Aprendizaje Profundo Localmente Ponderado

María Fernanda Hernández Luquin, Hugo Jair Escalante

Reporte Técnico No. CCC-20-002
16 de junio de 2020

Luis Enrique Erro 1
Sta. Ma. Tonantzintla,
72840, Puebla, México.

RESUMEN

En esta propuesta de investigación se propone la combinación de dos tipos de aprendizaje: el aprendizaje localmente ponderado y el aprendizaje profundo, para crear un esquema llamado *Aprendizaje profundo ponderado localmente, LWDL*. Al unir estos dos enfoques exploraremos el desempeño de los modelos propuestos en dominios de aplicación donde el aprendizaje local sea favorable, que incluyen el reconocimiento de emociones y la clasificación de ejemplos de clase minoritaria, con traslape de clases, datos con ruido y grano fino. La finalidad es obtener un desempeño competitivo con el estado del arte y un modelo robusto e interpretable en el reconocimiento de emociones y la clasificación de ejemplos dentro de los dominios de alcance del aprendizaje local. El esquema propuesto consiste en integrar el aprendizaje local en un modelo de aprendizaje profundo de extremo a extremo. La contribución principal se enfoca en la mejora de los modelos convencionales de aprendizaje profundo, bajo la hipótesis de que los métodos que incluyen el aprendizaje local tienen un mejor desempeño debido a que son capaces de generalizar a pesar de existir una gran similitud entre los atributos de las clases. Los resultados preliminares obtenidos con el esquema LWDL muestran ser competitivos con respecto al estado del arte en el dominio de aplicación como el ER.

Palabras Clave: Aprendizaje Profundo Localmente Ponderado, Aprendizaje Profundo, Aprendizaje Localmente Ponderado.

ABSTRACT

This research proposal proposes the combination of two types of learning: locally weighted learning and deep learning, to create a scheme called *Locally weighted deep learning, LWDL*. By combining these two approaches, we will explore the performance of the proposed models in application domains where local learning is promising, including emotion recognition and classification of: minority class examples, overlapping classes, data with noise and fine-grained. The aim is to obtain a competitive performance with state of the art and a robust and interpretable model in emotion recognition and the classification of examples within the scope of local learning. The proposed scheme is to integrate local learning into an end-to-end deep learning model. The main contribution focuses on the improvement of conventional deep learning models, under the hypothesis that the methods that include local learning have a notable performance due to it can generalize despite the existence of notable similarity between the attributes of the classes. The preliminary results obtained with the LWDL scheme, it shows to be competitive to state of the art in ER.

Keywords: Locally Weigted Deep Learning, Deep Learning, Locally Weighted Learning.

Contenido

1. Introducción

Las emociones son un fenómeno corto físico-psicológico que se presenta como modos de adaptación cuando un entorno demanda un cambio. Psicológicamente, las emociones alteran la atención, activando ciertos comportamientos en respuestas biológicas como las expresiones faciales, cambio en el tono de voz, movimientos musculares y la activación del sistema nervioso [44].

El reconocimiento de emociones (*Emotion Recognition*, ER) es la capacidad de identificar estados emocionales humanos mediante el análisis del habla, expresiones faciales, y gestos corporales [46]. Una forma de reconocer emociones se puede llevar a cabo mediante el análisis de imágenes de expresiones faciales. ER juega un papel importante en la computación afectiva y ha despertado un gran interés en muchas aplicaciones que incluyen, la interacción humano-computadora, inteligencia artificial, recomendación de vídeo, la comunicación paralingüística, psicología clínica, psiquiatría, neurología, evaluación del dolor, detección de mentiras, entornos inteligentes e interfaz de humano-computadora multimodal (HCI) [60, 49].

A través del tiempo, el estudio de las emociones ha llamado la atención en las ciencias biológicas y sociales. El estudio de las emociones ha establecido una hipótesis, llamada *hipótesis de la universalidad* [19]. La hipótesis de la universalidad afirma que todos los humanos comunican estados emocionales internos básicos. Se ha observado que estos estados emocionales usan los mismos movimientos faciales en virtud de sus orígenes biológicos y evolutivos. También establece que en cada cultura, se construyen modelos mentales que forman seis grupos distintos, uno por cada emoción básica. Esto es debido a que por cada emoción que se expresa, se utiliza una combinación específica de movimientos faciales comunes a todos los humanos. Así los modelos mentales construyen una representación que mide la intensidad emocional en todas las culturas [35].

Mediante un estudio basado en el análisis de culturas Ekman y Friesen definieron seis emociones básicas: felicidad, sorpresa, miedo, disgusto, enojo y tristeza, e indicaron que los humanos perciben emociones de la misma manera, independientemente de la cultura [18]. Aunque se determinaron en un contexto general seis emociones básicas, a través de la investigación, se han ido añadiendo un grupo de emociones compuestas [27]. Las emociones compuestas se forman a partir de la combinación de las emociones básicas.

El reconocimiento de emociones es una tarea compleja que se ha tratado de abordar con enfoques de aprendizaje automático (*Machine Learning*, ML) y de aprendizaje profundo (*Deep Learning*, DL). El aprendizaje automático (ML) permite a las computadoras la habilidad de aprender sin ser explícitamente programadas, siendo presentados muchos ejemplos relevantes sobre una tarea específica, para posteriormente, construir modelos capaces de hacer predicciones sobre nuevos ejemplos [22]. En los modelos de ML, se aplican técnicas de aprendizaje global y local. El aprendizaje global [1] contiene las fases de entrenamiento y prueba. En la fase de entrenamiento se construyen modelos utilizando todo su conjunto de datos de entrenamiento para generar un modelo que en la fase de prueba sea capaz de generalizar a instancias nunca antes vistas.

En contraste, el aprendizaje local hace predicciones a partir de un subconjunto de los datos de entrenamiento, creando un modelo aproximado a la instancia de consulta. El aprendizaje local está constituido de tres diferentes enfoques y son: las representaciones locales, la selección local y el aprendizaje localmente

[1] Para explicaciones posteriores se establece el término de *aprendizaje global* al aprendizaje automático computacional.

ponderado. Una representación local implica que cada nuevo punto de datos afecta a un pequeño subconjunto de parámetros y el responder una consulta también implicaría un pequeño subconjunto de parámetros. Algunos ejemplos de representaciones locales son las tablas de búsqueda y clasificadores basados en ejemplos o prototipos [5].
La selección local se refiere a métodos que almacenan todos (o la mayoría) de los datos de entrenamiento en memoria y usan una función de distancia para determinar qué puntos almacenados son relevantes para la consulta. La función de la selección local es ubicar una única salida usando el vecino más cercano o usando un esquema de votación basado en la distancia. El aprendizaje localmente ponderado almacena explícitamente los datos de entrenamiento (al igual que los enfoques de selección local) y solo ajusta los parámetros a los datos de entrenamiento cuando se conoce una consulta. La característica crítica del aprendizaje localmente ponderado es que se utiliza un criterio de ponderación local con respecto a la ubicación de la consulta para ajustar algún tipo de modelo paramétrico a los datos. Aquí surge la confusión de las estructuras de modelos aparentemente globales (por ejemplo, redes neuronales sigmoidales multicapa, o las redes de función de base radial) se llaman modelos locales debido al criterio de entrenamiento que establece. El criterio establece que todos los datos pueden participar en la construcción del modelo local, siempre que los datos distantes importen menos que los datos cercanos. Por lo tanto, existen enfoques y representaciones globales que se pueden transformar en enfoques ponderados localmente utilizando un criterio de entrenamiento local [4].

Los métodos de LWL son flexibles e interpretables y tienen una configuración de parámetros simple que mejora el rendimiento en la predicción. Además, pueden representar funciones no lineales con la ventaja de tener reglas simples en su entrenamiento como: el control de ajuste de parámetros, el suavizado, el rechazo de valores atípicos, entre otros. El proceso de modelado es fácil de entender y ajustar, debido a que se construye con puntos relacionados al punto de consulta. Una desventaja del método es que puede fallar en su generalización cuando se presenta una alta dimensionalidad en el espacio latente [51].

Retomando el concepto del reconocimiento de emociones en imágenes con expresiones faciales usando modelos de ML y DL están enfocados en detectar y analizar regiones de la cara. Los modelos se encargan de extraer características geométricas, de apariencia o un híbrido para ser procesadas por algún algoritmo de clasificación de ML como se ilustra en la Fig. 1. Algunos algoritmos usados para la extracción de estas ca-

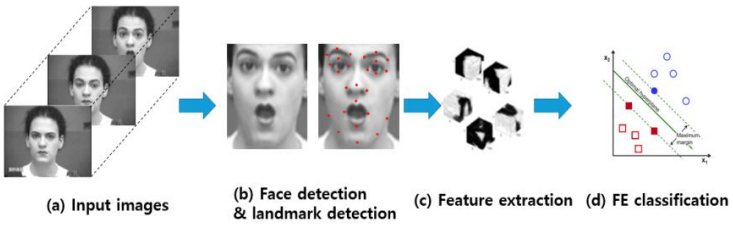


Figura 1: Enfoque convencional de ML usado en el reconocimiento de emociones en imágenes de expresiones faciales. A partir de imágenes de entrada (a), se detecta la región de cara y puntos de referencia faciales (b), para extraen de los componentes de la de las características espaciales y temporales de la cara (c) para utilizar algoritmos de clasificación. Figura reproducida de [39]

racterísticas son: *Histogram of Oriented Gradients (HoG)*, *local binary pattern (LBP)*, relación de distancia y ángulo entre puntos de referencia en la cara (*Facial Landmarks*). Los clasificadores usados son: *Support Vector Machine (SVM)*, *Random Forest (RF)* and *Multilayer Perceptron (MLP)* [39]. Estos algoritmos están

basados en el aprendizaje global. El aprendizaje local no ha sido ampliamente explorado en el contexto de ER. Una ventaja de los enfoques convencionales de ML es que requieren una capacidad de computo y memoria relativamente bajos en comparación con los enfoques basados en el aprendizaje profundo (*Deep Learning, DL*). Sin embargo, la extracción de características y los clasificadores deben ser diseñados por el programador y no pueden optimizarse en conjunto para mejorar el rendimiento del clasificador.

El aprendizaje profundo (*Deep Learning, DL*) se distingue por aprender característica de los datos a través de múltiples capas de abstracción. Los datos sin procesar se ingresan en el nivel inferior y la salida deseada se produce en el nivel superior. El resultado del aprendizaje se obtiene través de muchos niveles de datos transformados. El aprendizaje profundo es jerárquico en el sentido que en cada capa, el algoritmo extrae automáticamente características visuales desde niveles inferiores para ser procesadas en niveles más profundos como se muestra en la Fig. 2.

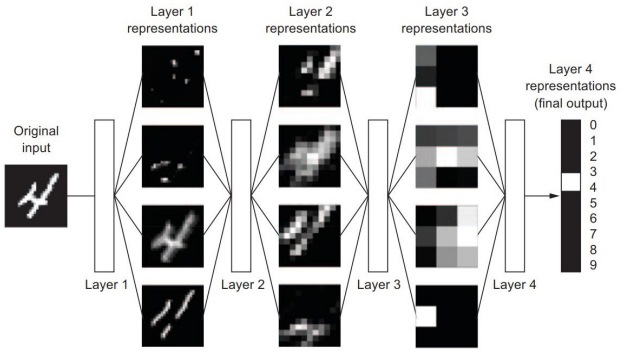


Figura 2: Representación de una red de aprendizaje profundo para la clasificación de dígitos. Figura reproducida de [21].

Actualmente, ER se ha abordado con enfoques de DL como: *Convolutional Neural Netowrks (CNNs)*, *Deep Neural Netowrks (DNNs)*, *Recurrent Neural Networks (RNNs)* y *Long short-term memory (LSTM)* o métodos híbridos [39]. Estos enfoques basados en el aprendizaje profundo se han utilizado para la extracción de características y en tareas de clasificación y regresión. Una de las ventajas de los enfoques de DL es que reducen la dependencia de modelos basados en la física o el uso de técnicas de pre-procesamiento. Esto se lleva a cabo al permitir el aprendizaje de extremo-a-extremo directamente desde los datos de entrada [75] (como se ilustra en la Fig 3).

Las técnicas de aprendizaje en un modelo de DL como las CNNs utilizan un método de aprendizaje global, es decir, que su modelo de entrenamiento se construye utilizando todo el conjunto de datos de entrenamiento. Actualmente, existen métodos que tratan de adaptar el aprendizaje local en una CNN con el uso de redes de función de base radial RBF o el de algoritmos basados en instancias como los k-vecinos más cercanos. Estas adaptaciones en su mayoría consiste en el apilamiento de arquitecturas que incluyen el aprendizaje local en combinación con CNNs, haciendo el aprendizaje independiente uno de otro.

Un modelo que integre el aprendizaje local de extremo-a-extremo sobre un enfoque del aprendizaje profundo puede tener un mejor desempeño en comparación con los modelos de DL basados en aprendizaje global. Además de ER, se espera que el modelo tenga un buen desempeño en dominios apropiados donde el aprendizaje localmente ponderado alcance resultados sobresalientes. Los dominios dentro del alcance

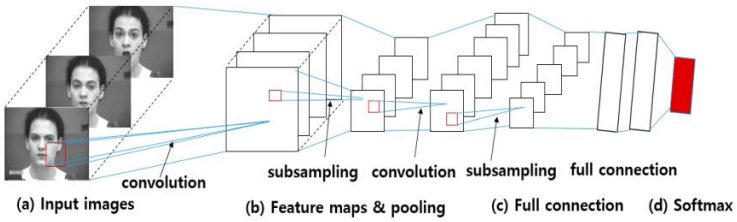


Figura 3: Enfoque convencional de DL usado en el reconocimiento de emociones en imágenes de expresiones faciales. A partir de imágenes de entrada (a), se aplica filtros que convolucionan a través de la imagen para construir mapas de características (b), que son mapeados a una resolución espacial demás baja dimensión para conectarse a capas de redes neuronales completamente conectadas detrás de las capas convolucionales (c), y se reconoce una sola expresión facial en función de la salida de softmax (d). Figura reproducida de [39]

identificados se refieren a la clasificación de ejemplos de clase minoritaria, con traslape de clases, datos con ruido y grano fino[2]. Esto debido a que el modelo de aprendizaje local permite ajustar la capacidad del algoritmo de aprendizaje a las propiedades locales de los datos.

En esta propuesta de investigación doctoral, se propone el desarrollo de un esquema de aprendizaje profundo que integre una técnica de aprendizaje local de extremo a extremo llamado *Locally Weigted Deep Learning, LWDL*. El esquema LWDL podría mejorar el desempeño en ER o también en algunos dominios dentro del alcance de LWL, debido a que la localidad favorecería en tareas que impliquen separar características donde la diferencia de los atributos entre clases sea muy sutil. El modelo local se encargará de agrupar aquellas instancias relacionadas entre sí mediante una métrica de distancia. La contribución principal de la investigación se orienta al desarrollo del esquema *LWDL* como una mejora en los modelos actuales del aprendizaje profundo para ER y dominios dentro del alcance de LWL, que basan su aprendizaje en métodos de aprendizaje global.

2. Marco Teórico

En esta sección se presentan los temas necesarios para el desarrollo de la propuesta de investigación doctoral relacionados al reconocimiento de emociones, el aprendizaje profundo y el aprendizaje localmente ponderado.

2.1. Reconocimiento de Emociones

Las emociones son estados de sentimiento con valencia afectiva negativa o positiva [54]. Juegan un papel importante en la vida humana y las interacciones sociales de cada persona, ya que constituyen una parte importante en la percepción y cognición humana [33]. Las investigaciones neurológicas y el estudio de las funciones utilitarias dentro del cerebro humano muestran una relación evidente entre las emociones humanas y la toma racional de decisiones [59, 64]. El reconocimiento automático de emociones se ha abordado usando diferentes modalidades que incluyen: voz, texto, signos vitales (EGG), reconocimiento de gestos, expresiones faciales e híbridos como se muestra en la Fig 4. Una de las formas más usadas es mediante el

[2]Nótese que inicialmente consideramos estos dominios dentro del alcance de la propuesta, sin embargo, una contribución del presente trabajo será la identificación de aquellos dominios en que el enfoque de LWDL puede tener mayor impacto.

análisis de las expresiones faciales. Las expresiones faciales pueden ser capturadas de forma simple mediante una cámara para su análisis en modelos computacionales y su etiquetado se puede llevar a cabo por expertos simplemente con observar la imagen. Las bases de datos de expresiones faciales que se usan para el reconocimiento de emociones contienen imágenes de emociones *planteadas* o *aparentes*. Para su construcción se pide a un grupo de participantes que expresen diferentes estados emocionales básicos. Una base de datos de expresiones espontáneas se dice que las expresiones son naturales. La diferencia radica en que las expresiones espontáneas difieren notablemente de las aparentes en términos de: intensidad, configuración y duración. En la mayoría de los casos, las expresiones aparente son exageradas, mientras que las espontáneas son sutiles y difieren en apariencia.

Si bien el reconocer emociones mediante signos vitales es una de las técnicas más acertadas, el medir estados emocionales involucra obtener señales vitales como: la presión arterial, la respiración, los electroencefalogramas, y los electrocardiogramas. La desventaja de este enfoque es que se requiere el uso de sensores físicos y usuarios con experiencia para el manejo de los equipos [33], limita la movilidad de los participantes y distrae las reacciones emocionales de la persona.

Figura 4: Diferentes fuentes para el reconocimiento de emociones.

2.1.1. Reconocimiento de Expresiones Faciales

La expresión facial es una de las características más importantes en el reconocimiento de las emociones humanas. Una expresión facial consiste en una secuencia de señales no verbales en la comunicación e interacción entre humanos [34] para comunicar una emoción. Una expresión facial implica la contracción de músculos en la cara y se puede reconocer a partir de imágenes estáticas o una secuencia de imágenes o vídeos [33]. El humano puede asumir la emoción de alguien con el sólo hecho de observar su rostro. El reconocimiento de expresiones faciales tiene varias aplicaciones como en la visión por computadora, el comportamiento humano no verbal y la interacción humano-computadora [36].

El objetivo del reconocimiento de expresiones faciales *(Facial Expression Recognition, FER)* es gene-

ralmente categorizar la expresión facial en diferentes clases, para distinguir entre distintos gestos faciales e interpretar incluso estados mentales [33]. Las expresiones faciales se deben a deformaciones temporales de los elementos faciales como: la boca, las cejas, los ojos y la nariz. El grado de cambios en todas las regiones faciales determina indirectamente la intensidad de la emoción. Existe un sistema que codifica las expresiones llamado Sistema de codificación de acción facial (FACS) [17]. FACS segmenta los efectos visibles de la activación muscular facial en unidades de acción (*Action Units, AU*) [29] y se utilizan particularmente un conjunto de 46 unidades de acción principales con respecto a su intensidad y ubicación. Esas unidades de acción codifican las acciones fundamentales de los músculos individuales o grupos de músculos que se ven involucrados cuando esta presente una expresiones faciales de una emoción en particular. FER se lleva

Figura 5: Ejemplos de la base de datos CK +. En la figura se presentan 8 emociones que expresan diferentes AU. Las emociones que se representan son: disgusto, felicidad, sorpresa, miedo, enojo, desprecio, tristeza y neutralidad. Figura reproducida de [47].

a cabo en aplicaciones como estudios psicológicos, animaciones faciales, ciencia cognitiva, neurociencia, comprensión de imágenes, videojuegos, robótica, dispositivos de visión por computadora y aprendizaje automático [14].

FER se ha abordado mediante el uso de modelos de aprendizaje profundo, alcanzado resultados sobresalientes en comparación con los métodos convencionales de ML, en la sección 3.1 se presenta una revisión del estado del arte en FER.

2.2. Aprendizaje Profundo

El aprendizaje profundo (*Deep Learning, DL*) es un sub-campo dentro del aprendizaje automático (ML) que aprende modelos en múltiples niveles de representación y abstracción a partir de los datos de entrada como imágenes, sonido y texto. Históricamente el concepto de aprendizaje profundo se originó a partir de la investigación de las redes neuronales artificiales [7]. La familia de métodos de aprendizaje profundo se ha vuelto cada vez más extensos abarcando también modelos probabilísticos jerárquicos y una variedad de algoritmos de aprendizaje supervisado y no supervisado [15].

Un ejemplo de modelo de aprendizaje profundo son las redes neuronales profundas *Deep neural network, (DNN)* (Ver Fig. 6). Se pueden definir como un perceptrón multicapa que consiste en una red neuronal artificial (*Artificial Neural Network, ANN*) formada por múltiples capas, de tal manera que tiene capacidad para resolver problemas que no son linealmente separables. Generalmente, este tipo de red consiste en una capa de entrada, una capa oculta y una capa de salida (cuando se tiene una configuración simple y este no se considera parte de una red profunda). Cada capa esta compuesta por neuronas que son conectadas entre las capas y se encargan de transferir los pesos a través de la red.

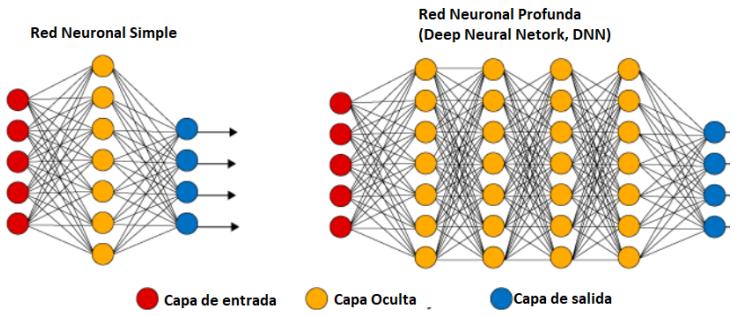


Figura 6: Representación de una arquitectura de red neuronal artificial.

Una arquitectura de aprendizaje profundo consiste en una representación de múltiples capas que aplican funciones de activación para realizar transformaciones no lineales de las entradas que se puede describir de la siguiente manera:

$$f_l^{W,b} = f_l\left(\sum_{j=1}^{N_l} W_{lj}X_j + b_l\right) = f_l\left(W_l X_l + b_l\right), l \leq l \leq L \tag{1}$$

Donde el número de unidades ocultas esta dato por N_l. El predictor se encarga de modelar un mapeo de alta dimensión F a través de la composición de funciones y se puede definir como:

$$Y(X) = F(X) = \left(f_1^{W_1,b_1} \circ \cdots \circ f_L^{W_L,b_L}\right) \tag{2}$$

La salida final es la respuesta de Y y puede ser categórica o numérica. La estructura explícita de una regla de predicción profunda es entonces:

$$\begin{gathered} Z^{(1)} = f^{(1)}\left(W^{(0)}X + b^{(0)}\right), \\ Z^{(2)} = f^{(2)}\left(W^{(1)}Z^{(1)} + b^{(1)}\right), \\ \vdots \\ Z^{(L)} = f^{(L)}\left(W^{(L-1)}Z^{(L-1)} + b^{(L-1)}\right), \\ Y(X) = W^{(L)}Z^{(L)} + b^{(1)} \end{gathered} \tag{3}$$

Aquí, se define como: $Z^{(L)}$ la L-ésima capa, $W^{(L)}$ la matriz de pesos y $b^{(L)}$ el sesgo. $Z^{(L)}$ contiene las características ocultas extraídas, dicho de otra manera, el enfoque profundo emplea predictores jerárquicos que comprenden una serie de transformaciones no lineales en L aplicadas a X. Cada una de las transformaciones L se refiere a una capa donde la entrada original es X, la salida de la primera transformación es la primera capa, y así sucesivamente hasta la salida Y como la capa $(L+1)$. Usamos $l \in \{1, \cdots, L\}$ para indexar las capas que se denominan capas ocultas. El número de capas L representa la profundidad de la arquitectura profunda.

A lo largo de la investigación, han surgido enfoques notables en DL como las *Convolutional Neural Network, (CNNs)*, *Recurrent Neural Network, (RNN)* (incluyendo *Long Short-Term Memory, (LSTM)* y *Gated Recurrent Units, (GRU)*), *Auto-Encoder, (AE)*, *Deep Belief Network, (DBN)*, *Generative Adversarial Network, (GAN)*, and *Deep Reinforcement Learning, (DRL)* [2]. DL tiende a generalizar mejor cuando se tiene

una gran cantidad de datos para entrenar y por lo tanto son modelos más complejos que requieren más recursos en hardware a diferencia de los modelos tradicionales de ML.

DL ha logrado una importancia excepcional en la comunidad científica debido a la aplicabilidad a casi cualquier dominio. Siendo capaz de resolver tareas asociadas con el campo del procesamiento de imágenes, visión por computadora, reconocimiento de voz, procesamiento del lenguaje natural, traducción automática, arte, imágenes médicas, procesamiento de información médica, robótica y ciberseguridad todos con notables resultados.

2.3. Componentes principales de una red de aprendizaje profundo.

Los componentes principales que se usan en redes de aprendizaje profundo son:

- Ajuste de capas. Las capas son una unidad fundamental en las redes profundas que van cambiando dependiendo del tipo de función de activación que use.
- Funciones de activación. Las funciones de activación son una función limitadora o umbral que modifica el valor de la salida de la neurona, poniendo un límite en el valor del cual no debe sobrepasar antes de propagarse a otra. Las funciones comúnmente usadas en DL son:
  - Función Sigmoid
  - Función Tanh
  - ReLU (*Rectified linear unit*) y sus variantes.
- Funciones de pérdida. Las funciones de pérdida cuantifican la salida predicha (o etiqueta) contra la salida real. Se utilizan funciones de pérdida para determinar la penalización por una clasificación incorrecta de un dato de entrada, algunas son:
  - Mean Squared Error Loss
  - Cross-Entropy Loss
  - Hinge loss
- Métodos de optimización. El entrenamiento de un modelo en aprendizaje automático implica encontrar el mejor conjunto de valores para el vector de parámetros como los valores de función de pérdida más bajo. El aprendizaje automático se puede ver como un problema de optimización, en el que se minimiza la función de pérdida con respecto a los parámetros de la función de predicción (según nuestro modelo). Algunos son:
  - *Adam*
  - *Gradient descent*
  - *Stochastic gradient descent*
  - *RMSprop*
- Ajuste de hiper-parámetros. Un hiper-parámetro se refiere a elegir libremente por el usuario algunas configuraciones que podrían mejorar el rendimiento. Los hiper-parámetros se dividen en varias categorías:

- Número de capas y neuronas.
- Magnitud (*momentum, learning rate*).
- Regularización. La regularización es una medida tomada contra el sobreajuste. Los modelos sobreajustados no tienen capacidad de predecir datos que no hayan visto antes, únicamente describe bien el conjunto de entrenamiento. La regularización ayuda a modificar el gradiente para que no se interponga en direcciones que lo lleven a un sobreajuste y algunos incluyen:
  - *Dropout*. Es un mecanismo utilizado para mejorar el entrenamiento de las redes neuronales al omitir una unidad oculta, permitiendo el aceleramiento del entrenamiento. *Dropout* es impulsado por la desactivación aleatoria de una neurona para que no contribuya al avance y la retro-propagación, es decir, toma un subconjunto de neuronas seleccionadas al azar y se establece en cero dentro de cada capa.
  - *Drop connect*. Hace lo mismo que *Dropout*, pero en lugar de elegir una unidad oculta, desactiva la conexión entre dos neuronas, estableciendo en su lugar un subconjunto de pesos seleccionado al azar dentro de la red a cero.
  - *L1 y L2 Penalty*. Los métodos de penalización L1 y L2, por el contrario, son una forma de evitar que el espacio de parámetros de la red neuronal sea demasiado grande en una dirección, haciendo pesos grandes más pequeños.
- Estrategias de inicialización de pesos.
- Definir número de épocas.
- Normalización de los datos de entrada.

Un enfoque de DL ampliamente usado en el procesamiento y clasificación de imágenes son las Redes Neuronales Convolucionales (CNNs). Las CNNs tienen la capacidad de funcionar como extractores automáticos de características en imágenes y han mostrado resultados superiores en tareas de clasificación de imágenes. En la siguiente sección 2.3.1, se explica el funcionamiento de las CNNs.

2.3.1. Redes Neuronales Convolucionales

Las Redes Neuronales Convolucionales (*Convolutional Neural Networks, CNNs*) son un tipo de red neuronal que aplica la operación matemática de convolución en sus capas iniciales de la red. Las CNNs constan de varias capas que permiten un aprendizaje automático de las características ya que su entrada son datos como imágenes. La imagen de entrada convoluciona a través de filtros, para producir características apropiadas y sean utilizadas por capas posteriores de la red para conducir a la etapa de clasificación [41]. La red se puede estructurar en dos componentes principales y son: *Extracción de características y Clasificación*.

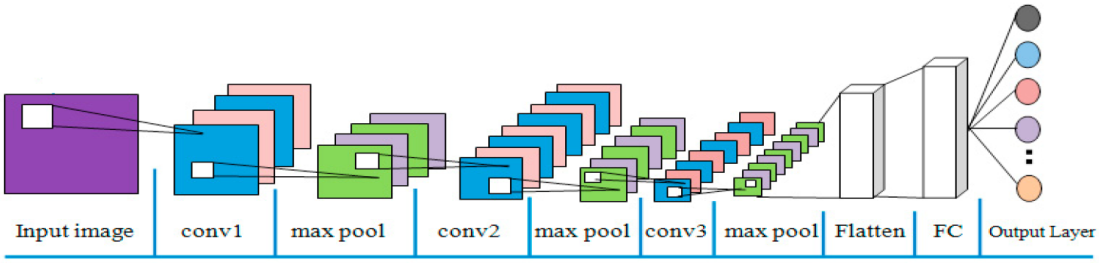


Figura 7: Modelo de una Red Neuronal Convolucional. Figura reproducida de [65].

Extracción de características

La capa convolucional implica la operación de convolución se le conoce como el detector de características de una CNN. La entrada a una capa convolucional son datos sin pre-procesar, por ejemplo imágenes y genera una salida de imágenes *mapa de características* que se usan como entrada a otra capa convolucional. Generalmente se interpreta como un filtro donde el núcleo filtra datos de entrada para cierto tipo de información, siendo capaz de analizar información acerca de la posición del objeto, la invarianza a rotaciones, el análisis de bordes y las texturas en la imagen. La operación matemática convolución en una imagen se denota como:

$$S(i,j) = (I * K)(i,j) = \sum_m \sum_n I(m,n)K(i-m, j-n) \tag{4}$$

Se toma como entrada una imagen I, se aplica un kernel de convolución K sobre cada píxel en la posición i, j en la imagen y nos da un mapa de características de la imagen como salida S.

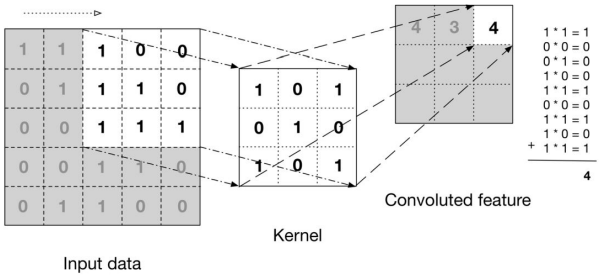


Figura 8: Ejemplo de la operación matemática de convolución sobre una imagen. Figura reproducida de [57].

Otro componente en la capa convolucional son las funciones de activación. En el caso de una CNN, comúnmente se usa la función de activación llamada unidad lineal rectificada o ReLu (*Rectified Linear Unit*). Esta función de activación calcula la salida como se muestra en la ecuación 5, la función calcula si la entrada esta por debajo de cero, la salida es cero.

$$R(x) = max(0, x) \tag{5}$$

Además de la operación de convolución en la fase de extracción de características, se usa una capa llamada *pooling layer*. La capa *Pooling* se inserta entre las capas convolucionales para reducir progresivamente el tamaño espacial (ancho y alto) de la representación de los datos y poder controlar el sobreajuste. La operación más usada en esta capa es Max-Pooling (como se muestra gráficamente en la Fig.9). Esta operación implica

la agrupación de un vecindario rectangular y su salida toma el valor máximo del píxel dentro del vecindario. Otras funciones de agrupación incluyen el promedio de un vecindario rectangular (Average-Pooling), la norma L2 de un vecindario rectangular y un promedio ponderado basado en la distancia desde el píxel central.

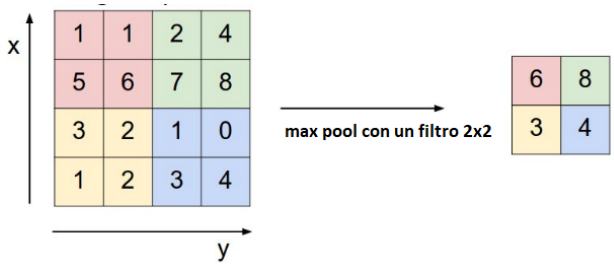


Figura 9: Ejemplo de la operación ***Max Pooling***. Figura reproducida de [65]

Fase de clasificación

La fase de clasificación en una CNN es llamada capa totalmente conectada. La capa se comporta al igual que las redes neuronales, todas las neuronas de la capa están conectadas con cada neurona de la capa anterior y se puede calcular como:

$$F(x) = \sigma(W * x) \tag{6}$$

Donde F es la salida de las unidades, $W \in \Re$ son los pesos de la red, y $\sigma : \Re \to \Re$ es la función de activación de la red. La capa final generalmente es la capa en la que el error se puede propagar usando el algoritmo de retropogación *Back Propagation* y el desempeño de la red se incrementa. Aquí la red generalmente usa la función *softmax*, donde la salida se calcula como sigue:

$$S(x)_j = \frac{x^{x_i}}{\sum_{i=0}^{N} e^{x_i}} \tag{7}$$

$S(x) : R \to [0, 1]N$, donde N es el tamaño del vector de entrada. Para $1 \leq j \leq N$. La capa de salida de una CNN tiene un tamaño igual al número de clases.

2.4. Aprendizaje Localmente Ponderado

En ML un enfoque que adapta algoritmos de aprendizaje local es el aprendizaje localmente ponderado (*Locally Weighted Learning, LWL*). LWL es una técnica de aproximación de funciones donde se realiza una predicción mediante el uso de un modelo local aproximado en torno al punto de interés. Los modelos locales adaptan modelos que para cada punto de interés, se va creando un modelo basado en las vecindades del punto. Es decir, a cada punto de los datos, se calcula un factor de ponderación que expresa la influencia entre sí de los datos para la predicción. En general, los puntos de los datos que están cerca del punto de consulta actual, reciben un peso mayor que los puntos de datos que están distantes [5], como se muestra en la Fig. 10.

Hay adaptaciones de LWL que implican la combinación de métodos de aprendizaje local y global, creando clasificadores como se muestran en la Tabla 1.

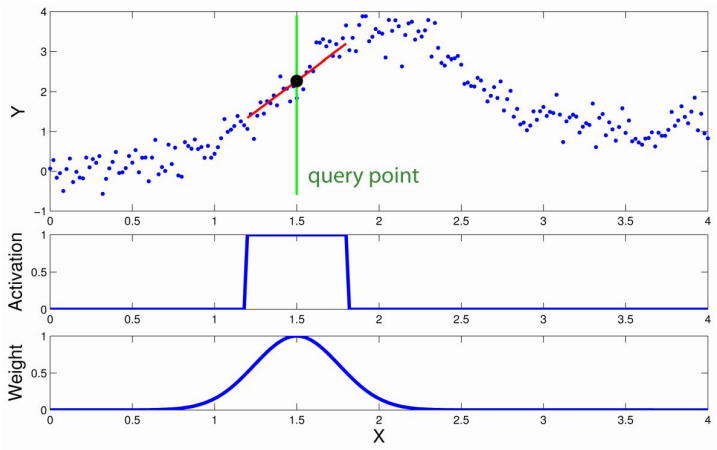


Figura 10: Ejemplo de Regresión Localmente Ponderada ***Locally Weighted Regression, (LWR)***, en la gráfica los puntos azules representan el conjunto de datos del entrenamiento (x, y) y los modelos lineales locales (líneas rojas). Figura reproducida de [20].

Modelos globales		Modelos locales e híbridos	
Suport Vector Machine	SVM	Support Vector Machine - kNN	SVM-kNN
Multi Layer Perceptron	MLP	Radia Basis Function Networks	RBF
Decision Tree	DT	Decision Tree - kNN	DT-kNN
Naive Bayes	NB	Naive Bayes - kNN	NB-kNN
Linear Regressor	LR	Locally Weighted Lineal Regressor	LWLR
		K-Nearest Neighbors	KNN
		Learning vector quantization	LVQ

Tabla 1: Clasificadores en aprendizaje automático basados en aprendizaje local, global e híbridos.

Existen algunos modelos de LWL que son no-paramétricos y la predicción actual se realiza mediante funciones locales que utilizan sólo un subconjunto de datos. Los modelos paramétricos aprenden una función que aproxima los datos de entrenamiento a la variable objetivo por un vector de parámetros cuyo tamaño es finito y fijado antes de observar cualquier dato. En los modelos no-paramétricos, la complejidad de su espacio de hipótesis crece según lo hace el número de instancias de datos a considerar. Por ejemplo, el algoritmo kNN hace que su complejidad sea una función del tamaño del conjunto de entrenamiento. El objetivo detrás de LWL es que en lugar de construir un modelo global para todo el espacio de instancias, para cada punto de consulta se construya un modelo local basado en datos vecinos al punto de consulta.

Una característica atractiva del LWL es que los modelos son interpretables. El proceso de modelado es fácil de entender y por lo tanto, fácil de ajustar o controlar algunos parámetros de entrenamiento en el clasificador. Hay dos categorías principales en las que puede dividir los métodos de LWL. La primera categoría incluye los métodos LWL basados en memoria donde todos los datos de entrenamiento se guardan en la memoria para hacer su predicción, por ejemplo los algoritmos: *k-Nearest Neighbor, Weighted Average, y Locally Weighted Regression*. La segunda categoría incluye métodos LWL incrementales que no necesitan recordar ningún dato explícitamente, por ejemplo las redes tipo RBF (*Radial Basis Functions Networks, RBF*) [4].

Las desventajas de estos métodos radica en que son sensibles cuando se presenta una alta dimensionalidad

en los datos, también tienen problemas para generalizar adecuadamente cuando se tiene un conjunto de datos extenso (para los métodos basados en memoria). En el caso de los métodos LWL incrementales, es complicado establecer parámetros iniciales en la configuración de los algoritmos, aunque una ventaja de los métodos incrementales es que pueden ser adaptados en tareas de clasificación donde el conjunto de datos es extenso y es capaz de generalizar adecuadamente [13].

2.4.1. Método de los *k* vecinos más cercanos

El método de los *k* vecinos más cercanos es un tipo de aprendizaje basado en instancias donde el algoritmo supone que todas las instancias corresponden a puntos en el espacio n-dimensional $\Re^n$, y la función objetivo se aproxima localmente [50]. Comúnmente, el vecino más cercano de una instancia se define en términos de la distancia euclidiana estándar, más precisamente, una instancia se puede describir como un vector de características en la forma:

$$\langle a_1(x), a_2(x) \cdots a_n(x) \rangle \tag{8}$$

Donde denotamos $a_n(x)$ como el valor del atributo n-ésimo de instancia x. Por lo tanto, la distancia entre dos instancias $d(x_i, x_j)$ se puede definir de la siguiente manera:

$$d(x_i, x_j) = \sqrt{\sum_{r=1}^{n} (a_r(x_i) - a_r(x_j))^2} \tag{9}$$

La función objetivo $f : \Re^n \rightarrow V$ del vecino más cercano se puede calcular como una valor discreto o un valor real, para un conjunto finito $V = v_1, \cdots, v_s$. El algoritmo del k vecino más cercano toma cada ejemplo del conjunto de prueba x_q y calcula los k ejemplos más cercanos del conjunto de entrenamiento. Por ejemplo, si $k = 1$ entonces el algoritmo retorna el 1-vecino más cercano y asigna a $f(x_q)$ el valor de $f(x_i)$ donde x_i es la instancia de entrenamiento más cercana a x_q. Cuando se tienen valores grandes de k, el algoritmo asigna a x_q al valor más común en el conjunto de los k más cercanos.
Las ventaja del algoritmo es que es fácil de interpretar sus resultados. Es insensible a los valores atípicos es decir, la precisión puede verse afectada por el ruido o las características irrelevantes. Las desventaja del algoritmo es que es un método basado en instancias, ya que no aprende explícitamente un modelo, en su lugar memoriza las instancias de entrenamiento, para ser posteriormente usadas como conocimiento en la fase de predicción.

2.4.2. Redes de Función de Base Radial

Las redes de Función de Base Radial, (*Radial Basis Function, RBF*), o Redes RBF son un tipo de red neuronal artificial construida a partir de funciones de kernel espacialmente localizadas. Las redes RBF pueden describirse como una combinación de los enfoques LWL (donde se hace una aproximación local en el momento de la consulta) y redes neuronales (donde se forma una aproximación global a la función objetivo en el momento del entrenamiento) [61]. Por lo tanto en las Redes RBF su modelo de aprendizaje supervisado se realiza bajo el concepto de aproximación local.
La arquitectura de una Red RFB es simple y esta compuesta de una capa de entrada, una capa oculta (en esta capa se definen las funciones RBF) y una capa de salida. En la Fig 11 se muestra la arquitectura típica de una Red RBF.

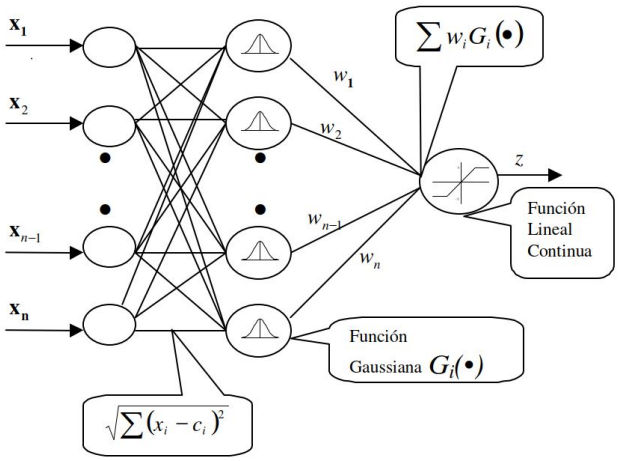


Figura 11: Arquitectura de Red de Función de Base Radial (RBFN). Consiste en un vector de entrada, una capa de neuronas RBF y una capa de salida. Figura reproducida de [13].

Su funcionamiento básicamente consiste en que la capa de entrada transmite los ejemplos o patrones de entrenamiento y prueba hacia las capas ocultas. Es decir, el número de unidades de entrada es exactamente igual a la dimensionalidad d de los datos. Los cálculos en la capa oculta están basados sobre comparaciones entre vectores prototipos. Los vectores prototipo se obtienen a partir de un agrupamiento previo sobre el conjunto de datos de entrada, tomando los centros del agrupamiento como los vectores prototipo. Cada capa oculta contiene d-dimensional vector prototipo. Para la i-ésima unidad oculta el vector prototipo es denotado por μ_i. Además, la i-ésima unidad oculta contiene un ancho de banda denotado por σ_i. Aunque los vectores prototipo son siempre específicos para unidades particulares, los anchos de banda de diferentes unidades σ_i a menudo se establecen en el mismo valor σ. Los vectores prototipo y los anchos de banda generalmente se aprenden de manera no supervisada o con el uso de una supervisión moderada. Entonces, para cualquier punto de entrada en el conjunto de entrenamiento X, la activación $\phi_i(X)$ de la i-ésima unidad oculta se define de la siguiente manera:

$$h_i = \phi_i(\overline{X}) = e^{\left(-\frac{\|\overline{X}-\overline{\mu}_i\|^2}{2\cdot\sigma i^2}\right)} \forall \in \{1, \cdots, m\} \tag{10}$$

El número total de unidades ocultas se denota por m. Cada una de estas unidades m está diseñada para tener un alto nivel de influencia con puntos cercanos a su vector prototipo. Por lo tanto, se puede ver a m como un número de grupos utilizados o centroides para modelar las unidades RBF. Para entradas de baja dimensión, es típico que el valor de m sea mayor que la dimensionalidad de entrada d, pero menor que el número de puntos de entrenamiento n.
Los pesos de las conexiones únicamente existen de los nodos ocultos a los nodos de salida y se establecen en w_i. Luego, la predicción $\overline{y}$ de la red RBF en la capa de salida se define de la siguiente manera:

$$\overline{y} = \sum_{i=1}^{m} w_i h_i = \sum_{i=1}^{m} w_i \phi_i(\overline{X}) = \sum_{i=1}^{m} w_i e^{\left(-\frac{\|\overline{X}-\overline{\mu}_i\|^2}{2\cdot\sigma i^2}\right)} \tag{11}$$

Una vez que se obtiene el valor predicho y, entonces se puede configurar una función de pérdida, como por ejemplo, mínimos cuadrados. Los valores de los pesos $w_1, \cdots, w_m$ son aprendidos de forma supervisada.

Las ventajas una red RBF es que tiene un mejor desempeño cuando el volumen de datos de entrenamiento es grande. También este tipo de red se le reconoce como una red con alta eficiencia en la fase de entrenamiento. Ya que su aprendizaje es más rápido debido a que el cambio de peso sólo afecta a la neurona oculta asociada a dicho peso, es decir, sólo a un grupo de patrones pertenecientes a la clase que representa a dicha neurona oculta. A diferencia de las MLP, las redes RBF requieren una mayor cantidad de neuronas en los nodos ocultos para que la red tenga un mejor desempeño. Las redes RBF no son comúnmente utilizadas en aplicaciones que impliquen un alto volumen de patrones de entrenamiento.

2.4.3. Otros métodos de aprendizaje local

Existen otros métodos de aprendizaje local como son las redes SOMs, el algoritmo LVQ y el método de regresión lineal localmente ponderado. En esta sección se muestra una descripción de los métodos.

Un mapa auto-organizado (*Self organizing maps*, SOMs) es un tipo de red neuronal artificial (ANN) que se utiliza para reducir la dimensionalidad y se entrenan utilizando el aprendizaje no supervisado para producir una representación discreta del espacio de entrada que son las muestras de entrenamiento de baja dimensión (típicamente bidimensional) llamada mapa. Los mapas SOMs difieren de otras redes neuronales artificiales, ya que aplican el aprendizaje competitivo en oposición al aprendizaje de corrección de errores (como la propagación hacia atrás con descenso de gradiente) y en el sentido de que usan una función de vecindario para preservar las propiedades topológicas del espacio de entrada.
La red SOM generalmente consta de dos capas de nodos: la capa de entrada y de salida. A diferencia de otras redes neuronales, La red SOM en la capa de entrada los nodos de origen están directamente conectados a la capa de salida sin ninguna capa oculta [3]. Los nodos en la capa de entrada denotan los atributos (características).

Otro método estrechamente relacionado a las redes SOM es el aprendizaje basado en prototipos llamado *Learning Vector Quantization (LVQ)*. La diferencia radica en que SOM es un método de agrupamiento y aprendizaje no supervisados en cambio LVQ es aprendizaje supervisado. LVQ utilizan uno o más prototipos para representar cada clase en el conjunto de datos. A nuevos puntos de datos, se les asigna la clase del prototipo más cercano a ellos. Por lo general, se usa la métrica de la distancia euclidiana. No hay limitación sobre cuántos prototipos pueden existir por clase, pero este debe ser al menos 1 para cada clase. Este algoritmo contiene fase de entrenamiento y prueba.

Un método no paramétrico basado en el aprendizaje localmente ponderado es la regresión lineal localmente ponderada (LWLR). Este método a diferencia del modelo de aprendizaje global (Regresión Lineal) ajusta muchos modelos sobre de regresión lineal un punto de consulta en vez de ajustar una sola línea de regresión. La curva resultante final es el producto de todos esos modelos locales de regresión como se ilustra en la Fig.10.

3. Estado del Arte

En esta sección presentamos una revisión de la literatura sobre los trabajos actuales relacionados a la investigación. Se abordan áreas como el reconocimiento de emociones en imágenes, el aprendizaje localmente ponderado y la adaptación de métodos locales en el aprendizaje profundo.

3.1. Reconocimiento de Emociones en imágenes

Actualmente, el reconocimiento de emociones en imágenes ha atraído una atención creciente debido a la complejidad de la tarea. El reconocimiento de emociones se realiza con la adaptación de enfoques convencionales de ML o DL. Por un lado, la adaptación de enfoques convencionales de ML incluye tres componentes como: la detección facial, la extracción de características y la clasificación (como se explicó en la Sec.1 y se ilustró en la Fig.1). En la extracción de características se emplean algoritmos que incluyen: Histogramas de gradientes (HoG) [25], Patrones Binarios Locales (LBP) [30], entre otros [23, 24]. En los métodos de clasificación se usan algunos métodos como SVM [70], AdaBoost [23] y DT [76], por mencionar algunos.

ER resolviendo con DL usa enfoques como las CNNs [8], RNNs, LSTM [11] y métodos híbridos [16]. Por ejemplo, en el trabajo propuesto por [36] se presenta un modelo basado en redes neuronales convolucionales (CNNs) donde adaptan módulos residuales en sus capas convolucionales. La red se evalúa sobre los conjuntos de datos Ck+ y JAFFE alcanzando un 95.23 % y 93.24 % de exactitud respectivamente.

La combinación de redes CNN y LSTM se presenta en [67], donde el autor combina la red VGG-Face con dos redes neuronales recurrentes tipo LSTM. La red VGG-Face se usa para extraer descriptores de características en las imágenes. Posteriormente, los descriptores se adaptan como secuencias de entrada en cada una las redes LSTM. La clasificación se hace tomando las salidas de celdas LSTM como entradas para un clasificador softmax. En el trabajo [31] se presenta un modelo con dos arquitecturas CNN, una CNN binaria (B-CNN) y una CNN que tiene como objetivo clasificar 8 emociones (E-CNN). La red B-CNN se entrena para crear un modelo que clasifique en imágenes una escena como positiva o negativa. Los pesos obtenidos del modelo B-CNN, son usados para entrenar la siguiente red E-CNN. La red E-CNN se encarga de entrenar un modelo que es capaz de reconocer 8 tipos de emociones, usando los pesos de las capas convolucionales de la red B-CNN para generar el modelo E-CNN. Otro trabajo que propone el uso de arquitecturas tipo CNN pero se adapta un pre-procesamiento de la imagen es en [60]. El pre-procesamiento consiste en la detección de rostros, cambio de tamaño, adición de ruido y normalización de datos previamente al entrenamiento de la CNN. El autor reporta que se obtuvo una mejora en el FER al adaptar el pre-procesamiento. EL método es evaluado usando los conjuntos de datos CK+, JAFFE y MUG.

Para el FER en [68] el autor propone tres modelos de CNN: Light-CNN, dual-branch CNN y una CNN pre-entrenada. La red Light-CNN es una arquitectura que consiste en 6 módulos de convolución residuales. La red dual-branch CNN consta de tres módulos: dos módulos de ramificación CNN individuales y un módulo de fusión. La primera rama toma la imagen completa como entrada y extrae las características globales. La otra rama toma la imagen de la característica de textura preprocesada por LBP como entrada. Finalmente, el tercer módulo es una red de fusión que toma como entrada las características globales y de textura. La red pre-entrenada que se utiliza es ResNet101 y la red esta entrenada sobre el conjunto de datos ImageNet. Se aplica la técnica Fine-Tuning para entrenar algunas capas y realizar un ajuste fino en algunas capas para extraer características más especificas. La salida se ajusta de acuerdo con el número de categorías de las emociones. Los tres modelos se entrenan para reconocer 7 emociones y se hace una comparativa entre ellos. El autor concluye que un modelo pre-entrenado presenta una mejora de exactitud en el reconocimiento de emociones. EL método es evaluado con CK+, BU-3DFE y FER2013 alcanzando una exactitud de 85.71 %, 48.17 % y 54.64 % respectivamente. Los resultados de exactitud alcanzados con este modelo, no muestran una mejora en comparación con los modelos mencionados anteriormente. Además ningún modelo se evalúa sobre conjunto de datos de ER que contengan emociones compuestas.

Todos estos trabajos han mostrado que los enfoques de DL mejoran el desempeño en el reconocimiento de

emociones en comparación de los métodos convencionales de ML. Otro punto interesante es que los métodos usados en ER demostraron que una forma de obtener resultados sobresaliente es mediante la integración de modelos que se enfoquen en el análisis geométrico y visual de la cara. Sin embargo, en los trabajos revisados no se pone tanto énfasis en la etapa de clasificación, que es igualmente importante. En esta trabajo nos enfocamos en tratar de mejorar el aspecto predictivo del modelo, mediante clasificación localmente ponderada.

Existe un trabajo basado en el aprendizaje localmente ponderado (LWL) que se enfoca al reconocimiento de emociones y hace una comparativa con un método de aprendizaje global. El trabajo presentado en [73] hace una comparativa de exactitud en el reconocimiento de 7 emociones básicas. Los métodos que usa son los k-vecinos más cercanos y una red MLP. El autor llega a la conclusión que un modelo local se ajusta mejor que un método global. Pero el trabajo únicamente se evalúa sobre un conjunto de datos de un modelo 3D de la cara. El hecho de que un modelo local funcione bien, se puede ser debido a que sea más sensible a separar los puntos de referencia faciales (*facial landmarks*) sobre un modelo 3D de la cara ya que directamente realiza cálculos de distancias entre estos atributos y procesar datos en 3D, permite hacer un agrupamiento favorable de los *facial landmarks*. La información procesada es únicamente de la cara y no hay una gran cantidad de variación en la información visual.

3.2. Aprendizaje Localmente Ponderado

El aprendizaje ponderado localmente se pueden dividir en cuatro categorías que incluyen aprendizaje basado en distancias, centroides, modelos locales ponderados e híbridos de modelos globales y locales, como ilustra la Fig.12. Los algoritmos basados en distancias son algoritmos de aprendizaje automático que clasifican las ejemplos calculando las distancias entre ellos y una serie de ejemplos almacenados internamente. Los ejemplos más cercanos a la consulta tienen la mayor influencia en la clasificación asignada a la consulta. El aprendizaje basado en centroides se refiere a algoritmos de agrupación que son no supervisados. Este trata de encontrar un número fijo k de agrupaciones en un conjunto de datos basados en las similitudes en sus características. Los modelos híbridos consisten en entrenar modelos de aprendizaje global para un conjunto local de instancias relacionadas a la instancia de consulta. Existen trabajos que incluyen adaptaciones híbridas del aprendizaje local y global para mejorar las tareas de clasificación. Por ejemplo, la adaptación SVM-kNN resuelve tareas relacionadas con la clasificación de personalidad [62], clasificación de imagen [66], tareas de reconocimiento de escritura a mano [81] y tareas relacionadas con la clasificación de texto [79]. El método SVM-KNN consiste en que para cada instancia de consulta, se toma un conjunto de los k-vecinos más cercanos; el conjunto se usa para entrenar un clasificador SVM que genere un modelo local para realizar la predicción. Las contribuciones anteriores de aprendizaje ponderado localmente se centran en cómo las funciones del clasificador pueden aproximarse utilizando cualquier esquema de codificación local. En el trabajo presentado por [42] el autor propone un clasificador SVM localmente lineal con un límite de decisión suave y una curvatura limitada. El esquema toma localmente un conjunto de datos y crea una función de decisión entre los datos para demostrar que aunque el problema no es linealmente separable, localmente en regiones suficientemente pequeñas el límite de decisión es casi lineal. Por lo tanto, los datos se pueden separar razonablemente bien utilizando un clasificador localmente lineal. Otra adaptación al aprendizaje local en SVM es la integración de un kernel no-lineal mediante el producto de un kernel local y un kernel global, para aprender características locales arbitrarias. El objetivo del aprendizaje del kernel es aprender conjuntamente los parámetros del kernel y SVM. En particular, el aprendizaje de múltiples núcleos locales aprende un núcleo diferente; por lo tanto, un clasificador para cada punto en el espacio de características [37]. La adaptación de métodos globales como los clasificadores árboles de decisión (*Deci-*

sion Tree, DT) y Multinomial Naive Bayes (*Multinomial Naive Bayes, MNB*) han sido combinados con el método local kNN. Estos métodos se han propuesto para resolver tareas relacionadas a la clasificación de textos [63] y la clasificación de datos incompletos [32].

Figura 12: Métodos de aprendizaje localmente ponderado.

En los trabajos presentados anteriormente, se concluye que el rendimiento de un clasificador que adapta un modelo híbrido reporta mejoras en ciertas tareas de clasificación. Esto puede ser debido a que en la práctica, entrenar un modelo global con todo el conjunto de datos es lento y tratar múltiples clases no es tan natural como en un método local. Sin embargo, en la vecindad de un pequeño número de ejemplos y un pequeño número de clases, los métodos globales a menudo funcionan mejor que otros métodos de clasificación. Esta combinación es la que hace que un método híbrido mejore el desempeño del clasificador. Hoy en día, se han hecho intentos como los presentados en [80, 74, 55, 45, 10] para adaptar el esquema

LWL en modelos de aprendizaje profundo. En la siguiente sección, se realiza una revisión de la literatura con las adaptaciones del aprendizaje local en modelos de aprendizaje profundo.

3.3. Aprendizaje Localmente Ponderado en modelos de Aprendizaje Profundo

Uno de los enfoques más explorados en DL con el aprendizaje local son las CNNs. Los trabajos se enfocan en resolver la tarea de clasificación de imágenes usando ejemplos adversos. Los ejemplos adversos se introducen en [72] y se refiere a que aplicando una imperceptible perturbación aleatoria sobre una imagen de entrada, la predicción de la red entrenada no es capaz de generalizar correctamente. En el trabajo [74] se presenta un método donde se realiza la combinación de una arquitectura de red neuronal profunda (*Deep Neural Network, DNN*) y una red de función de base radial (RBFN), para clasificar correctamente ejemplos adversos. A pesar de que se añade el concepto de aprendizaje local en enfoques de DL, este trabajo únicamente define una concatenación de una red neuronal profunda (DNN) y una red de función de base radial (RBFN). Las redes no aprenden de manera conjunta, el entrenamiento es independiente en cada configuración de red.
Un trabajo interesante que adapta una red RBF profunda (*Deep RBF*) se presenta en [80]. El autor propone una CNN y en su capa de salida aplica el concepto de unidades RBF, donde se establece una unidad RBF para cada clase (ver Fig. 13). Con la finalidad de sustituir la función *softmax* por las unidades RBF, tomando como salida la unidad RBF a la clase más cercana. Además, se propone una función de costo que se adapta para hacer que la red RBF profunda sea resistente a múltiples ataques adversos, es decir, que generalice correctamente los ejemplos adversos. Una desventaja a considerar en este modelo es que los métodos de aprendizaje local tienden a tener problemas con alta dimensionalidad en los datos y el método propuesto únicamente es evaluado sobre un conjunto de datos que puede ser llevado aun espacio latente de baja dimensión.

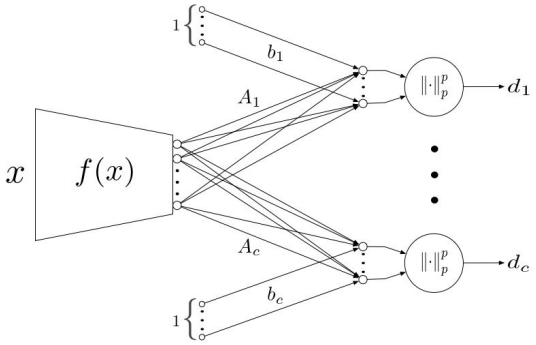


Figura 13: *Deep Radial Basis Function, Deep RBF* presentado por [80].

El algoritmo K-NN también se ha evaluado en modelos de aprendizaje profundo como una representación del aprendizaje local. En [55] se presenta una red profunda de los k-vecinos más cercanos (DkNN). Este clasificador híbrido combina el algoritmo K-NN con representaciones de los datos aprendidos por cada capa de la red neuronal profunda. Sus contribuciones fueron la demostración de la interpretabilidad de la red DkNN, la medida de no conformidad en una predicción y la solidez para identificar ejemplos adversos. El método DkNN consiste en calcular los k vecinos más cercanos de todo el conjunto de entrenamiento y usar una DNN entrenada para entrenar sobre ese conjunto. Otro trabajo que adapta el método DkNN es [69], este trabajo se basa completamente en el método de [55] la diferencia se basa en proponer una heurística para la

inicialización del conjunto de ejemplos que se encuentran cercanos al conjunto de entrenamiento.
Otras adaptaciones del aprendizaje profundo que impliquen técnicas de aprendizaje local como [10, 45] únicamente se han enfocado en dar una interpretabilidad a las decisiones tomadas por los clasificadores cuando se tienen ejemplos adversos. Estos trabajos presentan la desventaja de que el concepto de localidad no se puede llevar a cabo de extremo a extremo. Dado que se requiere del almacenamiento en memoria de las instancias, donde para los enfoques de DL no es viable; ya que se requiere una gran cantidad de datos para que un modelo de aprendizaje profundo generalice correctamente. Además en muchos casos, este tipo de aprendizaje no cuenta con una fase de clasificación.

Nótese que el aprendizaje profundo localmente ponderado no ha sido aprovechado para resolver tareas relacionadas al ER en imágenes. La adaptación puede ser benéfico para mejorar el desempeño en el reconocimiento de emociones. Debido a que se explotaría las ventajas de las CNNs que aprenden en conjunto aquellas característica visuales que permiten hacer una clasificación correcta de las instancias. El aprendizaje local debe contener una fase de entrenamiento y prueba para ser integrado en la parte de clasificación que adapte criterios de entrenamiento ponderados localmente.

4. Propuesta de Investigación

En esta propuesta de investigación se propone el esquema de aprendizaje profundo localmente ponderado *(Locally Weighted Deep Learning, LWDL)*, que consiste en integrar una técnica de aprendizaje local de extremo-a-extremo sobre un enfoque de aprendizaje profundo. El esquema puede ser aplicado a dominios en los cuales el aprendizaje localmente ponderado (LWL) es favorable. Los dominios dentro del alcance de LWL que se plantean son el ER y la clasificación de imágenes de grano fino, de datos con ruido y datos no-balanceados. La finalidad es mejorar el desempeño en ER y dominios dentro del alcance de LWL con respecto a métodos de DL basados en aprendizaje global. A continuación se detalla la investigación propuesta.

4.1. Motivación y Justificación

El aprendizaje localmente ponderado se aplica cuando se tiene problemas al construir clasificadores en los cuales es difícil separar las clases cuando sus atributos visuales son muy similares entre sí [78]. El aprendizaje local es capaz de crear un modelo robusto e interpretable que generaliza correctamente basándose en instancias muy cercanas entre sí.

En el caso del reconocimiento de emociones (ER) en imágenes (siendo una de las aplicaciones que van dentro del alcance del LWL), se puede llevar a cabo mediante el análisis de expresiones faciales. Actualmente existen técnicas más sofisticadas y precisas como el análisis de signos vitales (EGG) para el ER. Pero estás técnicas algunas veces son invasivas y se requieren aparatos especializados para capturar la información. En cambio, las expresiones faciales han sido consideradas por mucho tiempo un lenguaje universal para señalar estados emocionales en todas las culturas [35]. Estas son relativamente sencillas de capturar en imágenes, ya sea con una secuencia de imágenes o vídeos, pero esto nos da captura de emociones aparentes [3].

[3]Las emociones aparentes son aquellas que se perciben visualmente, de acuerdo a las convenciones establecidas en el reconocimiento de emociones, por lo que no es posible determinar si la emoción aparente es en realidad genuina.

Los conjuntos de datos en ER se capturan de dos maneras y son: en entornos controlados y no controlados. Los entornos controlados generalmente son conjuntos de datos poco extensos que se capturan bajos condiciones específicas donde los participantes tienen plena conciencia de la emoción que evoca y además se cuida la exposición, la oclusión, el enfoque, entre otros. En el caso de los no controlados, estos conjuntos de datos son extensos y generalmente no se controlan las condiciones de su captura.

El reconocimiento de emociones actualmente se aborda con enfoques del aprendizaje profundo obteniendo resultados sobresalientes. Los enfoques se dedican a resolver el ER sobre conjuntos de datos tomado bajos entornos controlados. Pero usualmente, este tipo de conjuntos de datos no son extensos. Esto es un problema en el momento de utilizar enfoques de DL, ya que los métodos tienden a generalizar mejor sobre conjuntos de datos extensos.

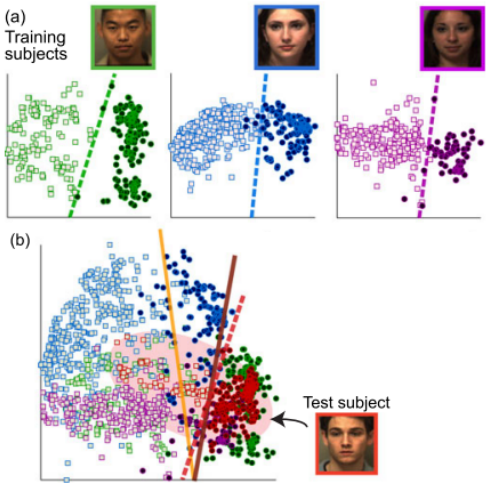


Figura 14: Ejemplo de un clasificador basado en aprendizaje local donde separa adecuadamente las expresiones faciales casi perfectamente para cada sujeto, para mejorar el reconocimiento de emociones. Figura reproducida de [12].

De aquí surge la importancia de construir modelos de DL que sean capaces de reconocer con exactitud emociones en conjuntos de datos tomados bajo entornos no controlados. Además, se enfrenta el reto donde existe una variación como: cultural, étnica, racial, de género, de edad y de intensidad emocional. Las variaciones presentan retos incluso en la identificación de emociones entre los observadores [48]. Las variaciones motivan a pensar en que una técnica de aprendizaje local puede ser prometedora en ER, para crear clasificadores que sean robustos a la detección de esta variantes (como se ilustra en la Fig.14). Debido a que la localidad se encargará de agrupar aquellos conjuntos de rasgos que tengan similitud en función a su distancia.

El aprendizaje local almacena un subconjunto de datos de entrenamiento cercanos al punto de consulta para construir un modelo local capaz de hacer predicciones. La predicción de un valor o clase para una nueva instancia, se basa inicialmente en el cálculo de distancias o similitudes entre instancias relacionadas al entrenamiento. Internamente el aprendizaje local hace transformaciones no lineales de los espacios, mediante el uso de aproximadores locales que se encargan del mapeo de entrada-salida de los datos cuando no son

linealmente separables. Este tipo de aprendizaje tiene la ventaja que los clasificadores son robustos al ruido en los datos y evita la dificultad de la construcción de una función global sobre todo el conjunto de datos. Una desventaja en el aprendizaje local es que los modelos presentan problemas ante la alta dimensionalidad, haciendo inmanejable la aplicación de los métodos locales en altas dimensiones de los datos y derivando en problemas de regularización [53]. Por lo que en este trabajo, se propone un esquema que explote las ventajas del aprendizaje local, pero desarrollando técnicas que resuelvan la problemática de alta dimensionalidad al integrar en un enfoque de DL.

Una motivación acerca del uso de métodos locales aplicados al reconocimiento de emociones en imágenes se basa en la hipótesis de la universalidad presentada en [19], dónde se establece que en cada cultura, se construyen modelos mentales capaces de identificar cada emoción básica en seis grupos distintos (como se mencionó en la Sec 1). En cada agrupación la emoción se expresa utilizando un método específico que implica una combinación de movimientos faciales comunes en todos los humanos. Pero cuando se da el caso de reconocer emociones inter-raciales, estos modelos mentales empiezan a fallar ya que no es tan evidente reconocer emociones cuando se presentan variaciones inter-culturales. Ya que no solo se agrupan esa combinación de movimientos faciales sino que también juega un papel importante la intensidad e incluso los los rasgos étnicos. Por lo tanto, esos modelos mentales necesitan hacer agrupaciones capaces de identificar en conjunto todas esas variaciones, para hacer una predicción exacta. En el caso de modelos computacionales para abordar el ER, un enfoque de DL que integre un aprendizaje local de extremo a extremo podría funcionar, de tal forma que mejoré el desempeño en la clasificación de emociones, bajo el principio de que los aproximadores locales se encargarán de construir funciones de decisión que separen los patrones basándose en la similitud entre ellos [35].

Actualmente, se han realizado intentos por adaptar el aprendizaje local en los modelos del DL. Pero estos esfuerzos no han sido explorados ampliamente, ni adaptados para aprender patrones complejos en los atributos. También, los métodos no se enfocan en el aprendizaje de extremo a extremo o su integración de un método local con un de DL es separada. Debido a que únicamente explotan el beneficio de la de extracción automática de características de un enfoque de DL y de forma separada (mediante un apilamiento de redes) se entrenan clasificadores basados en aprendizaje local.

Esto nos conduce al planteamiento de un esquema de aprendizaje profundo que integre un aprendizaje local de extremo a extremo en un enfoque de DL. El esquema consiste en el aprendizaje profundo localmente ponderado (LWDL) y se aplicará en el ER en imágenes para mejorar el desempeño en el reconocimiento sobre diversos conjuntos de datos. Tal esquema analizaría aquellos componentes faciales que permiten diferenciar cada emoción mediante el análisis de información geométrica y visual de la cara, basándose en métodos que impliquen la localidad en su aprendizaje para separar aquellos patrones mediante la distancia entre ellos. El esquema LWDL podría tener un desempeño competitivo en el reconocimiento de emociones aparentes en imágenes.

4.2. Planteamiento del problema

Los actuales esquemas de *Locally Weigted Deep Learning* adaptan el concepto más simple del aprendizaje local de extremo a extremo en un enfoque de DL. Este consiste en integrar en la capa final aproximadores locales que se encargan de computar la salida de la predicción. Tales esquemas presentan la problemática relacionada a la construcción de los aproximadores locales y se enfrentan a las siguientes condiciones:

- Los esquemas no generalizan correctamente cuando los aproximadores locales enfrentan una alta dimensionalidad en el espacio latente.
- La selección de parámetros para la creación de las unidades RBF no muestran un indicio de ser los parámetros que mejor se ajusten en el esquema LWDL.

El problema se abordará con el desarrollo de modelos de aprendizaje profundo ponderados localmente adaptando múltiples configuraciones de los aproximadores locales, creándolos con técnicas nuevas o que permitan reducir la alta dimensionalidad. Con el objetivo de superar las limitaciones de las soluciones ya existentes para obtener resultados que sean competitivos y alcancen un mejor desempeño el estado del arte en aplicaciones como ER y dominios en el alcance del LWL.

4.3. Preguntas de investigación

Esta propuesta de investigación plantea las siguientes preguntas:

1. ¿En que dominios de aplicación los métodos de aprendizaje local convencionales tienen un mejor desempeño en comparación con los métodos de aprendizaje global?
2. ¿Cuáles enfoques de DL y LWL conforman el esquema LWDL?
3. Los métodos de aprendizaje local presentan problemas para generalizar en altas dimensionalidades en el espacio latente por lo tanto, ¿Con qué técnicas se puede lidiar el problema de alta dimensionalidad en el esquema LWDL?
4. ¿En que dominios un esquema LWDL mejorara la exactitud en la clasificación de imágenes?
5. En el caso de aplicaciones como el reconocimiento de emociones, se tienen conjuntos de datos tomados en entornos controlados y no controlados, donde la heterogeneidad de identidad, género, edad, etnia, iluminación y pose es mucho mayor. ¿El esquema LWDL tiene un desempeño competitivo en el reconocimiento de emociones en imágenes mediante el análisis de expresiones faciales?

4.4. Hipótesis

La integración del aprendizaje local de extremo a extremo en un enfoque de aprendizaje profundo (LWDL) obtiene un desempeño competitivo en comparación con el estado del arte, generando un modelo que es interpretable y robusto en el reconocimiento de emociones y en la clasificación de ejemplos de clase minoritaria, traslape de clases o datos con ruido y grano fino.

4.5. Objetivos

Desarrollar un esquema de aprendizaje profundo localmente ponderado (LWDL), que obtenga un desempeño competitivo en comparación con los métodos tradicionales en la clasificación de imágenes dentro del alcance de LWL.

4.5.1. Objetivos Específicos

1. Evaluar las ventajas que ofrecen los esquemas de aprendizaje local y global en términos de rendimiento en ER y dominios dentro del alcance de LWL.

2. Determinar los componentes de la estructura del esquema LWDL de extremo-a-extremo.

3. Diseñar el esquema LWDL que contengan aprendizaje local de extremo-a-extremo aplicado a ER y dominios dentro del alcance de LWL.

4. Desarrollar una estrategia que resuelva la problemática de alta dimensionalidad en el espacio latente y la construcción de los aproximadores locales en el esquema LWDL.

5. Implementación y evaluación del esquema LWDL en ER y dominios dentro del alcance de LWL.

4.6. Contribuciones

En esta investigación doctoral se pretende obtener las siguientes contribuciones:

- Un esquema de *Locally Weighted Deep Learning*. La integración de la técnica de aprendizaje local de extremo a extremo sobre un enfoque del aprendizaje profundo.

- Una metodología para el reconocimiento de emociones y la clasificación de imágenes en el alcance de LWL.

4.7. Metodología

4.7.1. Evaluar las ventajas que ofrecen los esquemas de aprendizaje local y global en términos de rendimiento en ER y dominios dentro del alcance de LWL.

En esta sección se plantea la hipótesis de que los métodos convencionales de ML basados en aprendizaje local, tienen un mejor desempeño en el reconocimiento de emociones aparentes en imágenes.
Una aplicación del LWDL es el reconocimiento de emociones, aquí se establece la hipótesis de que los clasificadores globales no separan adecuadamente el conjunto de las unidades de acción que conforman las expresiones faciales, que a su vez componen una emoción. Debido a que tienden a generalizar mediante modelos que construyen sus funciones de decisión utilizando todo su conjunto de instancias de entrenamiento.

Para el reconocimiento de emociones un método local que construya sus límites de decisión basados en la similitud calculada mediante distancias, permitirán hacer una separabilidad adecuada de los patrones dados las variaciones que se presentan en la emoción. El reconocimiento requiere que los clasificadores hagan una separación de patrones más fina. Por ejemplo en el caso siguiente: tenemos una variación racial donde hay asiáticos y caucásicos. La forma de distinguir entre ellos emociones es diferente. Los asiáticos tienden mostrar signos tempranos característicos de intensidad emocional con los ojos. Los caucásicos involucran otros músculos faciales y no relacionados exactamente con los ojos [35]. Esto significa que en regiones específicas se brinda información importante para el reconocimiento de cierta emoción y que esta ligada la dependencia racial debido a los rasgos. En este caso, se puede pensar que un método de aprendizaje local sería pertinente dada esta y algunas variaciones culturales, ya que se encargaría de construir límites de decisión basándose en el cálculo de similitudes entre las variaciones que se presenten entre emoción.

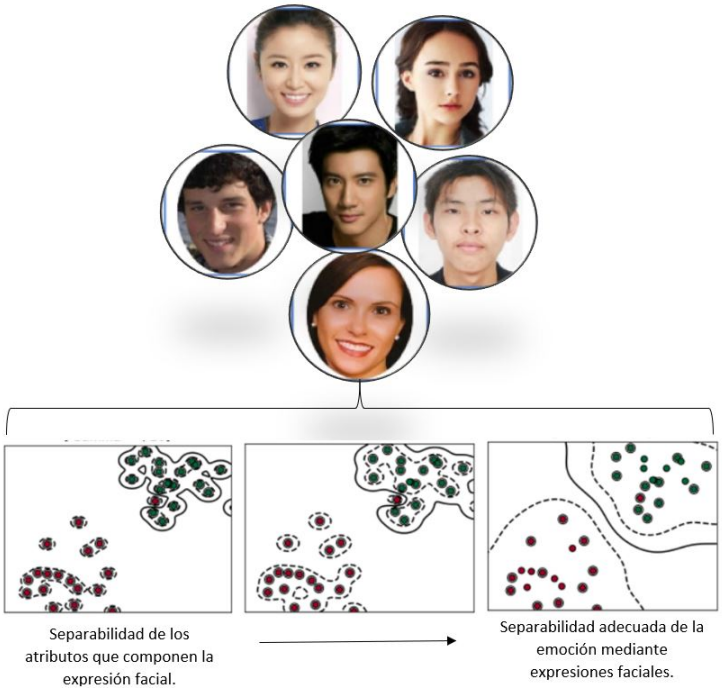


Figura 15: Representación gráfica de un clasificador basado en aprendizaje local para el reconocimiento de emociones en imágenes.

Para comprobar la hipótesis se debe formular una comparativa entre los métodos de aprendizaje local y global para reconocer emociones aparentes en imágenes.

Entregable:

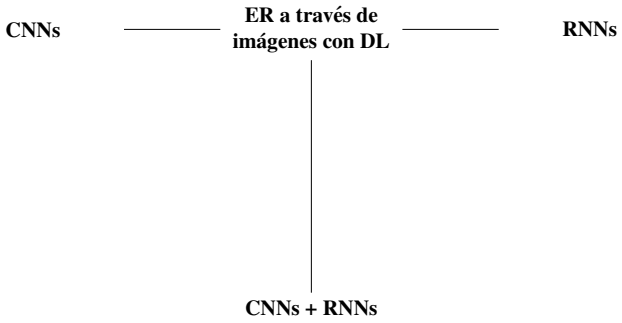
- Evaluación de clasificadores basados en el aprendizaje local y global en dominios dentro del alcance del LWL.

4.7.2. Determinar los componentes de la estructura del esquema LWDL de extremo-a-extremo.

En esta sección se hace una revisión de los enfoques de DL usados en ER. Por parte de LWL se hace una revisión y un análisis de métodos que se pueden integrar con DL, para hacer un aprendizaje local de extremo a extremo.

ER se ha llevado a cabo con enfoques como las CNNs, RNNs, e híbridos. Uno de los enfoques de aprendizaje profundo disponibles ampliamente usados, son las CNNs. La popularidad de estos enfoques radica en que reducen en gran medida la dependencia de las técnicas de pre-procesamiento en la imagen. Donde generalmente, se construyen modelos enfocados al análisis de la física de la cara y la aplicación de otras técnicas para el estudio de las expresiones faciales.

En el aprendizaje localmente ponderado se tienen dos variantes importantes, una consiste en métodos basados en instancias y métodos incrementales. Los métodos basados en instancias (como se mencionó en la Sec.2) son métodos que mantienen todas las instancias de entrenamiento son guardadas en memoria. Esto



presenta una desventaja cuando se tiene un conjunto de datos muy extenso, dado que se requiere mayor recurso computacional para hacer una predicción. En cambio, los métodos incrementales cuentan con una fase de entrenamiento y prueba. La construcción de su modelo se basa en crear límites de decisión en base a instancias relacionadas (muy cercanas) entre sí.

Por lo tanto se concluye que un método incremental se ajusta mejor para hacer una adaptación del aprendizaje local de extremo a extremo en un enfoque de DL. Esto es debido a que los métodos de DL tienden a generalizar mejor y evitar el sobre-ajuste cuando se entrena sobre conjuntos de datos muy grandes.

Una de las formas de abordar el esquema de *Locally Weigted Deep Learning, LWDL* (como se muestra en la Fig.16), es mediante la integración del enfoque tipo CNN, con algunos métodos de LWL como: Las Redes de Función de Base Radial (RBFN), o adaptando aproximadores locales basados en algoritmos de agrupamiento (*Clustering*), en incluso utilizando clasificadores basados en distancias ponderadas (*Distance Weighted Learning*).

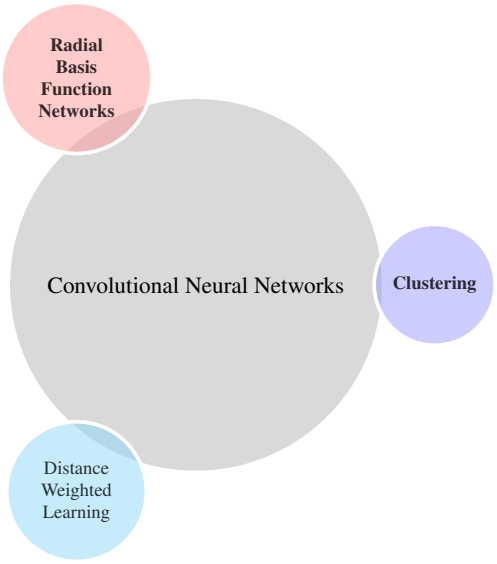


Figura 16: Métodos de aprendizaje localmente ponderado (LWL) que pueden ser adaptados sobre un enfoque de DL como las redes neuronales convolucionales (CNNs).

Entregable:

- Análisis de la pertinencia del aprendizaje local en enfoques de DL de extremo a extremo, aplicados al ER.

4.7.3. Diseñar el esquema LWDL que contengan aprendizaje local de extremo-a-extremo aplicado a ER y dominios dentro del alcance de LWL.

El enfoque de DL que se usará para adaptar el esquema LWDL son las CNNs. La razón es que se requiere tomar la ventaja de las CNNs que en gran medida reducen la dependencia de técnicas de pre-procesamiento y funcionan como extractores automáticos de características en conjunto con la clasificación de las imágenes. A continuación, se proponen algunas posibles arquitecturas de lo que comprende el esquema LWDL. Siendo estas algunas de las formas generales de abordar el ER con el esquema LWDL.

- **Red profunda de función de base radial (*Deep Radial Basis Function Network, Deep-RBF Network*).**

  Una forma de esquema LWDL es mediante la integración de un aprendizaje basado en redes RBF. Las redes RBF actúan como aproximadores locales y son semejantes a las redes MLP (en la Sec.2 se explica a detalle la diferencia entre cada una). Una forma de desarrollar el esquema LWDL es mediante la adaptación de un enfoque CNN que integre aproximadores locales basados en Redes RFB de extremo a extremo. Una CNN se puede considerar que contiene dos fases que son la extracción automática de características visuales y la fase de clasificación. La red aprenden características conforme a un perceptrón multicapa (MLP) que se considera son las capas totalmente conectadas de la CNN (siendo estas parte de la fase de clasificación). Las redes MLP funcionan globalmente, es decir, las salidas de la red son decididas por todas las neuronas. A diferencia de las redes de aproximación local que sus salidas están determinadas por unidades ocultas especializadas en ciertos campos receptivos locales.

  En la Fig.17 se presenta una arquitectura simple y la más general de una *Deep-RBF Network*. Esta configuración difiere de la ya existente en el estado del arte por la adaptación de multiples unidades RBF en capas intermedias y da pie a diseñar una configuración más compleja como la mostrada en la Fig. 28. La red integra unidades RBF para que el modelo clasifique conforme a unidades ocultas especializadas en ciertos campos receptivos locales (pero en capas iniciales de las totalmente conectadas).

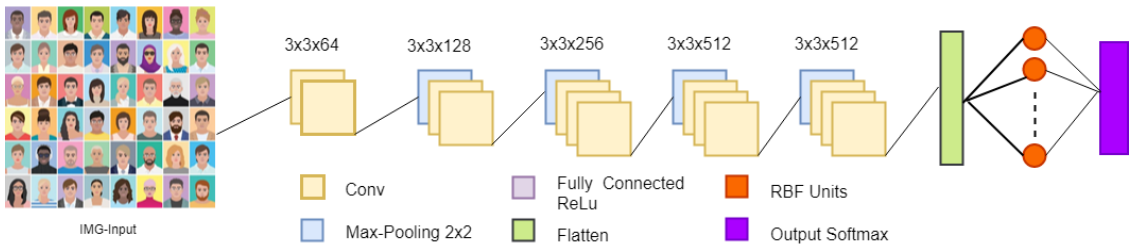


Figura 17: Descripción general una arquitectura de DL que integra el aprendizaje local de extremo a extremo usando unidades RBF.

A diferencia del primer esquema del aprendizaje local de extremo a extremo presentado en [80]. El autor aplica la localidad sobre la última capa para calcular la clase a la que pertenece la instancia, es

decir, en la capa de salida se adaptan unidades RBF conforme el tamaño de las clases. Se elige como salida el argumento mínimo de la neurona, significa que toma la clase que tenga la mínima distancia. En la Fig. 18 se muestra la arquitectura propuesta por [80].

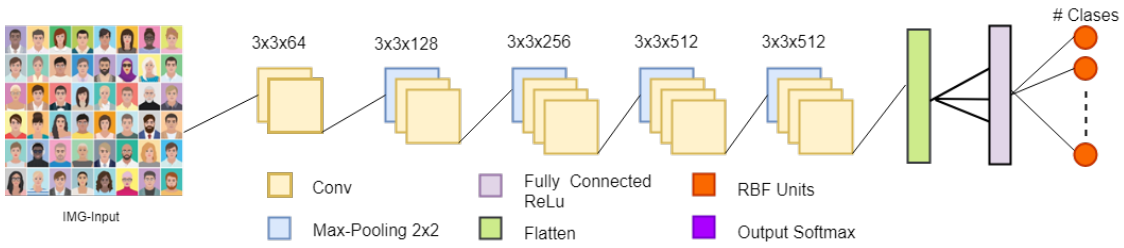


Figura 18: Descripción general una arquitectura de DL que integra el aprendizaje local de extremo a extremo usando unidades RBF.

De los modelos existentes, se observa que solo se ajustan bien en ejemplos donde el espacio latente no es de alta dimensión, por ejemplo en conjunto de datos con el que fue evaluado pertenece a un espacio latente de baja dimensión. Esa es la razón por la que el modelo se desempeña correctamente, pero si el modelo se prueba con un conjunto de datos de alta dimensionalidad, el modelo presentaría problemas para generalizar correctamente. Además, se presentaría otra problemática, y es el determinar los parámetros adecuados y el número de unidades RBF. Este tipo de problemas se deben enfrentar cuando se quiere adaptar el aprendizaje local de extremo a extremo sobre un enfoque de aprendizaje profundo para mejorar el desempeño en el reconocimiento de emociones en imágenes.

Entregable:

- Arquitecturas de DL que integren el LWDL.

4.7.4. Desarrollar una estrategia que resuelva la problemática de alta dimensionalidad en el espacio latente y la construcción de los aproximadores locales en el esquema LWDL.

En el caso de las redes tipo *Deep RBF* propuestas como uno de los esquemas LWDL, se presentan las siguientes problemáticas:

- Determinar las unidades RBF
- Reducir el espacio latente de alta dimensión.

Se ha planteado algunos métodos para para resolver estos problemas. A continuación se describen.

Determinar las unidades RBF

Este se puede llevar a cabo de dos formas:

1. A prueba y error.
   Las unidades RBF se pueden crear estableciendo algún número de unidades RBF ocultas, inicializándose de forma aleatoria los centros y radios.

2. Usando algoritmos de agrupamiento.
   Este determina las unidades RBF mediante la creación de prototipos. Los prototipos usan algoritmos de agrupamiento para construir los centros y anchos a partir del conjunto de datos de entrenamiento.

   Una estrategia puede ser el uso de algoritmos de agrupamiento como:

   - Agrupación espacial basada en densidad de aplicaciones con ruido (*Density-Based Spatial Clustering, DBSCAN*).
     El algoritmo determina un número de kernel basado en la densidad de la muestra con sus vecinos más cercanos. Estos kernel son usados para construir el agrupamiento. Se utilizarían estos kernel como el centro de cada unidad RBF, y se obtendría el radio como la distancia del kernel al vecino más lejano que es considerado como el punto kernel.
   - *K-Means Clustering*
     El número de unidades RBF será proporcional al número de centroides del algoritmo de agrupamiento, y será inicializado en el mismo punto. El radio será igual a la distancia del centroide al elemento más lejano del agrupamiento.
   - *Mean-Shift Clustering* El algoritmo puede determinar el número de clúster o ser un valor establecidos como parámetro, de igual forma, que los demás métodos de grupamiento, los centroides son usados para inicializar las unidades RBF.

Reducción de la alta dimensionalidad en el espacio latente.
La reducción de dimensionalidad del esquema LWDL, se puede llevar a cabo mediante dos técnicas:

- Autoencoders. En el aprendizaje profundo, se puede utilizar un tipo de red neuronal llamada *autoencoders*. Los autoencoders son redes neuronales que se pueden usar para reducir los datos en un espacio latente de baja dimensión al apilar múltiples transformaciones no lineales. Esta reducción se puede llevar a cabo dentro del modelo de aprendizaje profundo de extremo a extremo. El espacio latente de baja dimensión, se puede utilizar como entrada de la capa con aproximadores locales, para crear un aprendizaje local de extremo a extremo en un enfoque de aprendizaje profundo.

- Multibranch Deep Radial Basis Function. Una idea nueva para lidiar con la alta dimensionalidad en una CNN, es la adaptación de módulos de unidades RBF entre las capas convolucionales. En la Fig. 19 se presenta una arquitectura propuesta del esquema LWDL. La red reduce la dimensionalidad del espacio latente al adaptar múltiples conexiones ponderadas por aproximadores locales en los mapas de características. Los aproximadores locales consisten en múltiples módulos RBF que se conectan a un sub-conjunto de los mapas de características para hacer un aprendizaje local de las características extraídas. En esta configuración, se usan 16 módulos RBF de la última capa convolucional.

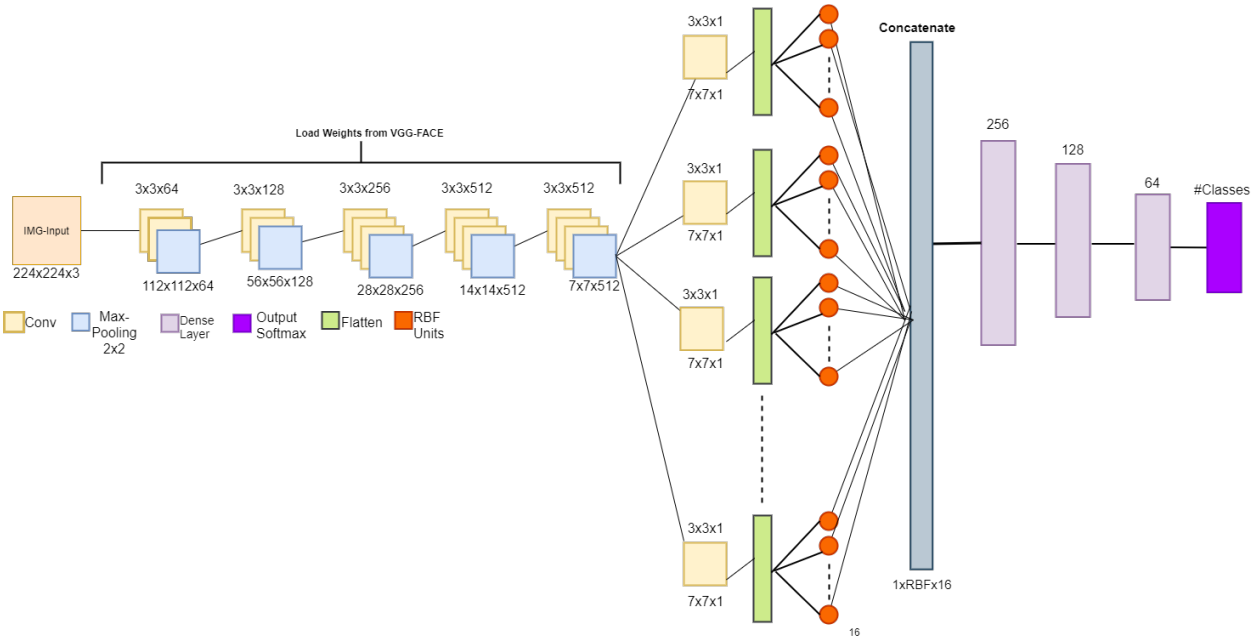


Figura 19: Multibranch Deep Radial Basis Function con 16 modulos RBF.

Otra configuración que podría ser funcional y que integra el aprendizaje local de extremo a extremo es la que se muestra en la Fig.20. Este modelo trata de aplicar un tipo de conexión residual, ya que entre las capas convolucionales, se hacen conexiones residuales ponderadas por aproximadores locales. Las ramas se adaptan a múltiples niveles jerárquicos de las características extraídas en la red neuronal convolucional.

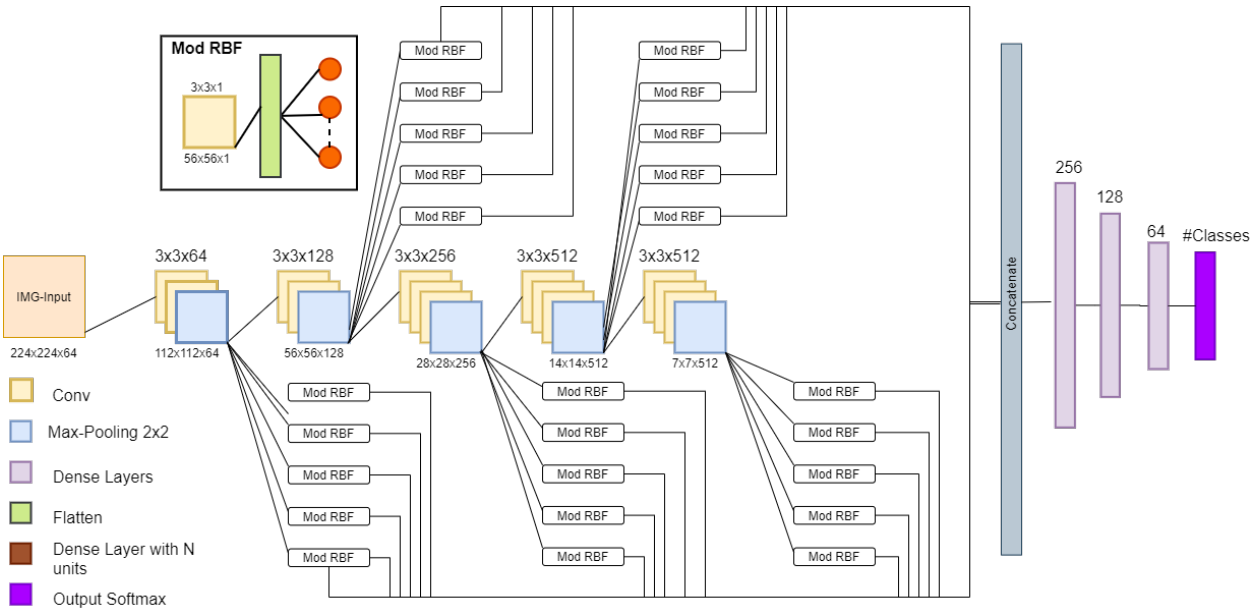


Figura 20: Multibranch Deep Radial Basis Function con 25 modulos RBF.

Entregable:

- Esquemas LWDL con estrategías para resolver la problemática de los métodos de aprendizaje local cuando se adaptan en un enfoque de DL de extremo a extremo.

4.7.5. Implementación y evaluación del esquema LWDL en ER y dominios dentro del alcance de LWL.

En esta sección se desarrolla el esquema LWDL integrando todas las técnicas propuestas para llevar a cabo el aprendizaje local de extremo a extremo en un enfoque de DL. También se muestran los conjuntos de datos relacionados al reconocimiento de emociones aparentes en imágenes, usados en el estado del arte y clasificación de objetos en imágenes enfocados al dominios dentro del alcance LWL.

- Implementación del esquema LWDL aplicado al ER.
- Evaluación del esquema LWDL con conjunto de datos que se usan comúnmente para entrenar algoritmos de aprendizaje automático y el aprendizaje profundo. Por ejemplo: Cifar10, Cifar100 y Tiny-Imagenet.
- Evaluación del esquema LWDL en conjuntos de datos usados en la clasificación de grano fino, datos no balanceados y datos con ruido.
- Evaluación del desempeño del esquema LWDL, usando conjuntos de datos relacionados al reconocimiento de emociones, comunmente usados en el estado del arte en el ER en imágenes de expresiones faciales. Los conjuntos de datos creados bajo entornos controlados comúnmente usados en el ER son los siguientes: CK+, JAFFE, MMI y iCV-MEFED. Los conjuntos de datos creados bajo entornos no controlados son los siguientes: EmotionNet y AffectNet, ExpW y RAF-DB.

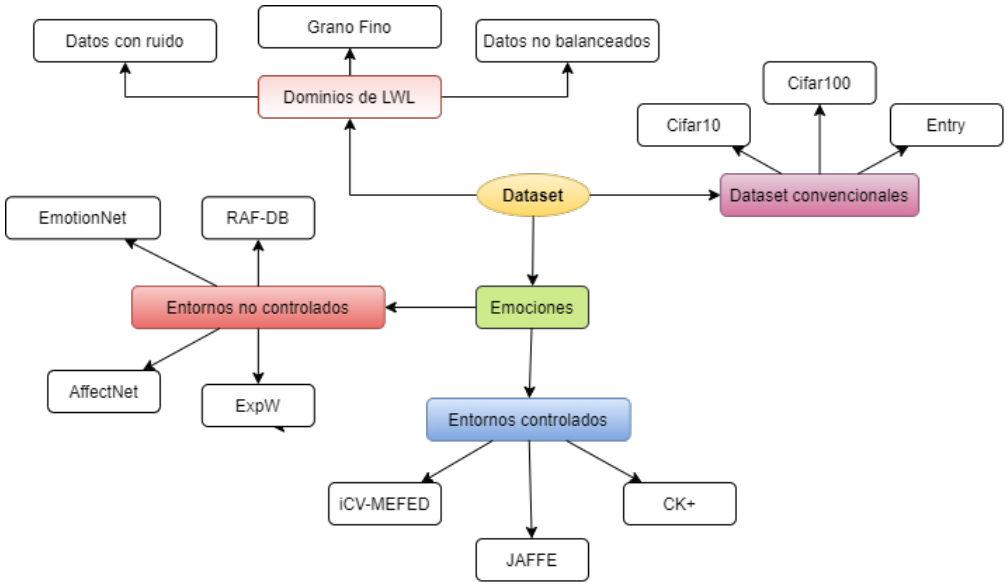


Figura 21: Conjunto de datos usados para evaluar el esquema LWDL.

- Comparativa entre el estado del arte relacionado con el reconocimiento de emociones en imágenes y los resultados obtenidos del esquema LWDL.

Entregable:

- Desarrollo, evaluación y comparación del esquema LWDL.

4.8. Cronograma de actividades

El cronograma de actividades para alcanzar los objetivos de la propuesta se presentan en la Tabla 22. El plan se divide en cuatro periodos por año, cada periodo de tres meses.

Actividades	Año															
	2019				2020				2021				2022			
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
Revisión del estado del arte.																
Desarrollo de la propuesta de investigación doctoral.																
Defensa de la propuesta de investigación doctoral.																
Evaluar las ventajas que ofrecen los esquemas de aprendizaje local y global en términos de rendimiento en la clasificación de imágenes.																
Determinar los componentes de la estructura de un aprendizaje local de extremo a extremo en un enfoque de DL.																
Proponer esquemas que contengan el aprendizaje local de extremo a extremo.																
Desarrollo de técnicas para resolver la problemática del aprendizaje local en enfoques de DL.																
Desarrollo del esquema LWDL.																
Evaluación del esquema LWDL en dominios como clasificación de imágenes en conjuntos de datos no balanceados o de grano fino.																
Evaluación del desempeño del esquema LWDL en conjuntos de datos capturados bajo entornos controlados y no controlados para el reconocimiento de emociones.																
Comparativa del esquema LWDL con el estado del arte en el reconocimiento de emociones y la clasificación de imágenes en conjuntos de datos no balanceados o de grano fino.																
Elaboración de artículos científicos.																
Escritura de Tesis.																
Defensa de Tesis.																

Figura 22: Cronograma de actividades para la investigación de doctorado.

4.9. Plan de publicaciones

Se tiene contemplado publicar al menos 2 artículos en conferencias internacionales y 2 artículos en revistas JCR. Los posibles foros de publicación son:

- IEEE Transactions on Affective Computing.
- Pattern Recognition Letters.
- IEEE International Conference on Automatic Face and Gesture Recognition.
- European Conference on Computer Vision.

5. Resultados Preliminares

En esta sección presentamos resultados preliminares que respaldan nuestra propuesta de investigación, enfocándose al reconocimiento de emociones como a una de las aplicaciones donde se puede aplicar el esquema LWDL. Los resultados se dividen en dos partes:

- Comparativa entre los métodos de aprendizaje local y global para reconocer emociones aparentes en imágenes. Cuyo objetivo es evaluar el desempeño de ambos tipos de aprendizaje para mostrar la pertinencia de la propuesta.
- Evaluación preliminar del esquema LWDL en el reconocimiento de emociones aparentes en imágenes mediante el análisis de expresiones faciales. Con el objetivo de demostrar que el esquema LWDL muestra resultados competitivos en ER en imágenes.

5.1. Comparativa entre los métodos de aprendizaje local y global para reconocer emociones aparentes en imágenes.

El experimento consiste en usar una CNN pre-entrenada como extractor de características visuales en imágenes para construir conjuntos de datos de entrenamiento, validación y prueba. Posteriormente se entrenan y evalúan algoritmos de clasificación usando con los conjuntos de datos obtenidos anteriormente para comparar el desempeño de los métodos que usan aprendizaje local y global, en la Fig.23 se muestra un diagrama de cada etapa.

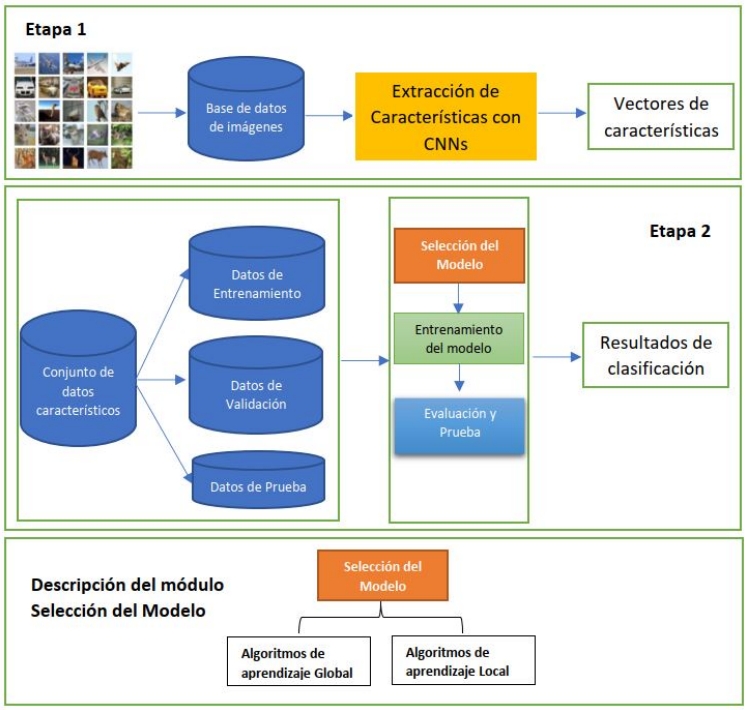


Figura 23: Diagrama del modelo de aprendizaje automático para clasificar imágenes

5.1.1. Extracción de características en imágenes

Para el experimento se usaron dos tipos de CNNs como extractores de características. Una de ellas es la red Inception V4 y la otra la red Modelo 9 presentada en el artículo [43]. Inception V4 [71] es una de las arquitecturas de CNNs más relevantes en el estado del arte por alcanzar resultados superiores a otras arquitecturas propuesta. El modelo 9 [43] se usó debido a que presenta resultados sobresalientes de exactitud al evaluar sobre el conjunto de datos Tiny-Imagenet que se usa para la experimentación. En la tabla 2 se describe las capas usadas de las CNNs.

Red Neuronal Convolucional (CNNs)	**Nombre de la capa**	**# Características Visuales**
Inception V4	FC1	1536
ConvNet Model 9	FC1	4096

Tabla 2: Extracción de características usando CNNs. Se extraen varias características dimensionales de la red. Se usa la primera capa totalmente conectada después de la última ReLU del modelo previamente entrenado.

El modelo 9 es una arquitectura que consta de 16 capas y recibe una imagen de entrada de 56 x 56 x 3. Es entrenada con un batch size de 200, un momentum de 0.9 y un dropout de 0.5. La Tabla 3 muestra en las columnas la profundidad de la red y las dimensiones de las capas.

Image Input	**ConvNet Configuración del Model 9**
56	conv3-64
	conv3-64
28	conv3-128
	conv3-128
14	conv3-256
	conv3-256
	conv3-256
7	(six layers) conv3-512
FC1	4096
FC2	4096
	dropout
FC3	200 Softmax
Deep (conv+fc)	16

Tabla 3: Arquitectura de la red convolucional del modelo 9. El modelo se usa para entrenar una CNN para cada conjunto de datos en el experimento.

La red Inception V4 se entrenó con imágenes de dimensiones de (299 x 299 x 3) y también se aplicó la técnica de *Fine-Tuning*. El esquema de la red Inception V4 se muestra en la Fig 24. La técnica *Fine-Tuning* consiste en usar un modelo de red profunda previamente entrenado, básicamente este descongela algunas capas superiores de un modelo congelado y vuelve a entrenar el modelo con nuevas capas. Para este caso, se uso los pesos de la red pre-entrenada sobre ImageNet. En la Fig. 25 se muestra el esquema de *Fine-Tuning* usado a la red Inception V4 para el experimento.

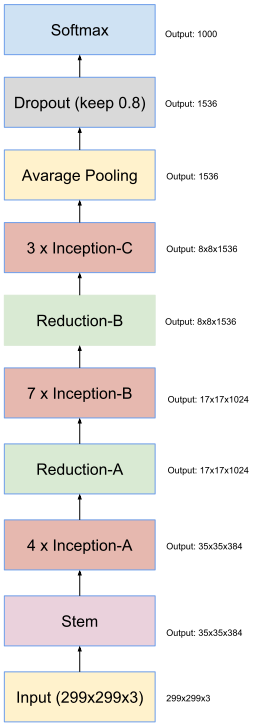


Figura 24: Arquitectura de la red Inception V4.

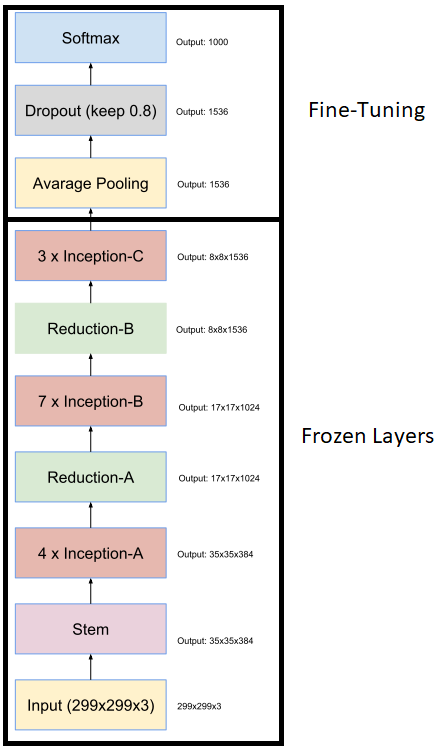


Figura 25: Fine-Tuning para la Red Inception V4.

5.1.2. Entrenamiento y prueba de los clasificadores basados en aprendizaje local y global.

Los algoritmos de ML usados en el experimento se muestran en la Fig.26. En cada uno se adapta ciertas configuraciones. Por ejemplo, SVM usa un kernel lineal, norma l2, con multi-clase (*one vs rest*) y un criterio de paro de 1e-5. MLP usa la función de activación ReLu, una capa oculta de 50 neuronas, momentum de 0.9, el optimizador Adam y una tasa de aprendizaje de 0.01. DT usa una profundidad de 5 con 10 estimadores sobre 5 características. NB se evalúa en un clasificador Gaussiano. SVM-kNN toma los 10 vecinos más cercanos y construye un modelo de maquina de soporte vectorial con los k vecinos más cercanos, este mismo principio se aplica para DT-kNN y NB-kNN. RBF se entrena con 120 unidades RBF y con el número de clases como centroides en sus capa de salida. kNN se evalúa con los $k = 1, ..., 10$, y se toma el mejor resultado para cada base de datos. LR y LVQ usan los parámetros por defecto presentados en la librería scikit-learn [58].

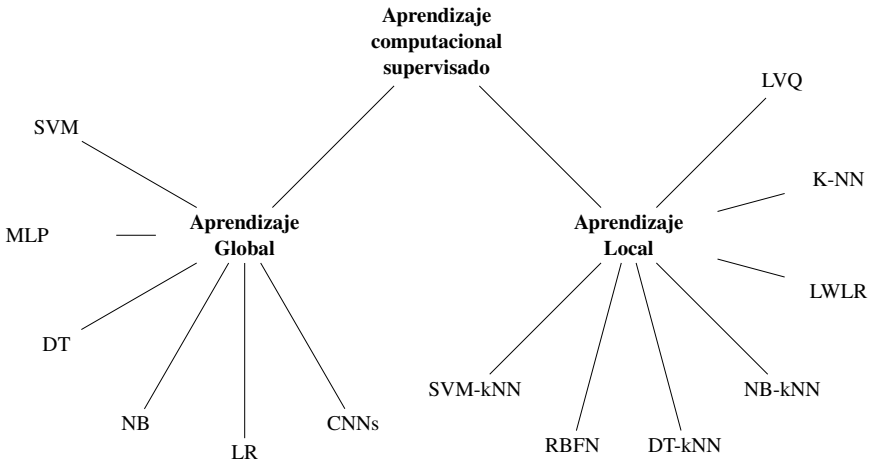


Figura 26: Algoritmos de aprendizaje supervisado usados en el experimento para hacer una comparativa entre aprendizaje local y global.

Los resultados de la clasificación se reportan sobre la métrica de exactitud Top-1. Top-1 es la exactitud convencional, es decir, toma la primera respuesta del modelo (a la que tiene la clase una mayor probabilidad) como la respuesta esperada.

5.1.3. Resultados de la evaluación de los métodos locales y globales en conjuntos de datos generales.

El experimento también lleva a cabo con algunos conjuntos de datos que son usados en el estado del arte, para evaluar el desempeño de los algoritmos y estos son: Cifar10, Cifar100 y TinyImagenet. En la Tabla 4 se muestra la distribución de los conjuntos de datos. Las colecciones de imágenes contienen diversos objetos que pertenecen a distintas categorías.

Dataset	#Training Samples	#Testing Samples	#Clases
Cifar10	50,000	10,000	10
Cifar100	50,000	10,000	100
Tiny Imagenet	100,000	10,000	200

Tabla 4: Distribución del conjunto de datos de entrenamiento y prueba para la clasificación de imágenes.

En las Tablas 5 y 6 los resultados obtenidos al evaluar los métodos de aprendizaje local y global indican que cuando se tiene una variación en las imágenes, un modelo local no representa una mejora. Y que depende más del tipo de imágenes que se presenten, por ejemplo en Tiny-ImageNet la variación en los objetos en las imágenes es mucho mayor que en Cifar10, modelo para el cual el aprendizaje local mejora usando un método híbrido.

Feature Extractor Inception V4: #Features 1536														
	Global Learning					Local Learning							CNN	
Database	**SVM**	**MLP**	**DT**	**MNB**	**LR**	**SVM-kNN**	**RBF**	**DT-kNN**	**NB-kNN**	**LWLR**	**KNN**	**LVQ**	**InceptionV4**	**InceptionV4 Fine-Tuning**
Cifar10	0.841	**0.861**	0.607	0.760	0.857	**0.891**	0.841	0.2	0.1	0.3	0.816	0.8	0.6541	0.85
Cifar100	0.639	**0.672**	0.591	0.661	0.652	0.610	**0.660**	0.102	0.092	0.124	0.654	0.551	0.1027	0.529
Tiny-Imagenet	0.410	0.443	0.318	0.412	**0.499**	0.483	0.460	0.085	0.053	0.078	**0.516**	0.5	0.2032	0.70

Tabla 5: Resultados de exactitud Top-1 en la clasificación usando aprendizaje local y global. Extrayendo 1536 características del modelo InceptionV4, pre-entrenado con el conjunto de datos ImageNet. Usando la primera capa totalmente conectada (FC1) después de la última ReLU.

Feature Extractor Model 9: #Features 4096 FC1														
	Global Learning					Local Learning							CNN	
Dataset	**SVM**	**MLP**	**DT**	**MNB**	**LR**	**SVM-kNN**	**RBF**	**DT-kNN**	**NB-kNN**	**LWLR**	**kNN**	**LVQ**	**Model 9**	**Model 9 Fine-Tuning**
Cifar10	0.821	**0.831**	0.597	0.740	0.794	**0.861**	0.812	0.2	0.110	0.397	0.803	0.473	0.763	0.837
Cifar100	0.624	0.632	0.571	0.636	**0.649**	0.596	**0.659**	0.07	0.075	0.147	0.644	0.491	0.702	0.794
Tiny-Imagenet	0.575	0.586	0.413	**0.591**	0.518	0.587	**0.596**	0.03	0.01	0.293	0.567	0.483	0.532	0.596

Tabla 6: Resultados de exactitud Top-1 en la clasificación de imágenes usando aprendizaje local y global. Extrayendo 4096 características del modelo 9 [43], pre-entrenado con el conjunto de datos TinyImageNet. Usando la primera capa totalmente conectada (FC1) después de la última ReLU.

5.1.4. Resultados de la evaluación de los métodos locales y globales en conjuntos de datos relacionadas al reconocimiento de emociones en imágenes.

El experimento enfocado al reconocimiento de emociones en imágenes usa una red CNN pre-entrenada llamada VGG-Face [56] como extractor de características visuales. La Tabla 9 muestra la distribución de los conjuntos de datos los comunes en el reconocimiento de emociones en imágenes.

Dataset	**#Training Samples**	**# Testing Samples**	**#Classes**
CK+	877	104	7
JAFFE+	149	64	7

Tabla 7: Distribución del conjunto de datos de entrenamiento y prueba para el reconocimiento de emociones usando el conjunto de datos Ck + de expresiones faciales.

La Red VGG-Face (Fig.27) es una CNN de 22 capas y 37 unidades de profundidad entrenada en más de 2 millones de imágenes de celebridades. VGG-Face ha demostrado tener desempeño sobresaliente en puntos de referencia relacionados al reconocimiento facial. La red utilizó un conjunto de datos en su entrenamiento que es similar al conjunto de datos CK +. Por lo cual la hace apropiada para el experimento y que su rendimiento sea más confiable en nuestra aplicación. La red extrae 4096 características visuales de la primera capa totalmente conectada (FC1) después de la última ReLU. Las bases de datos se dividen en ejemplos de entrenamiento y de prueba para cada ejemplo, se extrae un vector característico de la imagen $X = [x_1, x_2, \cdots, x_{4096}]$.

La comparativa entre lo métodos de aprendizaje local y global para ER en imágenes se muestra en la Tabla. 8. Los resultados obtenidos arrojan que los métodos locales tienden a generalizar mejor que los modelos globales, a excepción del clasificador LVQ. El clasificador que mejor se ajusta en el reconocimiento es el basado en instancias kNN. Los resultados brindan soporte de que los métodos locales se ajustan adecuadamente en el reconocimiento de emociones.

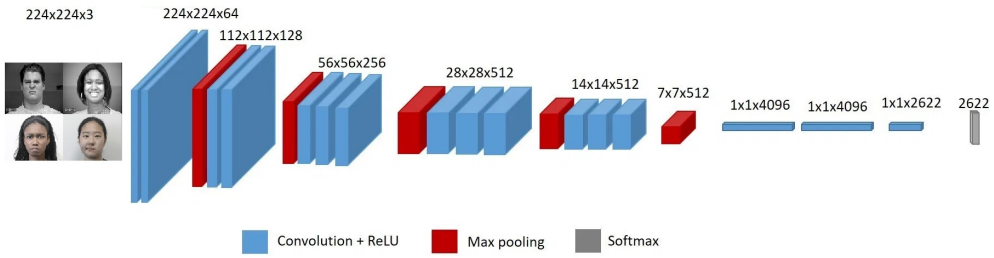


Figura 27: Arquitectura de la red CNN VGG-FACE.

Feature Extractor VGG FACE: #Features 4096 FC1														
	Global Learning					Local Learning							CNN	
Database	**SVM**	**MLP**	**DT**	**MNB**	**LR**	**SVM-kNN**	**RBF**	**DT-kNN**	**NB-kNN**	**LWLR**	**kNN**	**LVQ**	**VGGFACE**	**VGGFACE Fine-Tuning**
CK +	0.952	**0.962**	0.942	0.952	0.952	0.973	0.971	0.970	0.913	0.971	**0.981**	0.798	0.2113	0.8014
JAFFE	0.801	**0.912**	0.793	0.781	0.732	0.926	0.942	0.803	0.813	0.794	**0.953**	0.535	0.1428	0.2087

Tabla 8: Resultados de exactitud Top-1 en la clasificación usando aprendizaje local y global. Extrayendo 4096 características de la red VGG-Face pre-entrenada. Usando la primera capa totalmente conectada (FC1) después de la última ReLU.

5.2. Evaluación del esquema preliminar LWDL en el reconocimiento de emociones aparentes en imágenes mediante el análisis de expresiones faciales.

El esquema LWDL que se evalúa es el propuesto en la Fig.28. La arquitectura *Deep RBF 16-MOD* es tipo CNN que integra el aprendizaje local de extremo a extremo usando unidades RBF. La localidad se aplica mediante 16 módulos RBF en una CNN. La configuración de los módulos RBF se adaptan para reducir la alta dimensionalidad del espacio latente. Si no se implementaran la ramas en la arquitecta cada unidad RBF una dimensionalidad de entrada de 25,088. La elección de los 16 módulos se hace de forma arbitraria con la finalidad de solo tomar un subconjunto de los mapas de características de la capa convolucional final teniendo una dimensionalidad de entrada de 49.

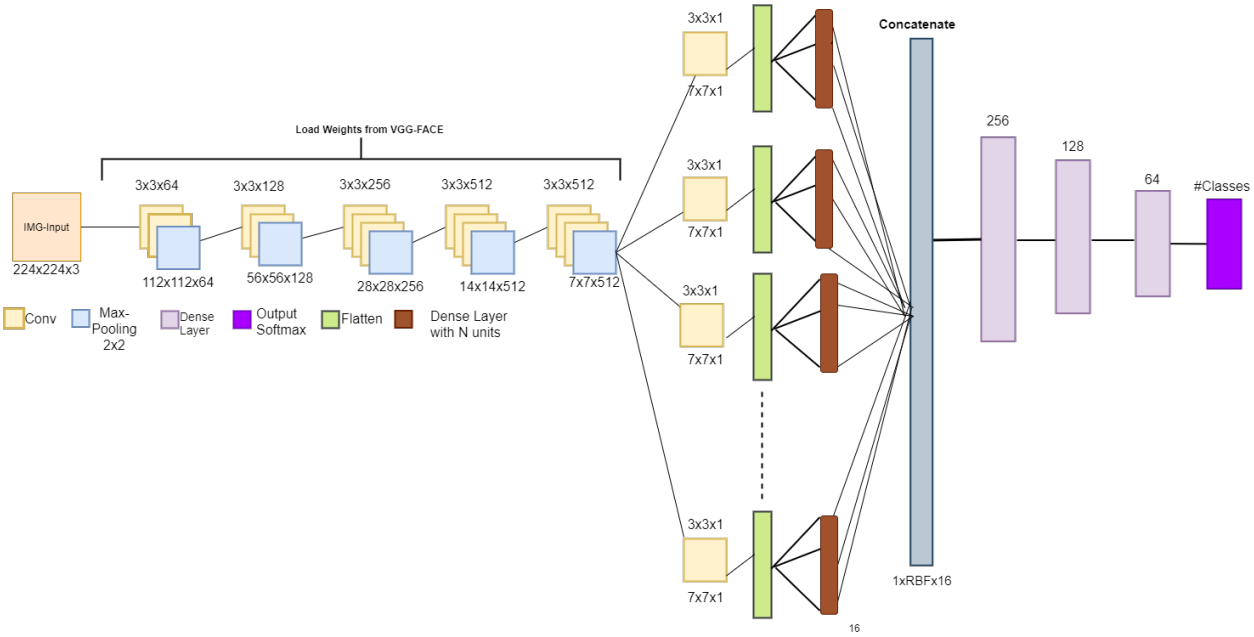


Figura 29: Convolutional Neural Networks con 16 mod, *CNN 16-MOD*.

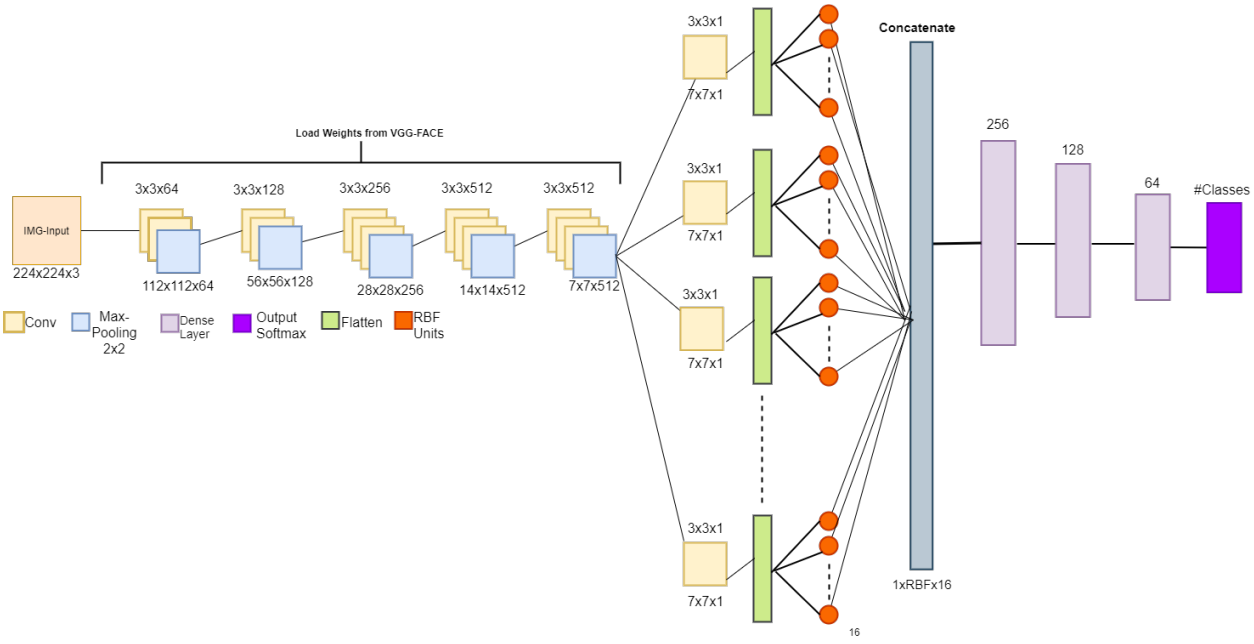


Figura 28: Deep Radial Basis Function con 16 modulos RBF.

Para la comparativa entre métodos que no adaptan el aprendizaje local, se evalúa una CNN contra un esquema LWDL. La CNN porpuesta tiene una arquitectura idéntica al *Deep RBF 16-MOD*. La diferencia radica en que los mod RBF se intercambian por capas densas y se establece el número de neuronas como parámetros. En la Fig. 29 se ilustra el esquema propuesto.

La evaluación de las arquitecturas se hace sobre conjuntos de datos de emociones. Los conjuntos de datos contienen imágenes de expresiones faciales, donde se expresan varias emociones aparentes. La Tabla 9 muestra la distribución de los conjuntos datos usados para el entrenamiento, validación y prueba.

Dataset	#Training Samples	#Validation Samples	#Testing Samples	Classes
CK+	877	94	123	7
JAFFE+	143	35	35	7
CK+JAFFE	700	219	221	7
iCV MEFED	15969	7000	5751	50

Tabla 9: Distribución del conjunto de datos de entrenamiento, validación y prueba para el reconocimiento de emociones en imágenes de expresiones faciales.

5.2.1. Resultados de la evaluación preliminar del esquema LWDL con 16 MOD.

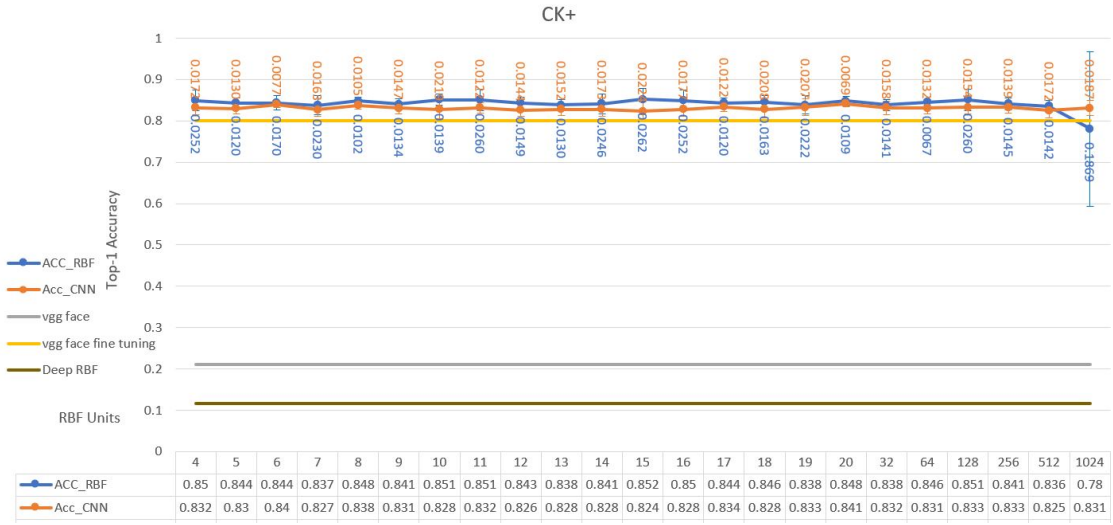
En la Tabla. 10 se muestran los valores de exactitud Top-1 y se hace una comparativa entre los modelos VGG FACE, Deep RBF, DRBF 16-MOD y CNN 16-MOD. El esquema LWDL al ser evaluado sobre los conjuntos de datos de emociones ha mostrado obtener resultados preliminares que son competitivos en el estado-del-arte, alcanzando una mejora en comparación con modelos de DL que no integran el aprendizaje local de extremo a extremo. Los modelos VGG FACE y Deep RBF se usa como marco de referencia del estado del arte. Deep RBF es un método de aprendizaje local de extremo a extremo en enfoque de DL presentado en [80].

Dataset	VGGFACE	VGGFACE FineTuning	Deep RBF	DRBF-16	CNN-16
CK+	0.2113	0.8014	0.1166	**0.8520**	0.8381
JAFFE	0.1428	0.2087	0.15625	**0.6429**	0.5971
CK+JAFFE	0.2511	0.4111	0.1435	**0.8726**	0.8594
iCV MEFED	0.0200	0.0484	0.02	**0.125**	0.0943

Tabla 10: Resutados *Top-1 Accuracy* al evaluar la red con VGG FACE (diagrama Fig.27), *Deep RBF* (diagrama Fig.13), *Deep RBF 16-MOD* (diagrama Fig.28) y *CNN 16-MOD* (diagrama Fig. 29).

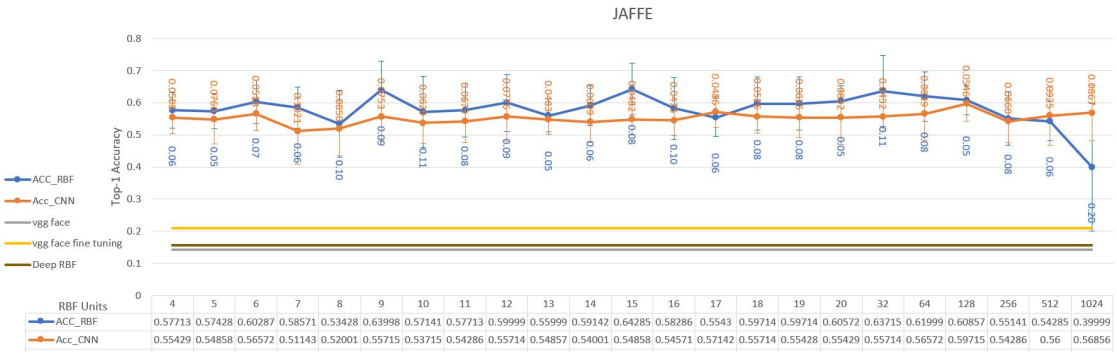
Las gráficas presentadas en la Figuras. 30, 31, 32 y 33 reportan resultados en *Top-1 Accuracy* de la evaluación de las arquitecturas *Deep RBF 16-MOD* (diagrama Fig.28) y *CNN 16-MOD* (diagrama Fig.29) para un grupo de neuronas RBF. La curva color azul representa los resultados de exactitud en el reconocimiento de emociones para cada cierto número de unidades RBF en cada conjunto de datos. La curva color naranja representa los mismos resultados pero sobre 16-Mod que adaptan capas densas con N número de neuronas.

En la Tabla 11 para CK+JAFFE se observa que la distribución de las predicciones por clase, los resultados tienen un sesgo en el modelo hacia CK+. Tal resultado se puede atribuir a la distribución de los datos, ya que en el caso de JAFFE se tiene un conjunto de datos muy reducido de imágenes en comparación con CK+.



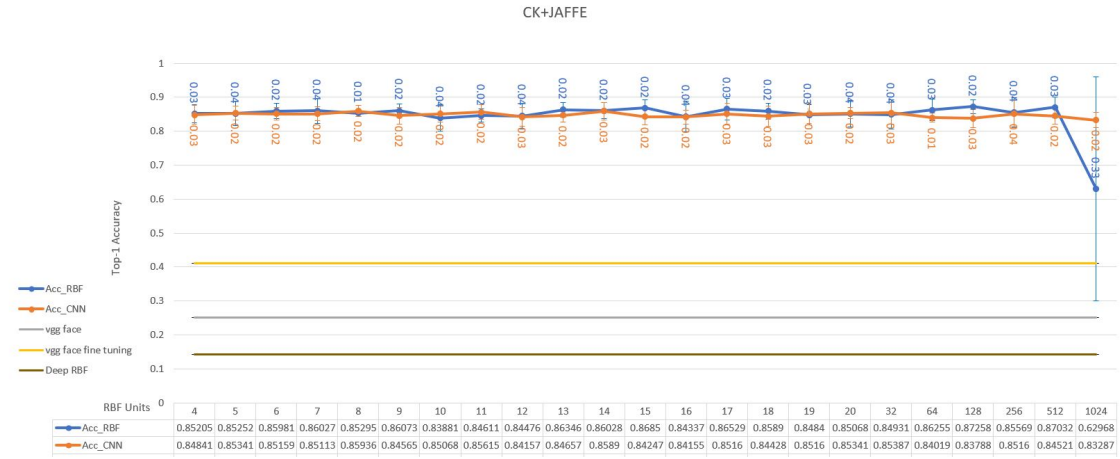
RBF Units	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	32	64	128	256	512	1024
ACC_RBF	0.85	0.844	0.844	0.837	0.848	0.841	0.851	0.851	0.843	0.838	0.841	0.852	0.85	0.844	0.846	0.838	0.848	0.838	0.846	0.851	0.841	0.836	0.78
Acc_CNN	0.832	0.83	0.84	0.827	0.838	0.831	0.828	0.832	0.826	0.828	0.828	0.824	0.828	0.834	0.828	0.833	0.841	0.832	0.831	0.833	0.833	0.825	0.831

Figura 30: Resultados *Top-1 accuracy* sobre el conjunto de datos CK+.



RBF Units	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	32	64	128	256	512	1024
ACC_RBF	0.57713	0.57428	0.60287	0.58571	0.53428	0.63998	0.57141	0.57713	0.59999	0.55999	0.59142	0.64285	0.58286	0.5543	0.59714	0.59714	0.60572	0.63715	0.61999	0.60857	0.55141	0.54285	0.39999
Acc_CNN	0.55429	0.54858	0.56572	0.51143	0.52001	0.55715	0.53715	0.54286	0.55714	0.54857	0.54001	0.54858	0.54571	0.57142	0.55714	0.55428	0.55429	0.55714	0.56572	0.59715	0.54286	0.56	0.56856

Figura 31: Resultados *Top-1 accuracy* sobre el conjunto de datos JAFFE.



RBF Units	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	32	64	128	256	512	1024
Acc_RBF	0.85205	0.85252	0.85981	0.86027	0.85295	0.86073	0.83881	0.84611	0.84476	0.86346	0.86028	0.8685	0.84337	0.86529	0.8589	0.8484	0.85068	0.84931	0.86255	0.87258	0.85569	0.87032	0.62968
Acc_CNN	0.84841	0.85341	0.85159	0.85113	0.85936	0.84565	0.85068	0.85615	0.84157	0.84657	0.8589	0.84247	0.84155	0.8516	0.84428	0.8516	0.85341	0.85387	0.84019	0.83788	0.8516	0.84521	0.83287

Figura 32: Resultados *Top-1 accuracy* sobre el conjunto de datos CK+JAFFE.

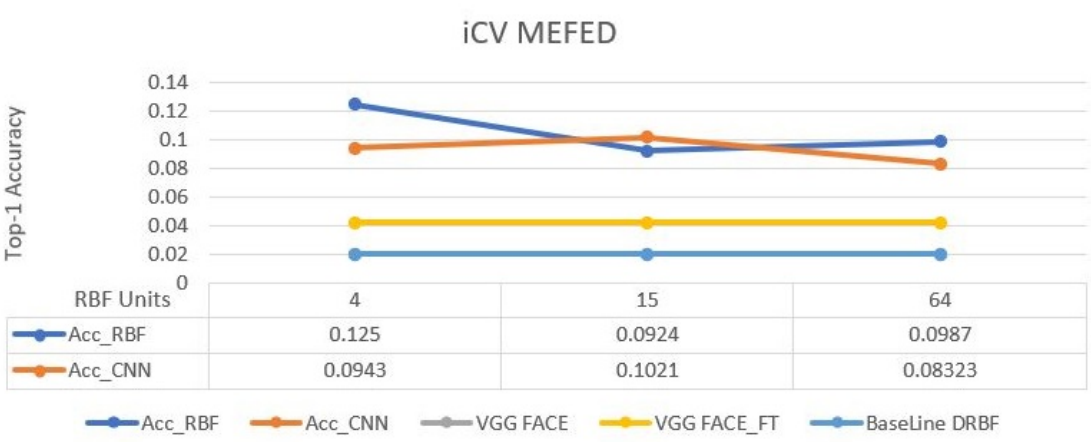


Figura 33: Resultados *Top-1 accuracy* sobre el conjunto de datos iCVMEFED.

	CK+			JAFFE		
Emoción	**Incorrectas**	**Correctas**	**Total Imágenes**	**Incorrectas**	**Correctas**	**Total Imágenes**
ANGRY	6	21	27	5	0	5
DISGUST	0	35	35	0	5	5
FEAR	6	9	15	4	1	5
HAPPY	0	41	41	0	5	5
SAD	9	7	16	3	2	5
SURPRISE	0	50	50	5	0	5
Total general imágenes	21	163	184	22	13	35

Tabla 11: Evaluación del modelo *Deep RBF 16-MOD* con 15 unidades RBF.

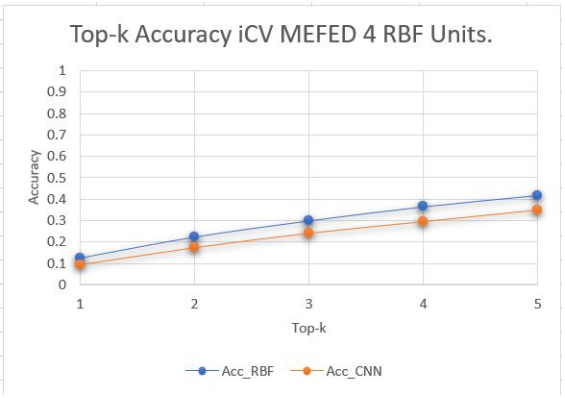
Las gráficas que se ilustran en la Fig. 35 son un grupo reducido de unidades RBF $RBF_Units = [4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 32, 64]$. Para observar más a detalle la dispersión de los datos. Se analiza que para CK+JAFFE los resultados son similares al probar en ambos modelos, tanto local como global. Para CK+ y JAFFE la dispersión muestra que la *Deep RBF 16-MOD* tiene un mejor desempeño al reconocer emociones.

Para el conjunto de datos iCV MEFED la evaluación se hace para el conjunto de unidades RBF *Units = [4, 15, 64]*. En la gráfica de la Fig.34 se muestran los resultados del *Top-k accuracy* para iCV MEFED obtenidos con el esquema LWDL y el actual estado del arte. El método que obtuvo la más alta exactitud en el reconocimiento de 50 emociones compuestas adapta la combinación de información geométrica de la cara con información de texturas. Básicamente su modelo consisten adaptar una CNN que concatena en su capa totalmente conectada los *facial landmark* como la información geométrica de la cara.

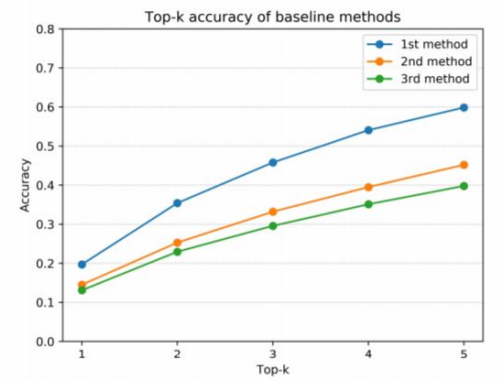
Los puntos de referencia faciales (*Facial landmark*) son un conjunto de puntos clave en las imágenes de rostros humanos. Estos puntos están definidos por sus coordenadas reales (x, y) en la imagen y pueden obtener información sobre las esquinas de la boca, las esquinas de los ojos, la silueta de las mandíbulas entre otros [6].

En la figura 34 se muestra la comparación de los resultados obtenidos con el esquema LWDL y el estado-del-arte para iCV MEFED. Existe una diferencia entre el conjunto de datos usado con nuestros esquemas

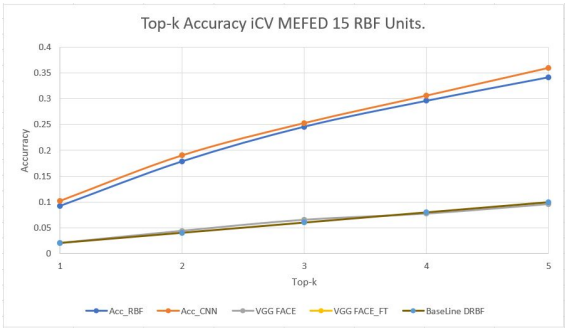
y con el de los modelos propuestos. En el caso del estado del arte para su entrenamiento toma el conjunto de entrenamiento y validación y lo dividen en aproximadamente el 10 % de las muestras. En el conjunto de validación hay 2,250 imágenes de 9 individuos. Las 20,719 imágenes restantes se usan en la fase de entrenamiento. Esto se hace para que todo los individuos estén contenidos en el conjunto de entrenamiento. El esquema LWDL usa el conjunto de datos sin hacer modificaciones a la base de datos original, tomando 15,969 imágenes de entrenamiento y 7,000 de validación. Al igual que en los métodos del estado-del-arte, se deja el resto para prueba y son 5,751. La gráfica 34c muestra los resultados alcanzados con *Deep RBF 16-MOD* y *CNN 16-MOD*.



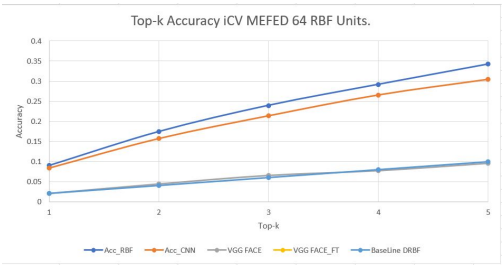
(a) Resultados *Top-k accuracy* de la evaluación de las arquitecturas *Deep RBF 16-MOD* y *CNN 16-MOD* sobre el conjunto de datos iCV MEFED.



(b) Resultados *Top-k accuracy* del actual estado-del-arte en el conjunto de datos iCV MEFED alcanzando un *Top-1 Accuracy* de 0.1980 en [28], [38] alcanza 0.1470 y [77] de 0.123. Figura reproducida de [27].

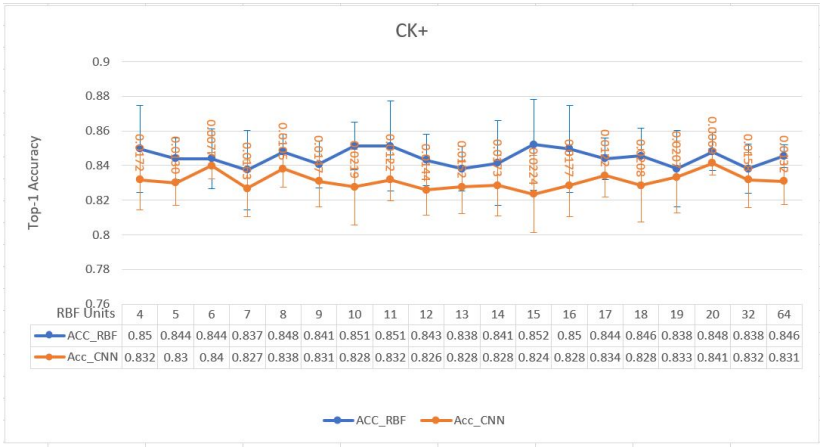


(c) Resultados *Top-k accuracy* de la evaluación de las arquitecturas *Deep RBF 16-MOD* y *CNN 16-MOD* sobre el conjunto de datos iCV MEFED con 15 unidades RBF.

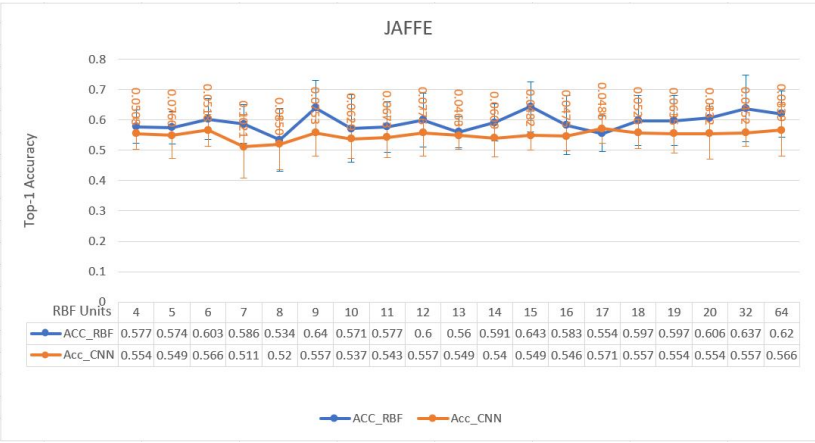


(d) Resultados *Top-k accuracy* de la evaluación de las arquitecturas *Deep RBF 16-MOD* y *CNN 16-MOD* sobre el conjunto de datos iCV MEFED con 64 unidades RBF.

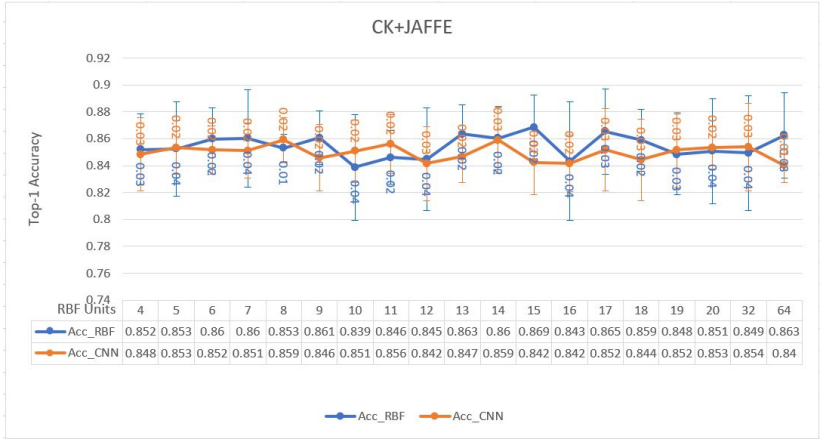
Figura 34: Gráficas de resultados obtenidos para el conjunto de datos iCV MEFED.



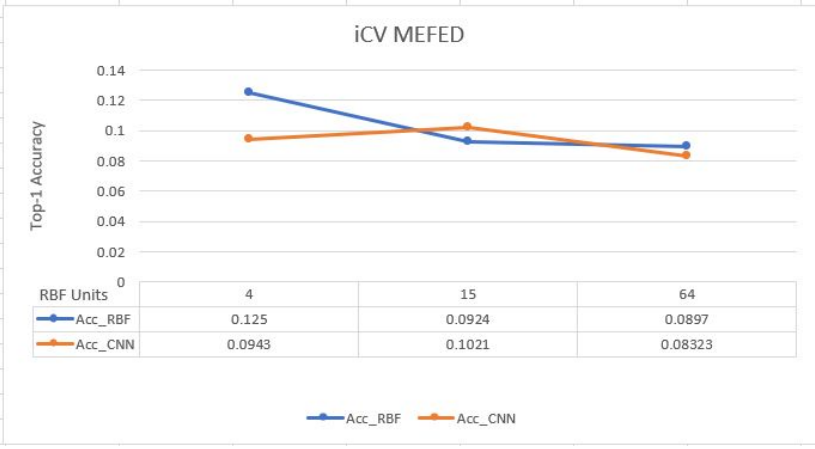
(a) Resultados *Top-1 accuracy* sobre el conjunto de datos CK+.



(b) Resultados *Top-1 accuracy* sobre el conjunto de datos JAFFE.



(c) Resultados *Top-1 accuracy* sobre el conjunto de datos CK+JAFFE.



(d) Resultados *Top-1 accuracy* sobre el conjunto de datos iCV MEFED.

Figura 35: Resultados *Top-1 accuracy* de la evaluación de las arquitecturas *Deep RBF 16-MOD* (diagrama Fig.28) y *CNN 16-MOD* (diagrama Fig.29) para un subconjunto de neuronas.

6. Conclusiones

6.1. Comparativa entre los métodos de aprendizaje local y global para reconocer emociones aparentes en imágenes.

En los experimentos iniciales se buscaba demostrar que los métodos de aprendizaje local tienen mejor desempeño si se elimina la fase de clasificación de la CNN, es decir, si se reemplaza el clasificador (MLP) por algún clasificador basado en aprendizaje local. Los resultados que obtuvimos (ver Tabla. 8) muestran que el usar una red CNN como extractor automático de características y se entrena un método de aprendizaje local tiene un buen desempeño en el reconocimiento de emociones en imágenes. Esto nos da pie a pensar que un método que englobe el aprendizaje local de extremo a extremo en una CNN, es decir que no sólo sea un apilamiento de arquitecturas puede incrementar el desempeño en el reconocimiento de emociones en imágenes.

6.2. Evaluación del esquema preliminar del LWDL en el reconocimiento de emociones aparentes en imágenes mediante el análisis de expresiones faciales.

Dado que la comparativa de la Sec.5.1 demostró que un método de aprendizaje local tiene un mejor desempeño en el ER. En este trabajo se presentó el análisis de 4 conjuntos de datos para el ER sobre enfoques que adaptan la localidad en DL. En los resultados mostrados anteriormente en la Tabla. 10 para dos de los conjuntos de datos el esquema LWDL (*Deep RBF 16-MOD*) mostraron un desempeño superior al alcanzado por (*CNN 16-MOD*). Si bien, el esquema LWDL esta motivado en las RBFN, el modelo parcial contempla la inicialización de las unidades RBF de forma aleatoria y se alcanzan resultados competitivos. Tales resultados nos dan pie a inferir que si inicializamos las unidades RBF utilizando algoritmos de agrupamiento podemos mejorar los resultados alcanzados. Así mismo se observó que la idea que se propuso para reducir la alta dimensionalidad de entrada en cada unidad RBF, mostró un buen desempeño no obstante queda trabajo por hacer para evaluar el desempeño de la *Deep RBF 16-MOD* cuando le anteceden encoders.

6.3. Trabajo Actual y futuro

Actualmente se esta trabajando en

- El diseño de una mejora del esquema LWDL.
- Experimentos usando la Deep RBF 16-MOD en conjuntos de datos de grano fino, no balanceados y para el reconocimiento de emociones.
- Análisis y preparación del conjunto de datos para emociones como EmotioNet.

El trabajo futuro se enfocará de acuerdo con la metodología propuesta en la Sec.4.7.

Referencias

[1] Osama Abu Abbas. Comparisons between data clustering algorithms. *International Arab Journal of Information Technology (IAJIT)*, 5(3), 2008.

[2] Md Zahangir Alom, Tarek M Taha, Chris Yakopcic, Stefan Westberg, Paheding Sidike, Mst Shamima Nasrin, Mahmudul Hasan, Brian C Van Essen, Abdul AS Awwal, and Vijayan K Asari. A state-of-the-art survey on deep learning theory and architectures. *Electronics*, 8(3):292, 2019.

[3] Umut Asan and Secil Ercan. *An Introduction to Self-Organizing Maps*, pages 299–319. 01 2012.

[4] Christopher G Atkeson, Andrew W Moore, and Stefan Schaal. Locally weighted learning. In *Lazy learning*, pages 11–73. Springer, 1997.

[5] Christopher G Atkeson, Andrew W Moore, and Stefan Schaal. Locally weighted learning for control. In *Lazy learning*, pages 75–113. Springer, 1997.

[6] Soufiane Belharbi, Clément Chatelain, Romain Hérault, and Sébastien Adam. Input/output deep architecture for structured output problems. *arXiv preprint arXiv:1504.07550*, 2015.

[7] Yoshua Bengio et al. Learning deep architectures for ai. *Foundations and trends® in Machine Learning*, 2(1):1–127, 2009.

[8] Ran Breuer and Ron Kimmel. A deep learning perspective on the origin of facial expressions. *arXiv preprint arXiv:1705.01842*, 2017.

[9] Zongwu Cai. Weighted nadaraya–watson regression estimation. *Statistics & probability letters*, 51(3):307–318, 2001.

[10] Dallas Card, Michael Zhang, and Noah A Smith. Deep weighted averaging classifiers. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pages 369–378. ACM, 2019.

[11] Wen-Sheng Chu, Fernando De la Torre, and Jeffrey F Cohn. Learning spatial and temporal cues for multi-label facial action unit detection. In *2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017)*, pages 25–32. IEEE, 2017.

[12] Wen-Sheng Chu, Fernando De la Torre, and Jeffrey F Cohn. Selective transfer machine for personalized facial expression analysis. *IEEE transactions on pattern analysis and machine intelligence*, 39(3):529–545, 2017.

[13] G Colmenares. Función de base radial. radial basis function (rbf)[en línea], 2007.

[14] Ciprian Adrian Corneanu, Marc Oliu Simón, Jeffrey F Cohn, and Sergio Escalera Guerrero. Survey on rgb, 3d, thermal, and multimodal approaches for facial expression recognition: History, trends, and affect-related applications. *IEEE transactions on pattern analysis and machine intelligence*, 38(8):1548–1568, 2016.

[15] Li Deng, Dong Yu, et al. Deep learning: methods and applications. *Foundations and Trends® in Signal Processing*, 7(3–4):197–387, 2014.

[16] Samira Ebrahimi Kahou, Vincent Michalski, Kishore Konda, Roland Memisevic, and Christopher Pal. Recurrent neural networks for emotion recognition in video. In *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction*, pages 467–474, 2015.

[17] Paul Ekman. Facial action coding system. 1977.

[18] Paul Ekman and Wallace V Friesen. Constants across cultures in the face and emotion. *Journal of personality and social psychology*, 17(2):124, 1971.

[19] Paul Ekman, E Richard Sorenson, and Wallace V Friesen. Pan-cultural elements in facial displays of emotion. *Science*, 164(3875):86–88, 1969.

[20] Peter Englert. Locally weighted learning. In *Seminar Class on Autonomous Learning Systems*, 2012.

[21] Chollet Francois. Deep learning with python, 2017.

[22] Aurélien Géron. *Hands-on machine learning with Scikit-Learn and TensorFlow: concepts, tools, and techniques to build intelligent systems*. .ºReilly Media, Inc.", 2017.

[23] Deepak Ghimire and Joonwhoan Lee. Geometric feature-based facial expression recognition in image sequences using multi-class adaboost and support vector machines. *Sensors*, 13(6):7714–7734, 2013.

[24] Deepak Ghimire, Joonwhoan Lee, Ze-Nian Li, and Sunghwan Jeong. Recognition of facial expressions based on salient geometric features and support vector machines. *Multimedia Tools and Applications*, 76(6):7921–7946, 2017.

[25] Alexander AS Gunawan et al. Face expression detection on kinect using active appearance model and fuzzy logic. *Procedia Computer Science*, 59:268–274, 2015.

[26] Gongde Guo, Hui Wang, David Bell, Yaxin Bi, and Kieran Greer. Knn model-based approach in classification. In *OTM Confederated International Conferences.ºn the Move to Meaningful Internet Systems*
, pages 986–996. Springer, 2003.

[27] Jianzhu Guo, Zhen Lei, Jun Wan, Egils Avots, Noushin Hajarolasvadi, Boris Knyazev, Artem Kuharenko, Julio C Silveira Jacques Junior, Xavier Baró, Hasan Demirel, et al. Dominant and complementary emotion recognition from still images of faces. *IEEE Access*, 6:26391–26403, 2018.

[28] Jianzhu Guo, Shuai Zhou, Jinlin Wu, Jun Wan, Xiangyu Zhu, Zhen Lei, and Stan Z Li. Multi-modality network with visual and geometrical information for micro emotion recognition. In *2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017)*, pages 814–819. IEEE, 2017.

[29] Jihun Hamm, Christian G Kohler, Ruben C Gur, and Ragini Verma. Automated facial action coding system for dynamic analysis of facial expressions in neuropsychiatric disorders. *Journal of neuroscience methods*, 200(2):237–256, 2011.

[30] SL Happy, Anjith George, and Aurobinda Routray. A real time facial expression classification system using local binary patterns. In *2012 4th International conference on intelligent human computer interaction (IHCI)*, pages 1–5. IEEE, 2012.

[31] Xuanyu He and Wei Zhang. Emotion recognition by assisted learning with convolutional neural networks. *Neurocomputing*, 291:187–194, 2018.

[32] Kuo-Wei Hsu and Jaideep Srivastava. An empirical study of applying ensembles of heterogeneous classifiers on imperfect data. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 28–39. Springer, 2009.

[33] Maryam Imani and Gholam Ali Montazer. A survey of emotion recognition methods with emphasis on e-learning environments. *Journal of Network and Computer Applications*, page 102423, 2019.

[34] Md IqbalQuraishi, J Pal Choudhury, Mallika De, and Purbaja Chakraborty. A framework for the recognition of human emotion using soft computing models. *International Journal of Computer Applications*, 40(17):50–55, 2012.

[35] Rachael E Jack, Oliver GB Garrod, Hui Yu, Roberto Caldara, and Philippe G Schyns. Facial expressions of emotion are not culturally universal. *Proceedings of the National Academy of Sciences*, 109(19):7241–7244, 2012.

[36] Deepak Kumar Jain, Pourya Shamsolmoali, and Paramjit Sehdev. Extended deep neural network for facial emotion recognition. *Pattern Recognition Letters*, 120:69–74, 2019.

[37] Cijo Jose, Prasoon Goyal, Parv Aggrwal, and Manik Varma. Local deep kernel learning for efficient non-linear svm prediction. In *International conference on machine learning*, pages 486–494, 2013.

[38] Vahid Kazemi and Josephine Sullivan. One millisecond face alignment with an ensemble of regression trees. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1867–1874, 2014.

[39] Byoung Ko. A brief review of facial emotion recognition based on visual information. *sensors*, 18(2):401, 2018.

[40] T. Kohonen. The self-organizing map. *Proceedings of the IEEE*, 78(9):1464–1480, Sep. 1990.

[41] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[42] Lubor Ladicky and Philip Torr. Locally linear support vector machines. In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, pages 985–992, 2011.

[43] Ya Le and Xuan Yang. Tiny imagenet visual recognition challenge. *CS 231N*, 2015.

[44] Robert W Levenson. The intrapersonal functions of emotion. *Cognition & Emotion*, 13(5):481–504, 1999.

[45] Shan Li and Weihong Deng. Deep facial expression recognition: A survey. *arXiv preprint arXiv:1804.08348*, 2018.

[46] Paul Pu Liang, Amir Zadeh, and Louis-Philippe Morency. Multimodal local-global ranking fusion for emotion recognition. In *Proceedings of the 2018 on International Conference on Multimodal Interaction*, pages 472–476. ACM, 2018.

[47] Patrick Lucey, Jeffrey F Cohn, Takeo Kanade, Jason Saragih, Zara Ambadar, and Iain Matthews. The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops*, pages 94–101. IEEE, 2010.

[48] David Matsumoto, Theodora Consolacion, Hiroshi Yamada, Ryuta Suzuki, Brenda Franklin, Sunita Paul, Rebecca Ray, and Hideko Uchida. American-japanese cultural differences in judgements of emotional expressions of different intensities. *Cognition & Emotion*, 16(6):721–747, 2002.

[49] Haotian Miao, Yifei Zhang, Weipeng Li, Haoran Zhang, Daling Wang, and Shi Feng. Chinese multimodal emotion recognition in deep and traditional machine leaming approaches. In *2018 First Asian Conference on Affective Computing and Intelligent Interaction (ACII Asia)*, pages 1–6. IEEE, 2018.

[50] Tom Mitchell, Bruce Buchanan, Gerald DeJong, Thomas Dietterich, Paul Rosenbloom, and Alex Waibel. Machine learning. *Annual review of computer science*, 4(1):417–433, 1990.

[51] Roman Neruda and Petra Kudová. Learning methods for radial basis function networks. *Future Generation Computer Systems*, 21(7):1131–1142, 2005.

[52] David Nova and Pablo A Estévez. A review of learning vector quantization classifiers. *Neural Computing and Applications*, 25(3-4):511–524, 2014.

[53] Mark JL Orr. Regularization in the selection of radial basis function centers. *Neural computation*, 7(3):606–623, 1995.

[54] Andrew Ortony, G Clore, and Allan Collins. The cognitive structure of emotions. cam (bridge university press. *Cambridge, England*, 1988.

[55] Nicolas Papernot and Patrick McDaniel. Deep k-nearest neighbors: Towards confident, interpretable and robust deep learning. *arXiv preprint arXiv:1803.04765*, 2018.

[56] Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, et al. Deep face recognition. In *bmvc*, volume 1, page 6, 2015.

[57] Josh Patterson and Adam Gibson. *Deep learning: A practitioner's approach*. .º'Reilly Media, Inc.", 2017.

[58] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[59] Rosalind W. Picard, Elias Vyzas, and Jennifer Healey. Toward machine emotional intelligence: Analysis of affective physiological state. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (10):1175–1191, 2001.

[60] Diah Anggraeni Pitaloka, Ajeng Wulandari, T Basaruddin, and Dewi Yanti Liliana. Enhancing cnn with preprocessing stage in automatic emotion recognition. *Procedia computer science*, 116:523–529, 2017.

[61] Michael JD Powell. Radial basis functions for multivariable interpolation: a review. *Algorithms for approximation*, 1987.

[62] Bayu Yudha Pratama and Riyanarto Sarno. Personality classification based on twitter text using naive bayes, knn and svm. In *2015 International Conference on Data and Software Engineering (ICoDSE)*, pages 170–174. IEEE, 2015.

[63] Antti Puurula and Albert Bifet. Ensembles of sparse multinomial classifiers for scalable text classification. In *Proceedings of the 2012 ECML/PKDD Discovery Challenge Workshop on Large-Scale Hierarchical Text Classification, Bristol*, 2012.

[64] David Sander, Didier Grandjean, Gilles Pourtois, Sophie Schwartz, Mohamed L Seghier, Klaus R Scherer, and Patrik Vuilleumier. Emotion and attention interactions in social cognition: brain regions involved in processing anger prosody. *Neuroimage*, 28(4):848–858, 2005.

[65] R Santhoshkumar and M Kalaiselvi Geetha. Deep learning approach for emotion recognition from human body movements with feedforward deep convolution neural networks. *Procedia Computer Science*, 152:158–165, 2019.

[66] Nicola Segata, Edoardo Pasolli, Farid Melgani, and Enrico Blanzieri. Local svm approaches for fast and accurate classification of remote-sensing images. *International journal of remote sensing*, 33(19):6186–6201, 2012.

[67] Alireza Sepas-Moghaddam, Ali Etemad, Paulo Lobato Correia, and Fernando Pereira. A deep framework for facial emotion recognition using light field images. In *2019 8th International Conference on Affective Computing and Intelligent Interaction (ACII)*, pages 1–7. IEEE, 2019.

[68] Jie Shao and Yongsheng Qian. Three convolutional neural network models for facial expression recognition in the wild. *Neurocomputing*, 355:82–92, 2019.

[69] Chawin Sitawarin and David Wagner. On the robustness of deep k-nearest neighbors. *arXiv preprint arXiv:1903.08333*, 2019.

[70] Myunghoon Suk and Balakrishnan Prabhakaran. Real-time mobile facial expression recognition system-a case study. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 132–137, 2014.

[71] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alexander A Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. In *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

[72] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.

[73] Paweł Tarnowski, Marcin Kołodziej, Andrzej Majkowski, and Remigiusz J Rak. Emotion recognition using facial expressions. *Procedia Computer Science*, 108:1175–1184, 2017.

[74] Petra Vidnerová and Roman Neruda. Deep networks with rbf layers to prevent adversarial examples. In *International Conference on Artificial Intelligence and Soft Computing*, pages 257–266. Springer, 2018.

[75] Robert Walecki, Vladimir Pavlovic, Björn Schuller, Maja Pantic, et al. Deep structured learning for facial action unit intensity estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3405–3414, 2017.

[76] Wei Wei, Qingxuan Jia, and Gang Chen. Real-time facial expression recognition for affective computing based on kinect. In *2016 IEEE 11th Conference on Industrial Electronics and Applications (ICIEA)*, pages 161–165. IEEE, 2016.

[77] Yandong Wen, Kaipeng Zhang, Zhifeng Li, and Yu Qiao. A discriminative feature learning approach for deep face recognition. In *European conference on computer vision*, pages 499–515. Springer, 2016.

[78] Aron Yu and Kristen Grauman. Fine-grained comparisons with attributes. In *Visual Attributes*, pages 119–154. Springer, 2017.

[79] Pingpeng Yuan, Yuqin Chen, Hai Jin, and Li Huang. Msvm-knn: Combining svm and k-nn for multi-class text classification. In *IEEE international workshop on Semantic Computing and Systems*, pages 133–140. IEEE, 2008.

[80] Pourya Habib Zadeh, Reshad Hosseini, and Suvrit Sra. Deep-rbf networks revisited: Robust classification with rejection. *arXiv preprint arXiv:1812.03190*, 2018.

[81] Cleber Zanchettin, Byron Leite Dantas Bezerra, and Washington W Azevedo. A knn-svm hybrid model for cursive handwriting recognition. In *The 2012 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2012.

[82] Hao Zhang, Alexander C Berg, Michael Maire, and Jitendra Malik. Svm-knn: Discriminative nearest neighbor classification for visual category recognition. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, pages 2126–2136. IEEE, 2006.