

# PREDICCIÓN DE TRÁFICO DE VÍDEO VBR CON REDES NEURONALES ARTIFICIALES

Carol Zuleimy Fernández Rodríguez.

Código: 20142005090

Inteligencia Computacional 2

## 1. Introducción

Las aplicaciones multimedia y, en particular, los flujos de vídeo de velocidad de bits variable VBR se están convirtiendo en componentes de tráfico importantes en las redes de alta velocidad y se ha recomendado ATM (modo de transferencia asincrónica) como vehículo de transporte para las redes digitales de servicios integrados de banda ancha (BISDN). Una aplicación importante en las redes ATM es proporcionar una transmisión en tiempo real, de baja pérdida y con un retardo mínimo de tráfico de velocidad de bits variable (VBR), que tiene muchas ráfagas, no es estacionario y está correlacionado. Para asegurar una buena calidad de servicio en este tipo de aplicaciones, es necesario conocer de antemano el tráfico para gestionar los recursos de la red, es decir, predecir el tráfico.

En este proyecto, adoptamos la metodología de redes neuronales para predecir el tráfico VBR representado en muestras de dos películas codificadas en MPEG 4, donde a partir de una trama, se predicen las 5 siguientes; se desarrollan y evalúan un conjunto de redes neuronales Feed Forward, donde se varía su arquitectura, funciones de activación y funciones de entrenamiento, para presentar una comparación de los resultados de cada red. Finalmente se presenta la red que se considera que mejor soluciona el problema y sea más conveniente de implementar.

## 2. Planteamiento del problema

La predicción del tráfico de red en redes integradas de banda ancha, como Internet y las redes de modo de transferencia asincrónica (ATM), ha atraído cada vez más actividades de investigación. Dado que diversas aplicaciones multimedia como videoconferencia, video a pedido y audio se vuelven cada vez más importantes en Internet; la asignación y administración del ancho de banda de la red, así como el control del tráfico, se han vuelto más complicados y desafiantes. El control de la red, por un lado, debe admitir diferentes tipos de clases de tráfico con su calidad de servicio requerida y, por otro lado, debe utilizar de manera eficiente los recursos de la red, como el ancho de banda de la red.

En general, hay dos categorías de aplicaciones de video de velocidad de bits variable (VBR), es decir, el video VBR pregrabado y las aplicaciones de video VBR en tiempo real. Para las aplicaciones de video pregrabadas, como video a pedido, todo el perfil de seguimiento de tráfico de video se puede analizar completamente con anticipación y determinar un ancho de banda efectivo cuando se entrega el video. Sin embargo, esto es difícil para situaciones de video en vivo, porque cualquier transmisión de video en vivo no se conoce de antemano y, por lo tanto, estimar su ancho de banda estático efectivo sería difícil e inexacto.

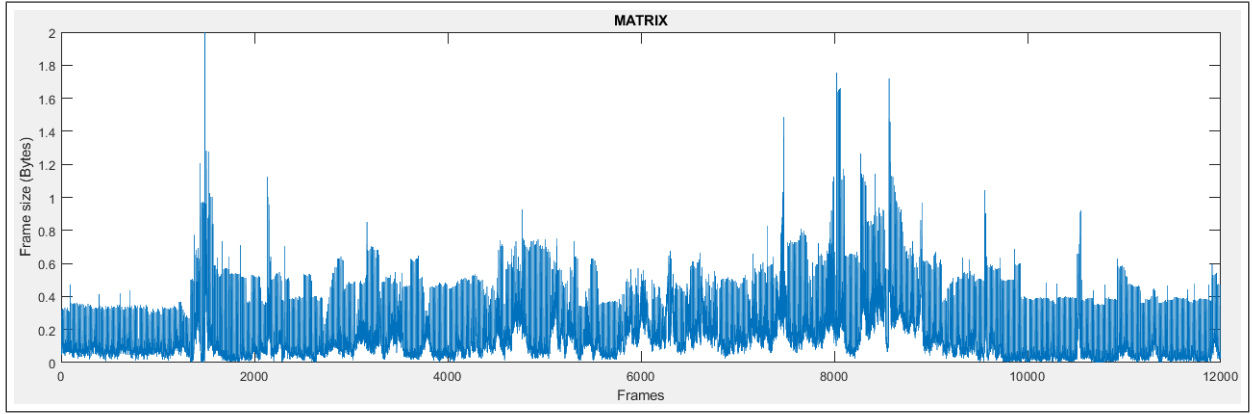


Figura 1: Muestra tráfico VBR para entrenamiento de la red (Matrix)

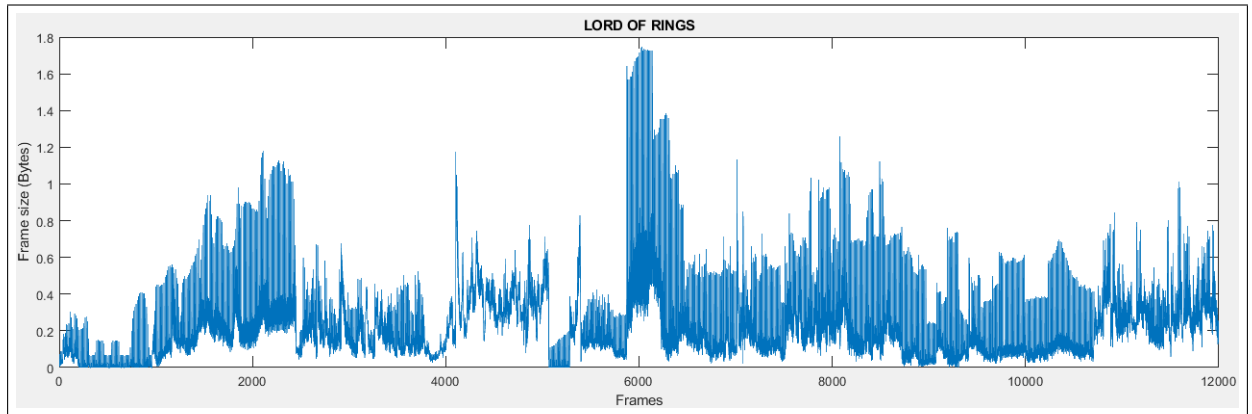


Figura 2: Muestra de tráfico VBR para prueba de red entrenada (Lord of the Rings)

Es por ello que la predicción del tráfico es importante para mejorar el funcionamiento de estas redes. Sin embargo, el tráfico de video VBR presenta altas fluctuaciones de tasa de bits durante períodos de tiempo cortos, lo que dificulta la predicción. Las redes neuronales se pueden utilizar eficazmente para superar este problema. En la literatura, el problema se ha evaluado principalmente utilizando redes neuronales Feed Forward, por lo que se buscará el mejor resultado usando este tipo de red neuronal, usando diferentes arquitecturas, funciones de activación y funciones de entrenamiento.

### 3. Desarrollo

#### 3.1. Obtención de los datos

Los datos que se utilizarán serán muestras de tráfico de vídeo de dos películas las cuales están codificadas en MPEG-4. MPEG-4 comprende un patrón alterno de tres cuadros (intracadro), (predictivo) y (bidireccional) con diferentes propiedades. Estos tres tipos de fotogramas se fusionan de forma determinista para formar la secuencia de vídeo MPEG-4 agregada. El tamaño de los fotogramas de vídeo varía drásticamente a medida que se genera la secuencia, lo que genera un tráfico de velocidad de bits variable (VBR).

Las dos muestras de tráfico de vídeo utilizadas, se codificaron en la Universidad Estatal de Arizona y se descargaron las tramas de su repositorio disponible en [1]. Se descargó una muestra de Matrix 1 y de Lord of the Rings 1; ambas con una longitud de 12000 tramas, tal como se muestra en las Figuras 1 y 2.

### 3.2. Transformación de los datos para la Red Neuronal

La predicción de tráfico de solo una trama adelante, es una predicción de tráfico a muy corto plazo. Cabe señalar que, al momento de administrar de ancho de banda, el proceso de reasignación de ancho de banda en sí mismo necesita tiempo de cálculo y otros recursos, por lo que no es práctico realizar dicha reasignación dinámica de ancho de banda con demasiada frecuencia. En otras palabras, no es práctico debido a tanta sobrecarga. En vista de esto, la predicción de tráfico a corto plazo de una trama por delante no es suficiente, así que, como primer intento de realizar la predicción a un plazo más amplio, se realizará a 5 tramas a partir de una trama actual.

Para ello se deben transformar los datos de forma tal, que el conjunto sea de una entrada y cinco salidas, es decir, que las muestras de 12000 tramas se deben dividir en 6 filas:

CONJUNTO	DIMENSIÓN
ENTRADA	1 Fila - 2000 Columnas
SALIDA	5 Filas - 2000 Columnas

### 3.3. Construcción Red Neuronal Feed Forward

Esta es la red que se utiliza con más frecuencia. En esta arquitectura, las señales pueden viajar solo hacia adelante y las neuronas en cualquier capa envían sus resultados solo a la siguiente capa. Normalmente, la red comprende una capa de entrada, una capa oculta de neuronas sigmoides y una capa de salida de neuronas lineales. La Figura 3 muestra la arquitectura de una red Feed-Forward con  $n$  entradas, una capa de  $m$  neuronas ocultas y  $p$  salidas.

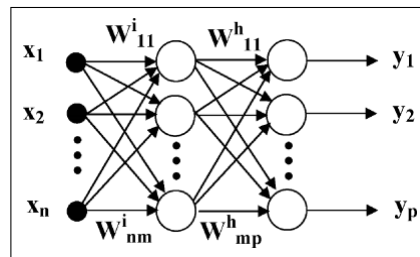


Figura 3: Red Feed Forward con  $n$  entradas y  $p$  salidas

Las redes neuronales que se construyen en este proyecto, se realizan en MATLAB, ya que posee herramientas, funciones y toolboxes especializadas para trabajar redes neuronales.

De acuerdo a las funciones de MATLAB con las que se pueden configurar las redes, en aras de experimentar con diferentes modelos para obtener el mejor, se creará un conjunto de redes neuronales donde se variará su estructura de capas y neuronas por capa, para dos funciones de activación diferentes y así mismo para tres funciones de entrenamiento, tal como se describe en la Figura 4.

Función de entrenamiento	Levenberg-Marquardt Backpropagation	Bayesian regularization Backpropagation	Resilient Backpropagation
Función de activación	Log-sigmoid	Hyperbolic tangent sigmoid	
Neuronas por capa	Número de capas		
5	1	2	3
10	1	2	3
15	1	2	3

Figura 4: Metodología de experimentación

### 3.4. Entrenamiento de la red

Además de las características descritas anteriormente, se ajustarán unos parámetros de la red que estarán fijos en todos los casos, tales como la división de los datos y criterios de parada del entrenamiento:

```
%División de datos
net.divideParam.trainRatio = 0.7; %Entrenamiento
net.divideParam.valRatio   = 0.2; %Validación
net.divideParam.testRatio  = 0.1; %Test

%Parámetros de entrenamiento (criterios de parada)
net.trainParam.max_fail = 10;    %Fallos máximos de validación
net.trainParam.epochs = 1000;   %Número máximo de épocas a entrenar
net.trainParam.goal = 0;        %Objetivo de rendimiento
net.trainParam.min_grad = 1e-10; %Gradiente de rendimiento mínimo
```

Figura 5: División de los datos y criterios de parada

La muestra de tráfico que se usará para entrenar la red, será la de *Matrix*, la cual se muestra en la Figura 1, para ello su entrada se denotará como  $x$  y las 5 salidas como  $t$ ; de esta forma, utilizando la función  $train(net, x, t)$ , se ingresan los datos mencionados. Finalmente, para observar los resultados de la red entrenada con los datos ingresados, se usa la función  $sim(net, x)$ , para simular la red con los datos de entrada  $x$ .

### 3.5. Validación de la red entrenada

La muestra de tráfico que se usará para validar la red entrenada, será una muestra distinta a la usada en el entrenamiento, para verificar que funciona bien ante datos nuevos, en este caso se usará la de *Lord of the Rings*, la cual se muestra en la Figura 2, para ello, sólo se usa la función  $sim(net, xx)$ , para simular la red entrenada anteriormente, con los datos de entrada de la muestra de validación, que se denota como  $xx$ .

### 3.6. Funciones de desempeño

El entrenamiento, cuyo objetivo es que la red neuronal sea capaz de reproducir el comportamiento subyacente en los datos aportados, consiste básicamente en la minimización de una función de coste o error, lo que equivale a que la salida de la red, se aproxima a la salida en los datos. Entre las funciones más usadas para problemas de este tipo, así como en la literatura de referencia, están el MSE y el MAE, los cuales están definidos en sus respectivas ecuaciones.

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - y_i')^2 \quad (1)$$

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - y_i'| \quad (2)$$

Donde  $n$  es el número de datos  $y_i$  representa los datos observados y  $y_i'$  representa los que predice la red neuronal.

Para la optimización de la red neuronal, un método muy usado es el del descenso del gradiente, donde calculan la variación del error al variar cada uno de los parámetros y luego modifican todos los parámetros de la red neuronal obteniendo un error menor. Se puede decir que es una búsqueda en serie de la solución o mínimo global. MATLAB cuenta con las funciones de entrenamiento, las cuales realizan dicho procedimiento y obtienen el gradiente resultante del entrenamiento de la red. Las que se usarán en este caso se ilustran en la Figura 4, en MATLAB, son denotadas como: *trainlm*, *trainbr* y *trainrp*, respectivamente.

## 4. Resultados

Se recopilaron todos los resultados de MSE, MAE y gradiente para la trama de entrenamiento y los resultados de MSE y MAE para la trama de validación. Para todos los casos se realizaron 5 experimentos, cuyos resultados al final se promediaron para obtener un único resultado. Las redes neuronales se variaron en cuanto a número de capas, capas por neurona, funciones de transferencia y funciones de entrenamiento, tal como se ilustró en la Figura 4. Finalmente, se graficaron los resultados con la trama de entrenamiento:

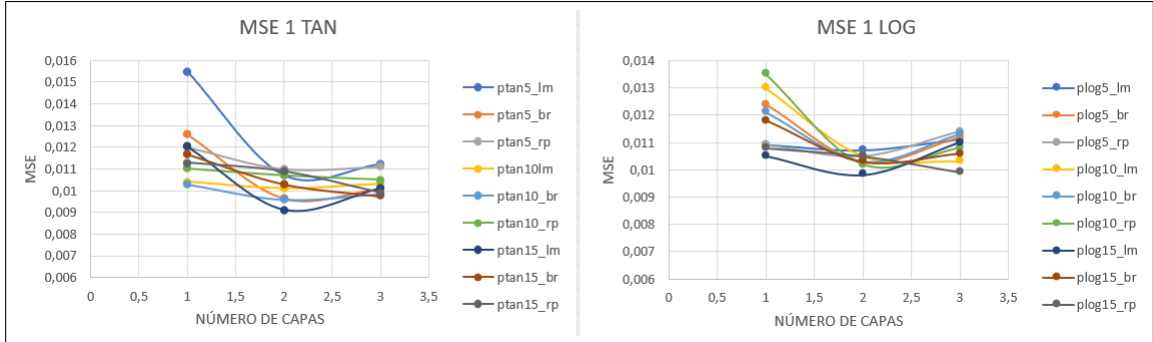


Figura 6: MSE trama de entrenamiento de todas las redes con función de transferencia Tangencial y Logarítmica

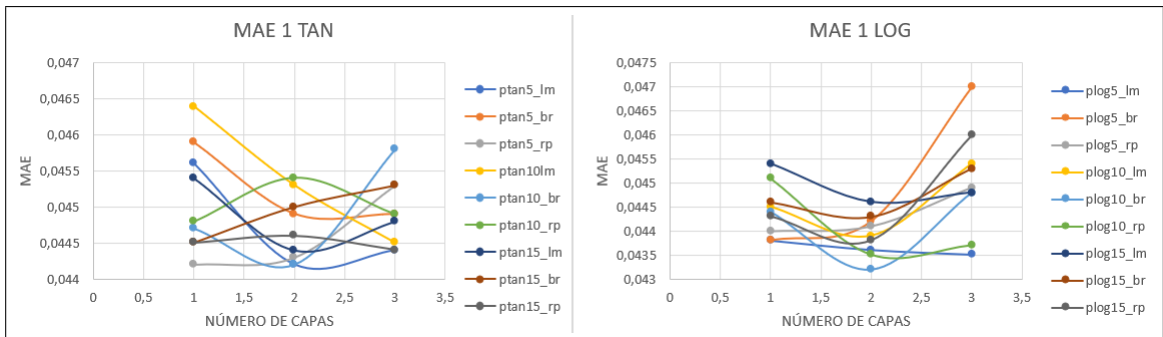


Figura 7: MAE trama de entrenamiento de todas las redes con función de transferencia Tangencial y Logarítmica

Así mismo, se grafican los de la trama de validación:

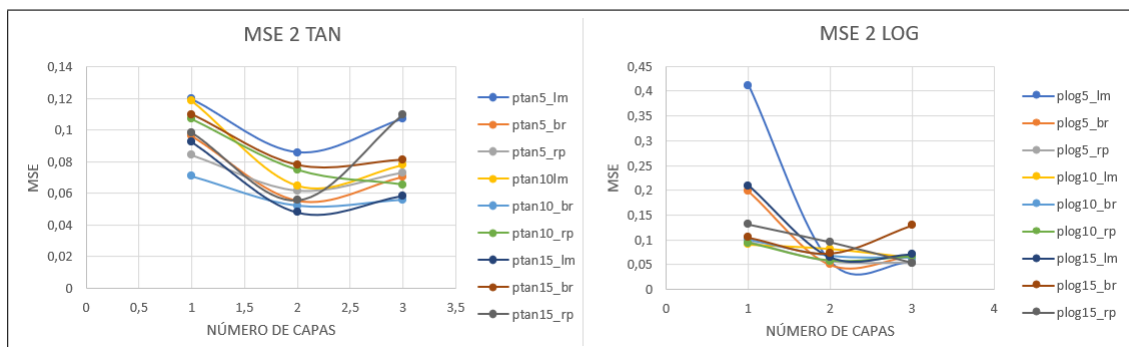


Figura 8: MSE trama de validación de todas las redes con función de transferencia Tangencial y Logarítmica

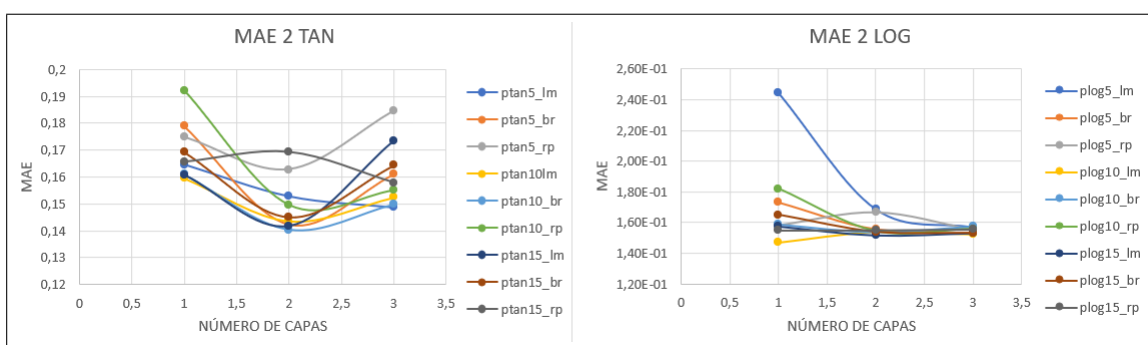


Figura 9: MAE trama de validación de todas las redes con función de transferencia Tangencial y Logarítmica

En las gráficas ilustradas anteriormente, se compilan los resultados de todas las redes con los diferentes números de neuronas y funciones de entrenamiento; tanto para la función de transferencia tangencial y logarítmica, donde el eje x indica el número de capas de cada una. De acuerdo a ello, lo más notable de los resultados, es que con 2 capas hubo un mejor desempeño en general, a partir de ello, se visualizaron las neuronas que obtuvieron los errores numéricamente más bajos en general, es decir, en su MSE y MAE para las dos tramas simuladas. Las dos redes que obtuvieron los mejores resultados son las siguientes:

#### 4.1. Mejores Soluciones

RED 1		RED 2	
Número de capas	2	Número de capas	2
Neuronas por capa	10	Neuronas por capa	15
Función entrenamiento	Bayesian Regularization	Función entrenamiento	Levenberg Marquardt
Función de transferencia	Tangencial	Función de transferencia	Tangencial
MSE trama entrenamiento	1.38 %	MSE trama entrenamiento	1.28 %
MSE trama validación	5.25 %	MSE trama validación	6.17 %
MAE trama entrenamiento	4.32 %	MAE trama entrenamiento	4.43 %
MAE trama validación	14.55 %	MAE trama validación	14.82 %
Gradiente	0.00178	Gradiente	2.46e -4
Iteraciones	18	Iteraciones	14

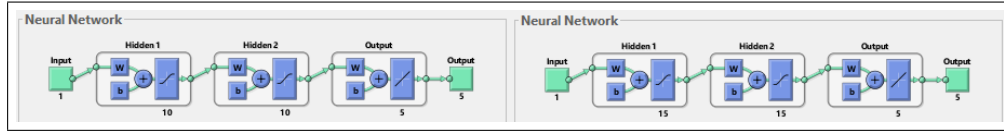


Figura 10: Arquitectura de las mejores redes

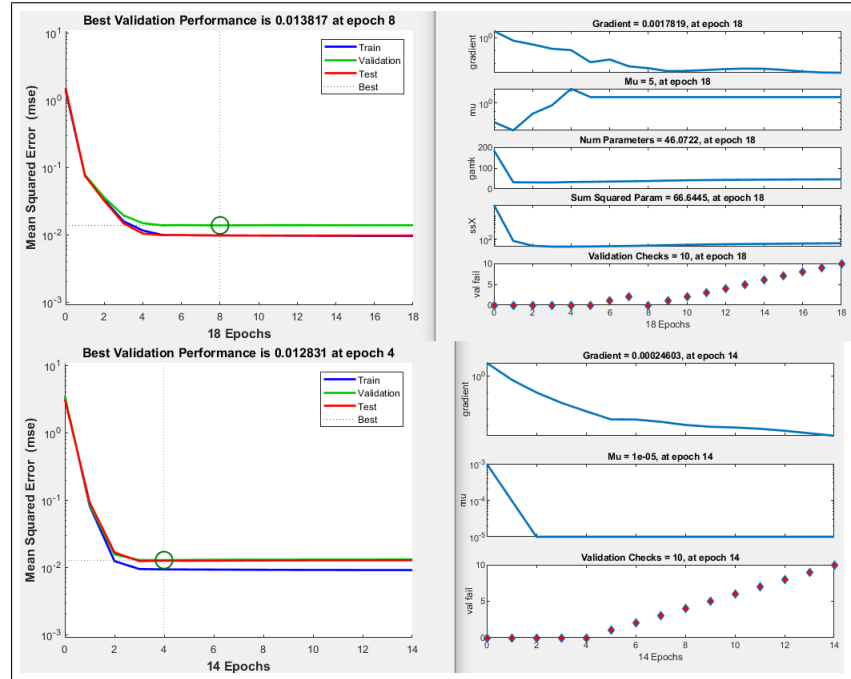


Figura 11: Performance y gradiente mejores redes

Los resultados de las tramas de simulación se vuelven a acomodar de forma tal de que vuelva a ser de una sola fila y así poder graficarlos sobre las tramas originales para comparar. Al ser una trama demasiado larga, se hacen dos acercamientos para observar mejor los resultados:

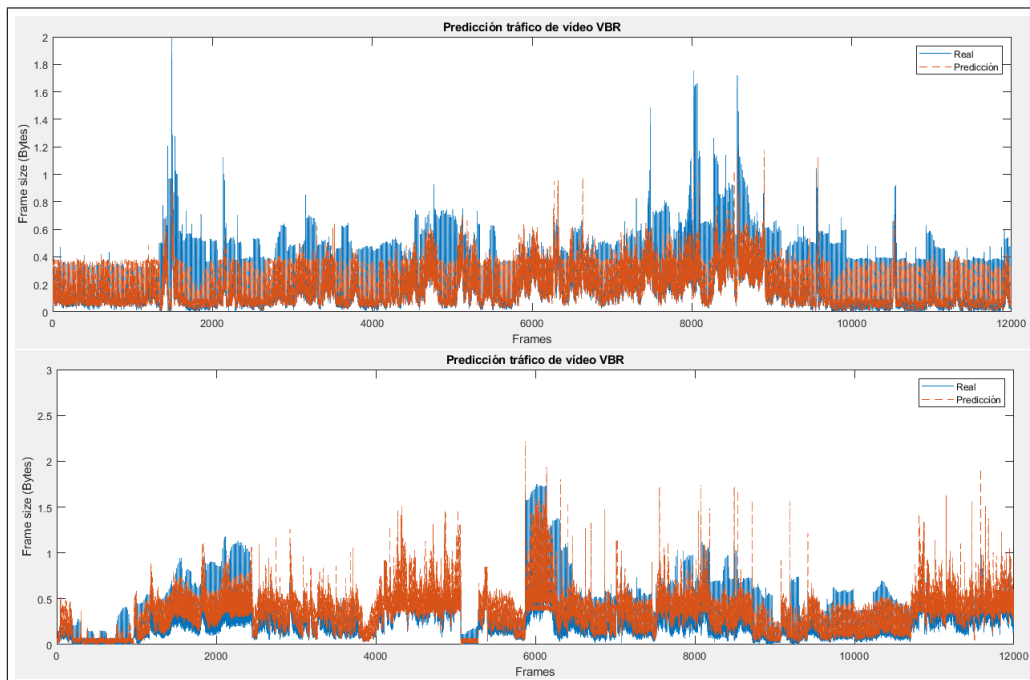


Figura 12: Comparación de resultados de las dos tramas con las originales

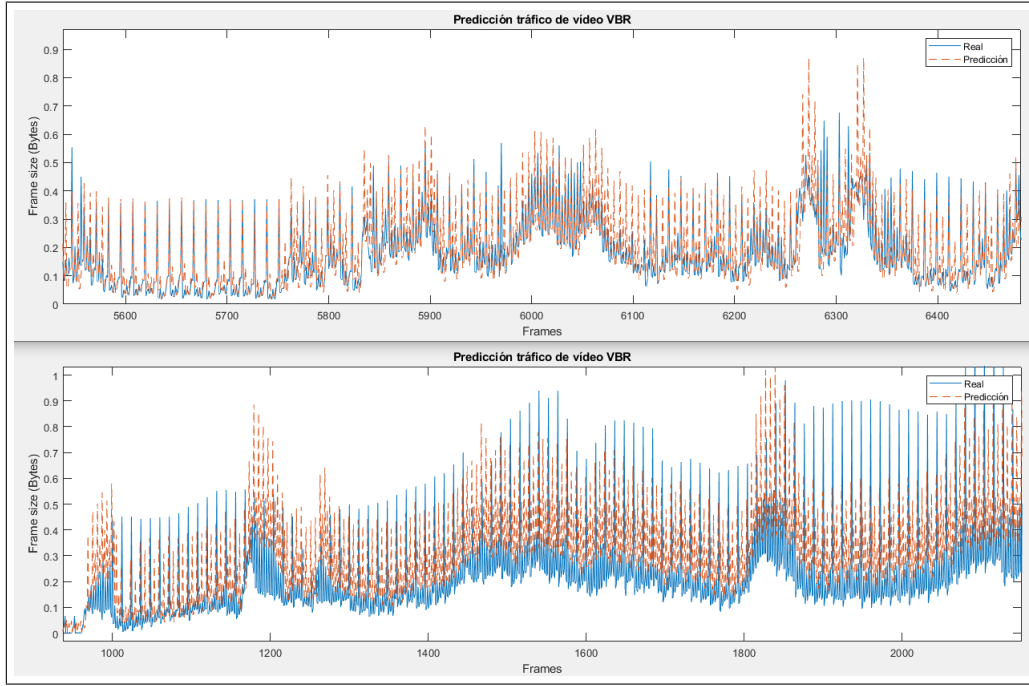


Figura 13: Acercamiento 1

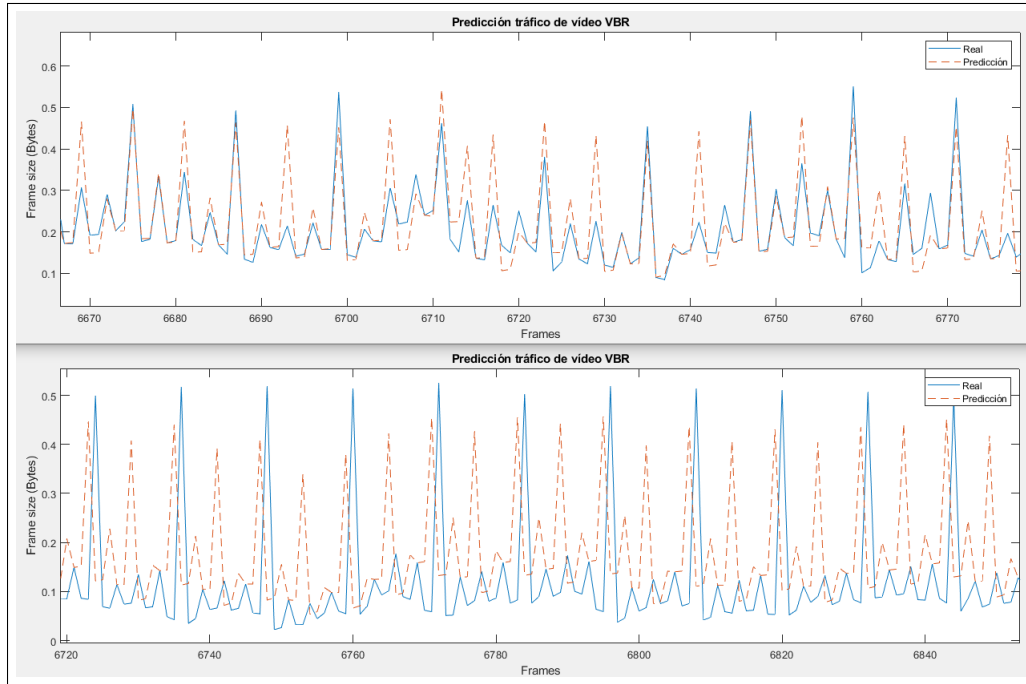


Figura 14: Acercamiento 2

En vista de la Figura 11, a pesar de ser los mejores resultados, no logra predecir adecuadamente las zonas donde el tráfico varía con más amplitud, sin embargo, sigue la forma del tráfico en general de buena manera aunque no sea exacto. Tanto en la Figura 12 como en la 13, se observa que los resultados de la trama usada en el entrenamiento (la ubicada en la parte superior) son más cercanos a la trama real que la trama usada de validación, cuyas razones son evidentes, siendo esta última trama, datos nuevos para la red entrenada.



## 5. Conclusiones

Se propuso un conjunto de 54 redes neuronales Feed Forward con diferentes configuraciones, para la predicción de tráfico de Tasa de Bits Variable (VBR), de las cuales se observó un mejor comportamiento en las configuraciones de 2 capas.

Es necesario seguir explorando más combinaciones de redes, así como otras configuraciones en sus parámetros, ya que, aún se observa que falta una mejor predicción en las zonas donde el tamaño entre tramas es más variable.

Por otra parte, el reto también está en realizar una predicción a un plazo más amplio que 5 tramas, sin embargo, la obtención de una red que logre la predicción adecuadamente, será más compleja que la actual.

En vista de los resultados de la predicción, es evidente que hay cosas que mejorar, ya que la red aún no es capaz de predecir adecuadamente las zonas donde el tráfico es más variable. Una posible razón, es que las redes fueron entrenadas con la forma del tráfico de una sola película, es decir, un único comportamiento, por lo que no generaliza muy bien sobre otras tramas. Por ello, para una próxima experimentación, se incluirán tramas de tráfico variadas en los datos de entrenamiento, es decir, de otras películas cuyo tráfico tenga distintos comportamientos, para hacer un entrenamiento más robusto.

## Referencias

- [1] P. Seeling, M. Reisslein, and B. Kulapala. Network Performance Evaluation with Frame Size and Quality Traces of Single-Layer and Two-Layer Video: A Tutorial. *IEEE Communications Surveys and Tutorials*, Vol. 6, No. 3, pages 58-78, Third Quarter 2004.
- [2] Liang, Y. (2004). Real-Time VBR Video Traffic Prediction for Dynamic Bandwidth Allocation. *IEEE Transactions on Systems, Man and Cybernetics, Part C (Applications and Reviews)*, 34(1), 32–47.
- [3] Moh, W. M., Chen, M.-J., Chu, N.-M., Liao, C.-D. (1995). Traffic prediction and dynamic bandwidth allocation over ATM: a neural network approach. *Computer Communications*, 18(8), 563–571.
- [4] Gavade, J. D., Kharat, P. K. (2011). Neural network based approach for MPEG video traffic prediction. 3rd International Conference on Advances in Recent Technologies in Communication and Computing (ARTCom 2011).
- [5] Lobejko, W. (n.d.). VBR traffic prediction in ATM system. *MILCOM 97 MILCOM 97 Proceedings*.