

$$\sum_{n=1}^{N} x_n(\varepsilon_i - y_n) = 0 \quad (13)$$

Siguiendo la estrategia de mínimos cuadrados ponderados [3] y efectuando la aproximación de función softmax en el caso de parámetros de función de compuerta se tiene:

$$W_j^T = (X^T \Pi_j X)^{-1} X^T \Pi_j \ ln(Y) \quad (14)$$

$$V^T = (X^T X)^1 X^T ln(\Pi_j) \quad (15)$$

Donde X representa los datos llanos de dimensión NxD, W_j es la matriz de parámetros del experto $j-esimo$ de QxI e Y es la matriz de las clases de largo NxQ. Finalmente Π_j es la matriz de las responsabilidades de un tamaño de KxD.

III. Mezcla superpuesta

El planteamiento se basa en el concepto de entropía en teoría de la información, y como esta influye en la determinación de que tan sesgado es un sistema estadístico, por ejemplo, se puede pensar que a priori, la elección de la mejor decisión en términos de bondad sobre el estado de un evento, esta relaciona en la medida de cuanta **información** posee en relación con los demás. Formalizando esta intuición, se establece que una distribución de probabilidad es menos sesgada, según ciertos parámetros fijos que permiten que la entropía sea máxima, es decir, cuando menos información especifica al problema contenga; esto se puede determinar con un método conocido en teoría de la información estadística como estimadores de máxima entropía [6].

Considerando la ecuación de la red de compuerta donde se adiciona la entropía de Shannon, se obtiene:

$$E_g = \sum_{n=1}^{N} \sum_{i=1}^{K} \pi_i^n \ ln(g_i(x_n,\nu)) + \lambda \sum_{n=1}^{N} \sum_{i=1}^{K} g_i(x_n,\nu) log_2(g_i(x_n,\nu)) \quad (16)$$

Por facilidad de operación, vamos a separar en E_g a los componentes de función de compuerta y entropía adicionada:

$$\underbrace{\sum_{n=1}^{N} \sum_{i=1}^{K} \pi_i^n log(g_i(x_n,\nu))}_{\alpha} + \underbrace{\sum_{n=1}^{N} \sum_{i=1}^{K} \lambda \ g_i(x_n,\nu) log(g_i(x_n,\nu))}_{\beta} \quad (17)$$

El primer sumando de la ecuación 3.16, en el cual se le ha dado la etiqueta de α, corresponde a la ecuación conocida de la red compuerta que suele ser la misma que la ecuación 2.76. El segundo sumando que ha sido denominado β, es el resultado de la inclusión del término de regularización entrópica al que se ha llegado. Ahora se procede a congelar β en la iteración anterior, que simbolizando como $t-1$ se obtiene:

$$\beta = \sum_{n=1}^{N}\sum_{i=1}^{K} \lambda\ g_i^{t-1}(x_n,\nu) log(g_i(x_n,\nu)) \quad (18)$$

Reemplazando la versión de β en la iteración anterior 3.17, se tiene:

$$E_g = \underbrace{\sum_{n=1}^{N}\sum_{i=1}^{K} \pi_i^n log(g_i(x_n,\nu))}_{\alpha} + \underbrace{\sum_{n=1}^{N}\sum_{i=1}^{K} \lambda\ g_i^{t-1}(x_n,\nu) log(g_i(x_n,\nu))}_{\beta} \quad (19)$$

Considerando que la red compuerta tiene una dependencia lineal de $s_i = \nu_i^T x$ obtenemos que:

$$\frac{\partial \hat{\ell}(\nu,\omega)}{\partial \nu} = \sum_n (\pi_i^n - g_i(x_n,\nu)) \quad (20)$$

Aproximando β de forma análoga a α, y siguiendo a [3], se tiene:

$$\sum_n (\pi_i^n - g_i(x_n|\nu))x_n \ + \lambda \sum_n (g_i(x_n,\nu) - g_i^{t-1}(x_n,\nu))x_n = 0 \quad (21)$$

Dado que la ecuación 21 no se puede resolver directamente, aplicaremos la misma solución previa usando los logaritmos de las salidas, obteniendo:

$$\sum_n (log(\pi_i^n) - s_{in})x_n \ + \lambda \sum_n (s_{in} - s_{in}^{t-1})x_n = 0 \quad (22)$$

Por lo tanto, el cambio en paso M de mezcla de expertos considerando la forma matricial está dada por:

$$V = \frac{1}{1-\lambda}(X^T X)^{-1} X^T log(\Pi) - \frac{\lambda}{1-\lambda} V^{T-1} \quad (23)$$

La ecuación previa es la modificación del paso M que se requiere para definir los nuevos parámetros ν en la red de compuerta. Se espera que este cambio tenga efectos en la superposición de los expertos con el fin de aumentar la tasa predicción.

IV. Experimentos

En esta sección se presentan los resultados de los experimentos sobre los conjuntos de datos reales y de imágenes. Los experimentos en los conjuntos de datos reales, su ejecución fue mediante validación cruzada estratificada, con 30 particiones o 30K-folds en entrenamiento y pruebas y 10 particiones o 10K-folds entre entrenamiento y validación. Para los conjuntos de imágenes se utilizó el mismo enfoque, pero con 10 particiones o 10K-folds entre entrenamiento y pruebas, y 5 particiones o 5K-folds en entrenamiento y validación, todo esto por la gran dimensionalidad y el tiempo que se emplea en procesar estos últimos conjuntos.

En nuestros experimentos, vamos a comparar considerando las siguientes:

- Log-verosimilitud entre los dos métodos a comparar: Es el cálculo de la típica log-verosimilitud entre ambos métodos, por cada iteración (puede variar en cada conjunto de datos).
- Nivel o tasa de predicción (accuracy) promedio: Es la exactitud de la predicción graficada al terminar el cálculo de la media aritmética en validación cruzada, según la cantidad de expertos (este número es fijo en cada media aritmética, y corresponde a 10,20,30,40 y 50 expertos).

A continuación se presentan los resultados obtenidos dentro de este trabajo:

A. Log-verosimilitud entre los dos métodos a comparar

Esta subsección esta dedicada a graficar el comportamiento de la Log-verosimilitud (Log-likelihood) como variable comparativa de convergencia entre ambos métodos.Esta conducta tendrá una vital importancia en la determinación de las iteraciones EM para la obtención de parámetros de la red compuerta y la red de expertos. La dimensionalidad empieza a ser notoria en la capacidad de cómputo de los equipos en que se ejecutan los experimentos, por ende, se pone especial énfasis en la observación de la verosimilitud con el fin de aumentar la velocidad y disminuir la carga de la ejecución. Los resultados obtenidos son los siguientes:

Ionosphere:

- Dicho conjunto de datos al poseer una dimensionalidad reducida, es de rápido computo en el servidor HP e incluso en el computador personal del alumno. Se puede observar para la primera cantidad de expertos ($k = 10$) la log-verosimilitud de MoE clásico es superior a través de todas las iteraciones, pero esta se regulariza en la iteración numero 100, y a partir de este punto, prácticamente los pasos de convergencia son muy lentos en ambos casos. Por ende se han seleccionado solo las primeras 100 iteraciones para la experimentación, en función de mejorar el tiempo de ejecución.
- Para $K = 20$ expertos, se encuentra una situación muy similar, ya que los pasos de convergencia después de la iteración 100 se vuelven muy bajos y se ha decidido en trabajar con dicho número en la experimentación.

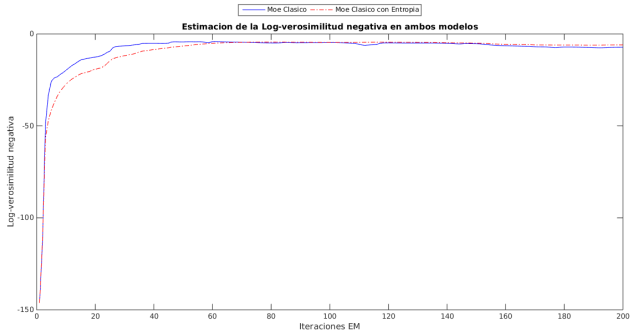

Fig. 2. Valor numérico de la log-verosimilitud para MoE clásico y MoEEntrópico con en el conjunto de datos Ionosphere, con 200 iteraciones de muestra y solo las primeras 100 seleccionadas como punto de convergencia, todo esto con 20 expertos.

Spectf:

- Este conjunto presenta un comportamiento ligeramente irregular con respecto a Ionosphere, pero se puede apreciar el punto de convergencia en general bordeando las 50 iteraciones para experimentación, que justamente concuerda con el número aproximado de los experimentos previos. También se incluye en todos los numero de expertos como muestra en 200 iteraciones.
  En $K = 20$ es en donde el comportamiento es más irregular, que crea confusión para determinar el punto de convergencia. En los experimentos previos se determinó a priori las 50 iteraciones también, por ende se respetara dicho número por la poca diferencia entre las 50 y 100 iteraciones en términos de magnitud de verosimilitud para MoE Clásico, excepto en MoEEntrópico que cae muy rápidamente al avanzar el algoritmo.

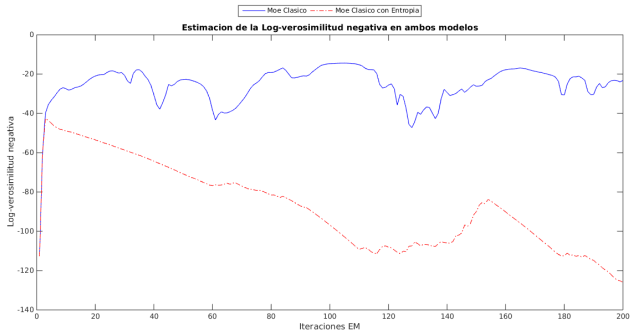

Fig. 3. Valor numérico de la log-verosimilitud para MoE clásico y MoEEntrópico con en el conjunto de datos Spectf, con 200 iteraciones de muestra y solo las primeras 50 seleccionadas como punto de convergencia, todo esto con 20 expertos.

Sonar:

- En el conjunto de datos Sonar se puede apreciar en la mayoría de los resultados según su cantidad de expertos asociado, un comportamiento regular ascendente manteniendo una asíntota cercana a las 30 iteraciones, a excepción de $K = 30$, que presenta un punto de convergencia cercano a la iteración 50. Al igual que los otros conjuntos de datos, se han mantenido las 200 iteraciones como muestra (se aplicó un zoom a los gráficos para poder apreciar mejor el comportamiento hasta la iteración numero 100) y las 50 iteraciones para experimentación.

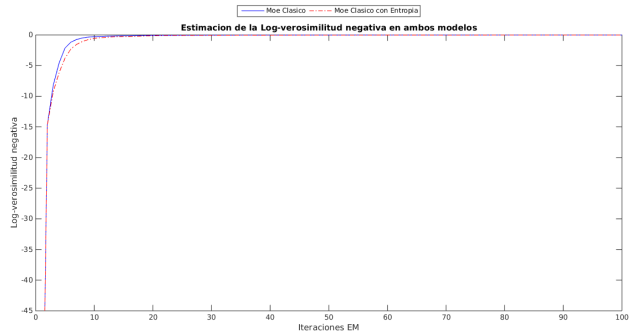

Fig. 4. Valor numérico de la log-verosimilitud para MoE clásico y MoEEntrópico con en el conjunto de datos Sonar, con 200 iteraciones de muestra y solo las primeras 50 seleccionadas como punto de convergencia, todo esto con 20 expertos.

Musk:

- Musk-1 es el conjunto de datos con más dificultad entre aquellos que poseen una dimensionalidad reducida (menor a 200). En casi todos casos se puede apreciar un comportamiento regular a las 50 iteraciones, a excepción de $K = 20$ que en el caso de MoEEntrópico presenta una curva bastante descendente, mostrando su mejor valor en las primeras iteraciones de la ejecución. En este caso también se acoge la premisa de las 200 iteraciones en muestra (con zoom en 100 iteraciones) y 50 en experimentación.

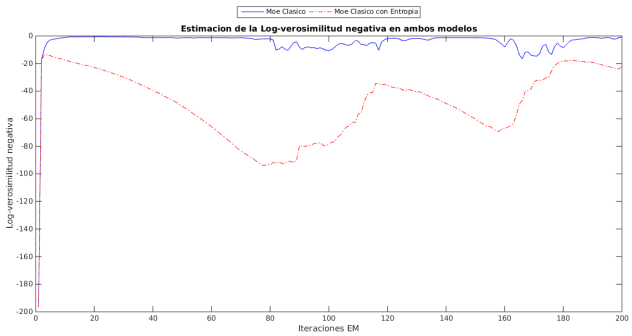

Fig. 5. Valor numérico de la log-verosimilitud para MoE clásico y MoEEntrópico con en el conjunto de datos Musk-1, con 200 iteraciones de muestra y solo las primeras 50 seleccionadas como punto de convergencia, todo esto con 20 expertos.

Secom:

- En este conjunto de datos los resultados son bastante similares unos de los otros, en el cual los puntos más

altos para ambos modelos se presentan en las primeras 3 iteraciones, y luego suben aproximadamente en la ultimas 5 (tomando en cuenta 50 iteraciones como muestra). Pero estos últimos repuntes de Log-verosimilitud no son lo suficientemente altos para considerar llegar a las casi 50 iteraciones, por ende se redondea 3 a 5 iteraciones para experimentación en el cual los resultados son los siguientes:

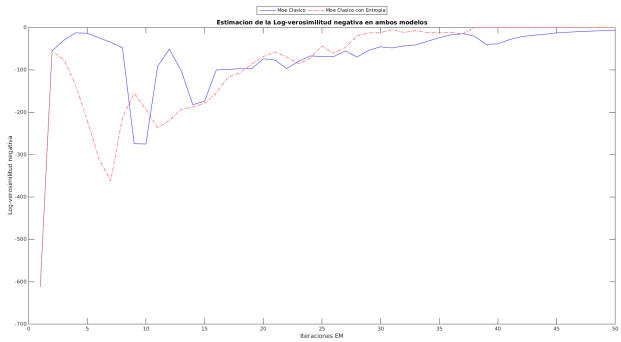

Fig. 6. Valor numérico de la log-verosimilitud para MoE clásico y MoEEntrópico con en el conjunto de datos Secom, con 50 iteraciones de muestra y solo las primeras 5 seleccionadas como punto de convergencia, todo esto con 20 expertos.

B. Parámetro λ encontrado según la mejor tasa de predicción y resumen general de la experimentación

En este apartado se presentan los valores de los parámetros encontrados entre los posibles candidatos del vector λ. Si analizamos nuevamente las dos expresiones fundamentales que impactan en el cálculo de los parámetros en la actual iteración ($(X^T X)^{-1} X^T log(\Pi)$) y los parámetros de la iteración anterior (V^{T-1}), corresponden a:

- $\frac{1}{1-\lambda}$ es el termino penalizador del calculo de parámetros en la iteración actual
- $\frac{\lambda}{1-\lambda}$ es el termino penalizador del calculo de parámetros en la iteración anterior.

Ya que cada conjunto de datos presenta patrones y formas distintas, el enfoque de búsqueda en grilla (Grid search) describe el procedimiento de búsqueda que consta de iterar sobre todos los posibles candidatos que mejor represente la relación del cálculo en iteraciones actuales y anteriores, todo en función de la meta propuesta, mejorar la tasa de predicción con respecto al esquema clásico. Se pueden encontrar todos los parámetros óptimos con su correspondiente tasa de predicción en la tabla 1, además de presentar el resumen general de la investigación por todas las cantidades de expertos en la tabla 2.

En resumen se encuentra que el modelo propuesta mejora los resultados de la mezcla clásica de expertos en casi la totalidad de casos con un rango aproximado de 1% a 4%. En las bases de datos de mayor dimensionalidad no se obtuvo mejoras, donde nosotros hipotetizamos que la alta dimensionalidad origina mayor posibilidad de sobreentrenamiento afectando a nuestro modelo ya que asume modelos mas complejos para los datos de entrada. Esto sugiere que el uso de mezclas de expertos penalizadas con seleccion embebida de variables puede mejorar el modelo propuesto.

V. Conclusiones

El método de Mezcla de Expertos superpuestos con penalización entrópica pone énfasis en los valores de los parámetros λ, en el cual los resultados de los experimentos han mostrando una tendencia negativa de estos parámetros en los conjuntos de datos de menor dimensionalidad (Ionosphere hasta Musk) y con valores en su mayoría positivos en los conjuntos de datos de dimensionalidad superior (Arrhythmia en adelante). Observando las gráficas asociadas a los términos de penalización $\frac{1}{1-\lambda}$ y $\frac{\lambda}{1-\lambda}$ podemos afirmar que la penalización disminuye en torno al intervalo $0 < \lambda < 1$ y esta aumentara para los parámetros de la actual y anterior iteración cuando λ se aleje de este intervalo, ya sea positiva o negativamente. Pese a esto, ya sea cualquier valor de λ escogido como el óptimo, causara un aumento en la entropía promedio en la arquitectura con respecto al esquema clásico, y además existe la tendencia hacia la uniformidad de las magnitudes de las salidas de la red compuerta, en contraste con el esquema clásico, en el cual una sola de las salidas tiene una ventaja numérica importante con respecto a las otras. En torno a los experimentos de rendimiento, estos proporcionan la evidencia de que MoEE mejora la tasa de predicción con respecto MoE clásico en 3 puntos porcentuales promedio, aunque también es importante recalcar que este comportamiento no esta presente en todos los conjuntos, especialmente los dos últimos (PIE10P y Leukemia) en el cual no hay ninguna diferencia en tasa de predicción o Log-verosimilitud. Como trabajo futuro, se planea implementar esta penalización en mezclas de expertos con selección embebida de variables.

Agradecimientos

Este trabajo fue parcialmente financiado por proyecto FONDECYT de Iniciación 11140892.

References

[1] C. M. Bishop, *Pattern Recognition and Machine Learning.* Oxford: Springer, 2006.

[2] R. A. J. . M. I. Jordan, "Adaptive mixture of local experts," *Department of Brain and cognitive science, Massachusetts Institute of Technology*, 1991. [Online]. Available: http://dx.doi.org/10.1002/andp.19053221004

[3] P. Moerland, "Some methods for training mixtures of experts," *Technical Report, IDIAP Research Institute*, 1997. [Online]. Available: http://publications.idiap.ch/downloads/reports/1997/com97-05.pdf

[4] M. I. Jordan and L. Xu, "Convergence results for the em approach to mixtures of experts architectures," *Department of Brain and cognitive science, Massachusetts Institute of Technology*, 1993. [Online]. Available: http://home.mit.bme.hu/ szekelyn/education/hibrid3D/jordan-xu.pdf

[5] B. Peralta and A. Soto, "Embedded local feature selection within mixture of experts," *Information Sciences*, 2014.

[6] E. Jaynes", ""information theory and statistical mechanics"," *"Standford University"*, "1957". [Online]. Available: "http://bayes.wustl.edu/etj/articles/theory.1.pdf"

TABLE I

RESUMEN DE LOS MEJORES PARÁMETROS ENCONTRADOS EN GRID SEARCH JUNTO A SU TASA DE PREDICCIÓN CONJUNTA, SEGÚN CADA UNO DE LOS CONJUNTOS DE DATOS ANALIZADOS.

Dataset	K=10		K=20		K=30		K=40		K=50	
	λ	predicción	λ	predicción	λ	predicción	λ	predicción	λ	predicción
Ionosphere	-32	93.6%	-128	94.3%	-16	95.7%	-32	94.3%	-128	96.4%
Spectf	128	100%	128	100%	-2	100%	-1.5	100%	8	100%
Sonar	-1	76.1%	-1	79.7%	64	80.9%	-1.5	83.3%	-2	78.5%
Musk	32	81.1%	-32	81.6%	-32	84.2%	-16	83.7%	-16	84.2%
Arrhythmia	0.5	61.3%	0.5	62.4%	0.5	65.1%	0.5	64.6%	0.5	65.1%
Secom	8	93.1%	4	93.6%	8	93.1%	8	93.3%	32	93.4%
PIE10P	128	100%	128	100%	128	100%	128	100%	128	100%
Leukemia	8	93.1%	128	96.5%	64	100%	32	100%	128	100%

TABLE II

RESUMEN DE LA TASA DE PREDICCIÓN PROMEDIÓ(MÁS SU DESVIACIÓN ESTANDAR), USANDO 30K-FOLDS EN VALIDACIÓN CRUZADA ESTRATIFICADA EN LOS CONJUNTOS DE DATOS REALES PARA MOE CLÁSICO Y MOEENTRÓPICO. DICHO PROMEDIO DE PREDICCIÓN PARA AMBOS MODELOS ESTÁ CLASIFICADO SEGÚN EL NÚMERO DE K EXPERTOS PROPUESTOS ALREDEDOR DEL TRABAJO ($K = 10, 20, 30, 40, 50$).

Dataset	K=10		K=20		K=30		K=40		K=50	
	MoE	MoEEntrópico	MoE	MoEEntrópico	MoE	MoEEntrópico	MoE	MoEEntrópico	MoE	MoEEntrópico
Ionosphere	85.1%(0.022)	**88.4**%(0.015)	87.9%(0.025)	**90.1**%(0.023)	86.9%(0.024)	**91.0**%(0.025)	87.3%(0.020)	**90.7**%(0.023)	87.6%(0.029)	**91.1**%(0.026)
Spectf	70.6%(0.067)	**72.8**%(0.073)	72.7%(0.044)	**78.0**%(0.127)	68.0%(0.067)	**73.2**%(0.155)	71.0%(0.086)	**75.5**%(0.075)	72.5%(0.082)	**74.8**%(0.093)
Sonar	**67.5**%(0.046)	**67.5**%(0.040)	67.2%(0.038)	**67.6**%(0.047)	**69.2**%(0.043)	69.0%(0.041)	69.24%(0.052)	**69.28**%(0.047)	67.5%(0.059)	**67.9**%(0.059)
Musk	75.7%(0.031)	**75.8**%(0.030)	75.9%(0.024)	**76.1**%(0.027)	75.8%(0.022)	**76.1**%(0.017)	76.6%(0.033)	**76.7**%(0.037)	**77.4**%(0.034)	77.2%(0.032)
Arrhythmia	48.2%(0.035)	**49.7**%(0.033)	51.3%(0.048)	**55.1**%(0.063)	48.3%(0.032)	**56.5**%(0.058)	49.8%(0.028)	**55.0**%(0.063)	50.3% (0.035)	**57.0**%(0.038)
Secom	88.8%(0.012)	**92.1**%(0.008)	89.1%(0.010)	**92.2**%(0.010)	89.2%(0.014)	**92.3**%(0.009)	89.0%(0.012)	**92.4**%(0.009)	89.6%(0.012)	**92.7**%(0.010)
PIE10P	**100**%(0)	**100**%(0)	**99.96**%(0.001)	**99.96**%(0.001)	**100**%(0)	**100**%(0)	**100**%(0)	**100**%(0)	**100**%(0)	**100**%(0)
Leukemia	**80.8**%(0)	**80.8**%(0)	**80.6**%(0.001)	80.5%(0.001)	**98.2**%(0)	**98.2**%(0)	**97.4**%(0)	**97.4**%(0)	**98.3**%(0)	**98.3**%(0)