Contest di Visione Artificiale: Gruppo 19

Demetrio Trimarco, Emilio Sorrentino, Francesco Rosa e Francesco Sabbarese

{d.trimarco1, e.sorrentino38, f.rosa5, f.sabbarese3}@studenti.unisa.it

Sommario

1 Introduzione

In questa documentazione viene affrontato il problema della *Age Estimation* (AE), riconosciuto come uno dei più complessi, anche per gli umani. La AE può essere approcciata utilizzando sia metodi di *regressione* che di *classificazione*. In quest'ottica, si è pensato di realizzare un confronto tra le due metodologie, prendendo in considerazione diverse architetture di rete, funzioni di costo e dataset. In particolare, dai paper analizzati, si è evidenziato come il successo di un metodo è influenzato dalla scelta della funzione di costo, nel nostro caso il *MAE* per la regressione e *Cross-Entropy* per la classificazione, e dal pre-processing del dataset. La procedura seguita è stata quella del "fine tuning" di reti già addestrate, sostituendo i livelli finali con blocchi studiati per i due diversi approcci. Tra le architetture di rete adottate per la *feature extraction* vi è senza dubbio la *VGG-Face*, ed in particolare la sua implementazione basata su "Resnet50", fornita di *weights* ottenuti in seguito al suo pre-addestramento sul task della *Face-recognition*. La metrica presa in considerazione per la valutazione delle prestazioni delle varie architetture è stata il MAE e la matrice di confusione, arricchita con heat-map, è stata usata come supporto all'analisi. Il dataset *VGGFACE2*, per via delle sue dimensioni e della ridotta potenza computazionale a nostra disposizione, è stato oggetto di analisi statistiche, volte a ridurne il numero dei campioni e ad uniformarne la distribuzione per età, sesso ed entità. Avendo riscontrato criticità in merito al numero di campioni per alcune fasce di età sono stati inclusi samples provenienti da dataset quali *UTKFace* ed *APPA-REAL*. Inoltre, tecniche di *Data Augmentation*, sono state adoperate a *runtime*, per aumentare la capacità delle reti di generalizzare.

Nel capitolo 2 ci si concentrerà unicamente sugli aspetti inerenti al modello meglio performante, mentre nel capitolo 3 verrà illustrata l'intera procedura di selezione che ci ha portati a scegliere il modello stesso.

2 Descrizione della soluzione

Di seguito vengono descritti i diversi aspetti che hanno caratterizzato il progetto.

2.1 Convolutional neural network

A valle di confronti tra diversi modelli (riportati nella sezione 3) si è deciso di usare la CNN VGG-Face [1], nello specifico la variante basata su ResNet50. Inoltre, sono stati applicati dei pesi ottenuti da un pre-addestramento che aveva come obiettivo il task di face-recognition.

La ResNet50 è un esempio di Residual Neural Network che a differenza delle Deep Neural Network classiche (come, ad esempio, la Vgg16) è caratterizzata dalla presenza di layer il cui output è input del layer immediatamente successivo e di un altro presente ad una profondità maggiore, permettendo così la realizzazione di reti più profonde ma risolvendo i problemi legati a queste ultime come per esempio il *vanishing gradient*.

La rete è stata progettata come regressore.

Al fine di realizzare un fine tuning dell'architettura, è stata implementata una funzione che, preso l'output della rete impiegata per la feature extraction ed il numero di neuroni voluti per gli hidden fully-connected layers, genera il blocco di layers che date le feature estratte, effettua la AE. Tale blocco, è costituito da:

- *BatchNormalizzation layer*: al fine di diminuire il rischio che differenti

distribuzioni in input ai diversi layer della rete possano influenzare l'addestramento, riducendo il fenomeno della *internal covariate shift;*

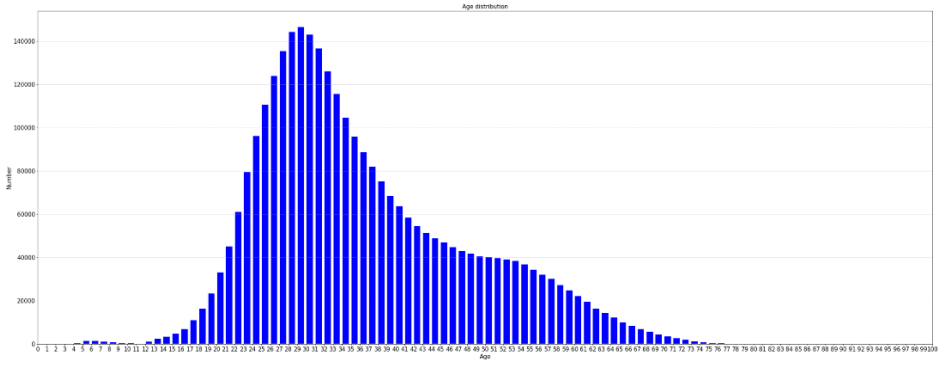
- *Dense layer*: layer fully-connected;
- *DropOut layer*: utilizzato per ottenere una maggiore generalizzazione in fase di training andando a disabilitare in maniera randomica dei neuroni dei layer fully connected, simulando quindi ad ogni batch una "rete diversa", riducendo quindi il fenomeno dell'*overfitting.*
- *PReLU*: come funzione di attivazione per gli hidden fully-connected layer, preferita alle altre così da non avere problemi di Dying *ReLU*.
- *ReLU*: come funzione di attivazione per il layer di output.

2.2 Procedura di addestramento

2.2.1 Dataset

Il dataset VggFace2 utilizzato presenta un numero di sample pari a circa 3000000, 9.131 diverse identità ed un rapporto maschio-femmina di 60/40.

È stata innanzitutto analizzata la distribuzione delle immagini presenti nel dataset suddivise per età ottenendo quanto segue:



Si può notare scarsità di sample in corrispondenza delle età che vanno da 0 a 15 anni e da 68 a 100 anni.

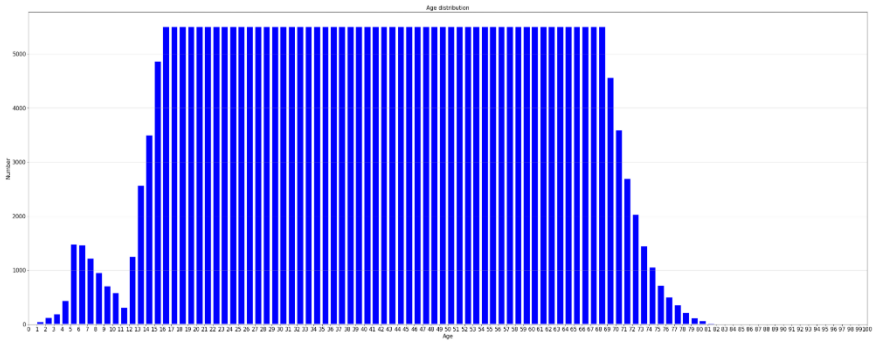
A causa delle elevate dimensioni del dataset e considerando anche i limiti dell'hardware utilizzato, si è deciso di utilizzare un subset del dataset.

Sono stati realizzati due subset, rispettivamente:

1. **SUBSET_1**: Un subset contenente 238323 immagini nel training set e 90251 immagini nel validation set. La creazione di tale subset ha avuto come obiettivo quello di effettuare una comparazione tra diverse architetture di base per comprendere quale meglio si prestasse al task preposto e per confermare delle ipotesi fatte. Il dataset è stato inoltre realizzato facendo riferimento ai seguenti criteri:
    - Ridurre le immagini solo in corrispondenza di quelle età per le quali il numero di sample supera una certa soglia (al fine di non penalizzare le età per le quali sono già presenti pochi sample).

- La scelta delle immagini da prelevare dalle fasce di età con numero di sample sopra la soglia è casuale (in quanto, in questo caso specifico, non è necessario mantenere la stessa rappresentatività del dataset iniziale).

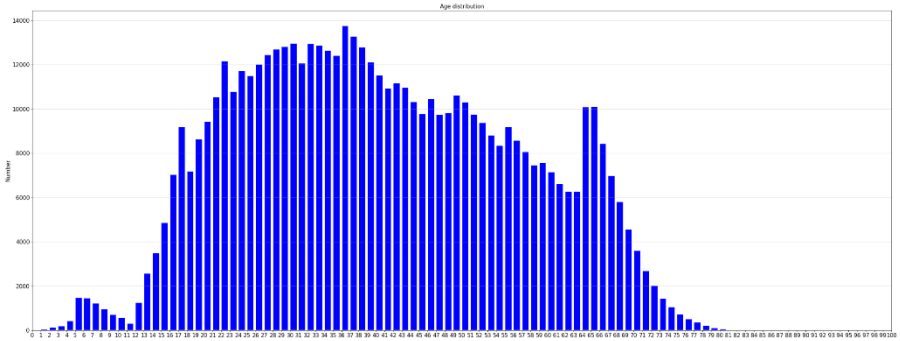
La distribuzione di tale subset risulta essere la seguente:



2. **SUBSET_2**: Un subset contenente 433525 immagini per il training set e 140173 immagini per il validation set. Tale dataset è stato creato in seguito ai test preliminari per la scelta della migliore architettura da utilizzare tra quelle inizialmente contemplate, al fine di effettuare il training vero e proprio delle migliori reti individuate. Di conseguenza è stato realizzato tentando di uniformare la distribuzione dei samples per età e sesso ed è stato quindi costruito facendo riferimento ai seguenti criteri:

   - Ridurre le immagini solo in corrispondenza di quelle età per le quali il numero di sample supera una certa soglia (al fine di non penalizzare le età per le quali sono già presenti pochi sample).
   - Mantenere invariato il numero delle identità presenti nel dataset.
   - Avere un numero medio di sample uguale per identità fissato il genere.

   Di seguito si mostra nuovamente la distribuzione dei sample:



   Si osserva che non è stato possibile uniformare effettivamente la distribuzione, questo per via della volontà di conservare, per ogni età, almeno un sample per ogni identità.

Si fa notare come, per entrambi i dataset, si è deciso di introdurre all'interno del validation set immagini di individui presenti anche nel training set, ma con **età diverse**, al fine di verificare se, durante la fase di training, la rete sia stata in grado di apprendere in modo sufficientemente generale da essere capace di classificare correttamente anche immagini della stessa persona in momenti diversi della sua vita.

A seguito dei primi addestramenti sul **SUBSET_2** si è notato come i risultati fossero fortemente inficiati dalla scarsità di sample sopra riportata per alcune fasce di età.

Si è tentato di risolvere tale problematica aggiungendo immagini a partire da altri due dataset. In particolare, si è fatto riferimento ai dataset **UTKFace** [2] e **APPA-REAL**. [3]

2.2.2 Face detection

Per effettuare il ritaglio dei volti si è deciso di fare riferimento alle bounding box rese disponibili tramite il **GenderRecognitionFramework** [4] ma anche alle funzioni di *face detection* messe a disposizione dallo stesso.

Più nello specifico, per la realizzazione dei subset riportati al paragrafo precedente, si riporta che:

- Dal training set fornitoci sono state scartate tutte le immagini per le quali non è presente una bounding box nei file csv disponibili.
- Dai dataset UTKFace [2] e Appa-Real [3] sono state prelevate tutte le immagini presenti effettuando face detection tramite le funzioni messe a disposizione dal **GenderRecognitionFramework**.

2.2.3 Face pre-processing

Nelle operazioni di pre-processing è stata utilizzata la libreria DLib. In particolare, per ogni immagine è stato applicato un riallineamento del volto e poi un ridimensionamento pari alla dimensione del layer di input della rete (ovvero 240x240). Per la procedura di allineamento sono stati utilizzati i bounding box presenti al framework **GenderRecognitionFramework** in modo tale da dare in input allo shape predictor la zona dell'immagine dov'è presente il volto. Si tenga presente che tali operazioni sono state effettuate al momento della preparazione del dataset in modo da non doverle applicare durante il training velocizzando di conseguenza questa fase.

Inoltre, ogni immagine ha subito un ulteriore processo di normalizzazione sulla base del modello utilizzato. Nel caso del modello VGGFace basata su ResNet50 pre-addestrata sul dataset VGGFace2, le immagini sono state normalizzate andando a sottrarre loro la media calcolata su tale dataset. Infine, ogni immagine è stata convertita in formato BGR in quanto la rete utilizzata è stata pre-addestrata su immagini che avevano questa disposizione dei canali.

2.2.4 Data augmentation

Al fine di aumentare la generalizzazione del modello e di sopperire al limitato numero di samples per determinate fasce di età, sono state prese in considerazione le seguenti tecniche di Augmentation dei dati:

- *random crop*: l'immagine viene ritagliata in maniera randomica, ovvero il punto su cui fare il *crop* è scelto casualmente. Dopo il taglio dell'immagine, viene effettuata una *resize* alla dimensione originaria. Tale trasformazione è utile per simulare la mancanza di informazioni, che può essere generata da un errore durante la fase di detection e allineamento;
- *random horizontal flip*: l'immagine viene specchiata lungo l'asse verticale così da simulare diversi orientamenti dei volti;

- *random brightness and contrast*: l'immagine subisce una modifica casuale all'interno di un intervallo prefissato dei valori di luminosità e contrasto, al fine di simulare diverse condizioni di illuminazione, che hanno un notevole impatto sulla stima dell'età;
- *random patch*: viene creata una patch di dimensione casuale e applicata in un punto casuale dell'immagine, portando a zero il valore associato ai pixel convoluti con la patch, per i tre canali dell'immagine, in modo da andare a simulare delle informazioni precluse dal volto;
- *gaussian blur*: applica un filtro gaussiano all'immagine al fine di simulare una mancanza di informazioni dovuti ad una sfocatura;
- *gaussian noise*: introduce nell'immagine un rumore gaussiano per modellare il rumore elettronico di fondo del sensore;
- *impulse noise*: introduce nell'immagine un rumore di tipo sale e pepe per simulare il rumore dovuto alle condizioni ambientali non perfette che si hanno durante l'acquisizione dell'immagine.

La data Augmentation viene applicata con una probabilità del 30%, ed ognuna delle trasformazioni ha una probabilità del 50% di essere applicata: in questo modo, ad ogni batch, la rete osserverà immagini trasformate sempre in maniera differente, e ad ogni epoca, le stesse immagini avranno una certa probabilità di ripresentarsi in maniera trasformata.

È stata selezionata la policy di Data Augmentation descritta sopra in quanto le trasformazioni applicate sono coerenti con le diverse condizioni di lavoro di un ipotetico sistema di cattura delle immagini in un ambiente non controllato, inoltre l'applicazione randomica di tali operazioni ci permette di ottenere un elevata variabilità non solo nell'applicazione di una singola trasformata ma anche di combinazioni di queste ultime.

Si noti che trasformazioni come rotazioni o traslazioni lungo gli assi non sono state applicati in quanto tali operazioni sarebbero state annullate dalle operazioni di face detection e allineamento.

2.2.5 Training from scratch o fine tuning

La rete selezionata (come anticipato nel paragrafo 2.1) è una **rete pre-addestrata**.

La scelta è stata guidata dagli esperimenti riportati nel capitolo seguente. In particolare, si riporta che i migliori risultati sono stati ottenuti, utilizzando i pesi pre-addestrati per task ***face recognition*** ed effettuando **fine-tuning su tutti i layer della rete**, piuttosto che solo sui layer aggiunti.

2.2.6 Procedura di training

Per la fase di training sono state definite le seguenti callback:

- ***early stopping***: sulla base dell'andamento della metrica sul *validation set,* causa l'early stop nel caso in cui questa non diminuisca di un valore inferiore ad un delta pari a 0.01 (min-delta) per 5 epoche (patience).
- ***reducing learning rate on plateau***: permette la riduzione automatica del learning rate sulla base dell'andamento della funzione di costo calcolata

durante la fase di training. In particolare, il learning rate iniziale è stato settato a 0.05 riducendolo di un fattore di 0.1 se per 3 epoche consecutive (patience) la *loss function* non presenta una riduzione pari ad almeno 0.1 (min-delta).

Per quanto riguarda la **funzione di loss** utilizzata, questa varia in base a come il problema è stato affrontato:

- Nel caso della **regressione** la funzione di costo che si è andata a minimizzare è stata la *Mean Absolute Error* (MAE), utilizzata nei problemi di *Real Age Estimation* trattati come problemi di Regressione, dove si punta alla minimizzazione dell'ampiezza dell'errore.
- Nel caso della **classificazione** è stata utilizzata come funzione di loss la *Sparse Categorical Cross Entropy*, variante della *Categorical Cross Entropy ,* particolarmente utilizzata nei problemi di classificazione, dove l'obiettivo è la minimizzazione dell'incertezza legata alla scelta della classe.

L'ottimizzatore utilizzato è stato lo **Stochastic Gradient Descent (SGD)**, ottimizzatore di tipo non adattivo, con *momentum* fissato, per tutta la durata del training, a 0.9 e *learning rate* variabile sulla base di quanto detto sopra.

Per la valutazione del modello è stata utilizzata come metrica la ***MAE***.

3 Risultati sperimentali

Di seguito si riporta la lista degli esperimenti effettuati, suddivisi sulla base dell'obiettivo per essi prefissato.

Si tenga presente che i primi esperimenti sono volti a comprendere quale fosse la rete più performante tra tutte per il tipo di problematica affrontata, pertanto il confronto è avvenuto solamente per il task di classificazione e in assenza di data augmentation. Decretata la rete migliore, gli esperimenti (dal 3 in poi) sono serviti per determinare il miglior utilizzo di questa rete.

- **ESPERIMENTO_1**: Comparazione tra VGGFace basata su Vgg16, Resnet50 e Senet50, al fine di selezionare la miglior versione della medesima rete.

ARCHITETTURA	TRAINING TYPE	DATASET	EPOCA	TRAINING	VALIDATION
VGGFace vgg16	Classificazione	SUBSET_1	10	3.70	4.07
VGGFace resnet50	Classificazione	SUBSET_1	10	3.09	3.90
VGGFace senet50	Classificazione	SUBSET_1	8	3.69	3.97

  - **RISULTATO:** Alla luce del MAE calcolato sul Validation set, si conclude che la versione di VGGFace meglio performante è quella basata su Resnet50. Un ulteriore vantaggio riscontrato è quello relativo alle tempistiche di addestramento, che nel caso di Resnet50 e Senet50 sono di circa 45 minuti, contro i circa 105 minuti della versione basata su Vgg16.

- **ESPERIMENTO_2**: In questo esperimento vengono analizzati diversi modelli, pre-addestrati sul task della Gender Recognition, effettuando fine-tuning, allenando solo i fully-connected layer aggiunti. Lo scopo è quello di andare ad effettuare un confronto con il MAE ottenuto nell'esperimento precedente.

ARCHITETTURA	TRAINING TYPE	DATASET	EPOCA	TRAINING	VALIDATION
* DENSENET	Classificazione	SUBSET_1	17	7.58	6.44
* MOBILENET	Classificazione	SUBSET_1	7	16.4	14.03
** INCEPTION-V3	Classificazione	SUBSET_1	14	8.38	7.94
* modelli pre-addestrati, prelevati dal **GenderRecognitionFramework**					
** pre-training per task "gender recognition", effettuato su sample prelevati dal dataset UTKFace					

  - **RISULTATO:** Si conclude che nessuno dei modelli esaminati riesce a restituire migliori performance rispetto al modello VGGFace basato su ResNet50. Questo ci porta a concludere che le reti pre-addestrate sul task di Gender Recognition non riescono ad ottenere prestazioni migliori rispetto alla VGGFace basata su Resnet50 pre-allenata su task face recognition.

- **ESPERIMENTO_3:** Sulla base dei due esperimenti precedenti si è concluso che l'architettura migliore da usare sia VGGFace basata su Resnet50. **Si è proceduto ad effettuare i successivi esperimenti solo su di essa.** Di seguito si riporta la valutazione dei risultati ottenuti con e senza data augmentation.

ARCHITETTURA	TRAINING TYPE	DATASET	EPOCA	TRAINING	VALIDATION
*VGGFace_Resnet50	Classificazione	SUBSET_1	10	3.09	3.90
**VGGFace_Resnet50	Classificazione	SUBSET_1	10	3.86	3.74
*senza data augmentation					
**con data augmentation					

  - **RISULTATO:** L'uso di Data Augmentation, in seguito allo stesso numero di epoche di addestramento, ha condotto ad un modello capace di generalizzare meglio, e di produrre delle migliori performance sul validation set.

- **ESPERIMENTO_4:** Gli esperimenti precedenti sono stati realizzati impostando i modelli come classificatori. Ci si è quindi chiesti se la rete potesse performare meglio come regressore. Si riportano di seguito i risultati:

ARCHITETTURA	TRAINING TYPE	DATASET	EPOCA	TRAINING	VALIDATION
VGGFace_Resnet50	Classificazione	SUBSET_1	10	3.86	3.74
VGGFace_Resnet50	Regressione	SUBSET_1	9	4.18	4.05
VGGFace_Resnet50	Classificazione	SUBSET_2	12	3.70	3.71
VGGFace_Resnet50	Regressione	SUBSET_2	12	3.39	3.79

  È interessante anche osservare le matrici di confusione legate ai primi due addestramenti riportati nella tabella.

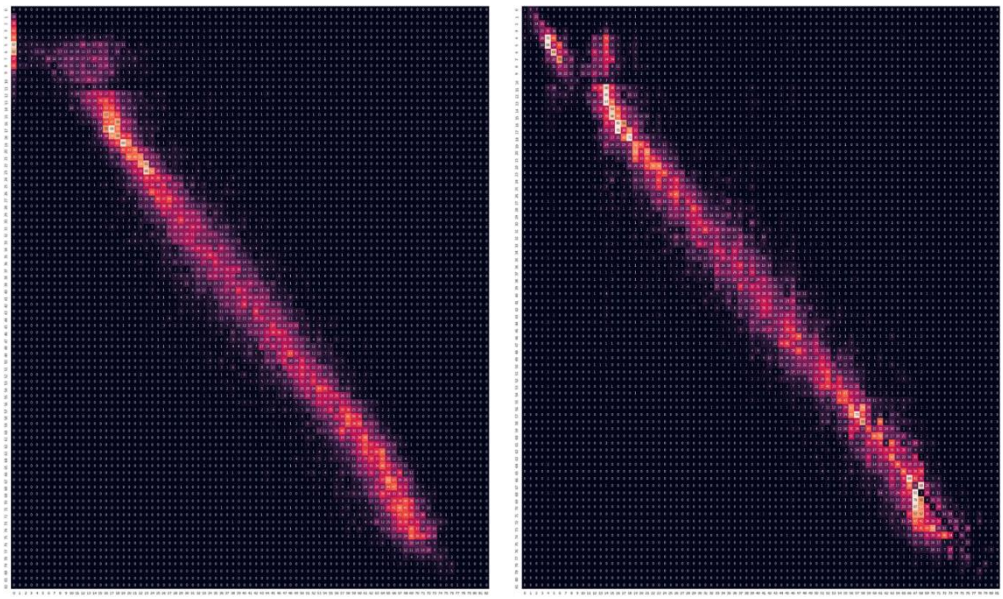


Figura 1: (Sinistra) Regressione su SUBSET_2- (Destra) Classificazione su SUBSET_2. [Asse X: età predetta, Asse Y: età corretta]

- **RISULTATO:** La metrica riporta risultati leggermente migliori per la **classificazione**, ma osservando anche le matrici di confusione, non si è ritenuto di avere dati a sufficienza per scartare la regressione. Di conseguenza, sono stati effettuati esperimenti aggiuntivi su di essa.

- **ESPERIMENTO_5:** Dalle matrici di confusione dell'esperimento precedente si nota che, sia per la classificazione che per la regressione, la scarsità di sample, per le fasce che vanno da 0 a 13 anni e da 72 a 100 anni, comporta un degradamento delle performance per tali intervalli. Inoltre, per quanto riguarda la regressione, osservando la figura sinistra di Figura1, si riscontra che numerosi individui con età tra 1 e 12 anni sono valutati erroneamente con un'età di 0 anni. Tale problema è probabilmente riconducibile al fenomeno della *dying ReLU*, il quale comporta una disattivazione dei neuroni nei layer intermedi associati a pesi negativi. Al fine di limitare tali problematiche, si è deciso di effettuare test ulteriori incrementando il numero di immagini per quelle fasce di età critiche e modificando la funzione di attivazione degli hidden fully-connected layer, effettuando test con *Leaky ReLU prima,* e *PReLU* poi.

ARCHITETTURA	TRAINING TYPE	DATASET	EPOCA	TRAINING	VALIDATION
* VGGFace_Resnet50	Regressione	SUBSET_1 + UTKFace + APPA_REAL	13	4.43	3.50
** VGGFace_Resnet50	Regressione	SUBSET_1 + UTKFace + APPA_REAL	14	5.18	3.86
*** VGGFace_Resnet50	Regressione	SUBSET_1 + UTKFace + APPA_REAL	14	4.42	3.39
* con funzione di attivazione leaky relu con slope 0.3					
** con funzione di attivazione leaky relu con slope 0.6					
*** con funzione di attivazione prelu con slope iniziale 0.6 e regolarizzazione L1L2					

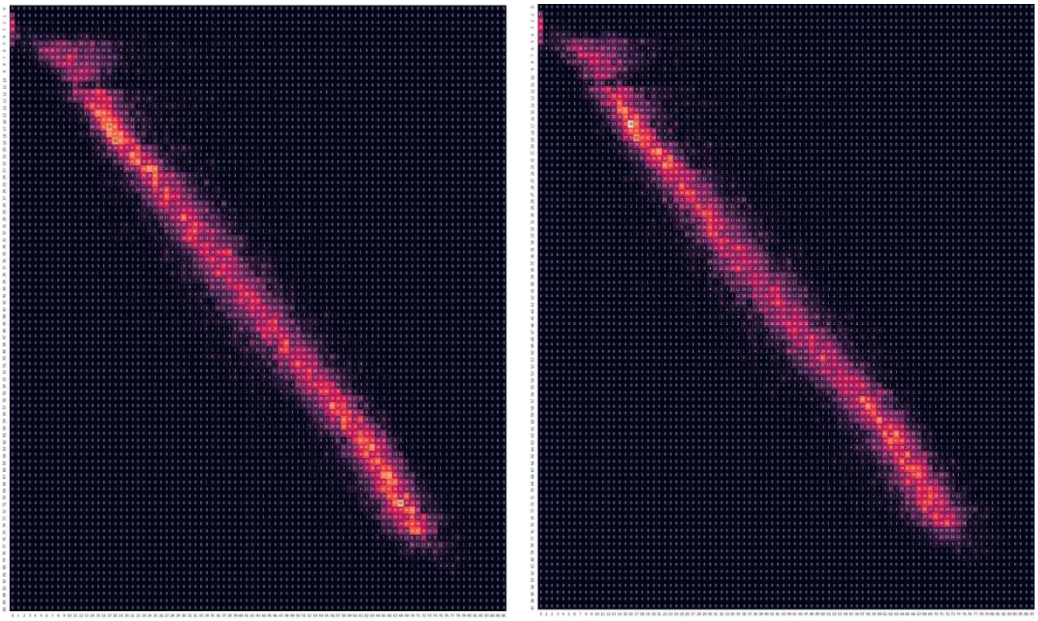


Figura 2: (Sinistra) Matrice di confusione ottenuta applicando leaky relu con slope = 0.3 - (Destra) Matrice di confusione ottenuta applicando prelu

- **RISULTATO:** L'aggiunta di immagini provenienti da altri dataset, così come l'utilizzo delle funzioni di attivazione Leaky *ReLU* e P*ReLU*, ha permesso di ottenere risultati migliori rispetto ai precedenti. Risulta interessante anche osservare la differenza tra le matrici di confusione in figura 2 e quella di sinistra in figura 1, da cui si nota un effettivo miglioramento nella fascia di età da 1 a 12 anni.

- **ESPERIMENTO_FINALE:** Confronto finale tra regressione e classificazione su SUBSET_2, con aggiunta dei dati provenienti dai dataset UTKFace e APPA-REAL. Si noti che:

  - Sono stati effettuati anche degli esperimenti in cui è stata variata l'input shape della rete ResNet50 in modo tale da avvicinarsi, per quanto possibile, alla distribuzione media, in termini di pixel, dei volti nel dataset VGGFACE2, scegliendo come dimensione 197x197 poiché vincolati dalla profondità della rete stessa.

  - Per tentare di avvicinarsi ancor di più alle dimensioni medie dei volti in termini di pixel (tra 50 px e 150 px), si è tentato di addestrare anche una rete MobileNet96, pre-addestrata su task gender recognition su dataset VGGFace2. Su quest'ultima, per via delle conclusioni a cui si è giunti dopo l'**ESPERIMENTO_2**, è stato effettuato *fine-tuning* riaddestrando i pesi di tutti i layer.

  - È stato effettuato anche un tentativo di addestramento della rete VGGFace basata su ResNet 50 effettuando fine-tuning su tutti i layer, anziché solo sugli ultimi come accaduto per tutti gli altri esperimenti sulla rete suddetta.

ARCHITETTURA	TRAINING TYPE	DATASET	EPOCA	TRAINING	VALIDATION
* VGGFace_Resnet50 (240x240)	Classificazione	SUBSET_2 + UTKFace + APPA_REAL	20	3.28	3.86
VGGFace_Resnet50 (197x197)	Classificazione	SUBSET_2 + UTKFace + APPA_REAL	15	3.97	3.86
MobileNet96 (96x96)	Classificazione	SUBSET_2 + UTKFace + APPA_REAL	31	2.96	4.85
* VGGFace_Resnet50 (240x240)	Regressione	SUBSET_2 + UTKFace + APPA_REAL	28	3.91	3.57
* VGGFace_Resnet50 (197x197)	Regressione	SUBSET_2 + UTKFace + APPA_REAL	17	3.85	3.63
** VGGFace_Resnet50 (240x240)	Regressione	SUBSET_2 + UTKFace + APPA_REAL	19	3.07	3.14
* con funzione di attivazione prelu con slope iniziale 0.6 e regolarizzazione L1 L2					
** fine-tuning su tutti i layer della rete e con f.a. prelu con slope iniziale 0.6 e regolarizzazione L1 L2					

- **RISULTATO:** Si conclude che la rete mobilenet96 non è riuscita a restituire risultati migliori rispetto alla rete VGGFace. Tra i training effettuati su quest'ultima, i migliori risultati sono stati ottenuti nel caso della regressione con applicazione di funzione di attivazione P*ReLU*, input shape di 240x240 e fine-tuning su tutti i layer della rete anziché solo su quelli finali, come successo invece per tutti gli altri esperimenti sulla stessa rete.

  È interessante comunque notare, tramite la matrice di confusione, la differenza in termini di predizione tra regressione e classificazione. In particolare, si considerano le due varianti VGGFace_Resnet50 con input shape 240x240 con fine-tuning solo sui layer finali.

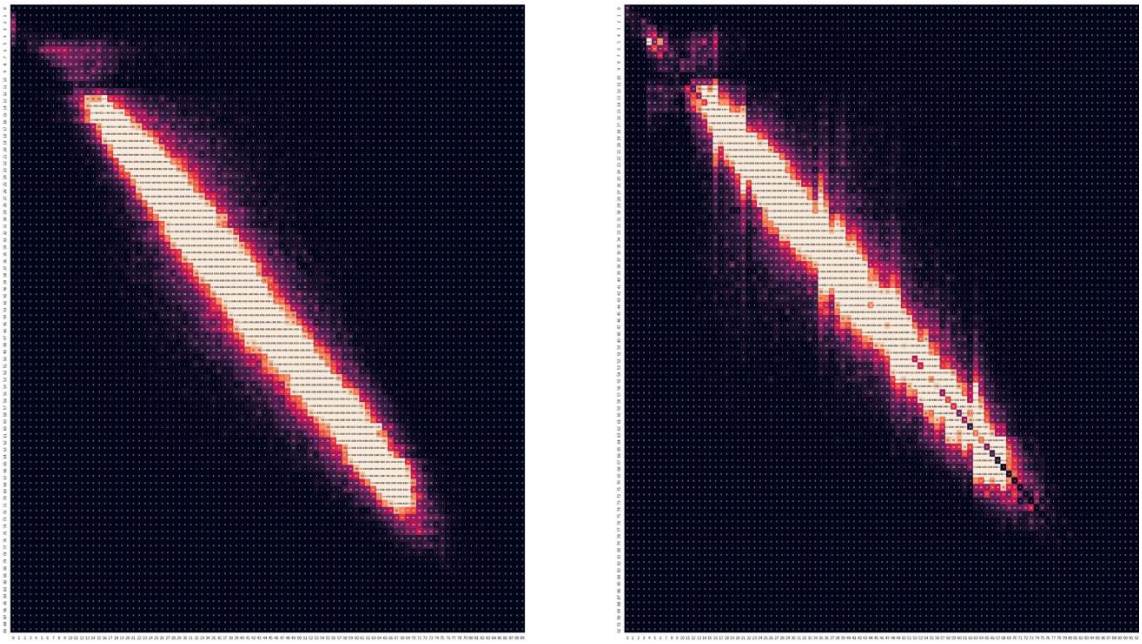


Figura 3: (Sinistra) Regressione VGGFACE_ResNet50 (240x240) - (Destra) Classificazione VGGFACE_ResNet50 (240x240)

Si osserva come, nonostante complessivamente la regressione riporti un MAE inferiore, la classificazione riesca a predire meglio le classi legate agli individui nelle fasce critiche (quelle aventi meno sample), ma produce un errore maggiore nel complesso, in particolare si osservi come, per determinate classi predette, vengano associati intervalli di età ampi della ground truth.

Si ricordi che nel validation set sono state inserite immagini di individui presenti anche nel training set, **ma con età diverse**. Si ritiene che ciò metta in luce il fatto che la rete possa essersi specializzata nell'associare l'età ad una identità nello specifico, e, probabilmente, su questo ha avuto impatto il pre-training su face recognition. Dal grafico di sinistra della *figura 3*, si nota come la regressione riesca a mitigare tale problematica fornendo un risultato complessivamente migliore, anche se limitato nelle fasce critiche.

Infine, si riporta di seguito la matrice di confusione del miglior modello ottenuto, ovvero VGGFace basata su ResNet50 con applicazione di funzione di attivazione P*ReLU*, input shape di 240x240 e fine-tuning su tutti i layer della rete.

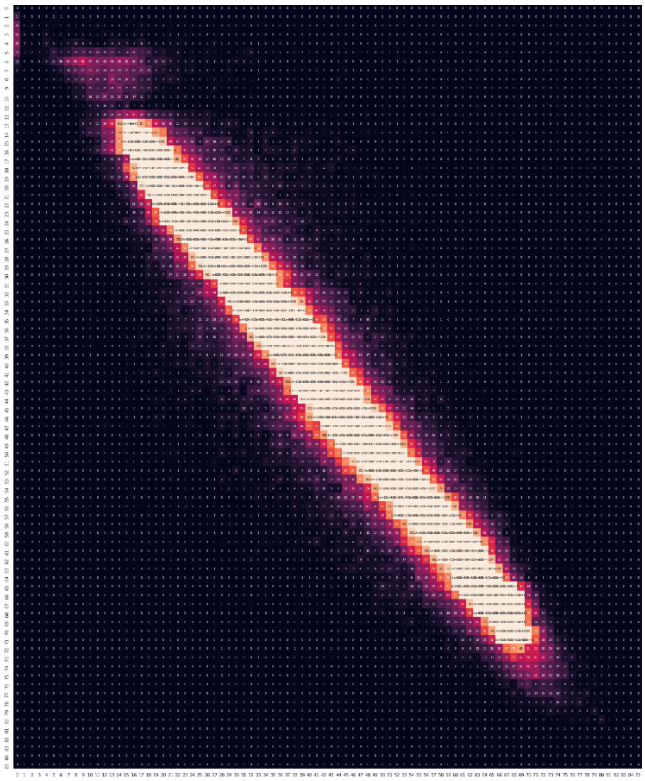


Figura 4: Matrice di confusione di VGGFace basata su ResNet50 con applicazione di funzione di attivazione PReLU e input shape di 240x240 e fine-tuning su tutti i layer della rete

Si nota un leggero miglioramento nella capacità di predizione rispetto alla regressione nella figura precedente. Ciò dimostra che effettuando fine-tuning su tutti i layer della rete si possono raggiungere performance ancora superiori ed in un numero di epoche inferiore rispetto a quanto osservato effettuando il freeze dei layer.

4 Conclusioni

Si conclude, sulla base degli esperimenti da noi effettuati, che:

- La miglior architettura è la VGGFace basata su ResNet50, si riportano di seguito i risultati:

ARCHITETTURA	TRAINING TYPE	DATASET	EPOCA	TRAINING	VALIDATION
* VGGFace_Resnet50 (240x240)	Regressione	SUBSET_2 + UTKFace + APPA_REAL	19	3.07	3.14
* fine-tuning su tutti i layer della rete e con f.d.a. prelu con slope iniziale 0.6 e regolarizzazione L1 L2					

- Eseguire fine-tuning sugli ultimi layer di una rete pre-addestrata su task *face recognition* permette di ottenere performance migliori rispetto alla stessa operazione effettuata su una rete pre-addestrata su task *gender recognition*, a parità di dataset.
- Fissato il modello VGGFace basato su Resnet50, si osserva come si ottengono delle performance migliori eseguendo fine-tuning su tutti i layer rispetto ad utilizzare la rete come *feature extractor*, anche se ciò comporta un aumento dei tempi di training per singola epoca.
- Si osserva che, sulla base di come il dataset è stato suddiviso, la regressione sembra essere più robusta al ripresentarsi della stessa persona con età diversa rispetto alla classificazione, in quanto non si nota sulla matrice di confusione un rilevante allontanamento dalla diagonale, rispetto a ciò che si ha con la classificazione. Invece quest'ultima ottiene performance migliori sulle fasce di età carenti in termini di sample (0-15 anni).
- L'utilizzo della *Data Augmentation* permette di avere un modello in grado di generalizzare meglio, osservando un miglioramento nelle performance.
- Particolarmente importante è stato considerare funzioni di attivazione alternative alla tipica ReLu, che ha permesso di migliorare le performance sulla regressione.

Riferimenti

[1] R. C. Malli, «keras-vggface,» [Online]. Available: https://github.com/rcmalli/keras-vggface.

[2] Z. S. Y. a. Q. H. Zhang, «Age Progression/Regression by Conditional Adversarial Autoencoder,» 2017. [Online]. Available: https://susanqq.github.io/UTKFace/.

[3] R. T. S. E. X. B. I. G. R. R. E Agustsson, «Apparent and real age estimation in still images with deep residual regressors on APPA-REAL database,» 2017. [Online]. Available: http://chalearnlap.cvc.uab.es/dataset/26/description/.

[4] M. Lab, «Gender recognition in the wild: a robustness evaluation over corrupted images,» [Online]. Available: https://github.com/MiviaLab/GenderRecognitionFramework.