RESCIENCE C

Replication / ML Reproducibility Challenge 2022

[Re] Reproducibility Study: Label-Free Explainability for Unsupervised Models

Sławomir Garcarz[1,2], Andreas Giorkatzi[1,2], Ana Ivășchescu[1,2], and Theodora-Mara Pîslar[1,2]
[1]University of Amsterdam, Amsterdam, Netherlands – [2]All authors contributed equally

Edited by
Koustuv Sinha,
Maurits Bleeker,
Samarth Bhargav

Received
04 February 2023

Published
20 July 2023

DOI
10.5281/zenodo.8173688

Reproducibility Summary

Scope of Reproducibility – This work studies the reproducibility of the paper ”Label-Free Explainability for Unsupervised Models” by Crabbé and van der Schaar to validate their main claims. These state that: (1) their extension of linear feature importance methods to the label-free setting is able to extract the key attributes of the data, (2) the adaptation of example importance methods to the unsupervised setting succeeds in highlighting the most influential examples, (3) different pretext tasks do not produce interchangeable representations and (4) the interpretability of saliency maps is uncorrelated to the level of disentanglement between individual latent units.

Methodology – The authors provided the code written in PyTorch needed to reproduce all the experiments. Some parts of the code were modified in order to extend the original experiments. The total computation time required to perform the original and extended versions of the experiments is 103 GPU hours. Most of the experiments were performed on NVIDIA TITAN RTX GPU.

Results – The plots supporting the label-free feature and example importance match the ones from the paper, except for the label-free feature importance experiment for CIFAR-10. Similarly, the Pearson correlation results were successfully reproduced. Due to the nature of the autoencoders used for evaluation, we could not obtain the exact numerical results. However, we visually and numerically compare the trends, and in most cases, we observe that our results are similar to the ones in the paper.

What was easy – The paper comes with publicly available code and an extensive appendix containing the setup for all experiments. With that, we were able to reproduce all the experiments with only minor changes to the code.

What was difficult – Despite the fact that running the original experiments was straightforward, extending them to new datasets or models was more challenging. Moreover, some of the experiments are more resource-consuming and require more time to run.

Correspondence should be addressed to Sławomir Garcarz (slawek.garcarz@gmail.com)
The authors have declared that no competing interests exist.
Code is available at https://github.com/ayiork/Label-Free-XAI. – SWH swh:1:dir:e76ce9ca64bef8b8ab34ef48336017ade33d40b9.
Open peer review is available at https://openreview.net/forum?id=sF_vYZSxSV.

Communication with original authors – We contacted the authors to resolve our concerns regarding some of the results. They were very helpful and answered all of our questions. Moreover, they provided us with a pre-trained SimCLR model. We used this model to validate our results.

1 Introduction

Recent developments of deep neural networks have resulted in black-box models whose transparency plays a key role in explaining decisions made in fields such as healthcare, finance, or justice. The demand for methods that interpret entangled models grows with their complexity, especially now when unsupervised learning takes over in the form of autoencoders ([1], [2], [3]). At this moment, there are ways for explaining models in a supervised setting, such as feature and example importance analysis. Feature importance analysis can highlight the contribution a feature has towards a decision made by a model ([4], [5], [6]). Similarly, example importance analysis highlights the major data points that influence the training ([7], [8], [9]). These methods are researched in supervised environments, yet in an unsupervised setting, the problem remains unsolved.

To address this issue, Crabbé and van der Schaar introduce ”Label-Free Explainability for Unsupervised Models” [10]. They extend previous explainability methods introduced in the context of supervised learning such as feature and example importance to the unsupervised and self-supervised regimes by defining a wrapper function to these methods. In addition, the authors claim that representation-based example importance introduced in supervised contexts can be easily extended to a label-free setting by replacing the supervised representation map with an unsupervised one. To show that their approach is effective and reliable the authors performed several experiments covering both unsupervised and self-supervised approaches.

In this research, we aim to evaluate the reproducibility of the paper by replicating their experiments, and by investigating further with different settings to reinforce their claims.

2 Scope of reproducibility

The authors developed a new framework called label-free explainability, which allows to extend linear feature importance and example importance methods to the unsupervised setting. They proved the following properties of this framework: completeness and invariance with respect to latent symmetries. Moreover, the authors experiment with pretext tasks as a use case for label-free explainability. Lastly, they evaluated qualitatively and quantitatively whether the generative factor associated with each latent unit is identifiable by using the saliency map of its latent unit with disentangled VAEs. In this study, we will verify the following claims of the original paper:

- **Claim 1**: Defining an auxiliary scalar function as a wrapper around the label-free black-box permits to compute feature importance by utilizing attributions such as Gradient Shap [4], Integrated Gradients [6], and Saliency [5].
- **Claim 2**: Label-free extension for example importance allows to identify salient training examples that are related to test examples one wants to explain by utilizing loss and representation-based methods.
- **Claim 3**: Given the notion of label-free explainability, different pretext tasks do not produce interchangeable representations in self-supervised learning.

- **Claim 4**: The ability to understand the significance of certain features in a model, represented by saliency maps, is not dependent on how well the latent units in the model are separated or disentangled from one another.

The rest of this report is organized in the following way: in section 3 we present the methodology by introducing the models, datasets, and the experimental setup used in our study, section 4 contains the results of the performed experiments, and in section 5 we discuss our experience and conclude on the results.

3 Methodology

The PyTorch implementation for reproducing the experiments is provided by the authors. By using their experimental settings, we were able to reproduce all the results from the paper. To conduct further experiments for the generalizability of the paper, we extended the provided code. We ran the experiments on Nvidia TITAN RTX GPU.

3.1 Model descriptions

Feature Importance — The extension to label-free feature importance methods proposed by the authors required defining an auxiliary scalar function g_x as a wrapper around black box function f. That function is then fed to any importance method a_i:

$$b_i\left(f, x\right) = a_i\left(g_x, x\right) \tag{1}$$

$$g_x : X \to \mathbb{R} \quad \text{such that for all } \widetilde{x} \in X :$$

$$g_x\left(\widetilde{x}\right) = \left\langle f\left(x\right), f\left(\widetilde{x}\right)\right\rangle_H . \tag{2}$$

Moreover, the authors proposed to weight importance scores in label-free environments by using activation scores from the previous layer. For a more detailed explanation refer to section 2.2 of the original paper [10].

Example Importance — In terms of the example importance, the authors split the methods into two families: loss-based and representation-based. For the former, in a supervised setting the importance score is assigned to each training example according to the influence on the loss when they are removed. In terms of the label-free setting, we want to train our model using a label-free loss $L = X \times \Theta \to H$. This is usually not enough as the authors explain, due to the fact that the importance scores that are computed can include irrelevant parts of the black-box. To solve that problem the authors split the parameter space $\Theta = \Theta_r \times \Theta_{irr}$, where *r* corresponds to the relevant parameters and *irr* to irrelevant parameters. This leads to the following equation:

$$c^n\left(f_{\theta_r}, x\right) = \delta^n_{\theta_r} L\left(x, \theta_*\right). \tag{3}$$

Using this equation we assume that the loss depends only on a single input example which is not true for contrastive losses [11]. Due to the fact that there is no apparent extension of loss-based example importance for settings with contrastive losses, the authors proposed a label-free modification of the representation-based example importance method. Representation-based methods attribute to each training example a score by analyzing the example's latent representation. For experiments with the CIFAR-10 dataset, we used the SimCLR model. For the ECG5000 dataset, a Recurrent Autoencoder with an encoder, two LSTMs, and a final linear layer was used. Moreover, for MNIST we used an autoencoder for each pretext task. Lastly, for disentangled VAEs we used β-VAE [12] and TC-VAE [13]. Descriptions of the models and hyperparameters used can be found in the Appendix A.1

3.2 Datasets

In order to check the consistency of the feature and example importance methods, the authors fitted the models described above on 3 datasets: MNIST [14], ECG5000 [15] and CIFAR-10 [16]. They also challenged the interpretability of disentangled representations by training the β-VAE and TC-VAE on both MNIST and dSprites [17]. We extended the analysis by also using the Fashion MNIST dataset [18] to perform consistency and pretext tests. An overview of the most important information on the datasets can be seen in Table 1 and more details about them can be found in Appendix A.2.

Datasets and links	Samples		Classes	Description
	Train	Test		
MNIST	60 000	10 000	10	Grayscale images of 0-9 digits.
CIFAR-10	50 000	10 000	10	RGB images of objects.
Fashion MNIST	60 000	10 000	10	Grayscale images of clothing items.
dSprites	737 280		-	Synthetic images of 2D shapes.
ECG5000	5 000		2	Time series of heart rates.

Table 1. Overview of the datasets used for validating the proposed methods in the label-free setting.

3.3 Hyperparameters

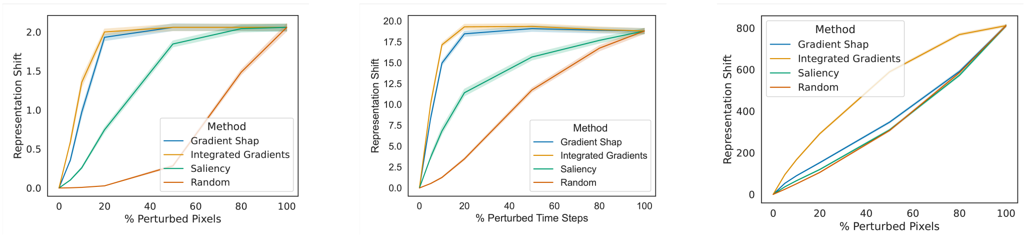
The authors of the original paper provided hyperparameters settings for all the experiments they performed. To match the results of the experiments we have decided to follow the setup used by them. For additional experiments, we used the same setup of hyperparameters to ensure the comparability of the results.

3.4 Experimental setup and code

The code developed by Crabbé and van der Schaar is available online on GitHub. It mainly reflects the setup described in Appendix B of their paper, and therefore constitutes the base we use for verifying their claims. We use the same metrics as in the original paper to compute the feature and example importance: latent shift for feature importance and similarity rate for example importance.
In addition, to support **claim 1** we experiment with a pixel-flipping approach for masking CIFAR-10 (method referenced by the authors in the paper), instead of blurring as masking. We visually analyze the results and observe the trends in the plots. For addressing **claim 1**, **claim 3** and **claim 4** we experiment with a different predefined attribution method from PyTorch, Integrated Gradients, in the context of the pretext and VAE experiments where Gradient Shap is initially used on the MNIST dataset. We use the Pearson correlation and the plots produced to quantitatively and qualitatively analyze the results and compare them with those from the original paper. To further challenge **claim 1** and **claim 2**, we evaluate the consistency checks on CIFAR-10 using the architecture of DenseNet121 (instead of ResNet 18 or 34) for the encoder, which is considered a more powerful network since all the layers are directly connected to each other. Lastly, we challenge **claim 3** on the Fashion MNIST dataset, compared to the MNIST dataset used in the paper for the experiments. The code used to produce the results for this paper is available on this GitHub repository.

3.5 Computational requirements

Due to limited GPU resources, we executed some of our experiments on Intel Core i7 - 10750H CPU, however, most of our experiments were performed on a cluster with NVIDIA TITAN RTX GPU. Table 2 illustrates the required times for each experiment.



(a) MNIST

(b) ECG5000

(c) CIFAR-10

Figure 1. Consistency check for label-free feature importance.

	Feature Consistency	Example Consistency	Pretext	DVAE
MNIST	1	4	4	11
ECG5000	2	23	-	-
Cifar10	2.5	2.5	-	-
dSprites	-	-	-	32
F-MNIST	1	4	4	-

Table 2. GPU hours for reproducing different experiments. ECG5000 features are CPU hours.

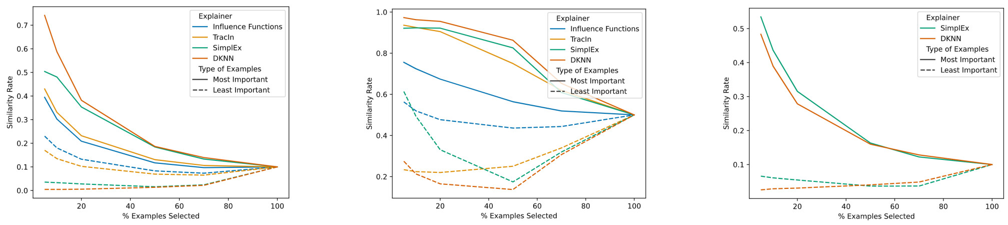
4 Results

In this section, we verify the main claims stated by reproducing the experiments from the paper, as well as challenging them to new experimental setups, such as using different datasets, models, or methods. Overall, the results from the experiments reproduced follow the same trends as the ones produced in the original paper.

4.1 Results reproducing original paper

Feature Consistency Checks — To verify **claim 1**, we reproduced feature importance experiments from section 4.1 of the original paper. In Figure 1 we present our results. Results for MNIST and ECG5000 datasets matched those from the original paper with only minor differences, however, for the CIFAR-10 dataset we found discrepancies with the original results. The authors claimed that the latent shift increases quickly when we perturb the most important pixels and decelerates when we perturb less relevant pixels. This is true for MNIST and ECG5000 datasets, but according to our results for CIFAR-10, this conclusion only holds for the Integrated Gradients method. For Gradient Shap and Saliency, we observe the opposite. The other claim is that perturbing pixels based on the importance score yields a bigger change in latent space than perturbing random pixels. Again for MNIST and ECG5000, we have been able to confirm that. For CIFAR-10 this is true only for the Integrated Gradients method. To conclude, **claim 1** made by the authors was partially confirmed by our experiments.

Example Consistency Checks — To check whether **claim 2** holds or not, the authors conducted experiments on three different datasets. The setup for these checks is to sample 1000 training examples and compute the importance score of each in predicting the latent representation of the test images. For computing the score, adaptations to the label-free setting of several methods (both loss and representation based) were used: Influence Functions [19][20], TracIn [21], SimplEx [22], DKNN [23]. In order to check the effectiveness of the methods they selected the M most important training examples and computed the similarity rates between them. They did the same for the M least important examples and expected the similarity between the least important samples to be



(a) MNIST **(b)** ECG5000 **(c)** CIFAR-10

Figure 2. Consistency check for label-free example importance.

lower than between the most important ones. Figure 2 shows the results we obtained after reproducing the original experiments on MNIST, ECG5000 and CIFAR-10 datasets. The plots display the distribution of similarity rates for different values for M and example importance methods. For MNIST and ECG5000 our results match the ones from the paper, but for CIFAR-10, even though the trend is similar, the scale of the similarity rate differs, as for us it peaks at 0.5, while in the original paper, the highest value is around 0.17. All in all, our results validate **claim 2** which states that label-free importance scores allow us to determine the training examples that explain the test ones best.

Comparing the Representations Learned with Different Pretext Tasks — To support **claim 1** and **claim 2**, the authors compared different pretext tasks, such as denoising, reconstruction, and inpainting, qualitatively and quantitatively. Through this analysis, they support **claim 3**. We reproduce the experiments in the same way, by using the MNIST dataset and averaging the Pearson correlation coefficients of five runs of different autoencoders, as stated in the original paper in Appendix C.2. For the **quantitative analysis**, we focus on interpreting and comparing the Pearson correlation obtained for feature and example importance shown in Table 3a and Table 3b, respectively, with the results from the original paper. In our case, the Pearson correlation coefficients for saliency maps range from .32 to .43 corresponding to the moderate positive correlations also obtained in the original paper. Furthermore, we have Pearson correlation coefficients for example importance ranging from .05 to .13 corresponding to the weak correlations obtained in the original paper.

Pear.	Rec.	Den.	Inp.
Den.	0.38 ± 0.02		
Inp.	0.33 ± 0.05	0.32 ± 0.02	
Clas.	0.43 ± 0.02	0.4 ± 0.01	0.35 ± 0.04

(a) Pearson correlation for saliency maps (avg +/- std).

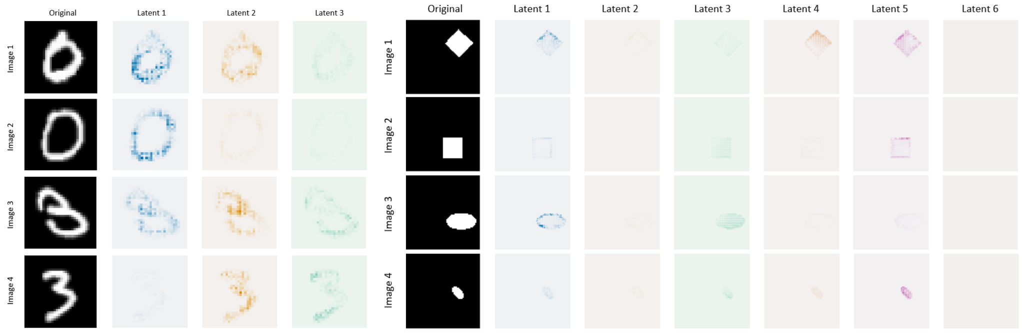
Pear.	Rec.	Den.	Inp.
Den.	0.08 ± 0.04		
Inp.	0.13 ± 0.05	0.09 ± 0.01	
Clas.	0.07 ± 0.02	0.05 ± 0.02	0.08 ± 0.02

(b) Pearson correlation for example imp. (avg +/- std).

Table 3. Pretext experiment results.

In terms of the **qualitative analysis**, we plot the most important examples and saliency maps for different encoders to support **claim 2** and **claim 3**. Visually, one can interpret the saliency maps as being different from one pretext task to another. Moreover, the top examples produced by different pretext tasks are hardly similar. Both conclusions reinforce the previous results from the quantitative analysis and compare positively to the results from the paper. Examples of these visualizations are shown in Appendix A.3. In conclusion, the representations of different pretext tasks are not interchangeable.

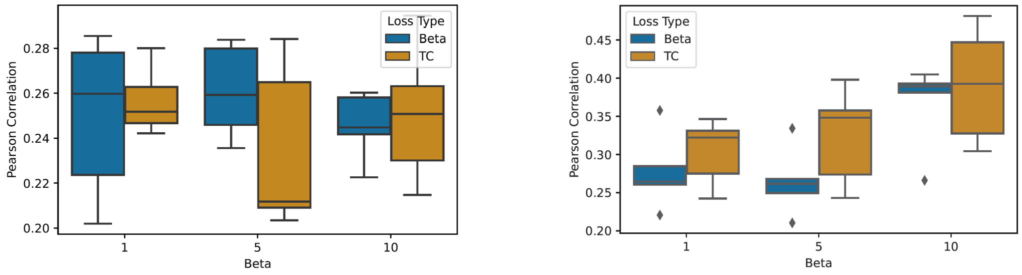
Challenging Assumptions with Disentangled VAEs — To investigate the paper's **claim 4**, we trained two disentangled VAEs, β-VAE and TC-VAE on MNIST and dSprites datasets. For the **qualitative Analysis**, we have validated the three paper's conclusions. Saliency maps of four test images are shown in Figure 3. Firstly, the latent unit can be sensitive and insensitive to similar images (Latent Unit 2 of the MNIST VAE is sensitive to image 1 but not to Image 2). Secondly, the focus of a latent unit can be completely different between similar images (Latent unit 5 of dSprites VAE in Image 1 focuses on the interior of the rectangle, and in Image 2 it focuses on the border of the rectangle). Lastly, some latent units focus on the same part of the image (Image 3 of MNIST). In terms of the **quantitative Analysis**, box plots observed in Figure 4 have some differences from the ones in the paper. That could be normal given how unstable and hard to seed VAEs are. Regarding the dSprites dataset, the plot shows a moderate increase in the Pearson correlation coefficients with β. Concerning the MNIST dataset, we observed a slight decrease in Pearson correlation with β. That leads us to the conclusion that increasing β does not suggest that latent units are paying attention to a specific part of the image, which is the same conclusion as in the paper.



(a) MNIST

(b) dSprites

Figure 3. Saliency maps for each unit of the disentangled VAEs.



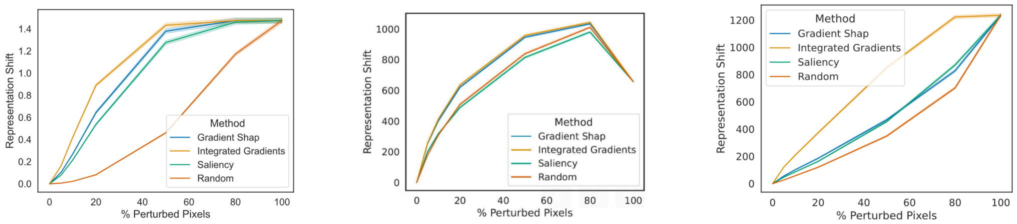
(a) MNIST

(b) dSprites

Figure 4. Pearson correlation between saliency maps for different values of β.

4.2 Results beyond original paper

Fashion MNIST – In order to test the generalizability of the methods, we run the feature and example importance experiments on a different dataset, Fashion MNIST. We also compared both quantitatively and qualitatively the representations learned for several pretext tasks and questioned how different the representations are from each other by computing their Pearson correlation coefficient and plotting the most important samples and the saliency maps. The feature importance graph can be seen in Figure 5a and it shows that the representation shift increases abruptly when we perturb the most important pixels, following the same trend as for the other datasets. The example importance graphs along with the correlation coefficients and plots can be seen in detail in Appendix A.4. The results for Fashion MNIST reinforce the idea that encoders trained



(a) Fashion MNIST. (b) CIFAR-10: black pixel masks. (c) CIFAR-10: DenseNet encoder.

Figure 5. Consistency check for label-free feature importance on additional experiments.

on different pretext tasks pay attention to distinct features in the image and that the learned representations are not interchangeable.

Additional experiments for CIFAR-10 dataset — In section 4.1 evaluating **claim 1**, we mentioned that we found inconsistencies between the original and our results for the CIFAR-10 dataset. We decided to investigate further this experiment. As a first step, we plotted the masked images to confirm that the quantitative analysis is correct. These results can be found in Figure 6. According to the results, the Integrated Gradients method worked best because most of the pixels covered are on the object, while for the other methods, a lot of pixels identified as salient are in the background. This observation matches the quantitative results presented in section 4.1. In addition, we experimented with the original pixel-flipping approach proposed by (Montavon et al., 2018)[24] to which the authors referred. This masking method, instead of blurring the most important pixels uses black pixels as a baseline. In Figure 5b we present the results of this experiment. As we can see Integrated Gradients and Gradient Shap methods performed better in this setup, however the relative difference from the Random baseline method is smaller than previously. We have also decided to plot the masked images which again matched quantitative metrics. Those results are presented in Figure 13 available in the appendix.

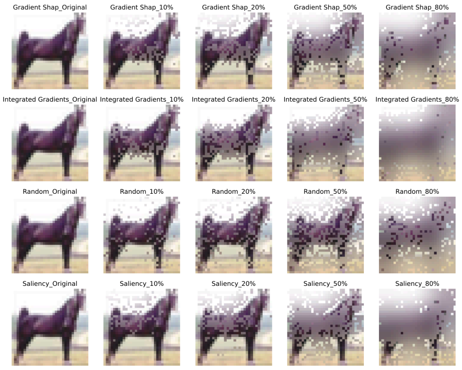


Figure 6. Image from CIFAR-10 with blur mask.

Explainability analysis with different attribution methods — The authors concluded after the feature consistency checks that the *label-free Integrated Gradients outperforms other methods for each model* tested. However, for their pretext and VAE tests, they used label-free Gradient Shap to produce the saliency maps. In this experiment, we want to observe the generalizability of the label-free feature importance in the context of self-supervised learning and disentangled VAEs. We experiment with label-free Integrated Gradients by using it as an attribution method for label-free feature importance. The experimental settings remain the same as described for the pretext and VAE experiment, respectively. The quantitative and qualitative results for each attribution method are shown in Appendix A.6. We obtain the same conclusion about the medium Pearson correlation for the feature importance and the low correlation for the example importance as in the original paper. For the VAE experiment, the Pearson correlation coefficients have higher values compared to the ones obtained using Gradient Shap in the original paper. However, we observe that as β grows, latent units are not paying attention to distinct parts of the image because the Pearson correlation does not decrease at the same time with β. Therefore, we could conclude both the pretext and VAE experiments similarly as in the original experiments, using label-free Integrated Gradients instead of label-free Gradient Shap, enhancing **claim 1**. This also proves the *generalizability* of the label-free feature importance, which is an easy-to-adapt method to practical examples and existing supervised explainability methods.

Evaluating CIFAR-10 experiments using DenseNet — To validate the integrity of **claim 1** and **claim 2** further in terms of the model used, we decided to change the encoder of the SimCLR network from a ResNet18 (or 34) to a DenseNet121. By running the same experiments we obtained identical results as can be seen in Figure 5c. The results can be found in the appendix (Figure 18). The trends are the same as the ones using the ResNet encoder, designating that the conclusion does not depend on the encoder.

5 Discussion

In this study, we carried out multiple experiments to replicate the key findings from the original research. Our reproducibility results lend credence to the original claims, as we were able to largely replicate the original findings. We validated the four claims of the authors, except for some minor discrepancies on CIFAR-10. Regarding these inconsistencies, we decided to contact the authors, and they provided us with the pre-trained model they used to perform experiments in the original paper. With the use of this model, we were able to obtain the same results as the authors, however, we couldn't reproduce them by training the model on our own. Additionally, we asked the authors why they choose to blur pixels instead of changing them to black. They justified it by pointing out that for the CIFAR-10 dataset, some black pixels may be salient, which will result in zero attribution. We confirmed that by looking at the results, however the parameters for Gaussian Blur were handcrafted and might not generalize well to different datasets. Aiming to prove the robustness of the proposed frameworks for a label-free feature and example importance we decided to test them in different settings. Firstly, we used a different dataset, the Fashion MNIST dataset, and we were able to prove that the conclusions still hold. Secondly, by swapping the encoder of SimCLR from ResNet to DenseNet we proved that the results are not dependent on the encoder. Furthermore, we experimented with a different attribution in the practical example of pretext tasks in self-supervised learning and VAE challenge, namely label-free Integrated Gradients instead of Gradient Shap, to support the generability of the label-free feature importance.

5.1 Reflection: What was easy, and what was difficult?

The original paper had all the newly introduced methods and experiments clearly stated and further explained in the appendix with mathematical proofs, detailed architectures of the models, values for hyperparameters and qualitative results. On top of this, having access to the original code implementation made it easy and straightforward to run all the experiments. The datasets were also publicly available. Even though the code was available and most of the reproducibility experiments were done without any modifications, the comments in the code were too sparse; therefore, understanding and extending the code demanded more time than expected. Moreover, the ECG5000 example importance experiments required more than the maximum time that we could use the GPU continuously. Thus, we modularized the code to save intermediate results which we merged together in the end.

5.2 Communication with original authors

We raised questions about some differences in the results, explored explanations for implementation decisions, and then got in touch with the authors for clarification. The authors replied immediately and provided satisfactory answers to most of our questions. However, a few of the answers were not sufficient. For instance, concerning the differences in the results using CIFAR-10 dataset, the authors provided the specific file with the trained parameters that were used for obtaining the results in the original paper. We were able to reproduce the original results using the pretrained model given by the authors. However, we were not able to find what exactly is causing the difference; we believe that they used different hyperparameters than specified in the paper.

References

1. J. Grill et al. "Bootstrap Your Own Latent: A New Approach to Self-Supervised Learning." In: **CoRR** abs/2006.07733 (2020). arXiv:2006.07733. URL: https://arxiv.org/abs/2006.07733.
2. T. Finke, M. Krämer, A. Morandini, A. Mück, and I. Oleksiyuk. "Autoencoders for unsupervised anomaly detection in high energy physics." In: **Journal of High Energy Physics** 2021.6 (June 2021). DOI: 10.1007/jhep06(2021)161. URL: https://doi.org/10.1007%5C%2Fjhep06%5C%282021%5C%29161%22.
3. S. Lu and R. Li. "DAC: Deep Autoencoder-based Clustering, a General Deep Learning Framework of Representation Learning." In: (2021). DOI: 10.48550/ARXIV.2102.07472. URL: https://arxiv.org/abs/2102.07472.
4. S. M. Lundberg and S.-I. Lee. "A unified approach to interpreting model predictions." In: **Advances in neural information processing systems** 30 (2017).
5. K. Simonyan, A. Vedaldi, and A. Zisserman. "Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps." In: (2013). DOI: 10.48550/ARXIV.1312.6034. URL: https://arxiv.org/abs/1312.6034.
6. M. Sundararajan, A. Taly, and Q. Yan. "Axiomatic attribution for deep networks." In: (2017), pp. 3319–3328.
7. N. Papernot and P. McDaniel. "Deep k-Nearest Neighbors: Towards Confident, Interpretable and Robust Deep Learning." In: (2018). DOI: 10.48550/ARXIV.1803.04765. URL: https://arxiv.org/abs/1803.04765.
8. G. Pruthi, F. Liu, S. Kale, and M. Sundararajan. "Estimating Training Data Influence by Tracing Gradient Descent." In: **Advances in Neural Information Processing Systems**. Ed. by H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin. Vol. 33. Curran Associates, Inc., 2020, pp. 19920–19930. URL: https://proceedings.neurips.cc/paper/2020/file/e6385d39ec9394f2f3a354d9d2b88eec-Paper.pdf.
9. J. Crabbé, Z. Qian, F. Imrie, and M. van der Schaar. "Explaining Latent Representations with a Corpus of Examples." In: (2021). DOI: 10.48550/ARXIV.2110.15355. URL: https://arxiv.org/abs/2110.15355.
10. J. Crabbé and M. van der Schaar. "Label-Free Explainability for Unsupervised Models." In: (2022). DOI: 10.48550/ARXIV.2203.01928. URL: https://arxiv.org/abs/2203.01928.
11. T. Chen, S. Kornblith, M. Norouzi, and G. Hinton. "A simple framework for contrastive learning of visual representations." In: (2020), pp. 1597–1607.
12. I. Higgins, L. Matthey, A. Pal, C. Burgess, X. Glorot, M. Botvinick, S. Mohamed, and A. Lerchner. "beta-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework." In: (2017). URL: https://openreview.net/forum?id=Sy2fzU9gl.
13. R. T. Q. Chen, X. Li, R. Grosse, and D. Duvenaud. "Isolating Sources of Disentanglement in Variational Autoencoders." In: (2018). DOI: 10.48550/ARXIV.1802.04942. URL: https://arxiv.org/abs/1802.04942.
14. Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. "Gradient-based learning applied to document recognition." In: **Proceedings of the IEEE** 86.11 (1998), pp. 2278–2323. DOI: 10.1109/5.726791.
15. A. L. Goldberg, L. A. Amaral, L. Glass, J. M. Hausdorff, P. C. Ivanov, R. G. Mark, J. E. Mietus, G. B. Moody, C. K. Peng, and H. E. Stanley. "PhysioBank, PhysioToolkit and PhysioNet: components of a new research resource for complex physiologic signals." In: **Circulation** 101.23 (2000). DOI: 10.1161/01.CIR.101.23.E215.
16. A. Krizhevsky. "Learning multiple layers of features from tiny images." In: Technical report (2009).
17. L. Matthey, I. Higgins, D. Hassabis, and A. Lerchner. "dSprites: Disentanglement testing Sprites dataset." In: (2017). https://github.com/deepmind/dsprites-dataset/.
18. H. Xiao, K. Rasul, and R. Vollgraf. "Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms." In: **arXiv preprint arXiv:1708.07747** (2017).
19. R. D. Cook and S. Weisberg. "Characterizations of an empirical influence function for detecting influential cases in regression." In: **Technometrics** 22.4 (1980), pp. 495–508.
20. P. W. Koh and P. Liang. "Understanding black-box predictions via influence functions." In: **International conference on machine learning**. PMLR. 2017, pp. 1885–1894.

21. G. Pruthi, F. Liu, S. Kale, and M. Sundararajan. "Estimating training data influence by tracing gradient descent." In: **Advances in Neural Information Processing Systems** 33 (2020), pp. 19920–19930.
22. J. Crabbé, Z. Qian, F. Imrie, and M. van der Schaar. "Explaining latent representations with a corpus of examples." In: **Advances in Neural Information Processing Systems** 34 (2021), pp. 12154–12166.
23. N. Papernot and P. McDaniel. "Deep k-nearest neighbors: Towards confident, interpretable and robust deep learning." In: **arXiv preprint arXiv:1803.04765** (2018).
24. G. Montavon, W. Samek, and K. Müller. "Methods for Interpreting and Understanding Deep Neural Networks." In: **CoRR** abs/1706.07979 (2017). arXiv:1706.07979. URL: http://arxiv.org/abs/1706.07979.
25. K. He, X. Zhang, S. Ren, and J. Sun. "Deep residual learning for image recognition." In: **Proceedings of the IEEE conference on computer vision and pattern recognition**. 2016, pp. 770–778.
26. G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger. "Densely Connected Convolutional Networks." In: (2016). DOI: 10.48550/ARXIV.1608.06993. URL: https://arxiv.org/abs/1608.06993.

A Appendices

A.1 Models

SimCLR — SimCLR model [11] is a self-supervised learning method that aims to learn representations of the input data by comparing different augmented versions of the same input via contrastive loss in the latent space. The authors used the combination of random crop and colour distortion as augmentation methods. Pre-trained ResNet-18 [25] or Densenet-121 [26] was used as an encoder on top of which a projection head with two linear layers and ReLU activation function was trained for 100 epochs. Hyperparameters used for training are in the table 4.

Name	Value
Optimizer	SGD
Learning rate	0.6
Momentum	0.9
Weight decay	1×10^{-6}
Temperature	0.5

Table 4. SimCLR hyperparameters.

ECG5000 autoencoder — We used the Recurrent Autoencoder that the authors used which consists of an encoder with an embedding dimension of 64, two LSTM layers and a decoder with two LSTMs and a final Linear layer. The model was trained to minimize the reconstruction loss given by $L_{rec}(x) = \sum_{t=1}^{T} |x_t - [\mathbf{f}_\mathbf{d} \circ \mathbf{f}_\mathbf{e}(\mathbf{x})]_t|$, where $\mathbf{x}$ is a vector representing one time series sample, T is the resolution of the heartbeat ($T = 140$) and $\mathbf{f}_\mathbf{e}$ and $\mathbf{f}_\mathbf{d}$ stand for the encoder and decoder functions. The model was trained for 150 epochs using the Adam optimizer.

MNIST autoencoder - Pretext Tasks — For each pretext task, a new autoencoder is trained. Each autoencoder is trained for 100 epochs, using the Adam optimizer, patience 10, and the same hyperparameters as presented in the paper. The classifier used follows the same architecture as the encoder used for the autoencoder, with an additional Softmax layer producing class probabilities. The pretext tasks tested are denoising, reconstruction, and inpainting, and for each pretext task the objective is to minimize their denoising, reconstruction, and inpainting loss accordingly as shown in equations 4, 5, 6. More details about these equations can be found in the original paper in Appendix C.1 and Appendix C.2.

$$L_{den}(x) = \mathbb{E}_\varepsilon[\mathbf{x} - \mathbf{f}_\mathbf{d} \circ \mathbf{f}_\mathbf{e}(\mathbf{x} + \varepsilon)]^2 \tag{4}$$

$$L_{rec}(x) = [\mathbf{x} - \mathbf{f}_\mathbf{d} \circ \mathbf{f}_\mathbf{e}(\mathbf{x})]^2 \tag{5}$$

$$L_{in}(x) = \mathbb{E}_M[\mathbf{x} - \mathbf{f}_\mathbf{d} \circ \mathbf{f}_\mathbf{e}(\mathbf{M} \odot \mathbf{x})]^2 \tag{6}$$

Disentangled Variational Autoencoder (VAE) — We used the provided code for VAE experiments and we ran two versions of disentangled VAEs: β-VAE [12] and TC-VAE [13], with beta values $\beta \in \{1,\ 5,\ 10\}$. We trained the models for 100 epochs on MNIST and dSprites datasets (90%-10% train-test split) using d_H = 3 and d_H = 6 latent units respectively. We ran 5 times for every disentangled VAE type for each β. Thus, in total, 30 models were trained.

A.2 Datasets

MNIST — This is a black-and-white image dataset containing digits from 0 to 9. Each image is 28x28 pixels. For the feature importance experiment the denoising autoencoder was trained on the entire training set and the attribution methods were tested on the whole training set. The example importance experiments were run on a subset of 1000 training and 1000 testing samples.

Fashion MNIST — It is similar to MNIST, but instead of digits, the images depict clothing items that belong to 10 different categories. The partitioning was done in the same way as for MNIST.

CIFAR-10 — This RGB image dataset contains 32x32 pixel pictures of objects corresponding to 10 categories. A 50000/10000 partition was used for the feature importance experiment, while for determining the most important examples, they used a subset of 1000/1000.

ECG5000 — Is a time series dataset that describes the heartbeat of a patient. Each time series describes one single heartbeat with a resolution of 140 time steps. This dataset was split into 4000 training and 1000 testing samples when conducting the experiments.

dSprites — This is a synthetic dataset of images showing 2D shapes generated from the following latent factors: colour, shape, scale, rotation, x and y positions of a sprite. The dataset has a total of 737280 images, from which 10% were used at test time.

A.3 Reproducing Pretext Experiments

Visualisations — We visualise the top examples produced by different pretext tasks as well as their saliency maps for qualitatively interpreting the results. The results are shown in Figure 7 and Figure 8.

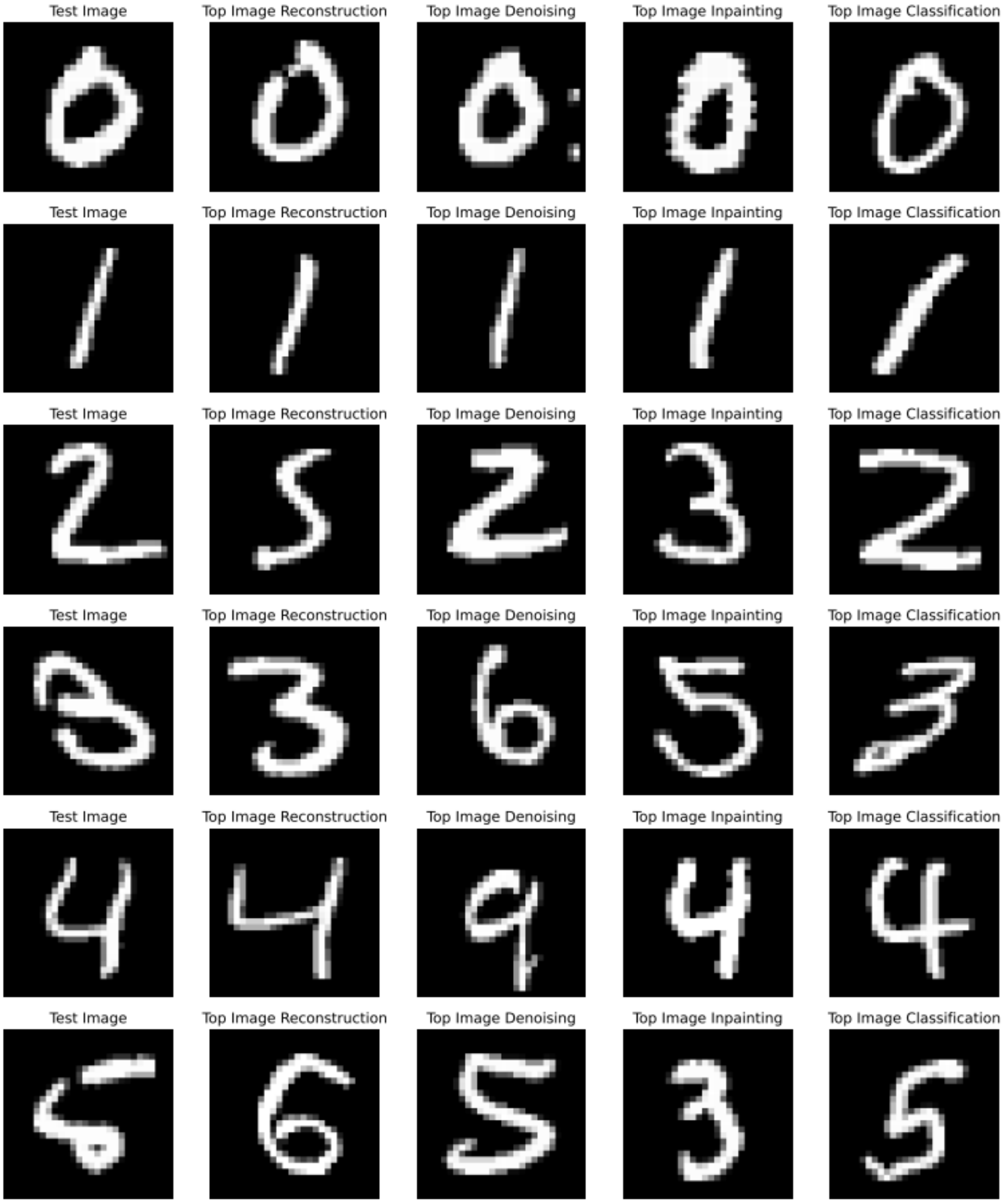


Figure 7. Label-free top example for various pretext tasks.

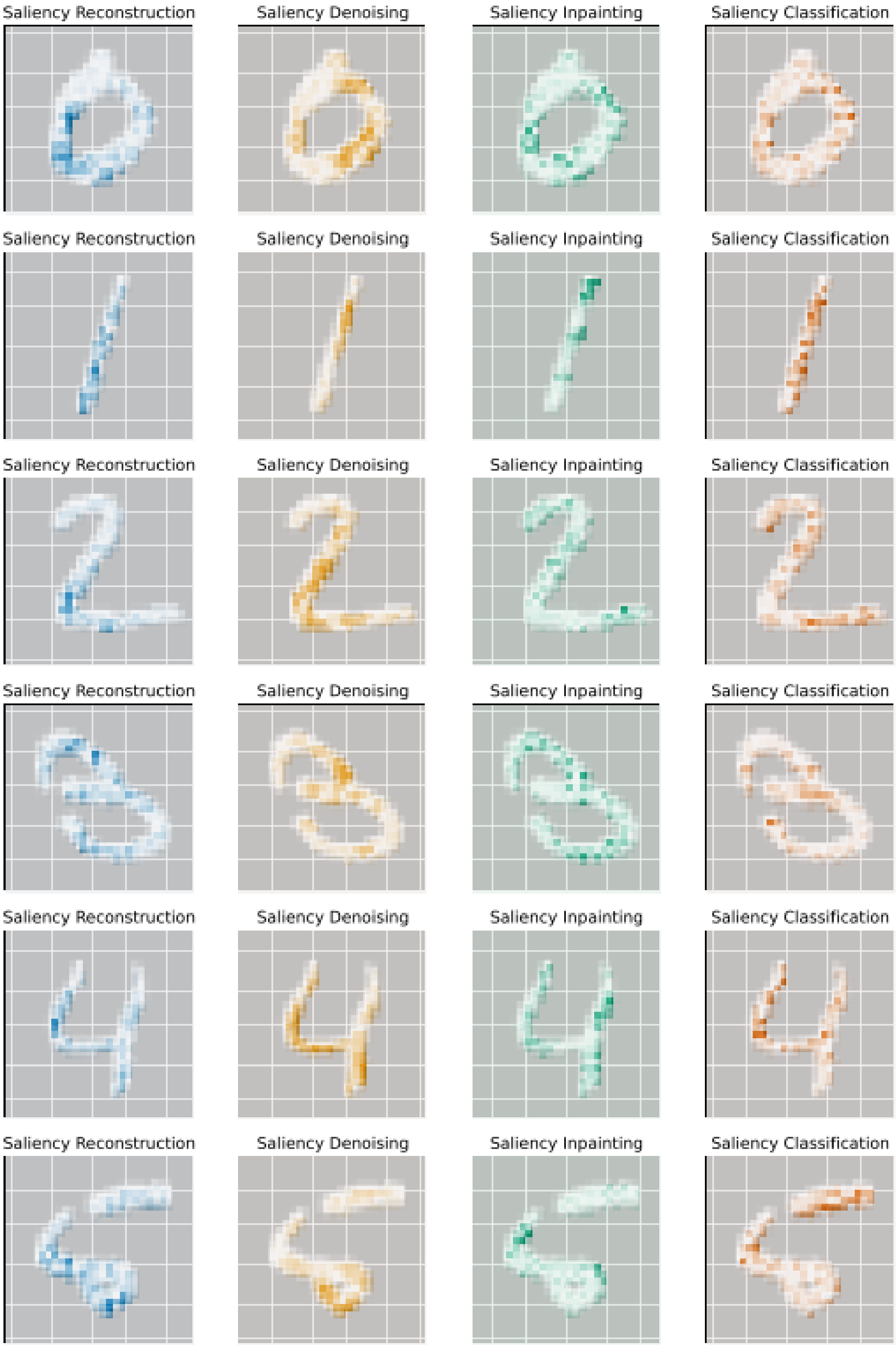
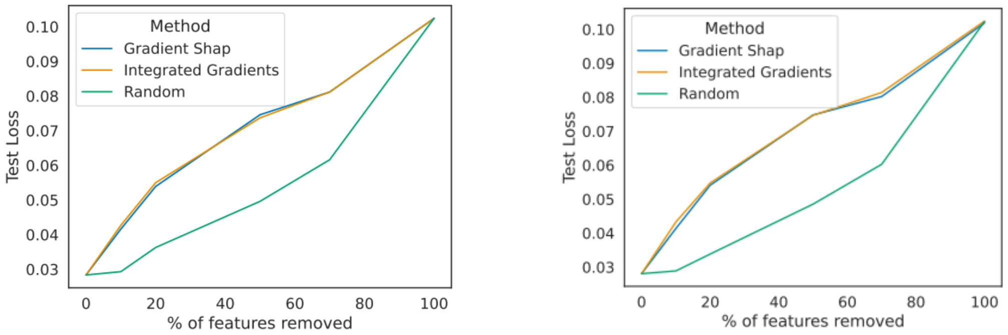


Figure 8. Label-free saliency for various pretext tasks.

ROAR Test — The original paper also computes an additional test supporting the label-free feature importance method. The test follows the same consistency check for feature importance, with a difference in first removing the most important pixels, and then training the autoencoder on the modified data. We compare results with the original paper in Figure 9 and obtain a similar trend.



(a) Our result.

(b) Their result.

Figure 9. Comparing ROAR test results.

A.4 Experiments on Fashion MNIST

Example consistency — It can be seen in Figure 10 that the same trend as for the other datasets holds for Fashion MNIST when testing the example consistency: the similarity rate between the most important examples is much higher than for the least important ones, showing that the method allows the identification of training samples related to test examples in the label-free setting.

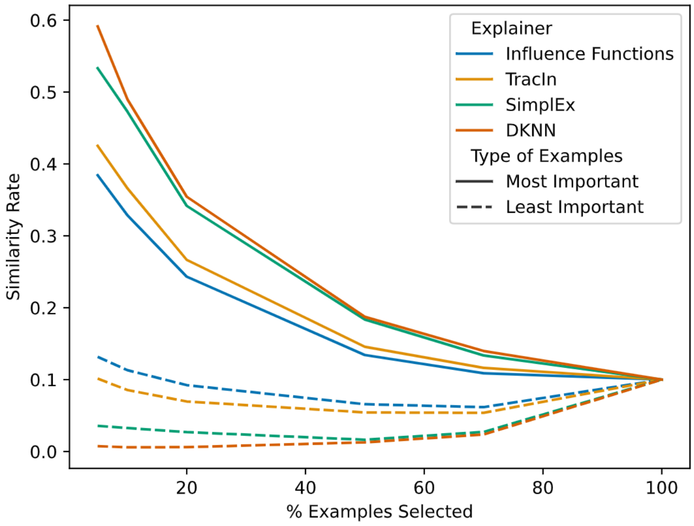


Figure 10. Consistency check for label-free example importance on Fashion MNIST.

Quantitative analysis for pretext tasks — Table 5 and Table 6 show the Pearson correlation coefficients of representations learned for different pretext tasks on Fashion MNIST dataset. The Pearson scores range from .31 to .49 corresponding to moderate positive correlation for saliency maps and from .07 to .31 corresponding to weak correlations for example importance.

PEARSON	RECON.	DENOIS.	INPAINT.	CLASSIF.
RECON.				
DENOIS.	$.49 \pm .05$			
INPAINT.	$.43 \pm .02$	$.45 \pm .02$		
CLASSIF.	$.37 \pm .01$	$.36 \pm .02$	$.31 \pm .03$	

Table 5. Pearson correlation for saliency maps (avg +/- std).

PEARSON	RECON.	DENOIS.	INPAINT.	CLASSIF.
RECON.				
DENOIS.	$.27 \pm .06$			
INPAINT.	$.30 \pm .03$	$.31 \pm .09$		
CLASSIF.	$.07 \pm .02$	$.07 \pm .03$	$.07 \pm .03$	

Table 6. Pearson correlation for example importance (avg +/- std).

Qualitative analysis for pretext tasks — The most important examples can be seen in Figure 11 and the saliency maps can be visualized in Figure 12. By plotting these images we can better understand the choice of most important examples: if we look at the saliency maps for the sneaker image, we see that only the inpainting and the classification representations focus on the midsole and outsole of the shoe, leading to having more relevant top examples to the test image for these tasks.

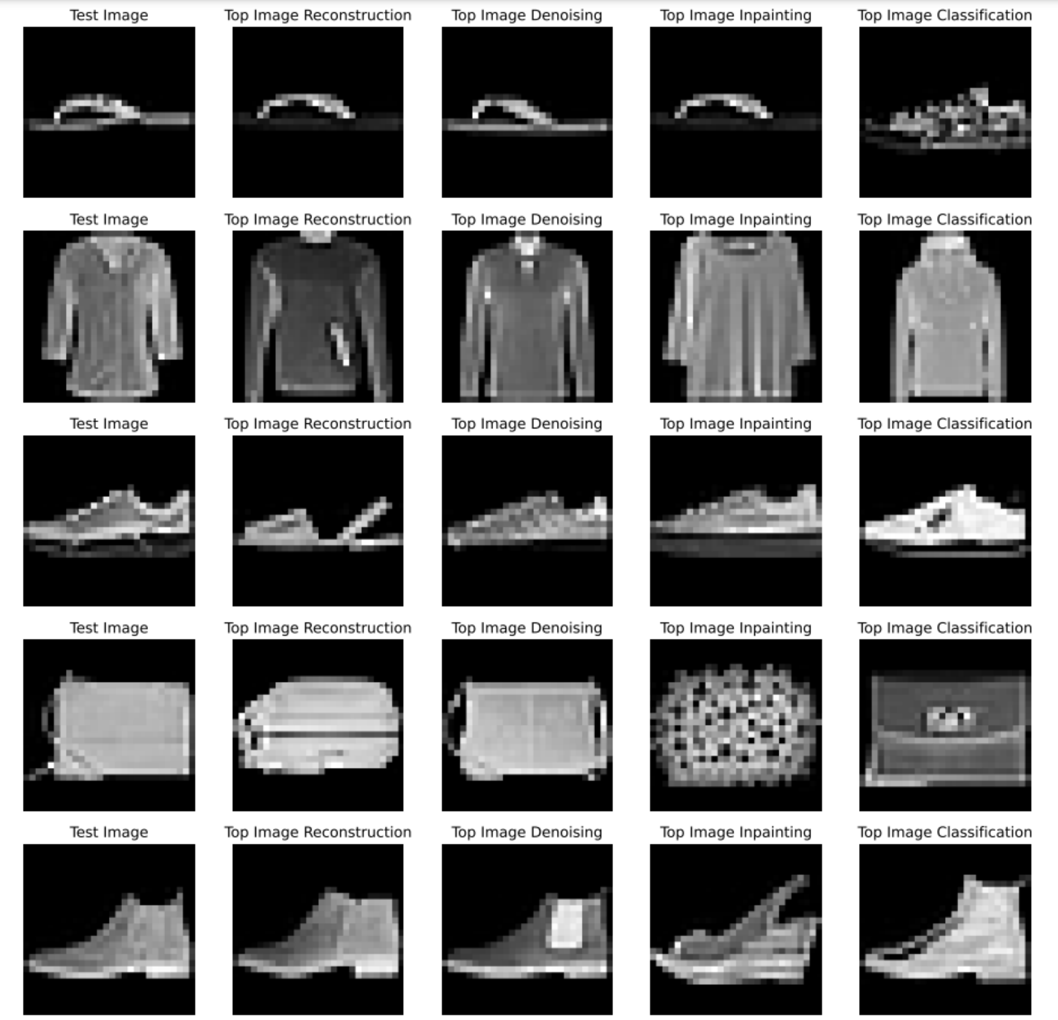


Figure 11. Top examples for sandal, shirt, sneaker, bag and ankle boot categories.

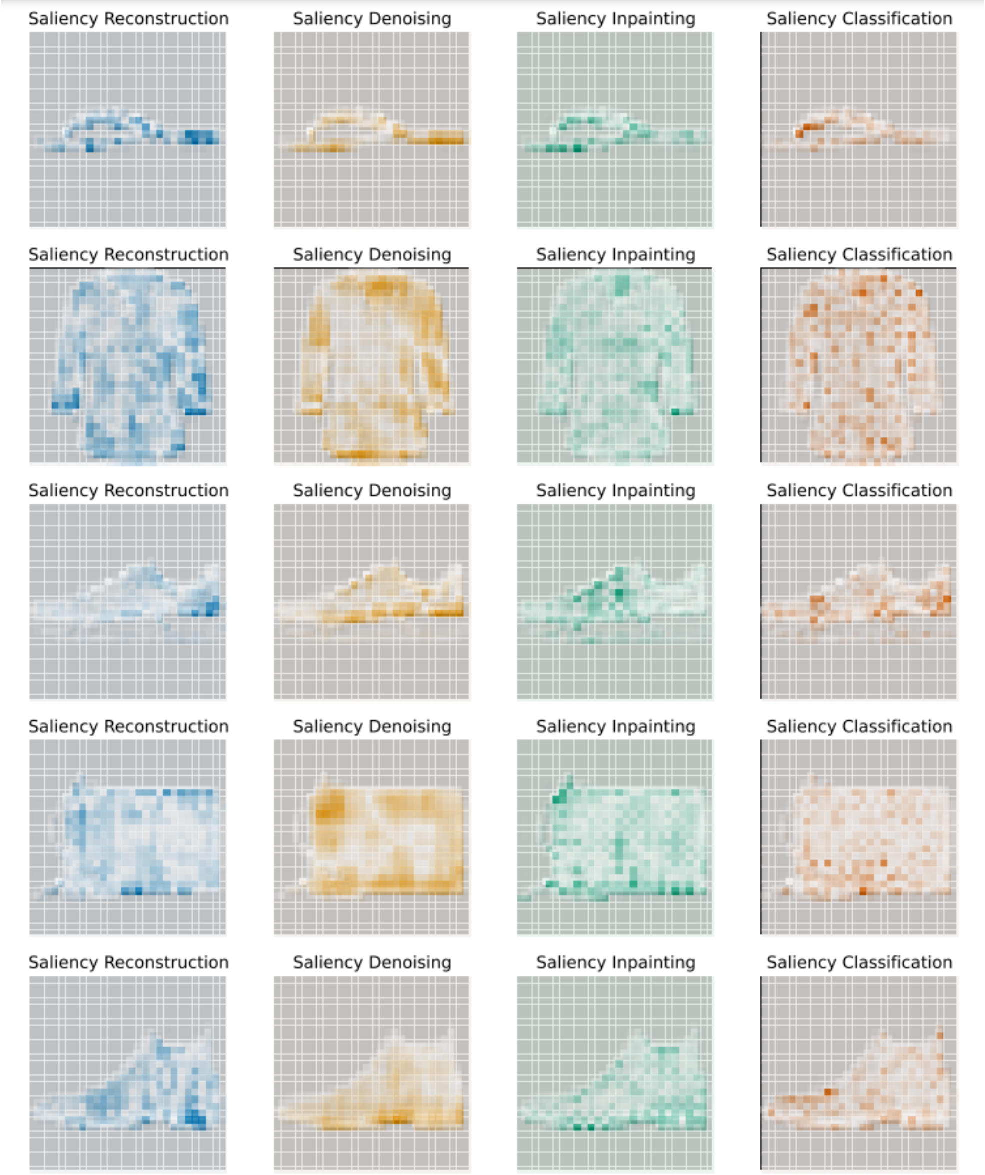


Figure 12. Saliency maps.

A.5 CIFAR-10 masked images

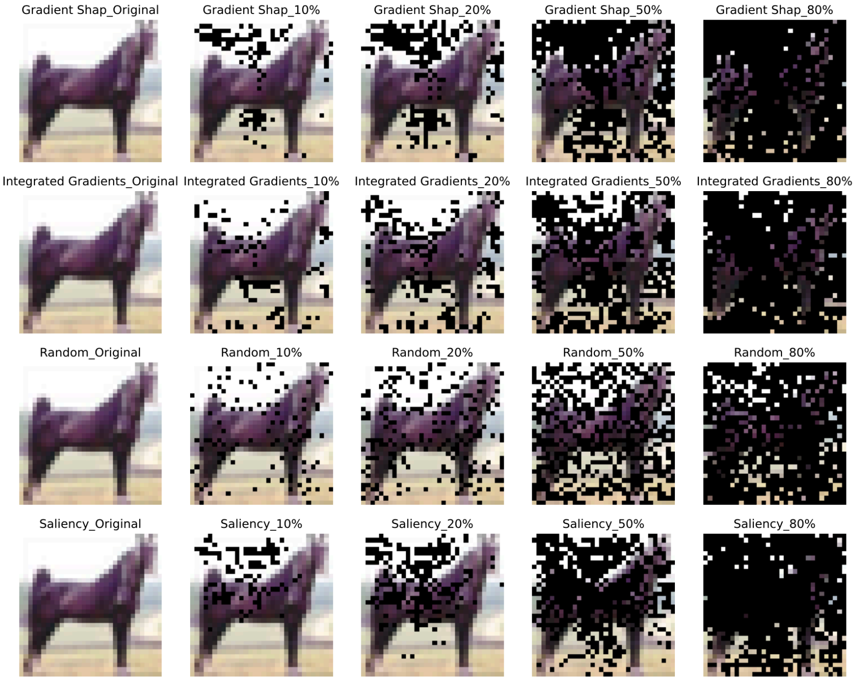


Figure 13. Example of masked image from CIFAR-10 dataset for black pixel mask

A.6 Explainability analysis with different attribution methods

Integrated Gradients — For the *pretext experiment*, we analyse the results **quantitatively** in Table 7 and Table 8 and **qualitatively** in Figure 16 and Figure 15. Moreover, for the *VAE experiment*, we analyse the results **quantitatively** in Figure 14 and **qualitatively** in Figure 17. Both experiments use label-free Integrated Gradients as their attribution method.

PEARSON	RECON.	DENOIS.	INPAINT.	CLASSIF.
RECON.				
DENOIS.	0.45 ± 0.06			
INPAINT.	0.43 ± 0.08	0.45 ± 0.05		
CLASSIF.	0.39 ± 0.03	0.4 ± 0.02	0.35 ± 0.05	

Table 7. Pearson correlation for saliency maps (avg +/- std) using label-free Integrated Gradients.

PEARSON	RECON.	DENOIS.	INPAINT.	CLASSIF.
RECON.				
DENOIS.	0.14 ± 0.04			
INPAINT.	0.18 ± 0.04	0.21 ± 0.04		
CLASSIF.	0.09 ± 0.02	0.09 ± 0.02	0.1 ± 0.01	

Table 8. Pearson correlation for example importance (avg +/- std) using label-free Integrated Gradients.

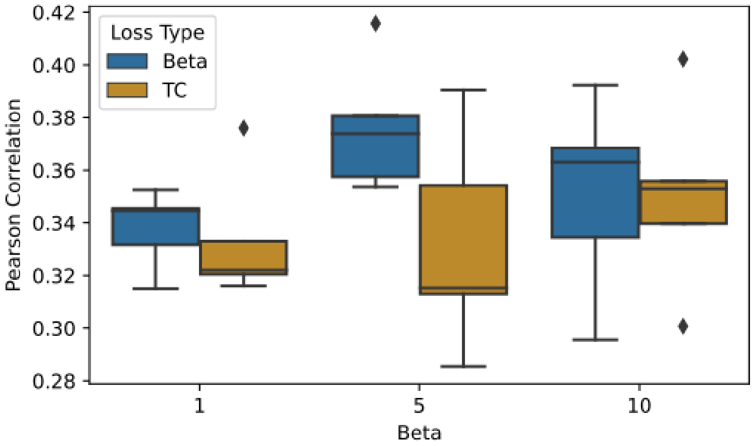


Figure 14. Pearson correlation between saliency maps for different values of β using label-free Integrated Gradients.

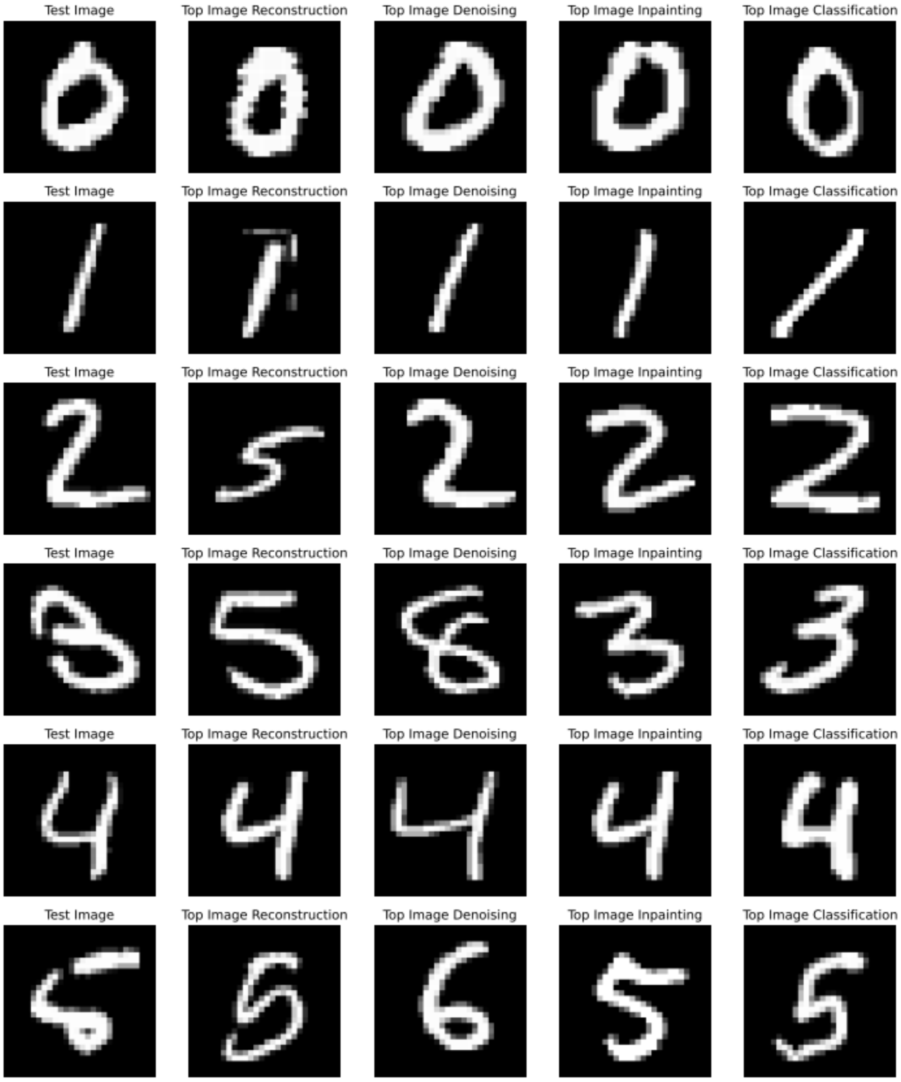


Figure 15. Top examples using label-free Integrated Gradients.

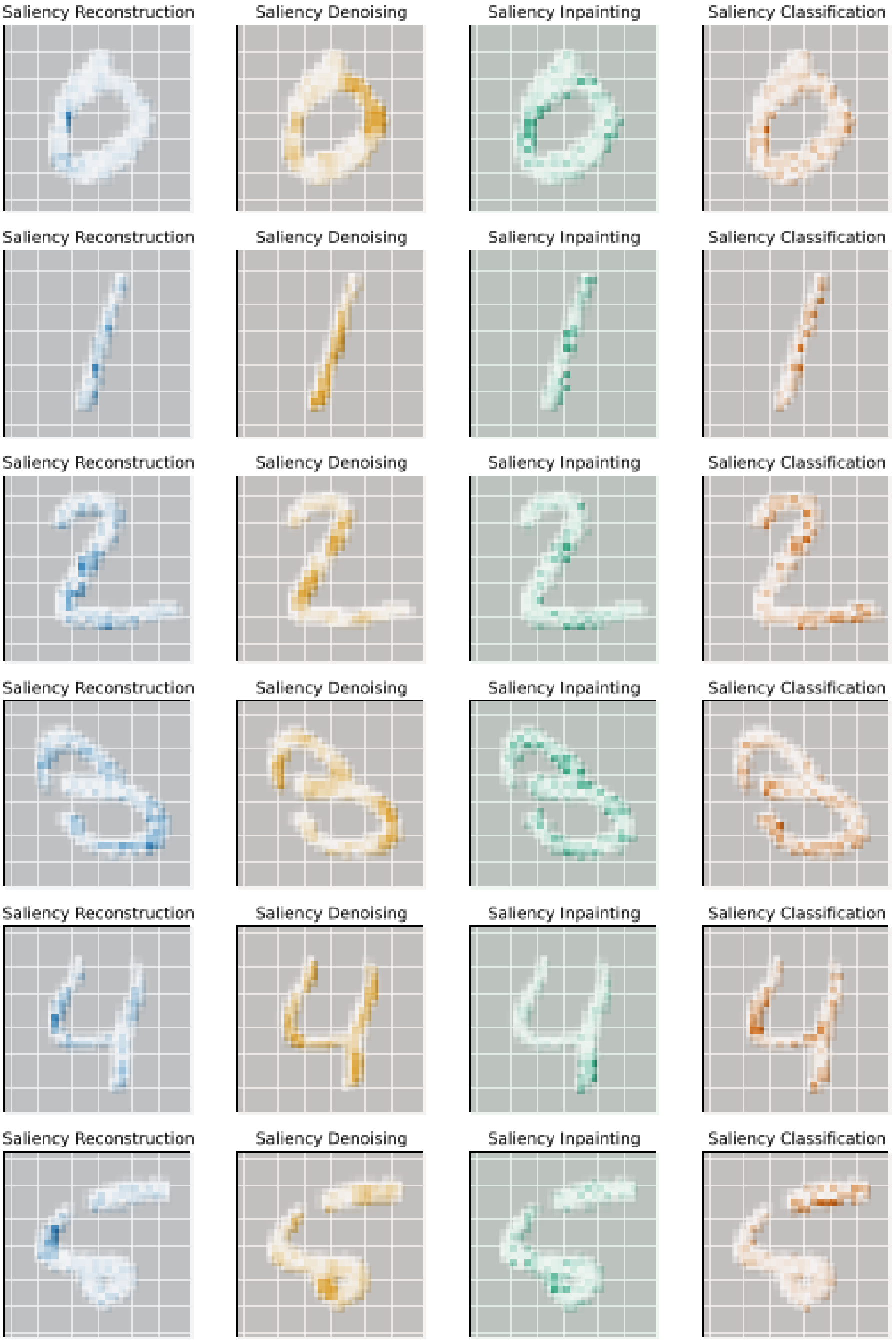


Figure 16. Saliency maps using label-free Integrated Gradients.

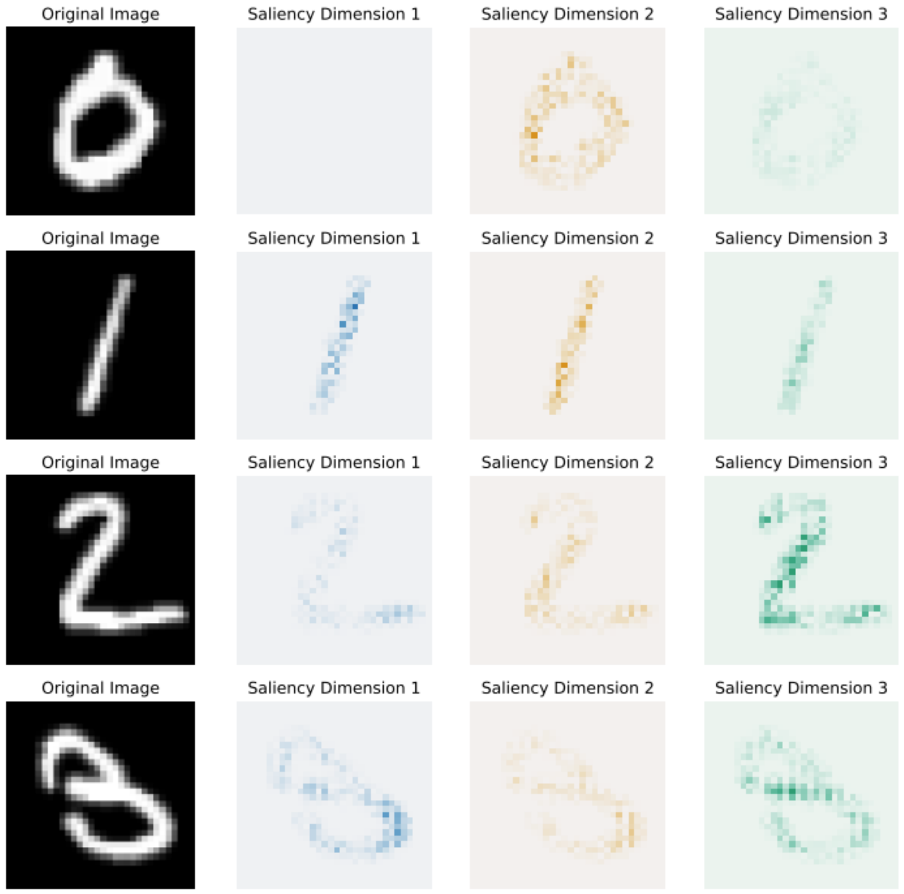


Figure 17. Saliency maps for each unit of the disentangled VAEs using Integrated Gradients.

A.7 DenseNet Experiments

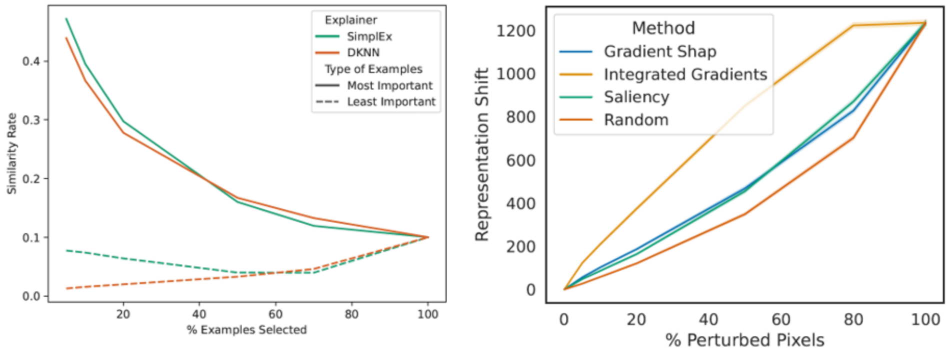


Figure 18. Consistency check for label-free example importance (left) and label-free feature importance (right) using DenseNet121 on CIFAR-10 dataset.