

Universidad de Concepción

FACULTAD DE INGENIERÍA INDUSTRIAL

# MACHINE LEARNING FOR BUSINESS INTELLIGENCE

*PROYECTO SEMESTRAL*

Autores:  
Ricardo Hernández  
Pablo Meza

09 de Agosto 2021

# 1 Introducción

El melanoma es la transformación maligna del melanocito. Este tumor tiene gran capacidad de invasión en la zona y frecuentemente produce la muerte en las personas que padecen esta malformación. Se estima que cerca del 80% de las personas que padecen de un melanoma o cáncer a la piel no se recuperan y terminan perdiendo la vida a causa de ello.

En este informe se plantean distintos modelos de clasificación basados en redes neuronales del tipo multilayer perceptrón, convolucionales y arquitecturas complejas de redes neuronales, para clasificar imágenes de naevus y melanomas como herramienta para la detección de melanomas.

## 2 Trabajos relacionados

La investigación realizada y plasmada en este informe tuvo base en otros diferentes estudios que tienen como propósito el mismo objetivo: Combatir las muertes por melanomas mediante su detección.

Dentro de los estudios revisados de otros equipos de trabajo se rescató la idea de usar estructuras CNN secuenciales y complejas para afrontar este problema. Donde los resultados obtenidos comprenden entre 60 y 95% de precisión en la clasificación. El nombre de los estudios se encuentran en el apartado "Referencias".

## 3 Metodología

### 3.1 Set de datos

El set de datos que se utilizará consta de 170 imágenes macroscópicas RGB (70 casos de melanoma y 100 naevus) extraídas del archivo de imágenes digitales del Departamento de Dermatología del Centro Médico Universitario de Groningen (UMCG) [1]. El número muy escaso de datos que se corregirá de la manera explicada en el siguiente apartado "Set de entrenamiento y validación". Ejemplo de los tipos de muestra se aprecian en la "Figure 1".

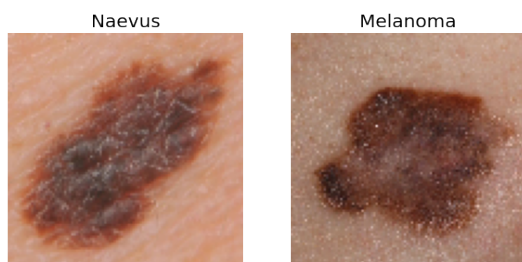


Figure 1: Sample types example

### 3.2 Set de entrenamiento y set de validación

Para construir un set de entrenamiento, inicialmente se obtuvo el 85% de las imágenes pertenecientes a los melanomas debido a que era el conjunto más pequeño, es decir, se obtuvieron 59 imágenes de melanoma, las cuales fueron concatenadas a 59 imágenes del set naevus, obteniendo así un set de datos de entrenamiento balanceado. Con los datos no considerados en el set de entrenamiento, se construye el set de validación, para validar los distintos modelos de la investigación. Sin embargo, dada la poca cantidad de imágenes se aplicó la técnica "Data Augmentation" que toma cada imagen y le aplicaba distintos tipos de transformaciones aleatorias como acercamiento, rotación, flip vertical, flip horizontal, desplazamiento horizontal y vertical para generar nuevas imágenes como se aprecia en la "Figure 2". Esto permitió incrementar cada set a 40 veces su tamaño, es decir, a 4720 imágenes en el set de entrenamiento y 2080 imágenes en el set de validación.

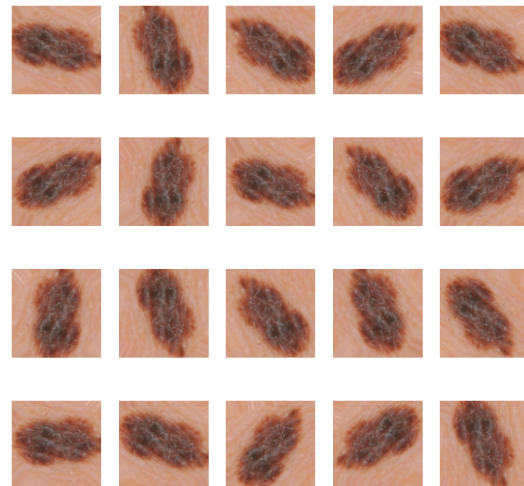


Figure 2: Data augmentation

### 3.3 Obtención de los Modelos

Se realizaron cinco modelos de redes neuronales, estos son un multilayer perceptrón, dos basados en redes neuronales convolucionales y dos en arquitecturas complejas de redes neuronales.

El primer modelo, es un modelo multilayer perceptrón (MLP), cuya arquitectura consiste en una capa de entrada de tipo Flatten que toma y aplanar la imagen de entrada para su posterior análisis en las tres capas densas que conforman la red, con dos de ellas regularizadas con L1 y L2, adicionalmente a cada capas densa se le aplica regularización mediante Dropout, lo anterior con el objetivo de evitar sobre ajuste y obtener modelos con mejor poder de predicción. Finalmente, dado que es un problema de clasificación binaria, se termina la estructura con una capa densa

compuesta solamente por una neurona y activación del tipo "sigmoid". Esto al igual que en los próximos modelos permitirá que la salida de estos sea la probabilidad de clasificar al dato de entrada como melanoma si es mayor al 50% o no. Se muestran en "Table 1" los hiperparámetros que se probarán para la elección del mejor modelo.

Table 1: Grid search MLP

HP	Values
Nº neuronas en capas	{75,80,85}
Batch size	{1180,2360,4720}
LR	{1e-3,1.2e-3,1.5e-3}
Optimizer	{Adam,SGD}
Dropout in LSTM	{0.25,0.35,0.45}

El segundo modelo fue hecho sólo con capas convolucionales, el cual consta con una capa de entrada, seguida de cinco capas convolucionales, de las cuales tres poseen una operación Maxpooling. Finalmente, se le aplica una capa Flatten para trabajar la información obtenida de las capas convolucionales, aplanarla y poder clasificar a través de la capa densa de salida compuesta de una neurona y activación "sigmoid". Se muestran en la "Table 2" los hiperparámetros que se probarán para la elección del mejor modelo.

Table 2: Grid search CNN-1

HP	Values
Batch size	{1180,2360,4720}
LR	{1e-4,1.5e-4,1e-3,1.5e-3}
Optimizer	{Adam,SGD}
Kernel size	{2,3}
Pool size	{2,3}

El tercer modelo fue una combinación de redes MLP y CNN. La entrada igual a la del segundo modelo, daba paso a cinco capas convolucionales, de las cuales tres tenían operación Maxpooling. A la estructura mencionada le sigue una capa Flatten y se agrega un número indeterminado de capas densas con regularización L1, L2 y Dropout, los cuales se establecen como hiperparámetros a probar. Al igual que todos los modelos, se termina con una capa densa de una neurona y activación Sigmoid para que sea posible la clasificación. Se muestran en la "Table 3" los hiperparámetros que se probarán para la elección del mejor modelo.

Cabe mencionar que en los primeros tres modelos se enfocaron los esfuerzos en presentar buenos modelos en base a la combinación "ideal" de hiperparámetros que en conjunto hicieran un buen trabajo de entrenamiento y predicción, utilizando hypertuning con la Clase Tuner de Keras.

Los hiperparámetros que "compiten" para ser escogidos en los respectivos modelos, resultan de

Table 3: Grid search CNN-2

HP	Values
Dense Layers	{1,2,3}
L1	{1e-4,1e-3,1e-2}
L2	{1e-4,1e-3,1e-2}
Batch size	{1180,2360,4720}
LR	{1e-7, 1e-6, 1e-5, 1e-4}
Optimizer	{Adam,SGD}
Dropout in LSTM	{0.25, 0.35,0.45}
Kernel size	{2,3}
Pool size	{2,3}

la experimentación previa con distintos hiperparámetros, evaluando en cada experimento el entrenamiento y predicción.

El cuarto y quinto modelo, CM1 y CM2 respectivamente, son redes neuronales que posee una estructura no secuencial (estructuras complejas). En estos modelos no se ajustaron los hiperparámetros mediante la técnica de hypertuning. Tanto la estructura como los hiperparámetros, se buscaron de forma manual mediante la observación de la evolución de los entrenamientos y resultados.

### 3.4 Class Tuner

Para realizar el proceso de hypertuning mencionado las subsecciones anteriores, se utilizó una estructura de Class Tuner donde es importante mencionar que para la elección de los parámetros de cada modelo, se evaluó bajo la métrica de "Val. Accuracy", escogiendo finalmente los valores que generaban el modelo con precisión más alta.

Otro punto importante a mencionar en esta estructura es que se consideró un valor al parámetro beta alto con el fin de probar modelos muy diferentes entre sí y los parámetros se repetirían lo menos posible.

Finalmente, al aplicar el Class Tuner, siempre se tuvo en cuenta la cantidad de posibles combinaciones dado la cantidad de hiperparámetros para establecer la magnitud de la cantidad de intentos para probar diferentes modelos.

### 3.5 Entrenamiento y validación

En todos los entrenamientos se han incluido dos callbacks, Stop early y Reduce lr. La función de pérdida utilizada es Binary Cross. Se entrena los modelos con el set de entrenamiento y se valida con el set de validación. Con respecto al callback Stop Early se utilizó para detener los entrenamientos cuando el valor del "Val. Accuracy" no ha mejorado en una cantidad de épocas predeterminada, y se utilizó el callback Reduce lr, que disminuye el learning rate cuando en el entrenamiento el "Val. Loss" no ha mejorado en

una cantidad de épocas determinada.

## 4 Resultados

### 4.1 Modelos obtenidos

Se presentan a continuación en la "Table 3", "Table 4" y "Table 5" los hiperparámetros seleccionados en el proceso de hypertuning para los tres primeros modelos. Las arquitecturas resultantes se pueden apreciar visualmente en la "Figure 3", "Figure 4" y "Figure 5" en la sección "Anexos".

Table 4: Results MLP

HP	Values
Nº neuronas en capas	80
Batch size	4720
LR	1.5e3
Optimizer	SGD
Dropout	0.25

Table 5: Results CNN1

HP	Values
Batch size	4720
LR	1.5e-3
Optimizer	Adam
Kernel Size	2
Maxpooling Size	3

Table 6: Results CNN2

HP	Values
Dense Layers	1
L1	1e-2
L2	1e-2
Batch size	1180
LR	1e-4
Optimizer	Adam
Dropout	0.25
Kernel Size	3
Maxpooling Size	3

Con respecto al modelo CM1 resultante, este se compone de una capa de entrada que se conecta con 2 diferentes secciones de extracción de características. Una sección contiene 5 capas convolucionales (32, 64, 128, 256, 512 filtros respectivamente, activación relu) con kernel size de 5x5. La otra sección contiene 4 capas convolucionales (32,64,128,256 filtros respectivamente, activación relu) con kernel size de 2x2. Cada capa convolucional es seguida por una capa Maxpooling, con pool size de 2. Ambas secciones de extracción de características finaliza con una capa flatten que se concatenan formando una sola capa que pasa por 3 capas densas (3000, 1000 y 1 neurona respectivamente, con activación "sigmoide"). En

donde las 2 primeras utilizan regularización L1 de 0.001 y L2 de 0.001. El modelo se ha entrenado con el optimizador Adam y un learning rate de 0.0001. La arquitectura resultante se pueden apreciar visualmente en la "Figure 6" en la sección "Anexos".

Con respecto al modelo CM2 resultante se compone de una capa de entrada que se conecta con 2 secciones idénticas de extracción de características. Por una sección entra el input original, y en la otra sección entra el input rotado en una cantidad de grados fija por una semilla. Cada sección contiene 4 capas convolucionales (32, 64, 128, 256 filtros respectivamente, activación relu) con kernel size de 2x2. Las 3 primeras capas convolucionales son seguidas por una capa Maxpooling, con pool size de 2. Ambas secciones de extracción de características finaliza con una capa flatten que se concatenan formando una sola capa que pasa por 2 capas densas ( 1000 y 1 neurona respectivamente, con activación sigmoide). En donde la primera utilizan regularización L1 de 0.001 y L2 de 0.001. El modelo se ha entrenado con el optimizador Adam y un learning rate de 0.0001. La arquitectura resultante se pueden apreciar visualmente en la "Figure 7" en la sección "Anexos".

### 4.2 Desempeño de los modelos

Se puede observar según los desempeños de los distintos modelos mostrados en la "Table 7" que el modelo CNN1 arrojó mejores resultados en todas las métricas, es decir, en "Train Accuracy", "Val. Accuracy", "Val. TP" y "Val. TP". Lo que implica que a la hora de predecir los datos de entrenamiento y datos jamás vistos en el entrenamiento, el modelo CNN1 es más confiable y preciso. Esto a su vez implica que las personas diagnosticadas de melanoma, tengan una gran probabilidad de padecer esta enfermedad, o bien, cada vez que se "diagnostique" no tener melanoma, haya una gran posibilidad de que el paciente no padezca de esta enfermedad.Sin embargo, como se puede ver en la "Figure 9" en la sección "Anexos", el entrenamiento no es regular, por lo que no se asegura que este modelo siempre esté dentro de los más competitivo a la hora de acertar.

El siguiente mejor modelo en cuanto a resultados es el modelo CM2, el que a su vez, como se aprecia en la "Figure 12" en la sección "Anexos" se puede observar que tiene un entrenamiento regular, lo que implica que probablemente generalmente dentro de los modelos más competitivos a la hora de predecir o diagnosticar.

En cuanto a aspectos interesantes del estudio, se puede observar y mencionar que el modelo MLP tiene un valor inferior pero cercano en cuanto a la métrica "Val. Accuracy" en comparación a

modelos más complejos como los dos CNN's y los CM's. También se puede observar la misma cercanía en resultados a los valores "Val. TN" y una superioridad en el valor "Val. TP" cuando se le compara con el modelo CNN2 y CM1. Resultados inesperados por parte del equipo debido a que es una estructura simple en comparación a los demás.

Table 7: Model performance metrics

Metrics	MLP	CNN1	CNN2	CM1	CM2
Train Accuracy (%)	70.55	96.63	95.55	83.81	81.65
Val. Accuracy (%)	78.61	84.86	82.21	81.01	80.82
Val. TP (%)	70.0	79.0	57.0	59.0	75.0
Val. FP (%)	19.0	14.0	11.0	13.0	17.0
Val. TN (%)	81.0	86.0	89.0	87.0	82.0
Val. FN (%)	30.0	21.0	42.0	41.0	25.0

NOTA:

TP: Diagnóstico melanoma y estado melanoma.

FP: Diagnóstico melanoma y estado no melanoma.

TN: Diagnóstico no melanoma y estado no melanoma.

FN: Diagnóstico no melanoma y estado melanoma.

## 5 Discusiones

En la experimentación se pudo apreciar que los distintos modelos tenían gran capacidad para aprender de los datos, por lo tanto se cree que la complejidad de los modelos para este reducido set de imágenes no es tan fundamental para la tarea de clasificación, como una correcta aplicación de regularizadores y técnicas como lo es el callbacks Stop Early para evitar el sobre ajuste.

Se puede visualizar en los gráficos de entrenamiento que el comportamiento de la métrica "Val. Accuracy" en la mayoría de los modelos, presenta movimientos bruscos e irregulares. Dado esto, la paciencia determinada en el callback toma un rol fundamental, ya que si bien el comportamiento es irregular, la paciencia debe ser suficiente para dejar al algoritmo mejorar, pero no demasiada para poder parar a tiempo el entrenamiento, antes de que el sobre ajuste aumente considerablemente. Se estima que las irregularidades en el comportamiento de esta métrica puede deberse a las pocas imágenes que se utilizan en la base de datos, tanto para entrenar como para validar y también debido a que se generaron imágenes que son transformaciones de las mismas, en donde la actualización de pesos debido al error de una muestra, pueda generar un gran impacto en la predicción para las transformaciones con el mismo origen.

Ahora bien para un futuro análisis se podría responder la pregunta ¿Hasta qué grado puede ser sencillo un modelo para seguir siendo competi-

tivo en comparación a los presentados? y ¿Qué parámetro o estructura podrán lograr una notable diferencia entre modelos simple MLP y complejos como los CNN y CM?

Otra idea a discutir es la probabilidad de que los datos y modelos obtenidos mejoren a medida que el data set crezca en volumen. Una pregunta a responder en el futuro es ¿Hasta qué punto es una mayor cantidad de datos no generaría un impacto positivo en la obtención de modelos con buenos resultados?

Finalmente se supone en el presente estudio que la técnica de data augmentation ha funcionado y aportado considerablemente y de forma positiva a la obtención de datos, parámetros y modelos obtenidos, sin embargo, esto sólo se podría comprobar si se hiciera el mismo procedimiento con un set de dato del mismo tamaño sin haber aplicado una técnica de la misma índole.

## 6 Conclusiones

Dado los resultados obtenidos se puede concluir que la complejidad de un modelo no determina su precisión en la detección del melanoma, debido a que un modelo MLP simple tuvo resultados sorprendentes, sin embargo, como se mencionó anteriormente no lo hace seguro, para ello es necesario un buen estudio de los elementos y conocimiento del área. En base a lo anterior, se puede agregar en base a la experiencia y estudios leídos que con conocimientos amplios de Machine Learning se puede lograr modelos muy precisos y seguros, los cuales llegan a tener una precisión de hasta el 95% como se puede ver en los estudios revisados previamente antes de la presente investigación.

También se concluye que para tener un buen modelo y que genere con alta probabilidad buenos resultados es necesario aplicarle ciertas técnicas de regularización y búsqueda de parámetros que lo mejores a la hora de predecir o categorizar un input.

Finalmente se concluye que el proceso de data augmentation fue útil para resolver el problema del precario data set que se tenía en un comienzo. Esto permitió que se pudieran entrenar los modelos y obtener resultados para tener las conclusiones presentadas en el presente informe.

En cuanto a los modelos presentados, se infiere en base a los resultados que podrían servir para combatir a través de la detección los melanomas, sin embargo, es importante mencionar que en ningún momento (por ahora) se debe reemplazar la opinión profesional para el diagnóstico de esta enfermedad.

## 7 Referencias

- 1) Kalyanakumar Jayapriya, Jeena Jacob, I. (2019). Hybrid fully convolutional networks-based skin lesion segmentation and melanoma detection using deep feature. WILEY.
- 2) Roberto Mantas, Julio Prieto, Jose Antonio López, Francisco Varela. (2018). Clasificación de melanomas por imágenes.
- 3) Julia K. Winkler, Katharina Sies, Christine Fink, Ferdinand Toberer, Alexander Enk, Teresa Deinlein, Rainer Hofmann-Wellenhof, Luc Thomas, Aimilios Lallas, Andreas Blum, Wilhelm Stolz, Mohamed S. Abass, Tobias Fuchs, Albert Rosenberger, Holger A. Haenssle. (2020). Melanoma recognition by a deep learning convolutional neural network. Performance in different melanoma subtypes and localisations. ELSEVIER.
- 4) Chanki Yu, Sejung Yang, Wonoh Kim, Jinwoong Jung, Kee-Yang Chung, Sang Wook Lee, Byungho Oh. (2018). Acral melanoma detection using a convolutional neural network for dermoscopy images. PLOS.
- 6) Eduardo Pérez, Oscar Reyes, Sebastián Ventura. (2020). Convolutional neural networks for the automatic diagnosis of melanoma: An extensive experimental study. ELSEVIER.
- 5) Hybrid fully convolutional networks-based skin lesion segmentation and melanoma detection using deep feature. (2018). WILEY.
- 6) Skin melanoma classification using ROI and data augmentation with deep convolutional neural networks. (2020). Springer.
- 7) Department of Dermatology of the University Medical Center Groningen (UMCG). (2015, 26 julio).

## 8 Linkografía

[1] Dermatology database used in MED-NODE.  
Dermatology database used in MED-NODE.  
[http://www.cs.rug.nl/imaging/databases/melanoma\\_naevi/](http://www.cs.rug.nl/imaging/databases/melanoma_naevi/)

## 9 Anexos

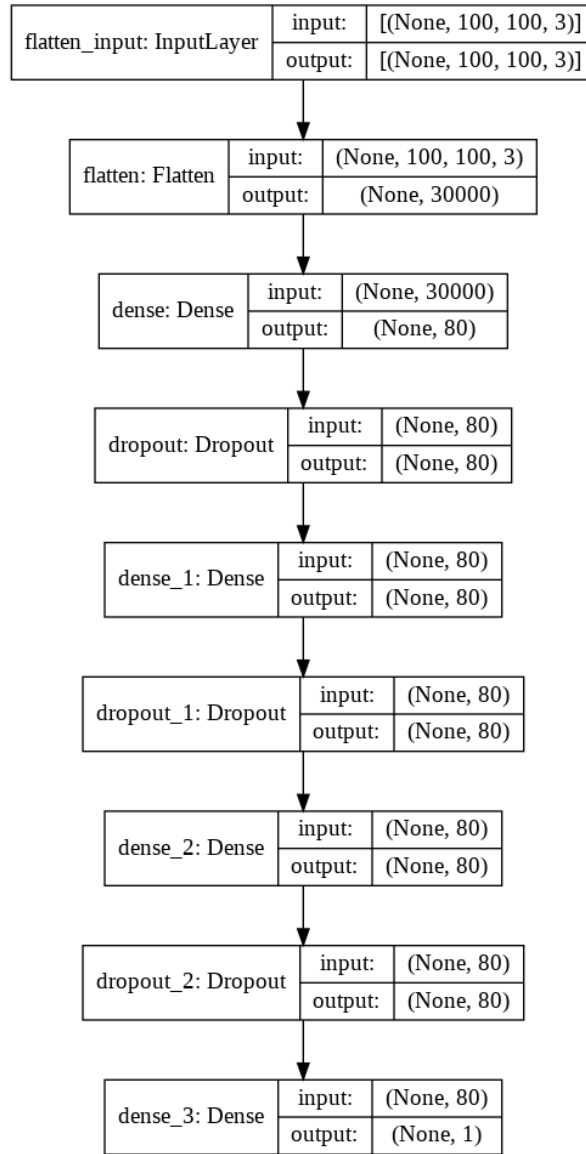


Figure 3: MLP Model

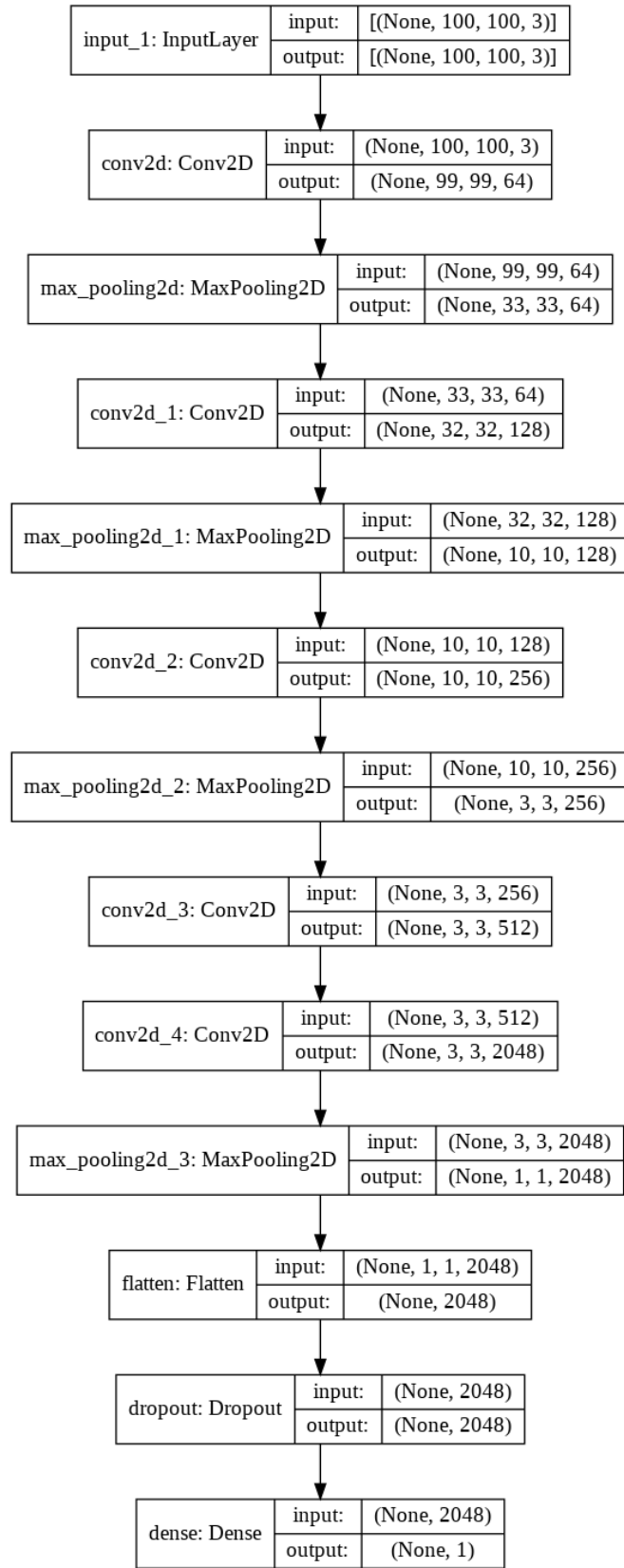


Figure 4: CNN1 Model



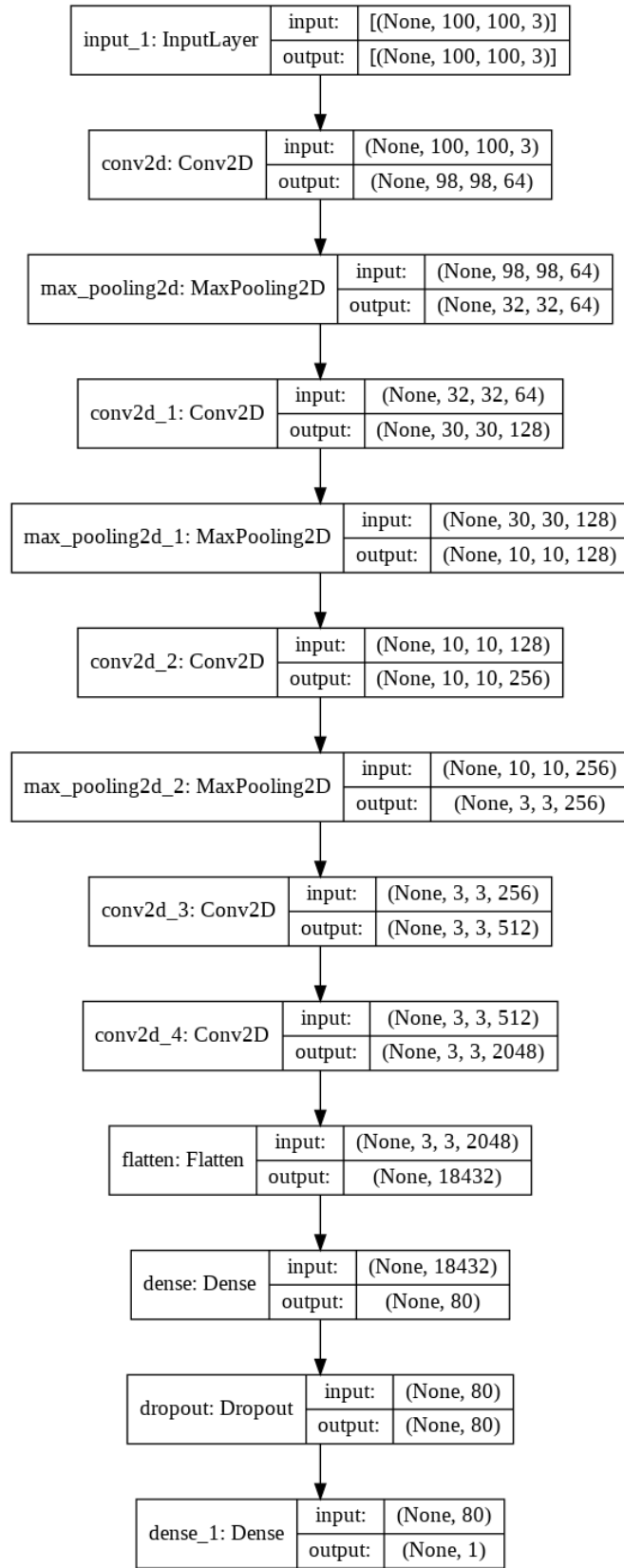


Figure 5: CNN2 Model

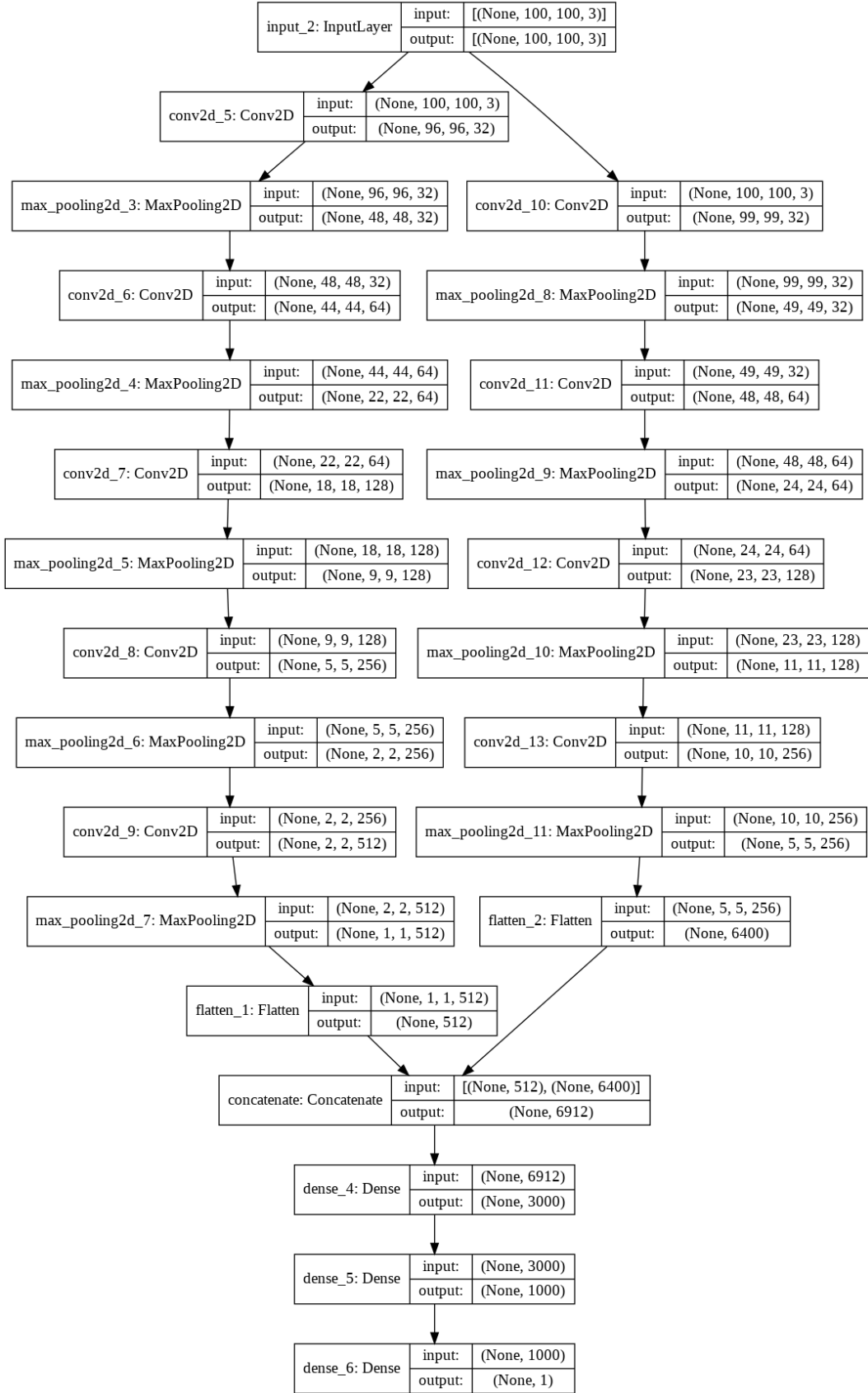


Figure 6: CM1 Model

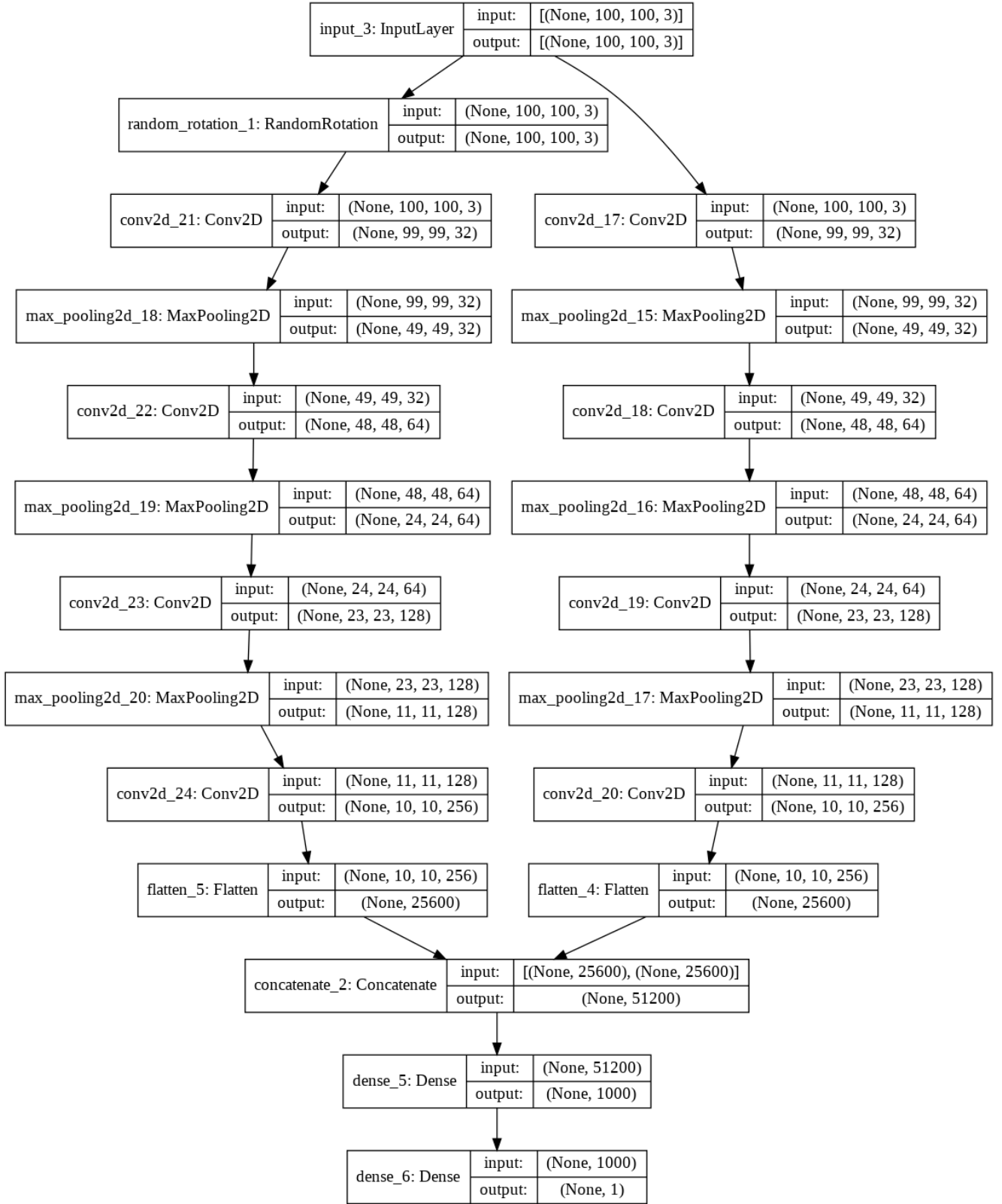


Figure 7: CM2 Model

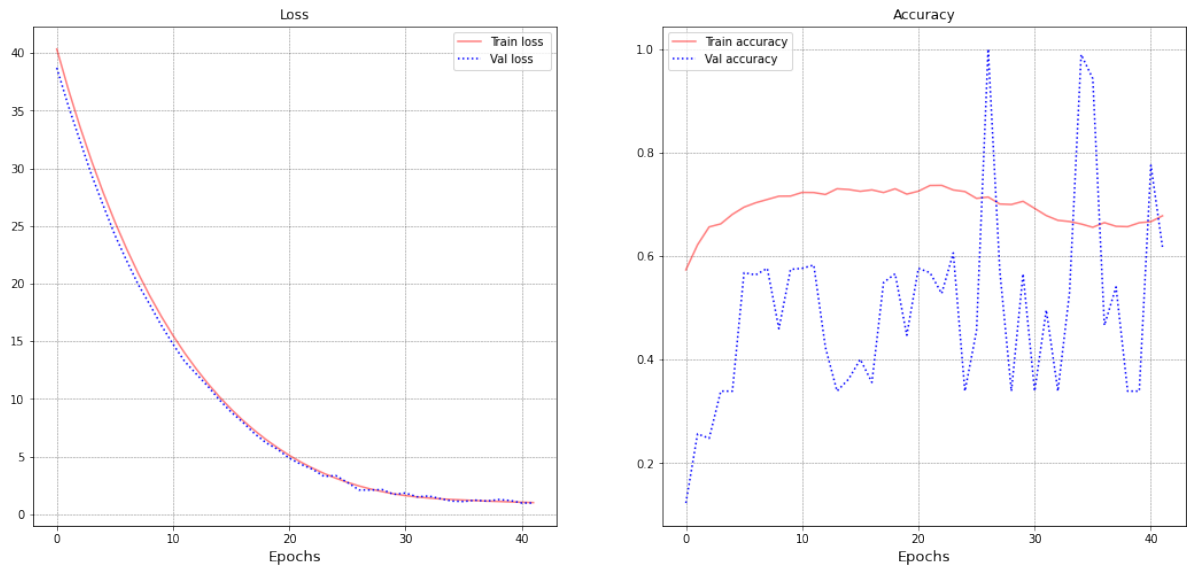


Figure 8: Loss and Accuracy MLP

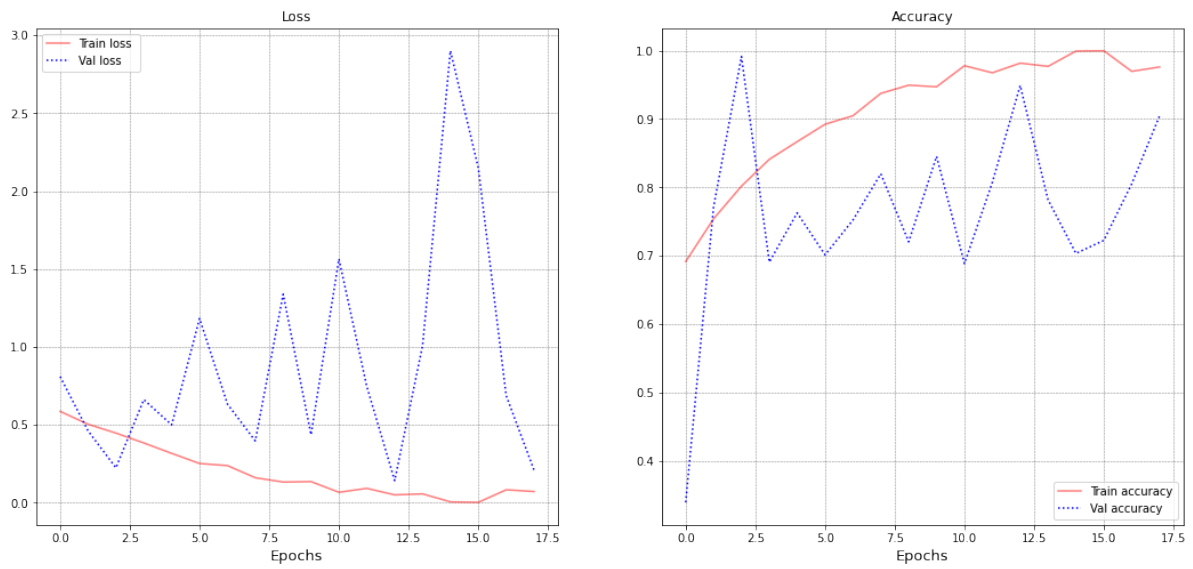


Figure 9: Loss and Accuracy CNN1

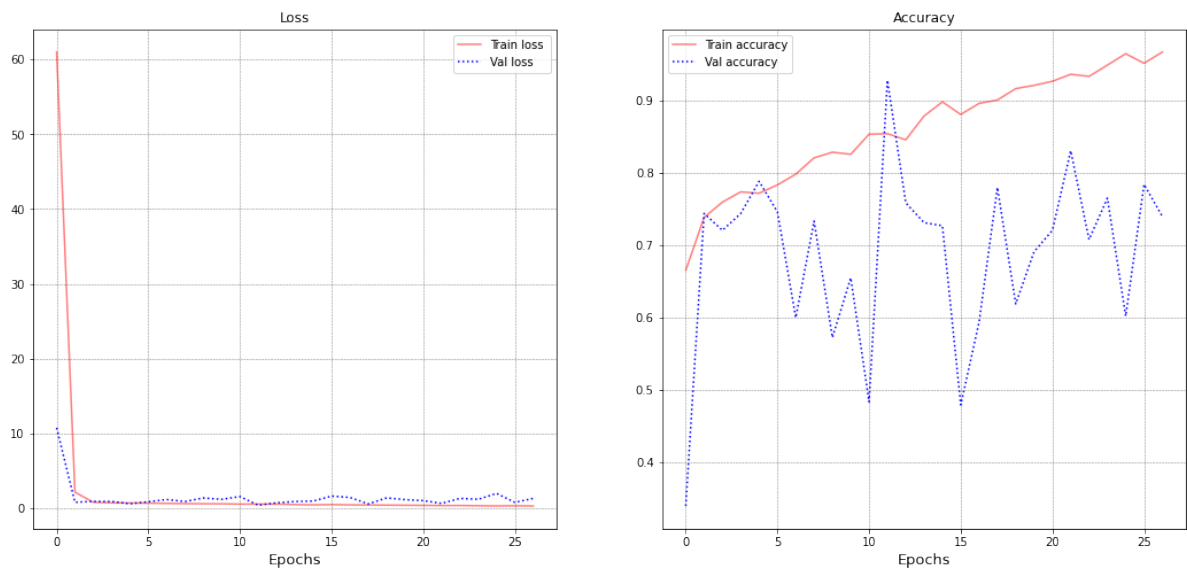


Figure 10: Loss and Accuracy CNN2

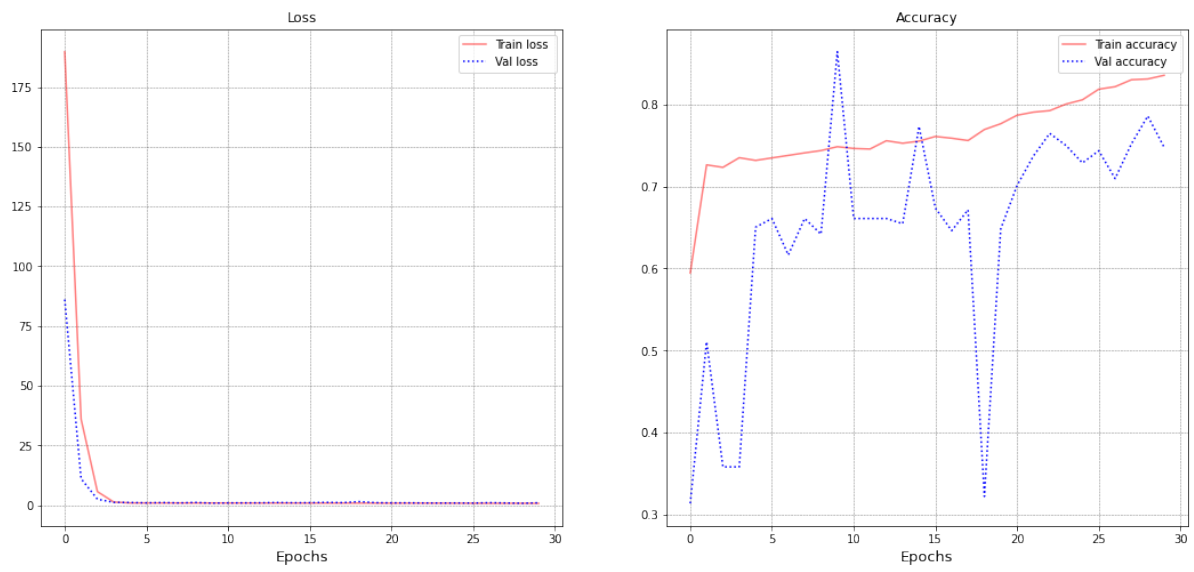


Figure 11: Loss and Accuracy CM1

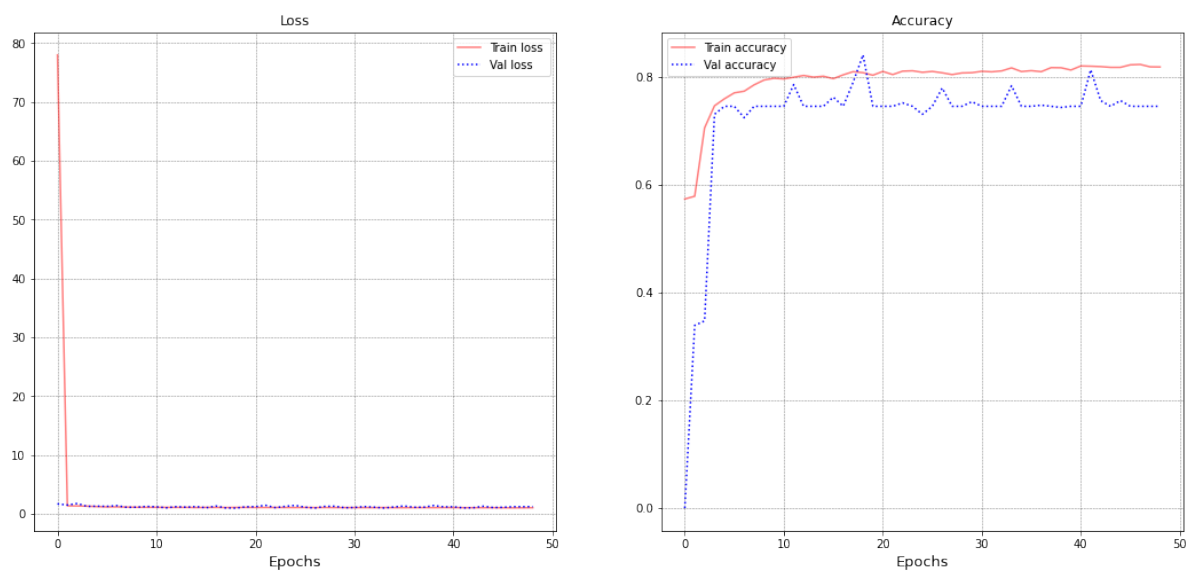


Figure 12: Loss and Accuracy CM2