DOI: 10.1109/CISS56502.2023.10089673 https://ieeexplore.ieee.org/document/10089673

Tools and Visualizations for Exploring Classification Landscapes

William Powers
Queens College, CUNY
New York, NY, USA
william.powers81@qmail.cuny.edu

Lin Shi
Queens College, CUNY
New York, NY, USA
lin.shi66@qmail.cuny.edu

Larry S. Liebovitch
Queens College, CUNY; Columbia University
New York, NY, USA
lsl2140@columbia.edu

***Abstract*—Neural networks and deep learning systems find the correct classification of input data by locating the corresponding local minima in the hyper-dimensional, classification landscape. An increasing number of adversarial examples have now shown that these networks sometimes find an unexpected and incorrect minimum and so make an incorrect classification. To understand those results requires a better understanding of the nature of these classification landscapes. Previous studies have explored the properties of the landscape of back propagation in training these networks. In our studies here, we explore the classification landscape of already trained networks. We present some novel procedures and analytical tools to study the classification landscape and visualizations to meaningfully represent those results. We apply these methods to study the classification landscape in classic examples, including image classification in the MNIST data set and flower classification from numerical feature values in the Iris data set.**

***Index Terms*—machine learning; image classification; data visualization; deep learning networks; loss landscapes**

I. Introduction

Machine learning models are capable of task automation, anomaly detection, and predicting outcomes based on data and training algorithms. However, it should be understood that these tools are prone to errors. For example, while certain given articles of clothing such as a shoe and a coat are easily distinguishable by a human being, a machine learning model's capacity to distinguish between these choices can depend on the training data it is supplied and whether it has encountered factors within this training data that would allow it to categorize the article of clothing. There is a case of a neural network classifier incorrectly predicting an image of a revolver handgun as a mousetrap and an image of a vulture as an orangutan, after an image rotation and translation [1]. Finding new means in which the vulnerabilities of these artificial intelligence classifiers are exposed can lead to better understanding the limitations of these tools, and how they can be improved.

An approach to testing the capabilities of machine learning technology is to present it with images that have been modified from their original format when supplied as a test set to the machine learning model. One method of modification involves using a morph sequence generator. This tool takes two or more images as input and creates a specified number of newly created images in between them as shown in Fig. 1. This generates a sequence of images in which it appears that the selected input images progressively morph into one another. While an image classifier may have a high accuracy in predicting an unmodified original image of a shirt or a dress, a combination of the two of those images mid-morph can have an adversarial effect on ability of the classifier to make that prediction. Therefore, choosing an appropriate neural network and dataset is necessary in order to determine whether or not prediction models are susceptible to uncertainty. In our first tests, we used prediction models generated from the Keras API with TensorFlow to create fully connected artificial neural networks (ANN) to classify clothing elements from their images in the MNIST Fashion dataset, which is a collection of grayscale 28x28 pixel images of articles of clothing inspired by the Modified National Institute of Standards and Technology database which uses images of handwritten numeric digits. [2] The remaining task of obscuring target variables would be performed using the autoimagemorph python tool [3] which automatically creates a morphing sequence between two or more select images using Delaunay triangulation, projective/affine transformation, application of projections through matrices, masking, and alpha blending [4].

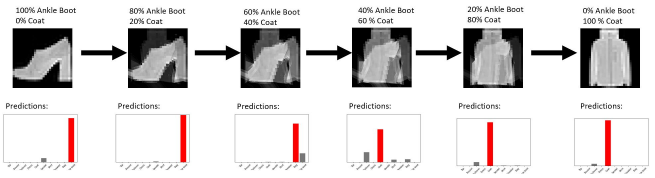

Fig. 1. The path of the evaluations in the horizontal direction through the images of an ankle boot morphing into a coat.

A similar approach in testing whether these machine learning models are prone to error can be made using interpolations created with select pairs of datapoints. The Iris dataset [5] is a compilation of 150 instances of iris plants that are categorized according to whether they belong to the setosa, versicolor, or virginica classes. [6] These individual specimens are measured by their sepal length, sepal width, petal length, and petal width. Applying combinations of these attributes on 2-dimensional scatter plots can lead to identifying characteristics unique to

Supported in part by PSC-CUNY Research Award 63037-00 51.

each class. For example, assessing a plot measuring the petal width along the x-axis and petal length along the y-axis reveals the trend that the petals of virginica flowers are typically long and wide, the petals of setosa plants are thin and narrow, and the petals of versicolor plants range between the other two. This is in contrast with the sepal plot, which has a greater sense of inconsistency between classes, except that the sepal width of the setosa is typically greater than the other classes. Randomly selecting a single flower from each of the three classes, and processing each of its identifying attributes in pairs by interpolation can be used as a technique in which predictions can be made on new data that has been generated from the original dataset, and can test whether the machine learning models are accurate [7].

II. Methods

A. Image Morphing

Image morphing is a technique in which these vulnerabilities can be demonstrated by collecting the accuracy results of image predictions using 5 groups of 2 sets of 3 randomly selected articles of clothing from 3 distinct classes from the 10 available within the Fashion-MNIST dataset, as shown in Fig. 2. This first set of clothes is used to produce a second set using different articles belonging to the same classes as the first set. Each permutation of pairs of clothing from these sets of three are used as input for the autoimagemorph application, and produces a collection of 21 morph sequences between pairs of items midsequence into the third item within the set. These morph sequences consist of 21 images of the transition between the selected combination into the third item. The first paired combination consists of 100 percent of the first image and 0 percent of the second image morphing into the third image. Then, the compositional allocation between the two images changes in 5 percent increments, until reaching 0 percent of the first image and 100 percent of the second image morphing into the third image within the set.

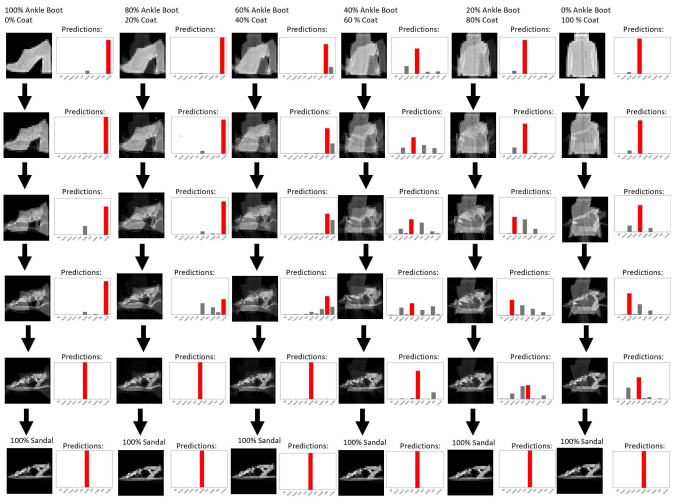

Fig. 2. The path of the evaluations in the vertical direction through the images of combinations of an ankle boot and coat morphing into a sandal.

ANN are used to predict the classes of each image within a morph sequence they belong to. The accuracy rating of the predictions pertaining to the three classes of clothing used within the group are then collected within dataframes, and used to present the information on a 2-dimensional line plot using the plotly express library of the matplotlib python tool [8]. As each line plot utilizes a different combination of two images into a third, there is an expectation that the image predictions would less accurately predict either of the paired articles of clothing as their combinations become evenly split, and more accurately predict the article of clothing when their selected combination favors a greater percentage of one item over another, or near the end of a morph sequence as it morphs completely into the third image within the group.

The slices of each morph sequence are then utilized in creating 3-dimensional landscapes of each class of clothing for each set within the group (A and B). The x-axis of the landscape is determined by the sequential combinations between the first and second article of clothing, and the y-axis is each of the frames within each combination. The z-axis, or height of the 3-dimensional landscape, is the prediction accuracy of the specified class used to label the landscape. These landscapes are generated using the graph objects package within the plotly library.

B. Interpolation

Vulnerabilities can also be understood using interpolation. This involves creating five groups of 3 selected distinct items belonging to each class. The attributes of each item within a group are paired with the same attribute of another item belonging to a different class, which are both used to generate 9 evenly distributed points of data between the selected pair. The collection of each of these datapoints with respect to their attribute can determine a new unclassified item outside the original dataset, which can be predicted using the ANN model trained on the original unmodified dataset.

The permutations of pairs between each of these 3 distinct flowers created in total 362 unique items. In order to make predictions on these items, a prediction model was compiled using a single hidden layer of size 16, categorical cross-entropy loss, and the Adam optimizer. After a train-test-split was performed on the original iris dataset, this model performed with an accuracy of 76 percent on the test set. This model was then used in making predictions on the complete original dataset by removing the target values.

In order to determine whether the pattern of predictions generated were uniquely specific to the iris dataset, the interpolation method and separate machine learning models were also trained to make predictions using both the Measurements on Three Hawk Species dataset from the Stat2Data R package [9] as well as the Wheat seed prediction dataset [10], which similarly collects data belonging to three distinct classes. The selected parameters within the hawks dataset were the wing length, tail length, weight, and hallux. The selected parameters within the seeds dataset were the kernel length, kernel width, kernel groove length, and asymmetry coefficient. In order to

maintain consistency with the size of the iris dataset, 50 items from each of the three classes in each dataset were randomly selected prior to graphing the data.

A comparison between the accuracy of this model using the original iris dataset and the accuracy of the model using predictions on the interpolated datapoints were illustrated by creating ternary plots using the plotly express library. Each vertex of the triangular ternary plot represents 100 percent certainty among the classes of flowers within the iris dataset. Such plots are very useful in illustrating the composition mix of metal alloys and other situations where three variables are plotted on a plane [11]. The advantage of these ternary plots over 3-dimensional plots, is that any view of a 3-dimensional plot projects higher level points onto lower level points, while each point in the ternary plot is a unique and unambiguous combination of the three variables. The diameter and color of each point vary according to the prediction made using the machine learning model.

III. Results

A. Image Morphing

The visual 2-dimensional and 3-dimensional representations of image predictions between each of the 5 groups, while similar, present some discerning characteristics. Group 1A.3 in Fig. 3, which presents all combinations of a coat and ankle boot morphing into a sandal, shows typical results given the methodology. The first frame of the first morph sequence is accurately predicted as an ankle boot, which by the 21st frame is morphed completely and accurately predicted as a sandal. Each successive sequence then begins with a less accurate prediction for the ankle boot, and by the 220th frame begins with a more accurate prediction for the coat. This pattern reversal in accuracy predictions as the composition shifts from one image within the pair to the other while being morphed into the third image is normal and expected.

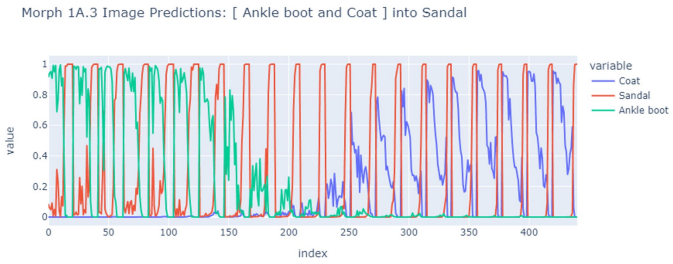

Fig. 3. Group 1A predictions shown on line plot with typical characteristics.

However, other groups have shown to demonstrate unorthodox results. For example, the line plot representations of Groups 3A.2 to 3A.3 in Fig. 4 are of the same images of a shirt, a pullover, and a bag. However, each group uses a different permutation of two of the images morphing into the third image. The graph of 3A.3 shows typical fluctuations between the shirt and pullover as they gradually morph into a complete image of the bag, which is distinctly predicted at the end of each of the 21 cycles. Group 3A.2, instead using combinations of the bag and pullover morphing into a shirt, is not so clearly defined. The shirt is never so accurately predicted as was the bag in the previous figure. The accuracy ratings between the first two images are also more pronounced, as the predictions for the bag are almost entirely eliminated in favor of the shirt and pullover even while the composition of the frames between the bag and pullover are roughly 50 percent.

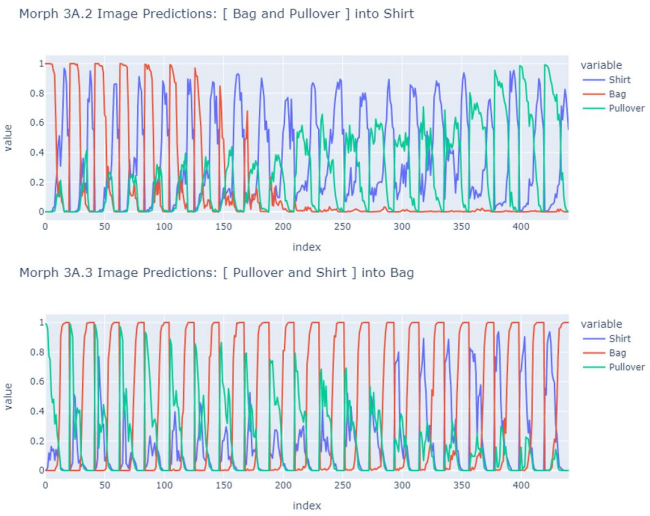

Fig. 4. Group 3A predictions shown on line plots with different ordered permutations.

The allocation of image predictions was also utilized in creating 3-dimensional landscapes in Fig. 5, which also revealed similar patterns, now with the predictions for each unique article of clothing separated onto an individual graph, with each slice overlapping with the end of each cycle. Isolating the prediction accuracy of each article of clothing within a group separately on a surface plot allows a better visualization of ambiguity, according to whether the surface is smooth or rough. The prediction landscapes of the shirt belonging to Groups 3A.2 and 3B.2, and the bag belonging to Group 3B.2 fluctuate rapidly with rough edges. The prediction landscapes of the bag belonging to Group 3A.2, and the pullover belonging to Groups 3A.2 and 3B.2 demonstrate less fluctuations and an overall smoother transition between less accurate and more accurate predictions.

B. Interpolation

The interpolation predictions for Group E illustrates the random selection of three points belonging to each class within the iris dataset, using pairs of features to determine the x and y axes within a scatter plot. The interpolation and predictions made using these features fill a triangular region in Fig. 6 between the points with distinct patterns indicating that the classification criteria does not allow for overlap. These gradient patterns found in each scatter plot reveal clear boundary lines that are apparent, and that predictions shift when crossing a threshold given certain combinations of features.

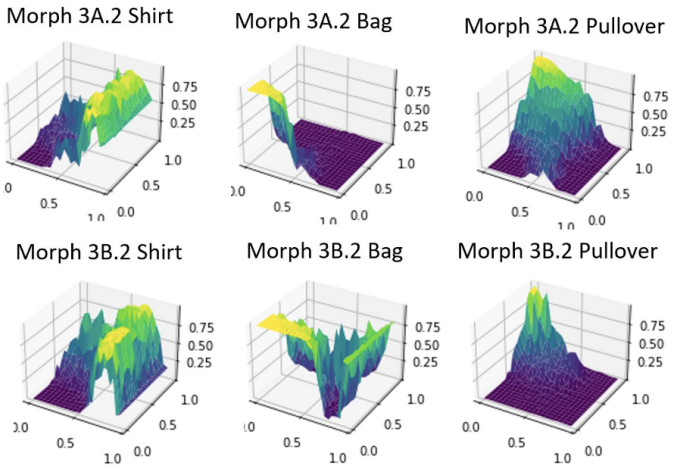

Fig. 5. Group 3A and 3B predictions shown on 3-dimensional landscapes.

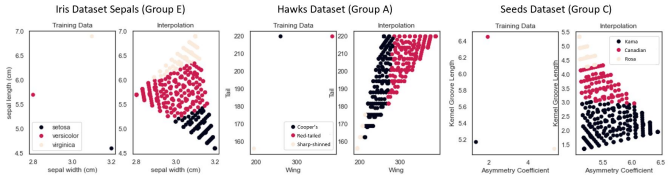

Fig. 6. Iris, Hawks, and Seeds interpolated data shown on scatter plots.

The accuracy percentage values of the model's predictions using newly interpolated data can also be interpreted using slices on a line plot. The line plots for each group reveal a trend in which favorable predictions for setosa generate the highest accuracy ratings among the three classes throughout the interpolated field. While the model projects the highest accuracy ratings when predicting the setosa class in certain slices, it conversely predicts against the setosa class which has the lowest accuracy ratings in other portions of the line plot. As a result, the model is often undecided between predicting either the versicolor or virginica classes throughout each of the 3 cycles demonstrated by the line plot in Fig. 7

These graphical depictions of predictions using interpolated data made with scatter plots and line plots illustrate visual patterns when making comparisons among the three groups. Visualizing these predictions using ternary plots in Fig. 8 also determine a distinguishing characteristic among each of the groups. The setosa is the most favorably predicted among the classes, and therefore has points reaching near 100 percent accuracy. This is in stark contrast with predictions made for virginica or versicolor, which are unable to reach the vertex of either region and follow a curved pattern between the two groups. The ternary plot with predictions made using the original iris dataset had a wide gap dividing the setosa from the versicolor and virginica classes. The ternary plots using predictions from the interpolated groups are densely concentrated throughout the plot, and especially at the bend of the curve nearest the versicolor vertex.

Applying the same method of model training, random

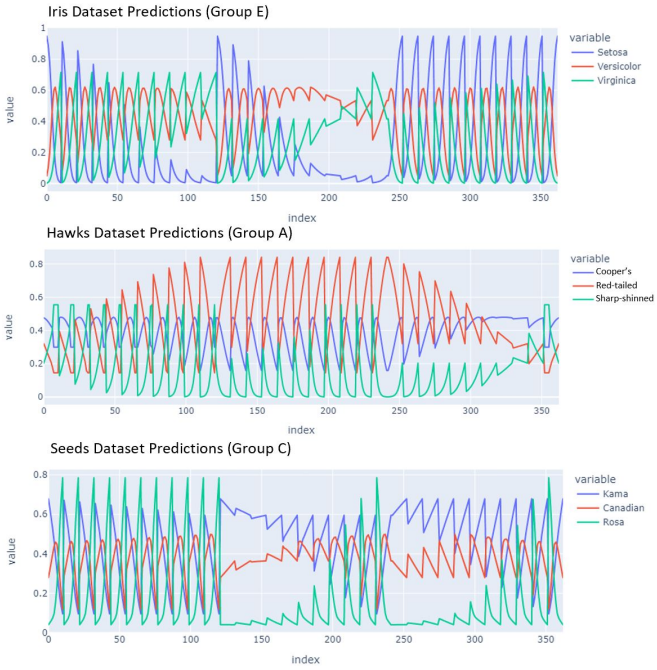

Fig. 7. Iris, Hawks, and Seeds interpolated data predictions shown on line plots throughout each of the 3 cycles.

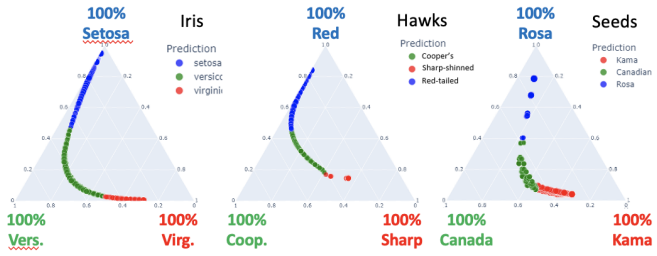

Fig. 8. Iris, Hawks, and Seeds interpolated data predictions shown on ternary plots.

selection, and interpolation using the hawks and seeds dataset reveal that the resulting patterns from making predictions using these techniques is not isolated to the iris dataset. Interpolated points scatter across a triangular figure, and the predictions made are in noticeably separate and parallel regions in the scatter plots belonging to each dataset. The line plots also similarly favor predictions for and against one element of the group, while the other two are more contested. The ternary plots also reveal a similarly curved distribution of points.

IV. Discussion

A. Context

We first put our work in a general context and then discuss specifics about it. Previous studies have visualized the loss function in the training of a neural network, that is, how the energy surface depends on the connection weights being determined in back propagation [12] [13] [14]. In our case, we are interested in something different, the energy surface of

the network when the weights are fixed and we compare the classifications for different input values. However, what has been learned about the energy surface in studying backpropagation is also instructive for our problem. LeCun, Bengio, and Hinton [15] write that "the landscape is packed with a combinatorially large number of saddle points where the gradient is zero, and the surface curves up in most dimensions and curves down in the remainder. The analysis seems to show that saddle points with only a few downward curving directions are present in very large numbers, but almost all of them have very similar values of the objective function." We are asking, if in our case, the landscape also has similar properties. If so, that would mean from an input, a neural network could bounce around through many saddle points in a high dimensional space, and find similarly good minima at very different locations corresponding to very different classifications. We studied only small, local, neighborhood changes in the inputs rather than the global structure directly. We are asking, if those small continuous changes in the inputs do, or do not, lead to small continuous changes in the output classification. That is, are "adverserial examples" or "hallucinations" reported in AI systems atypical of trained neural networks, or are they typical properties of a high dimensional landscape with many saddle points. By varying the input in incremental amounts, we found that sometimes there are rough landscapes with unexpected classifications (Figure 5) and sometimes the transitions between classifications are well defined (Figure 8). Other valuable approaches to understanding classification results include "explainable AI" [16] which is successful at visualizing the importance of different features in the classification. We don't know if those approaches may also be susceptible to the roughness in the classification landscape.

B. Details

The process of generating predictions using morph sequences, while successful for these selected groups of images, were sometimes prone to error. It appeared that while the high granularity and low pixelation of the Fashion-MNIST dataset images would be beneficial to the ANN model given the small field to assess, it served problematic for the autoimagemorph tool using its default settings. From its documentation, the tool is more capable of generating a morph sequence using images with a higher pixel count and greater level of detail. It became necessary to adjust the *featuregridsize* option to 3, which allows compatibility with a greater number of MNIST images. Another issue involved the amount of time required to generate each morph sequence for each permutation within each group and set, which could require the program to run for several hours, even if TensorFlow is running using the GPU. Memory limitations along with error occurrences led to intermittently saving data in .pkl format after each morph sequence as safe practice.

The prediction results generated varied according to the specific images and permutation choice utilized for each group. Choosing different combinations of the same images within a group dramatically affected whether or not the prediction model could accurately predict the image during each morph sequence. There often occurred sharp changes among the prediction accuracies within each cycle, even though the cycles themselves were partitioned in 5 percent margins in the image collection.

The project also at an early stage involved the creation of 3-dimensional landscapes using a random walk function with a vector field in order to visualize the effect that magnitudes of force, fixed weights, and slope can have on a randomly moving object across a landscape. These movements were recorded across landscapes of different shapes, including symmetrical bowls and randomly generated fractal landscapes as the LOG of the average distance vs. the LOG of the number of steps. However, we were unable to utilize this tool with regard to image predictions using the ANN model in order to demonstrate ambiguity within landscapes with a smooth or rough surface.

The similarity amongst certain articles of clothing may also contribute to the level of uncertainty that can occur using the ANN prediction model. There exists a strong resemblance between many of the grayscale images of clothing within the Fashion-MNIST dataset, and certain types may appear to be more alike than others.

The interpolated data between randomly selected points within the iris dataset can also distribute across wide margins, and creates many unclassified datapoints using only a few which belong to the original iris dataset. This may have an effect on predictions generated according to the test set chosen when training the model, the accuracy of the model compiled, or the points that were selected for interpolation. The strong gradient that exists between predicted classes within the interpolated field, when compared with often overlapping clusters that exist in the sepal and petal scatter plots of the original iris dataset, demonstrate the strict level of specificity the model must adhere to when making predictions with so few points available from the training set.

Another approach to revealing the vulnerability of neural networks may involve prediction models beyond image classification. For example, the vulnerability of a neural network classification tool using audio or textual data may have a different potential for mischaracterization than would the ANN image prediction model using the Fashion-MNIST dataset. HuggingFace is one data platform which offers an expansive library of datasets and training models in order to classify audio, visual, and textual data. Some of these models include the BLOOM multilingual data model [17], which uses 59 languages and 1.6TB of training data in order to predict which text prompts would be input using textual data. Speechbrain is an Automatic Speech Recognition toolkit utilizing PyTorch models in order to transcribe audio to text format [18]. The amount of information available within these expansive datasets may serve to increase the accuracy of predictions, but may also lose the precision afforded by more limited datasets such as Fashion-MNIST, and may have an ever greater propensity for error.

V. Conclusion

The use of the ANN model in order to make predictions on information outside of the original dataset can exhibit the vulnerabilities of neural networks by graphing the results using various approaches to data visualization. Deep learning models are exceptional within the limited scope of the framework they are designed with, and the boundaries of the collected data elements they are trained with. Testing the interface between machine learning models across different collections of data can exhibit parallel issues and deviations from typical expectations. Exploration of these susceptibilities through different means of data manipulation can signal where these barriers exist and allow for future refinement.

References

[1] L. Engstrom, D. Tsipras, L. Schmidt, & A. A. Madry. (2017). Rotation and a Translation Suffice: Fooling CNNs with Simple Transformations. *ArXiv* https://openreview.net/forum?id=BJfvknCqFQ

[2] H. Xiao, K. Rasul, & R. Vollgraf. (2017). Fashion-MNIST: a Novel Image Dataset for Benchmarking Machine Learning Algorithms. (2017,8,28)

[3] A. Jankovics. (2020). autoimagemorph. *GitHub Repository*. https://github.com/jankovicsandras/autoimagemorph

[4] D. Dowd. (2021). Python-Image-Morpher. *GitHub Repository*. https://github.com/ddowd97/Python-Image-Morpher

[5] D. Dua, & C. Graff. (2017). UCI Machine Learning Repository. (University of California, Irvine, School of Information,2017), http://archive.ics.uci.edu/ml

[6] R. Fisher. (1936). The use of multiple measurements in taxonomic problems. *Annual Eugenics*, 7:179–188

[7] W. Powers, L. Shi, L. Liebovitch. (2023). VulNeuralNetworks. *GitHub Repository*. https://github.com/wpqc21/VulNeuralNetworks

[8] P. Inc. (2015). Collaborative data science. (Plotly Technologies Inc), https://plot.ly

[9] A. Cannon, G. Cobb, B. Hartlaub, J. Legler, R. Lock, T. Moore, A. Rossman, & J. Witmer. (2021). *GitHub Repository*. https://github.com/statmanrobin/Stat2Data

[10] M. Charytanowicz, J. Niewczas, P. Kulczycki, P. A. Kowalski, S. Lukasik, & S. Zak. (2010). 'A Complete Gradient Clustering Algorithm for Features Analysis of X-ray Images', in: Information Technologies in Biomedicine, Ewa Pietka, Jacek Kawa (eds.), Springer-Verlag, Berlin-Heidelberg, 2010, pp. 15-24.

[11] Ternary Plot, Wikipedia. (2022). https://en.wikipedia.org/wiki/Ternary

[12] I. J. Goodfellow, O. Vinyals, & A. M. Sale. (2015). Qualitatively Characterizing Neural Network Optimization Problems. arXiv:1412.6544v6, https://doi.org/10.48550/arXiv.1412.6544

[13] D. J. Im, M. Tao, & K. Branson. (2017). An Empirical Analysis of the Optimization of Deep Network Loss Surfaces. arXiv:1612.04010v4, https://doi.org/10.48550/arXiv.1612.04010

[14] H. Li, Z. Xu, G. Taylor, C. Studer, & T. Goldstein. (2018). Visualizing the Loss Landscape of Neural Nets. 32nd Conference on Neural Information Processing Systems (NeurIPS 2018), Montreal, Canada

[15] Y. LeCun, Y. Bengio, & G Hinton. (2015). Deep Learning. Nature 521:436-444 (28 May 2015)

[16] IBM. (2023). Explainable AI (XAI). https://www.ibm.com/watson/explainable-ai

[17] M. Al. (2022). BigScience Large Open-science Open-access Multilingual Language Model. *BigScience*. https://huggingface.co/bigscience/bloom

[18] M. Ravanelli, T. Parcollet, P. Plantinga, A. Rouhe, S. Cornell, L. Lugosch, C. Subakan, N. Dawalatabad, A. Heba, J. Zhong, J. Chou, S. Yeh, S. Fu, C. Liao, E. Rastorgueva, F. Grondin, W. Aris, H. Na., Y. Gao, R. Mori. & Y. Bengio. (2021). SpeechBrain: A General-Purpose Speech Toolkit. arXiv:2106.04624