Revitalizing regression tasks through modern training procedures: applications in medical image analysis for Covid-19 infection percentage estimation

Radu Miron[1,2] and Mihaela Elena Breaban[1]

[1] Alexandru Ioan Cuza University, Faculty of Computer Science, Iasi, Romania
[2] SenticLab, Iasi, Romania
radu.miron@senticlab.com
pmihaela@info.uaic.ro

Abstract. In order to establish the correct protocol for COVID-19 treatment, estimating the percentage of COVID-19 specific infection within the lung tissue can be an important tool. This article describes the approach we used in order to estimate the COVID-19 infection percentage on lung CT scan slices within the Covid-19-Infection-Percentage-Estimation-Challenge. Our method frames the regression problem as a multi-tasking process and is based on modern training pipelines and architectures that correspond to state of the art models on image classification tasks. It obtained the best score on the validation dataset and ranked third in the testing phase within the competition.

Keywords: deep regression · multi-tasking · Covid-19 · medical image analysis

1 Introduction

The COVID-19 pandemic has become a healthcare crisis around the world since its start in 2019. Quick discovery of the infected patients is key to positive outcome. Methods like RT-PCR, X-Ray or CT-scans are the to go choice for diagnosis of COVID-19 infection. The last two methods mentioned not only can correctly diagnose a patient with the infection, but they can also give insights into the stage the disease is progressing. The downside of these two methods is the burden an expert radiologist might be put through in order to evaluate a great amount of X-rays or CTs. CT scans have a clear advantage in comparison with X-rays due to their more detailed structure. Signs of early or late stage of infection can be easily detected in CT scans, thus making the decision to follow a certain protocol an easier task for the doctors. Having this into consideration, several AI solutions have been proposed in order to come to the aid of radiologists. The Covid-19 Infection Percentage Estimation competition [1], [2] establishes a new benchmark that may offer real help into depicting the evolution stage of COVID-19 infection. The organizers have publicly released a dataset which

consists of several CT scan slices and their corresponding Covid-19 infection percentage. In the following, we present our solution to the challenge which, on the validation set beats the second place by a large margin and improves with more than 1 the MAE score of the baseline solution provided by the organizers of the competition.

2 Related work

Regression analysis taking as input image data is much less reported in the literature compared to classification, object detection or segmentation tasks, especially when it comes to the medical domain; nevertheless, it can greatly benefit from pre-trained deep models developed to solve the most popular tasks in computer vision. Such methods, that use deep learning (mostly convolutional networks) to build a model able to estimate a numerical response variable given an input image, are generally called deep regression methods.

The latest advancements recorded in deep regression seem to be mostly related to a few datasets that were published as part of some challenges or for benchmarking purposes.

In this regard, Age Estimation or Attractiveness Estimation given as input facial images, attracted a great deal of interest. The authors of [3] improve several results on datasets like ChaLearn(2015/ 2016) [10], MORPH [11], FGNET [15] or UTKFace [12] for age estimation and SCUT-FBP [13] or CFD [14] for face attractiveness. They jointly learn to maximize the similarity between the target distribution and the generated distribution at training stage and to regress a real number in an end-to-end fashion. The output value for an input x is quantized into a range of possible values instead of just one label. The authors also mention that they pretrain their model on a large corpus of facial images before training on the downstream tasks. The method proposed by these authors is an extension of [4]. For attractiveness estimation no other model was found to report performances on benchmarks. In [5], the authors propose again a multi-tasking approach, but this time they use extra-training data and infer a posterior distribution for the ages of images given the results of multiple observed events of an annotation process. They use ordinal hyperplane [16] methods which are furthered mapped into posterior distribution using a linear layer with softmax activation. In [6], the authors extend the regression task into binary tasks used for rank prediction, where each task indicates whether the predicted output lies in a certain range or not. For robust results, the authors use for the binary tasks the same weight parameters, but different bias ones. They use weighted cross-entropy to optimize the learning process. In [7] the authors give a two point representation to the age, and consider it as an approximation of adjacent ends of certain bins which split equally the entire domain of ages. Instead of learning directly the age, the model learns the distribution of probability of the input to be in a certain bin. They also use multi-tasking in an end-to-end fashion, regressing from the learnt distribution the age through a linear layer and a softmax activation function. In [8], the authors use a GAN-like architecture to

reconstruct facial images with certain ages. They use the sum of 4 losses in order to finally regress the age from an image. In [9], the authors use an approach similar to Regression via Classification, but instead of projecting the continuous target values into one discrete representation through bins, they do it in multiple ways.

A systematic evaluation and statistical analysis of vanilla deep regression, (i.e. convolutional neural networks with a linear regression top layer) is presented in [28]. The authors use as base architectures VGG-16 and ResNet-50 in the context of three distinct problems: head-pose estimation, facial land- mark detection and human-body pose estimation. They analyze the impact of different network optimizers, batch sizes, batch normalization, dropout and they compare three distinct loss functions: Mean Squared Error, Mean Absolut Error and the Huber loss.

Related to the medical domain, the most popular regression task is estimating bone age from pediatric hand xRays, which was framed as a challenge in 2017, releasing a dataset developed by Stanford University and the University of Colorado that was annotated by multiple expert observers [27]. The best approaches made use of well known pretrained architectures as Inception3 and ResNet-50 along with data augmentation and ensembling.

3 Investigated approaches

Motivated by the recent great results obtained for image classification task due to new architectures and new training techniques, we try to revitalize the regression problem starting from an image through these new state of the art methods. We have experimented with several neural networks, like *ResNet* [17], *ResNeXt* [18], *SE-ResNet* [19], *EfficientNet* [20], *SK-ResNeXt* [21] and *ResNeSt* [22]. We try different methods for adjusting the final layers of our models.
The first approach just adds a linear layer on top of the feature extractor mentioned above which outputs a single number, between 0 and 100. For this approach we used the smooth *L1* loss, with parameter $\beta = 1$.
The second approach adds on top of the feature extractor a linear layer with 101 output cells, followed by a softmax activation layer, thus predicting the probability distribution of each integer percentage. On top of the linear layer we put another layer with 101 input features (the output of the previous step) and 1 output feature (the number we must regress from the input image). We use as loss function the sum of two losses:

$$loss_1 = L1_{smooth}(\sum_{i=0}^{i=100} i \cdot softmax(f_1)(i), gt) \tag{1}$$

and

$$loss2 = L1_{smooth}(f_2(0), gt). \tag{2}$$

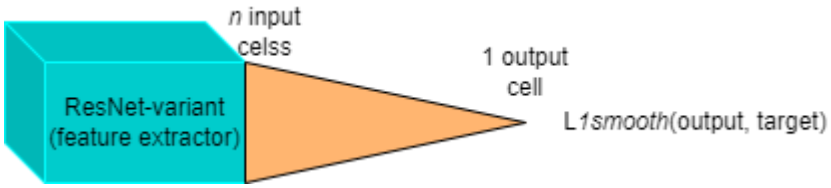
The first loss is used in order to learn the expectancy of the number we must regress, whereas the second loss is the loss used in the first method. gt stands

for ground truth for the current input image. $softmax(f_1)(i)$ represents the i^{th} element of the output of the first added layer on top of the feature extractor after softmax application. $f_2(0)$ represents the output number of the second added layer.
We also try another trick, where instead of approximating the probability expectation, we approximate the probability distribution itself through the KL-divergence loss. If an input image has $p\%$ target, the distribution will be

$$P(output = y) = \begin{cases} 0 \text{ if } y \leq p-3 \text{ or } y \geq p+3 \\ 0.6 \text{ if } y = p \\ 0.15 \text{ if } y = p-1 \text{ or } y = p+1 \\ 0.05 \text{ if } y = p-2 \text{ or } y = p+2 \end{cases} \quad (3)$$

Thus, other than the two losses presented in the previous method, we compute the third loss by being the KL-divergence between the predicted probability distribution and the target one. The final loss will be the sum of the three losses.
We also try to improve the power of the feature extractor, according to [3], and we replace its global average pooling with a hybrid pooling mechanism.

Fig. 1. The first approach, with a simple linear layer with exactly one output cell



3.1 Training procedure

We believe this step is very important, as we bring modern training techniques used for image classification task into estimation tasks.
As training procedure we use *SAM + SGD* [23] as optimizer and cosine annealing with warm-up [24] as learning rate scheduler. The initial learning rate is *1e-3*. We train every model for 50 epochs. In order to avoid overfitting, we use Random Augmentation [25] as a strong regularization with the following list of augmentations: rotation between 0 and 30 degrees, Color, Contrast, Brightness, Sharpness, ShearX, ShearY, Cutout, TranslateX, TranslateY. We do not rescale the input and keep the original size of 512×512 .
Out of all the feature extractors we tried, we notice that *SK-ResNeXt* and *ResNeSt* (both are based on branch attention [26]) give the best results, no matter what last layer method we use. We stick with *ResNeSt* for final architecture. The final ensemble also contains a few *ResNeXt* models, as we noticed it boosted the score a little bit more.

Fig. 2. The second approach, with two linear layers, learning from two tasks

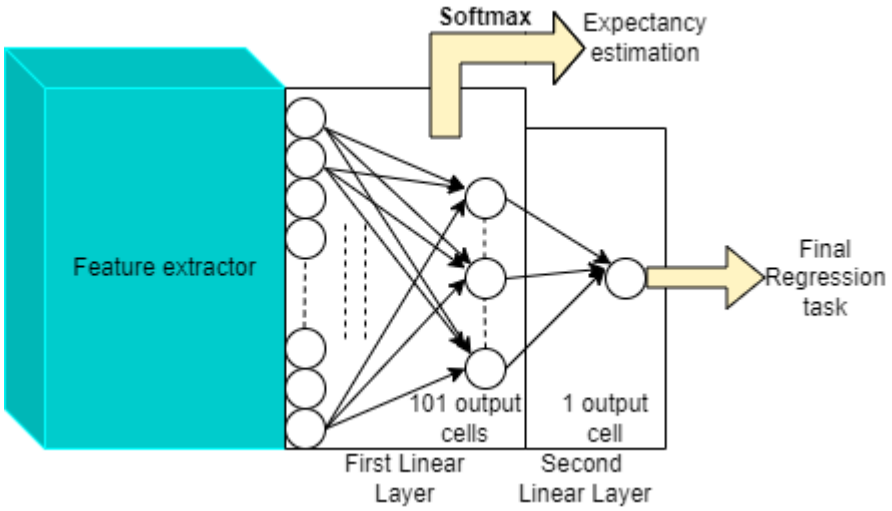
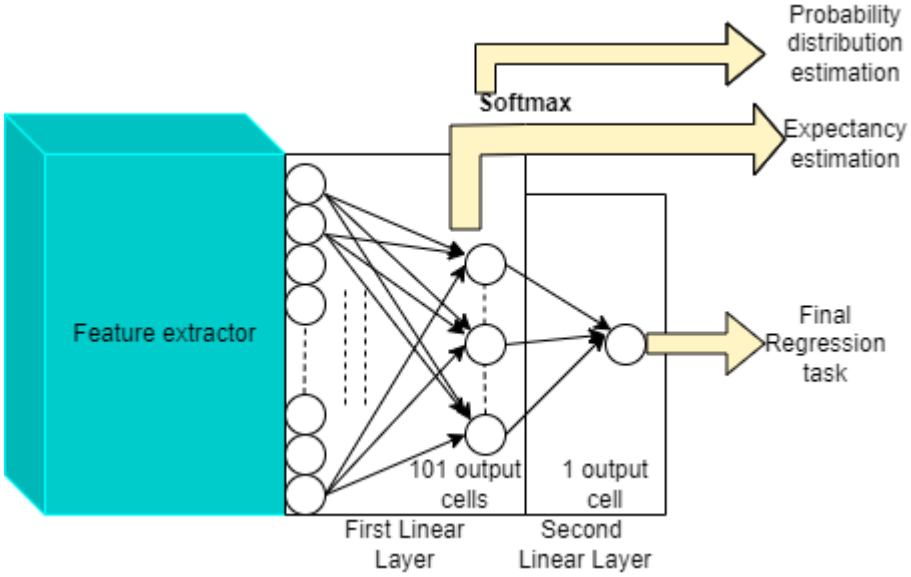


Fig. 3. The third approach, with two linear layers, learning from three tasks



3.2 Inference procedure

During inference, the output is always rounded to the closest integer. We only consider the output of the regression task during inference. We use ensemble models, gathering predictions from models trained on all our 5 folds. For each model trained, we use the checkpoints from the last 5 epochs at inference time. We noticed the results are slightly better when we use just two of our folds. When combining predictions from different models into a single prediction, we simply compute the mean of the predictions and round it to the closest integer. We also notice that the results get a little bit better if we combine *ResNeSt* models with one *ResNeXt* model.

4 Experimental analysis

4.1 Dataset

The dataset provided in the competition consists of several slices from 183 CT scans. The organizers of the competition mention in [3], the importance of COVID-19 percentage estimation in order to establish the severity of the case. We split the training set into 5 folds, taking into consideration the distribution of the labels Normal, Minimal, Moderate, Extent, Severe, and Critical as described in [3].

4.2 Ablation study

Table 1 presents the results on the validation set obtained with the various setups for individual models and training procedures that were described above.

Table 1. Results on validation dataset, standalone models

Model	Method	MAE
ResNeXt50 (1)	5 folds, 30 epochs, only the regression task	4.601076
ResNeXt50 (2)	2 folds, 30 epochs, only the regression task	4.521138
ResNeSt50 (3)	5 folds, 30 epochs, only the regression task	4.654881
ResNeSt50 (4)	2 folds, 30 epochs, only the regression task	4.516526
ResNeXt50 (5)	2 folds, 50 epochs, 2 tasks, without hybrid pooling	4.554189
ResNeSt50 (6)	2 folds, 50 epochs, 2 tasks, without hybrid pooling	4.343582
ResNeSt50 (7)	2 folds, 50 epochs, 2 tasks, with hybrid pooling	**4.285934**
EfficientNet-b4 (8)	2 folds, 50 epochs, 2 tasks, with hybrid pooling	4.8701
SE-ResNet50 (9)	2 folds, 50 epochs, 2 tasks, with hybrid pooling	4.797079
SK-ResNeXt50 (10)	2 folds, 50 epochs, 2 tasks, with hybrid pooling	4.405842
ResNeSt101 (11)	2 folds, 50 epochs, 2 tasks, with hybrid pooling	4.33359
ResNeSt50 (12)	2 folds, 50 epochs, 3 tasks, without hybrid pooling	4.408916
ResNeSt50 (13)	2 folds, 50 epochs, 3 tasks, with hybrid pooling	4.335127

We can notice that *ResNeSt* and *ResNeXt* compete on par when designed with the most simple method of training, that of adding only a linear layer with

one output cell. We can also conclude that using only 2 folds (carefully selected) out of the 5 constructed, improves the results. For the further experiments we only use the two folds that provided the best results from the first 4 experiments. From model (5) we can see that adding extra tasks for *ResNeXt*, does not bring any improvements, whereas for *ResNeSt* (model (6)), the improvements are clear. Using hybrid pooling, instead of global average pooling before the added linear layers, also adds some improvement to the overall result (model(7, 13)). Adding the third task to the training procedure does not seem to bring benefits for the standalone model itself, but brings small improvements when used in an ensemble.
After deciding which were the best models, we started assembling models to improve the overall result. The result obtained on the validation set, which placed us on the first position in the competition in the validation phase, as well as the constituents of the ensemble are reported in Table 2. This ensemble was used in the test phase where it recorded a 4.61 MAE on the test set, being ranked third in the competition.

Table 2. Results on validation dataset, best ensemble

Model	Method	MAE
models (5, 7, 11, 13)	each model trained with the configs mentioned in 1	**4.171407**

Self-supervision can be further used to improve the results [29]. We applied pseudo-labelingand extended the original training set with the inclusion of the validation dataset for which we use instead of the ground truth (which is not available) the percentages predicted by the best ensemble model. All models retrained on the extended dataset provided better results compared with their counterpart trained just with the training set, as illustrated in Table 3.

Table 3. Results on validation dataset, best ensemble

Model	Method	MAE
ResNeSt50	2 folds, trained for 50 epochs, 2 tasks, with hybrid pooling	4.189854
ResNeSt50	2 folds, trained for 50 epochs, 3 tasks, with hybrid pooling	4.172175
ResNeSt101	2 folds, trained for 50 epochs, 2 tasks, with hybrid pooling	4.156034
ResNeSt200	2 folds, trained for 50 epochs, 2 tasks, with hybrid pooling	4.448885
ResNeSt101	2 same configuration, without rounding when inferencing	**4.133743**

We noticed during our trials to ensemble the new models that no ensemble can top the best single model trained on the the extended training set using pseudo-labeling.

5 Conclusions

Deep regression methods, built on existing deep learning models pre-trained for classification tasks in computer vision, may be important tools for assisting medical diagnosis. In this context, reframing the regression task as multi-task learning proves once again to bring a significant increase in performance.

References

1. Bougourzi, Fares and Distante, Cosimo and Ouafi, Abdelkrim and Dornaika, Fadi and Hadid, Abdenour and Taleb-Ahmed, Abdelmalik: Per-COVID-19: A Benchmark Dataset for COVID-19 Percentage Estimation from CT-Scans; Journal of Imaging, Volume 7, 2021, 9-189 https://doi.org/10.3390/jimaging7090189
2. Vantaggiato, Edoardo and Paladini, Emanuela and Bougourzi, Fares and Distante, Cosimo and Hadid, Abdenour and Taleb-Ahmed, Abdelmalik: Covid-19 recognition using ensemble-cnns in two new chest x-ray databases; Sensors, Volume 21, 2021, 5-1748
3. Bin-Bin Gao and Xinxin Liu and Hong-Yu Zhou and Jianxin Wu and Xin Geng: Learning Expectation of Label Distribution for Facial Age and Attractiveness Estimation; CoRR, Volume abs/2007.01771, 2020
4. Bin-Bin Gao and Chao Xing and Chen-Wei Xie and Jianxin Wu and Xin Geng: Deep Label Distribution Learning with Label Ambiguity, CoRR, Volume abs/1611.01731, 2016
5. Yunxuan Zhang and Li Liu and Cheng Li and Chen Change Loy: Quantifying Facial Age by Posterior of Age Comparisons, CoRR, Volume abs/1708.09687, 2017
6. Wenzhi Cao and Vahid Mirjalili and Sebastian Raschka: Consistent Rank Logits for Ordinal Regression with Convolutional Neural Networks, CoRR, Volume abs/1901.07884, 2019
7. Chao Zhang and Shuaicheng Liu and Xun Xu and Ce Zhu: C3AE: Exploring the Limits of Compact Model for Age Estimation, CoRR, Volume abs/1904.05059, 2019
8. Haiping Zhu and Qi Zhou and Junping Zhang and James Z. Wang: Facial Aging and Rejuvenation by Conditional Multi-Adversarial Autoencoder with Ordinal Regression, CoRR, Volume abs/1804.02740, 2018
9. Axel Berg and Magnus Oskarsson and Mark O'Connor: Deep Ordinal Regression with Label Diversity, CoRR, Volume abs/2006.15864, 2020
10. S. Escalera et al., "ChaLearn Looking at People 2015: Apparent Age and Cultural Event Recognition Datasets and Results," 2015 IEEE International Conference on Computer Vision Workshop (ICCVW), 2015, pp. 243-251, doi: 10.1109/ICCVW.2015.40.
11. K. Ricanek and T. Tesafaye, "MORPH: a longitudinal image database of normal adult age-progression," 7th International Conference on Automatic Face and Gesture Recognition (FGR06), 2006, pp. 341-345, doi: 10.1109/FGR.2006.78.
12. Zhifei Zhang and Yang Song and Hairong Qi: Age Progression/Regression by Conditional Adversarial Autoencoder, CoRR, Volume abs/1702.08423, 2017
13. D. Xie, L. Liang, L. Jin, J. Xu and M. Li, "SCUT-FBP: A Benchmark Dataset for Facial Beauty Perception," 2015 IEEE International Conference on Systems, Man, and Cybernetics, 2015, pp. 1821-1826, doi: 10.1109/SMC.2015.319.
14. Ma, D.S., Correll, J. & Wittenbrink, B. The Chicago face database: A free stimulus set of faces and norming data. Behav Res 47, 1122–1135 (2015). https://doi.org/10.3758/s13428-014-0532-5

15. Y. Fu, G. Guo and T. S. Huang, "Age Synthesis and Estimation via Faces: A Survey," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 32, no. 11, pp. 1955-1976, Nov. 2010, doi: 10.1109/TPAMI.2010.36.
16. K. Chang, C. Chen and Y. Hung, "Ordinal hyperplanes ranker with cost sensitivities for age estimation," CVPR 2011, 2011, pp. 585-592, doi: 10.1109/CVPR.2011.5995437.
17. Kaiming He and Xiangyu Zhang and Shaoqing Ren and Jian Sun: Deep Residual Learning for Image Recognition, CoRR, Volume abs/1512.03385 2015
18. Saining Xie and Ross B. Girshick and Piotr Dollár and Zhuowen Tu and Kaiming He: Aggregated Residual Transformations for Deep Neural Networks, CoRR, Volume abs/1611.05431, 2016
19. Jie Hu and Li Shen and Gang Sun: Squeeze-and-Excitation Networks, CoRR, Volume abs/1709.01507, 2017
20. Mingxing Tan and Quoc V. Le: EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks, CoRR, Volume abs/1905.11946, 2019
21. Xiang Li and Wenhai Wang and Xiaolin Hu and Jian Yang: Selective Kernel Networks, CoRR Volume abs/1903.06586, 2019
22. Hang Zhang and Chongruo Wu and Zhongyue Zhang and Yi Zhu and Zhi Zhang and Haibin Lin and Yue Sun and Tong He and Jonas Mueller and R. Manmatha and Mu Li and Alexander J. Smola: ResNeSt: Split-Attention Networks, CoRR, Volume abs/2004.08955, 2020
23. Pierre Foret and Ariel Kleiner and Hossein Mobahi and Behnam Neyshabur: Sharpness-Aware Minimization for Efficiently Improving Generalization, CoRR, Volume abs/2010.01412, 2020
24. Ilya Loshchilov and Frank Hutter: SGDR: Stochastic Gradient Descent with Restarts, CoRR, Volume abs/1608.03983, 2016
25. Ekin D. Cubuk and Barret Zoph and Jonathon Shlens and Quoc V. Le: RandAugment: Practical data augmentation with no separate search, CoRR, Volume abs/1909.13719, 2019
26. Meng-Hao Guo and Tian-Xing Xu and Jiangjiang Liu and Zheng-Ning Liu and Peng-Tao Jiang and Tai-Jiang Mu and Song-Hai Zhang and Ralph R. Martin and Ming-Ming Cheng and Shi-Min Hu: Attention Mechanisms in Computer Vision: A Survey, CoRR, Volume abs/2111.07624, 2021
27. Halabi SS, Prevedello LM, Kalpathy-Cramer J, et al. The RSNA Pediatric Bone Age Machine Learning Challenge. Radiology 2018; 290(2):498-503.
28. Lathuilière, Stéphane, et al. "A comprehensive analysis of deep regression." IEEE transactions on pattern analysis and machine intelligence 42.9 (2019): 2065-2081.
29. Miron Radu, Moisii Cosmin, Dinu Sergiu, Breaban Mihaela. Evaluating volumetric and slice-based approaches for COVID-19 detection in chest CTs. InProceedings of the IEEE/CVF International Conference on Computer Vision 2021 (pp. 529-536).