CNN Hyperparameter Optimization for Pulmonary Nodule Classification

Anthony E. A. Jatobá[1], Lucas L. Lima[2], Lucas B. V. Amorim[3], Marcelo C. Oliveira[1]

[1]Programa de Pós-Graduação em Informática (PPgI),
Instituto de Computação, Universidade Federal de Alagoas (UFAL)
Maceió, AL – Brasil

[2]Programa de Pós-graduação Interunidades em Bioengenharia,
Universidade de São Paulo – Ribeirão Preto, SP – Brasil

[3]Instituto de Computação, Universidade Federal de Alagoas (UFAL)
Maceió, AL – Brasil

{aeaj,lucas,oliveiramc}@ic.ufal.br, lucaslima7@usp.br

Abstract. *Convolutional Neural Networks (CNNs) are a powerful tool to develop image-based computer-aided diagnosis systems, but as these models become more complex, manual configuration becomes unfeasible. Automatic Hyperparameter Optimization is a promising approach to model tuning, but there is no agreement on what algorithm is the right choice. In this work, we compared direct search, probabilistic search and bayesian optimization for tuning 2D and 3D CNNs for lung nodule classification. Our models achieved an AUC of 0.88, sensitivity of 87.03%, and specificity of 78.66%. Moreover, our experiments brings evidence on the weak performance of grid search, while showing that simple techniques such as random search can match probabilistic approaches.*

Resumo. *Redes Neurais Convolucionais (RNCs) são uma técnica poderosa para sistemas de diagnóstico auxiliado por computador, mas a configuração manual de redes complexas é inviável. A otimização automática de hiperparâmetros é uma abordagem promissora, mas não há consenso sobre a técnica mais adequada. Neste trabalho, comparamos busca direta, probabilística e otimização bayesiana na otimização de RNCs 2D e 3D para classificação de nódulos pulmonares. Foram obtidas AUC de 0,88, sensibilidade de 87,03% e especificidade de 78,66%. Nossos experimentos demonstram o fraco desempenho da busca em grade, enquanto mostram que técnicas simples, como a busca aleatória, pode ter desempenho comparável a abordagens probabilísticas.*

1. Introduction

Lung cancer is the primary cause of cancer-related death worldwide, with 1.8 million estimated deaths in 2018 [Bray et al. 2018, World Health Organisation 2019]. Early detection can improve the effectiveness of patients' treatment, being a decisive survivability factor. If identified in its early stages, survival rates achieve results above 90%, in contrast to a mere 15% when diagnosed in its last phases [Blandin Knight et al. 2017]. Computed Tomography (CT) scan is the preferred method for early lung cancer detection, producing a volume of slices in high definition and contrast that allows to find small, abnormal

areas (called nodules or masses) in the lungs, especially in current or former smokers [Blandin Knight et al. 2017, Hua et al. 2015].

However, the lung cancer diagnosis is still quite challenging, as each image generated by the CT exam has to be carefully evaluated, a time-consuming task subject to a series of internal and external factors, such as equipment quality, expertise level, and work-related fatigue [Chuquicusma et al. 2018]. Furthermore, most lung nodules seen on CT scans are not cancer. They are more often the result of old infections, scar tissue, or other causes. Those factors combined often lead to inconsistencies in the diagnosis between different specialists or even the same specialist in different circumstances [Kang et al. 2017]. Computer-Aided Diagnosis (CADx) tools try to relieve this problem by providing a second opinion to the diagnosis, enhancing its speed and accuracy [Hua et al. 2015, da Silva et al. 2017, Kumar et al. 2015].

The typical pipeline for designing CAD systems constitutes of: 1) extracting features from the nodules' volumes or slices; 2) using the features to train a Machine Learning (ML) model for detection (CADe) or diagnosis (CADx) [Ferreira et al. 2018]. In the past few years, Deep Learning (DL) emerged as a promising approach for CAD systems design [Litjens et al. 2017, Sun et al. 2016, Zhu et al. 2018]. This family of techniques is capable of learning high-level representations directly from the data, without need for a feature extracting step.

As CNN models evolve, they have become much more complex, requiring increasingly higher amounts of time to be designed, trained, and evaluated. Since these models are very susceptible to their settings, proper configuration is a challenge [Montavon et al. 2012]. Manual configuration of these parameters through experimentation is becoming each time less feasible, still, much of the recent work on DL consists in proposing hand-designed architectures [Miikkulainen et al. 2019]. In this scenario, automatic Hyperparameter Optimization (HO) is a promising approach for model tuning, presenting competitive results to manual tuning made by specialists and allowing them to focus on other aspects of the model development such as data acquisition and processing [Bergstra and Bengio 2012]. Nonetheless, in the face of a myriad of options for HO, it is still desirable to choose one that can lead to better results within limited time constraints.

In this study, we intended to discuss the impact of different HO techniques in optimizing 2D and 3D CNNs for pulmonary nodule classification in regards to performance and time consumption.

The remainder of this paper is organized as follows: section 2 presents related works on pulmonary nodules classification; section 3 describes the data and methodology used in this work; section 4 presents the obtained results and its discussion; section 5 concludes this work.

2. Related Work

Until the last decade, lung nodule classification was performed by extracting features from medical images for a machine learning classifier, such as SVM, Random Forest and Artificial Neural Networks. With the growth of Deep Learning as a viable approach, the attention quickly switched to this family of techniques, leading to significant improvements [Litjens et al. 2017, Yang et al. 2018].

Hua et al. [Hua et al. 2015] proposed using Deep Belief Networks (DBNs) and CNNs for the classification of pulmonary nodules into benign and malignant. A set of 2,545 CT scans containing nodules larger than 3mm were selected from the LIDC-IDRI dataset. The results were obtained through leave-one-out cross-validation for DBN and CNN, as well as a K-nearest neighbor and support vector machine models implemented as baselines. The DBN model reached a sensitivity of 73.4% and specificity of 82.2%, while the CNN got 73.3% and 78.7%, endorsing deep learning effectiveness in pulmonary nodules classification. However, as a seminal work using deep learning to classify medical images, the results were quickly outperformed.

Shen et al.[Shen et al. 2015] proposed a multi-scale CNN (MCNN) approach, where different sized patches were extracted from each nodule in the CT scans, each patch being fed into a 3D CNN and combined to extract features to be used by SVM or Random Forest (RF) classifiers. The approach was validated in 1,375 nodules (880 benign and 495 malignant) from the LIDC-IDRI dataset using 5-fold cross-validation. The best model achieved an accuracy of 86.64% and was able to deal with noisy input. Eight different CNN configurations were evaluated, and the SVM and RF classifiers were optimized with grid-search. Nevertheless, the authors concluded that better results could be obtained with a more comprehensive optimization strategy.

Kang et al.[Kang et al. 2017] evaluated distinct multi-scale 3D CNN architectures for binary (benign and malignant) and ternary (benign, malignant and metastatic malignant) classification of pulmonary nodules. The models were trained on 776 nodules (186 benignant and 590 malignant) from the LIDC-IDRI dataset. The models used were multi-view CNN, a 3D CNN with chain architecture, and a 3D CNN with a Direct Acyclic Graph (DAG) architecture. The best models reached error rates of 4.59% in binary classification and 7.70% in ternary classification. The results were obtained by performing 10-fold cross-validation. Nonetheless, the authors performed data augmentation in the test sets, which is not a common practice.

Dey et al. [Dey et al. 2018] proposes four multi-view 3D CNN architectures for nodule classification into benign and malignant. The models were evaluated on a private dataset of 147 nodules (37% benign and 63% malignant). Since the number of samples is small, the networks were pre-trained with 686 nodules (46% benign and 54% malignant) from the LIDC-IDRI dataset. The best model achieved an AUC of 0.86 without transfer learning and 0.90 after the pre-training. The results were obtained through 5-fold cross-validation. The size of the dataset and its privacy makes it difficult to compare other results with this work.

Onishi et al.[Onishi et al. 2019] used Generative Adversarial Networks (GAN) to create synthetic samples of pulmonary nodules for training a 3D CNN. The GAN was trained on 60 nodules (27 benign and 33 malignant) from a private dataset. This approach lead to a classification accuracy of 81.7%, a 20% increase compared to using only the original data. The small size of the dataset may be detrimental to this work generalization.

3. Material and Methods

Figure 1 summarizes the steps of our methodology. We used a subset of nodules from the LIDC-IDRI dataset (Section 3.1), then we segmented the nodules slice by slice. Data was

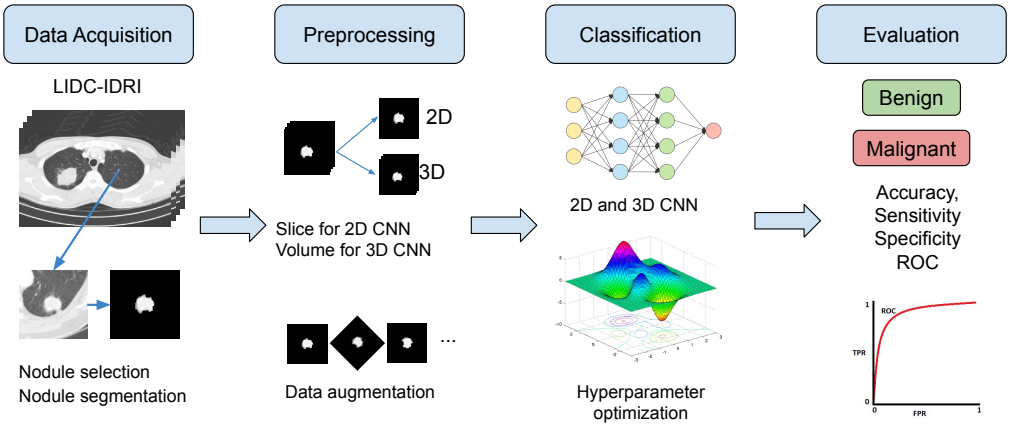

Figure 1. Overview of our methodology.

prepared for our models (Section 3.2) by selecting slices for 2D CNNs and assembling a volume for the 3D CNNs and then balanced and augmented via successive rotations. Our models settings are chosen by four HO techniques (grid search, random search, simulated annealing and Tree-structured Parzen Estimator - Section 3.4) and then, the models were evaluated according to a set of metrics and validated with 10-fold cross-validation (Section 3.5).

3.1. Data Acquisition

We used the LIDC-IDRI dataset, a public repository containing 1,018 CT scans from 1,010 patients [Armato et al. 2011]. Four experienced radiologists reviewed each scan, identifying and evaluating the lesions in regards to a series of pathologic features: calcification, internal structure, lobulation, margins, sphericity, spiculation, subtlety, malignancy, and texture. The malignancy probability is defined on a five-point scale, where a 1 is a high chance of being benign and a 5 is a high chance of being malignant. The annotations also include a freehand outline of nodules with a diameter larger than 3mm.

Table 1. Nodules distribution according to its malignancy probability.

	Benign		Malignant	
Malignancy Probability	1	2	4	5
Number of nodules	304	394	171	137
Total by class	698		308	

A pulmonary nodule is defined as a focal opacity with a diameter between 3mm and 30mm, so we considered the lesions within these dimensions [da Silva et al. 2017]. We discarded the nodules with an undefined malignancy (probability value of 3). Lastly, we selected only the solid nodules, as their contour is drawn with a higher precision by the radiologists. As a result, a total of 1,006 nodules were extracted from the LIDC-IDRI database. Table 1 shows how they are distributed according to their malignancy.

We then applied a greyscale lung windowing by setting the window in 1,600 and level in -600 Hounsfield units to standardize the images contrasts. Then, we used the outline drawn by the radiologists to segment each nodule slice by slice.

3.2. Preprocessing

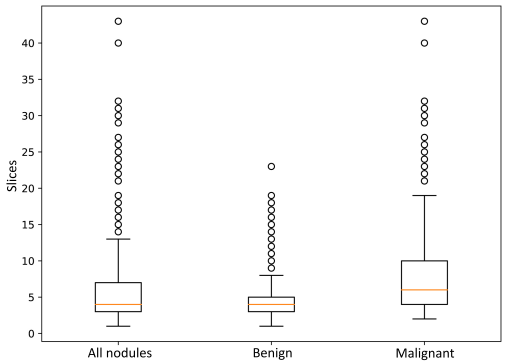

Figure 2. Distribution of slice numbers in our nodules set.

For this work, we used 2D and 3D CNNs for lung nodule classification. It is important to notice that each model requires an input of distinct dimensions: a 2D CNN expects an image as input while a 3D CNN expects a volume. So, for the 2D CNN, we selected the slice with the largest diameter as obtained in Section 3.1 for each nodule. Then, each slice is cropped to an image measuring 64x64 pixels. On the other hand, for the 3D CNN, we could not feed the nodules directly into the CNN, as each nodule had a different number of slices, as shown in Figure 2.

In this particular case, we needed an approach to assemble a volume for the model. We did so by selecting the first 5 slices of each nodule. In the cases of nodules with less than 5 slices, we added all-black images to the end of the volume, until completing the desired number of slices.

As Table 1 shows, benign and malignant classes are unbalanced, with 698 benign nodules and 308 malignant ones. This imbalance may lead to models biased to the classification of benign cases. Hence we solved this problem by performing consecutive rotations in the slices for the 2D CNN and volumes for the 3D CNN, as a way to both balance and augment the data. This method is commonly used in literature [Kang et al. 2017, Onishi et al. 2019]. The algorithm was performed as follows:

1. 10% of the nodules were selected for composing the test set, which was kept unchanged; the 90% remaining nodules were used to compose the training set.
2. We rotated the benign nodules from the training set in intervals of 72°, making a total of 3,140 samples;
3. the malignant nodules were rotated in intervals of 30°, generating 3,324 samples.

For the HO step, data augmentation was performed once, while on the validation step it was necessary to execute the process for each iteration of the 10-fold cross-validation.

3.3. Classification

Convolutional Neural Networks (CNN) constitute a class of neural networks able to learn multi-level hierarchies of features [Goodfellow et al. 2016]. The network extracts features of spatial features from the input and classifies patterns from these features. A typical CNN consists of convolutional, pooling and dense layers that can extract multi-level learnable representations to solve a particular task.

Table 2. Search Space

Layer	Parameter	Grid Search	Random Search, TPE Simulated Annealing
Convolutional	Number of filters	{32, 64, 96}	[32, 96]
Dense 1	Number of units	{32, 128, 256}	[32, 256]
	Dropout	{0.3, 0.6}	[0, 0.6]
Dense 2	Number of units	{16, 32, 48}	[16, 48]
	Dropout	{0.3, 0.6}	[0, 0.6]
-	Epochs	{5, 15, 30}	[5, 30]

A conventional CNN operates over 2D inputs, so each channel on the network is a 2D feature map. With 3D CNNs, unlike the 2D ones, the operations are performed in a cubic manner, generating 3D feature volumes.

3.4. Hyperparameter Optimization

Neural networks can perform a wide range of applications, but not every network architecture is capable of performing a given task successfully [da Silva et al. 2017]. Therefore, network architecture needs to be tuned to obtain the best results. The objective of Hyperparameter Optimization (HO) is to find a set of hyperparameters for a ML algorithm that results in a better performance in a validation set [Bergstra et al. 2011]. It can be formalized by:

$$x^* = arg \min_{x \in X} f(x) \tag{1}$$

Where $f(x)$ is the objective function to be optimized, x^* is the optimal set of hyperparameter values that minimize the objective function $f(x)$ and x can assume any combination of values in a X domain of hyperparameters values.

A wide variety of techniques for HO are available, however, the cost of training neural networks makes several authors to focus on other steps on the model design pipeline [Claesen and De Moor 2015]. In this work, we evaluated four distinct algorithms for the task: two direct search algorithms (Grid Search and Random Search), as well as a probabilistic (Simulated Annealing) and a Bayesian Optimizer (Tree-structured Parzen Estimator).

We defined the search space for each algorithm empirically. On Random Search, TPE, and Simulated Annealing, the optimization occurred within the limits of the search space. Using Grid Search, we had to define a discrete set of possibilities for each hyperparameter, as a way to keep the search feasible. The search space for each algorithm is presented in Table 2. For random search, TPE and Simulated Annealing optimization, the number of trials was limited to 70, while Grid Search was allowed to explore all its 324 possible combinations.

Our network architecture was defined empirically and contains a single convolutional layer, a single max-pooling layer, followed by two fully connected layers and then a single unit for the output. The activation function is *ReLU* for the hidden layers and *sig-*

moid for the output layer. We used the *RMSProp* optimizer with a learning rate of 10^{-4}. The loss function utilized was *binary cross-entropy*.

3.5. Results Evaluation

After training our models, it is necessary to validate its results. Our evaluation methodology uses a set of metrics commonly used in CADx systems: accuracy, sensitivity and specificity, Receiver Operating Characteristic (ROC) curve and area under the ROC curve (AUC). ROC curve and AUC are well-established metrics in the literature of CADx systems, so, they will be our choice metric of comparison. This set of metrics is well-known in classification problems in health and are proper descriptors of the model's behavior [Fawcett 2006]. Model validation was performed with 10-fold cross-validation. For every network, we utilized the same training and test sets, making our comparisons fairer.

4. Results and Discussion

This section presents and discusses the results obtained from the proposed methodology. All experiments were implemented in Python using Keras deep learning library [Chollet et al. 2015] with Tensorflow as backend [Abadi et al. 2016]. For hyperparameter optimization, we used Hyperas [Pumperla 2019], which encapsulates the algorithms implementations from Hyperopt [Bergstra et al. 2013]. The hardware consisted of a system equipped with a Intel Core i7-5960X, 128 GB of RAM and a 12 GB GeForce GTX Titan X GPU.

4.1. Hyperparameter Optimization

Our first result consists of the network architectures provided by the HO step. The values obtained for each hyperparameter are shown in Table 3. There, *Filters* stands for the number of filters in the convolutional layer, *Dense1* and *Dense2* to the number of units in the dense layers, *Dropout1* and *Dropout2* are the dropout rate for the *Dense1* and *Dense2* layers, and *Epochs*, the number of training epochs.

Table 3. Hyperparameters obtained by each HO algorithm.

	Optimization	Filters	Dense1	Drop1	Dense2	Drop2	Epochs
2D CNN	Grid Search	64	256	0.30	48	0.60	5
	Random Search	80	106	0.35	40	0.25	21
	Sim. Annealing	47	145	0.58	40	0.05	24
	TPE	78	124	0.55	42	0.13	25
3D CNN	Grid Search	64	256	0.30	48	0.60	5
	Random Search	69	225	0.36	38	0.50	7
	Sim. Annealing	38	153	0.48	41	0.25	24
	TPE	41	68	0.55	38	0.14	17

Grid Search optimization resulted in the same architectures for both 2D and 3D CNNs. The 2D models needed 18.75 epochs on average, slightly more than the 13.25 epochs required by the 3D CNNs. Table 4 summarises the amount of time spent by each algorithm in optimizing our models.

Grid search took the highest amount of time, but we should remind that this technique evaluated a higher amount of models than the other algorithms. Random Search and TPE had an equivalent cost. Simulated Annealing took the least amount of time with either 2D or 3D CNN.

Table 4. Time spent by each technique in optimizing 2D and 3D CNNs.

	2D CNN	3D CNN
Grid Search	3 hours and 34 minutes	12 hours
Random Search	30 minutes	1 hour and 24 minutes
Sim. Annealing	8 minutes	20 minutes
TPE	30 minutes	1 hour and 36 minutes

4.2. Classification Results

The results obtained for each performance metric from our methodology are presented in Table 5.

Except for the 3D CNN optimized with Simulated Annealing, which presented an AUC of 0.86, every other model obtained an AUC of 0.88.

The models also presented a similar accuracy; however, this metric is not ideal to evaluate classification of unbalanced classes. Instead, sensitivity and specificity are better candidates to assess model performance.

The 2D CNN optimized with Simulated Annealing had the best accuracy of 79.13%, a consequence of its high specificity. This model in particular is well suited to the classification of benign nodules, to the cost of exhibiting the worst sensitivity of all models. The 3D CNN optimized with Random Search reached a sensitivity of 87.03%, but the lowest specificity value. This model is more adequate for classifying malignant nodules. Figure 3 presents the ROC curves for these models.

From these results, we can draw a few conclusions about the impact of the HO technique over the models' performance. Despite making more trials than the other algorithms, grid search results didn't stand out in any particular way, being a poor choice for the task.

Table 5. Classification results for the models provided by each HO technique.

	Optimization	Accuracy	Sensitivity	Specificity	AUC
2D CNN	Grid Search	77.43%	81.84%	75.49%	**0.88**
	Random Search	76.83%	84.12%	73.62%	**0.88**
	Sim. Annealing	**79.13**%	80.18%	**78.66**%	**0.88**
	TPE	78.14%	81.15%	76.80%	**0.88**
3D CNN	Grid Search	77.93%	85.41%	74.64%	**0.88**
	Random Search	77.55%	**87.03**%	73.36%	**0.88**
	Sim. Annealing	76.35%	81.91%	73.92%	0.86
	TPE	78.04%	82.17%	76.22%	**0.88**

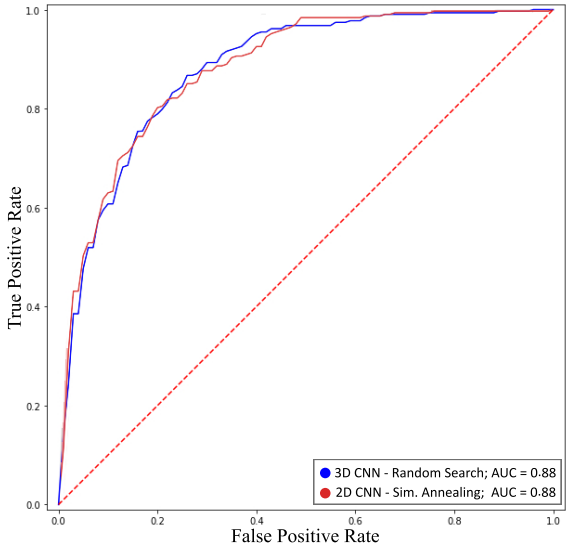

Figure 3. ROC Curve for 2D CNN and 3D CNN optimized by Simulated Annealing and Random Search, respectively.

Random search models had higher sensitivity, but some of the lowest specificity values, with a negative impact on accuracy. This algorithm has one of the most straightforward implementations, and its results are competitive with its probabilistic and bayesian competitors.

Simulated annealing took the least amount of time to optimize both models. A possible explanation is that the temperature mechanism allowed for the model to focus the search in a region of the search space that contains simpler and faster to train models. As for the models, this algorithm led to the best accuracy and specificity values with 2D CNN, but also the lowest AUC with 3D CNN.

TPE models presented good results and with a balance between sensitivity and specificity and some of the best accuracy values. Despite the extra cost involved in bayesian optimization, the time spent was the same as with random search.

5. Conclusion

In this paper, we presented a comparison between four Hyperparameter Optimization algorithms, covering direct search (grid search and random search), a probabilistic search (simulated annealing) and a bayesian optimizator (TPE) on the task of optimizing a 2D and 3D CNN for pulmonary nodule classification.

The optimized models presented satisfactory results, with the majority of the models presenting an AUC of 0.88, a strong indicator of performance on distinguishing the benign and malignant nodules. The recurrence of this value may suggest a limitation on our dataset. Furthermore, our models achieved an accuracy of 79.13%, sensitivity of 87.03%, and specificity of 78.66%, allowing the choice for models with different capacities on classifying benign or malignant lesions.

In regards to HO, grid search has proved to be a poor choice for the task. Despite leading to models with equivalent performance as the other techniques, it requires an

increasing number of trials as the search space grows and can be impractical for complex models such as the 3D CNNs.

Simulated Annealing led to the model with higher specificity, but also presented the model with the worst AUC. Further investigation has to be done concerning this algorithm robustness.

Random Search tended to find models with higher sensitivity, as TPE presented models with a better balance between the conflicting metrics sensitivity and specificity. Both algorithms also required equivalent amounts of time.

Both 2D and 3D CNN presented a similar performance in classification, so the 2D model is preferable, as it requires less training and preprocessing. However, the results are very dependent on our methodology, so this can not be generalized.

In future work, we plan to evaluate evolutionary algorithms to the task, as well as allowing the HO techniques to optimize the number of layers and other parameters on the network, such as activation functions and convolution and pooling kernel size.

References

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., et al. (2016). Tensorflow: A system for large-scale machine learning. In *12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)*, pages 265–283.

Armato, S. G., McLennan, G., Bidaut, L., McNitt-Gray, M. F., Meyer, C. R., Reeves, A. P., Zhao, B., Aberle, D. R., Henschke, C. I., Hoffman, E. A., et al. (2011). The lung image database consortium (lidc) and image database resource initiative (idri): a completed reference database of lung nodules on ct scans. *Medical physics*, 38(2):915–931.

Bergstra, J. and Bengio, Y. (2012). Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, 13(Feb):281–305.

Bergstra, J., Yamins, D., and Cox, D. D. (2013). Hyperopt: A python library for optimizing the hyperparameters of machine learning algorithms. In *Proceedings of the 12th Python in science conference*, pages 13–20. Citeseer.

Bergstra, J. S., Bardenet, R., Bengio, Y., and Kégl, B. (2011). Algorithms for hyper-parameter optimization. In *Advances in neural information processing systems*, pages 2546–2554.

Blandin Knight, S., Crosbie, P. A., Balata, H., Chudziak, J., Hussell, T., and Dive, C. (2017). Progress and prospects of early detection in lung cancer. *Open biology*, 7(9):170070.

Bray, F., Ferlay, J., Soerjomataram, I., Siegel, R. L., Torre, L. A., and Jemal, A. (2018). Global cancer statistics 2018: Globocan estimates of incidence and mortality worldwide for 36 cancers in 185 countries. *CA: a cancer journal for clinicians*, 68(6):394–424.

Chollet, F. et al. (2015). Keras. `https://github.com/fchollet/keras`.

Chuquicusma, M. J., Hussein, S., Burt, J., and Bagci, U. (2018). How to fool radiologists with generative adversarial networks? a visual turing test for lung cancer diagnosis. In *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*, pages 240–244. IEEE.

Claesen, M. and De Moor, B. (2015). Hyperparameter search in machine learning. *arXiv preprint arXiv:1502.02127*.

da Silva, G. L., da Silva Neto, O. P., Silva, A. C., de Paiva, A. C., and Gattass, M. (2017). Lung nodules diagnosis based on evolutionary convolutional neural network. *Multimedia Tools and Applications*, 76(18):19039–19055.

Dey, R., Lu, Z., and Hong, Y. (2018). Diagnostic classification of lung nodules using 3d neural networks. In *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*, pages 774–778. IEEE.

Fawcett, T. (2006). An introduction to roc analysis. *Pattern recognition letters*, 27(8):861–874.

Ferreira, J. R., Oliveira, M. C., and de Azevedo-Marques, P. M. (2018). Characterization of pulmonary nodules based on features of margin sharpness and texture. *Journal of digital imaging*, 31(4):451–463.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. MIT Press. `http://www.deeplearningbook.org`.

Hua, K.-L., Hsu, C.-H., Hidayati, S. C., Cheng, W.-H., and Chen, Y.-J. (2015). Computer-aided classification of lung nodules on computed tomography images via deep learning technique. *OncoTargets and therapy*, 8.

Kang, G., Liu, K., Hou, B., and Zhang, N. (2017). 3d multi-view convolutional neural networks for lung nodule classification. *PloS one*, 12(11):e0188290.

Kumar, D., Wong, A., and Clausi, D. A. (2015). Lung nodule classification using deep features in ct images. In *2015 12th Conference on Computer and Robot Vision*, pages 133–138. IEEE.

Litjens, G., Kooi, T., Bejnordi, B. E., Setio, A. A. A., Ciompi, F., Ghafoorian, M., Van Der Laak, J. A., Van Ginneken, B., and Sánchez, C. I. (2017). A survey on deep learning in medical image analysis. *Medical image analysis*, 42:60–88.

Miikkulainen, R., Liang, J., Meyerson, E., Rawal, A., Fink, D., Francon, O., Raju, B., Shahrzad, H., Navruzyan, A., Duffy, N., et al. (2019). Evolving deep neural networks. In *Artificial Intelligence in the Age of Neural Networks and Brain Computing*, pages 293–312. Elsevier.

Montavon, G., Orr, G., and Müller, K.-R. (2012). *Neural networks: tricks of the trade*, volume 7700. springer.

Onishi, Y., Teramoto, A., Tsujimoto, M., Tsukamoto, T., Saito, K., Toyama, H., Imaizumi, K., and Fujita, H. (2019). Automated pulmonary nodule classification in computed tomography images using a deep convolutional neural network trained by generative adversarial networks. *BioMed research international*, 2019.

Pumperla, M. (2019). Hyperas.

Shen, W., Zhou, M., Yang, F., Yang, C., and Tian, J. (2015). Multi-scale convolutional neural networks for lung nodule classification. In *International Conference on Information Processing in Medical Imaging*, pages 588–599. Springer.

Sun, W., Zheng, B., and Qian, W. (2016). Computer aided lung cancer diagnosis with deep learning algorithms. *Medical Imaging 2016: Computer-Aided Diagnosis*, 9785(March):97850Z.

World Health Organisation (2019). Vision impairment and blindness, Fact Sheet N°282. `http://www.who.int/mediacentre/factsheets/fs282/fr/`, Last accessed on 2019-02-14.

Yang, Y., Feng, X., Chi, W., Li, Z., Duan, W., Liu, H., Liang, W., Wang, W., Chen, P., He, J., et al. (2018). Deep learning aided decision support for pulmonary nodules diagnosing: a review. *Journal of thoracic disease*, 10(Suppl 7):S867.

Zhu, W., Liu, C., Fan, W., and Xie, X. (2018). Deeplung: Deep 3d dual path nets for automated pulmonary nodule detection and classification. In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 673–681. IEEE.