Lab 4 Explainability

Duruo Li

May 22, 2023

1 Introduction

This report is about explainability for models, especially or "black-box" models. There are two main ideas: 1) focus on local rather than global explainability, i.e., certain locations/images 2) artificially perturb the input and observe outputs (kind of like the control variate methods)

This report is divided into two parts: Part 1, Tabular Data; Part 2, Image Data, in which I try different methods to explain complex models, e.g., LIME, SmoothGrad, etc.

2 Tabular Data

2.1 Imbalance Test

Question: 1

Diabetes: 34.896%

Not diabetes: 65.104%

There is a slight imbalance w.r.t the outcome variable, but it is not severe. In reality, the percentage of people with diabetes is typically less than 35%. Therefore, it's possible that this dataset has already been modified.

2.2 Feature importance

Questions: 2-6

Linear Model (L1 Penalty)

As for Lasso linear model, to check the feature importance, we check the coefficients/weights of each features, see table 1: None of the eight features have been completely eliminated. However, if we disregard features with a weight magnitude below 0.001, SkinThickness and Insulin would be ruled out.

Random Forest

For random forest model, there are two ways to check feature importance: forest importance and LIME.

Feature	Weight
Pregnancies	0.018288
BMI	0.013085
Glucose	0.006288
BloodPressure	-0.002529
Age	0.002344
Insulin	-0.000078
SkinThickness	-0.000026
DiabetesPedigreeFunction	0.000000

Table 1: Features selected by Lasso linear model

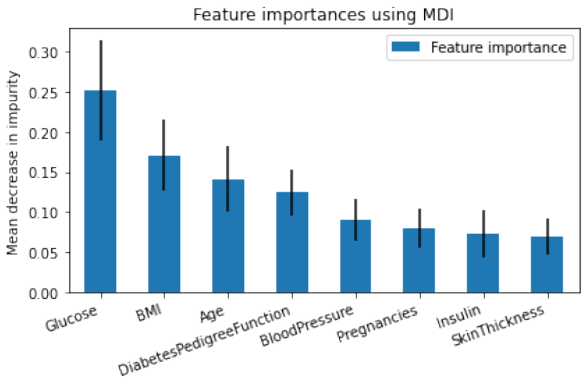


Figure 1
Forest importance using MDI

According bar plot 1, which shows the **forest importance** based on Mean Decrease in Inpurity(MDI) of each features, Glucose, BMI, Age and DiabetesPedigreeFunction are the top 4 most important features.

p.s. black error bar is the deviation of the impurity decrease within each tree i.e., uncertainty of the feature importance across different trees

As for **LIME**, since it's a local method, I choose two instances, namely the 5th and the 12th samples from test dataset, to check feature importance respectively, and then make comparisons, see figure 2 and figure 3.

For sample 5, the predicted probability of having diabetes is 78%. In terms of feature importance, BMI (greater than 36.38) has the highest score for predicting class 0 (not having diabetes). On the other hand, Age (less than or equal to 24), Preganancies (less than or equal to 1), SkinThickness (greater than 22), and DiabetesPedigreeFunction (greater than 0.24) are the top four most important features for predicting class 1 (having diabetes).

In comparisons, for sample 12, the most significant feature for predicting class 0 (not having diabetes) is Glucose (greater than 141). However, this feature is not even among the top 5 important features for sample 5. Additionally, both SkinThickness and DiabetesPedigreeFunction contribute to class 0 (having diabetes) for sample 12, whereas they contribute to class 1 (not having diabetes) for sample 5.

In conclusion, the LIME outcomes aren't stable across different data points, which is reasonable since it's a local method.

Comparisons: Weights v.s Forest importance v.s LIME
I choose Sample 5 to compare the top 5 most important features selected by the 3 methods:

- Linear model weights: DiabetesPedigreeFunction, Pregnancies, BMI, Glucose, Age
- Forest importance: Glucose, BMI, Age, DiabetesPedigreeFunction, BloodPressure
- LIME: BMI, Age, Preganancie, SkinThickness, DiabetesPedigreeFunction

These models have DiabetesPedigreeFunction, BMI, and Age as common features among their top 5 selections. The feature Pregnancies appears in two of the models. Overall, they share similar features in 3 out of 5 selections, i.e., roughly speaking, they select similar features.

However, it's worth noting that LIME may not be stable and might not share as many selected features for certain samples.

3 Image Data

Questions: 7

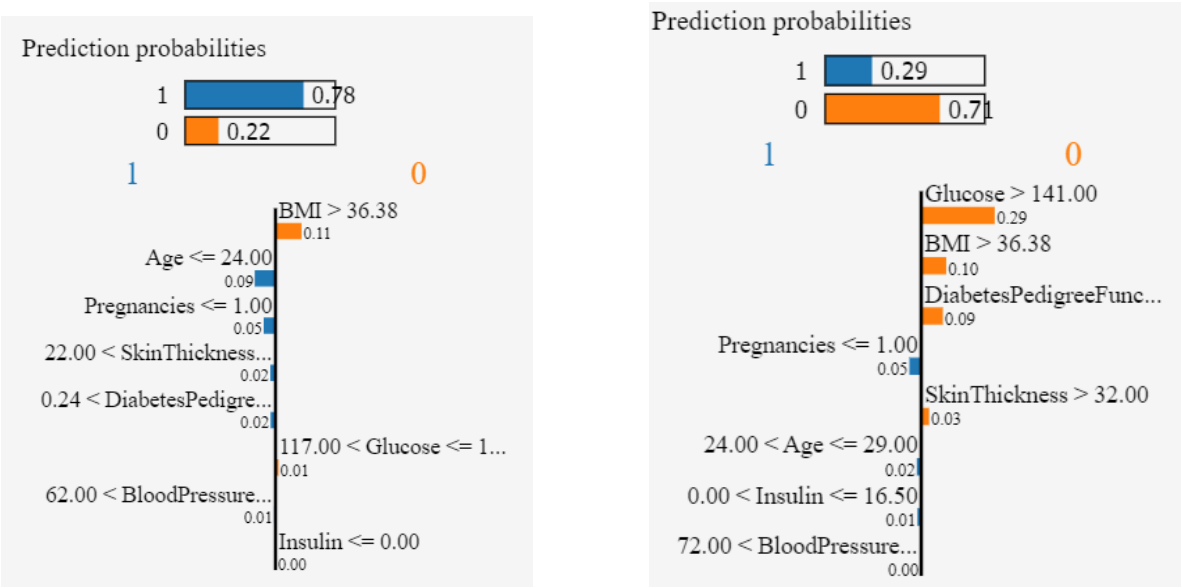


Figure 2: LIME outcomes for Sample 5(left) & Sample 12(right)

Feature	Value
BMI	38.80
Age	22.00
Pregnancies	0.00
SkinThickness	27.00
DiabetesPedigreeFunction	0.26
Glucose	119.00
BloodPressure	66.00
Insulin	0.00

Feature	Value
Glucose	180.00
BMI	59.40
DiabetesPedigreeFunction	2.42
Pregnancies	0.00
SkinThickness	63.00
Age	25.00
Insulin	14.00
BloodPressure	78.00

Figure 3: LIME outcomes for Sample 5(left) & Sample 12(right)

Animal_10N dataset includes 10 classes of images, see figure 4.

3.1 Modeling

Questions: 8-10

To classify the images, I try different models, including one linear model and 3 CNN models.
To be specific, hyper-parameters(some of them) for the 4 models are as follows:

- Linear: one linear layer
- VanillaCNN1: 2 convolutional layers + one linear layers; ReLU activation funtion; Adam optimizer
- VaniilaCNN2: 2 convolutional layers + 2 linear layer; ReLU activation function; Adam optimizer
- VaniilaCNN3: 2 convolutional layers + 2 linear layer; ReLU activation function; SGD optimizer

p.s. I have also tried other hyper-parameters to tune the CNN model, such as ELU activation function, SGD optimizer, different batch sizes, epochs, etc. However, none of them performs better than VanillaCNN2. Therefore, I choose not to show them.

To compare their performances, I make **learning curves** based on loss and accuracy for Linear, CNN1, and CNN2 models, These two metrics provide insights into different aspects of the models. Loss entropy cares about how exactly it predicts correctly i.e., not only demanding "relatively" correct, but also "absolutely" correct. While accuracy solely assesses whether the model makes correct predictions through comparisons, without considering the level of uncertainty in the guesses.

According to the curves, see figure 5, the VanillaCNN2 model achieved the highest validation accuracy of approximately 50%.

Note that, as the number of epochs increases, the validation performance initially improves and then diminishes. Therefore, it is advisable to limit the number of epochs to prevent overfitting.

During the tuning process, I discovered that the number of epochs, number of layers, optimizer, activation function all play crucial roles.

The choice of optimizer, in particular, has a significant impact. To illustrate, comparing VanillaCNN2 and VanillaCNN3, with other parameters held constant, changing the optimizer from "Adam" to "SGD" resulted in even worse performance than that of a linear model with a single layer. See figure 6.

3.2 Feature atribution

Questions: 11, 12

SmoothGrad

By adding different magnitudes of noise (standard deviation), the salience maps are shown in figure 8. It can be observed that neither too low nor too high noise levels are visually appealing. Optimal results, in terms of human vision, are achieved when adding noise at certain levels, such as 5% to 10% in this case.

LIME

By using LIME to show what features, i.e., certain regions of the image, are influential in my CNN, see figure 7.

During tuning, it was observed that increasing the number of superpixels leads to a more fine-grained examination of the image. I noticed that superpixels can overlap with each other and may not be of equal size. However, there is a point of saturation where further increasing the number of superpixels does not result in any noticeable changes, i.e., unable distinguish finer details.

When applying SmoothGrad and LIME to identify important features in predicting the "Cat" class, no specific influential regions were consistently observed across different cat images. Some

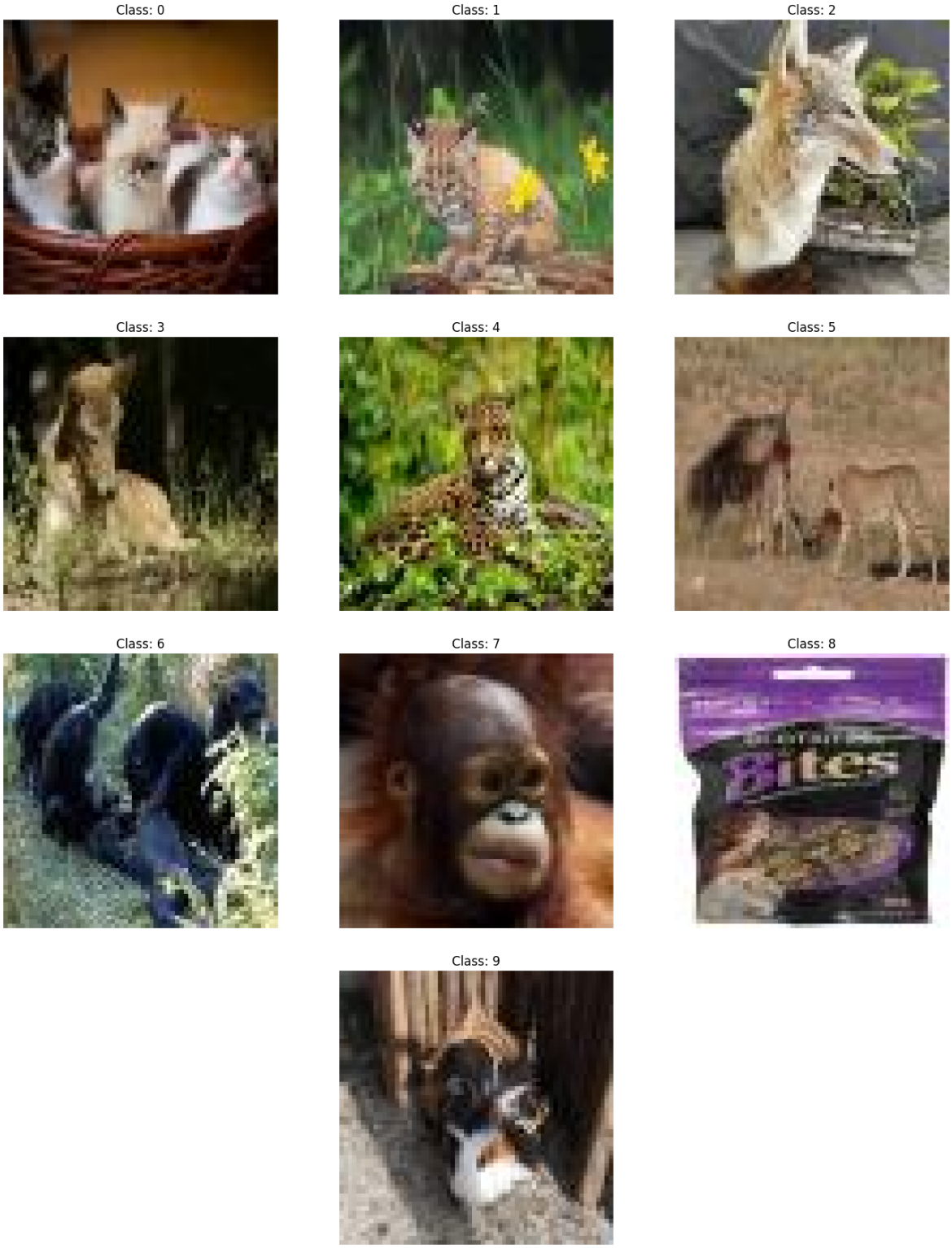


Figure 4: Sample figures for each class in Animal_10N dataset

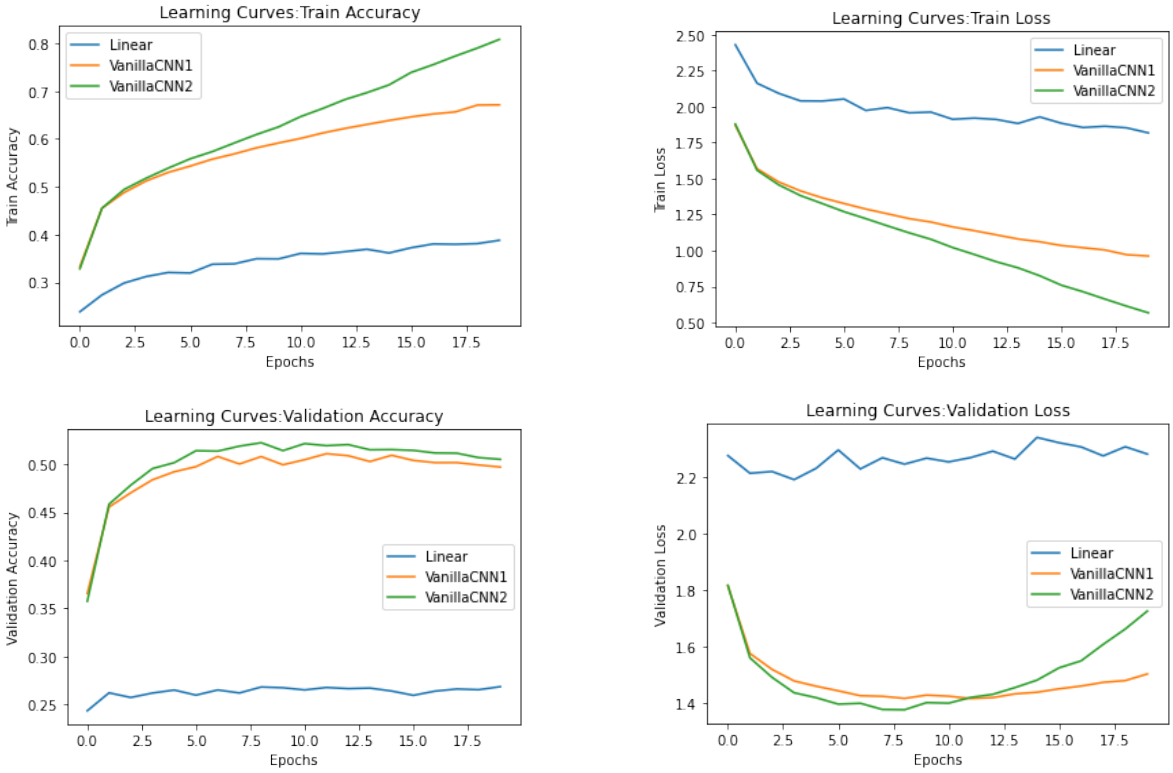


Figure 5: Learning curves: training and validation

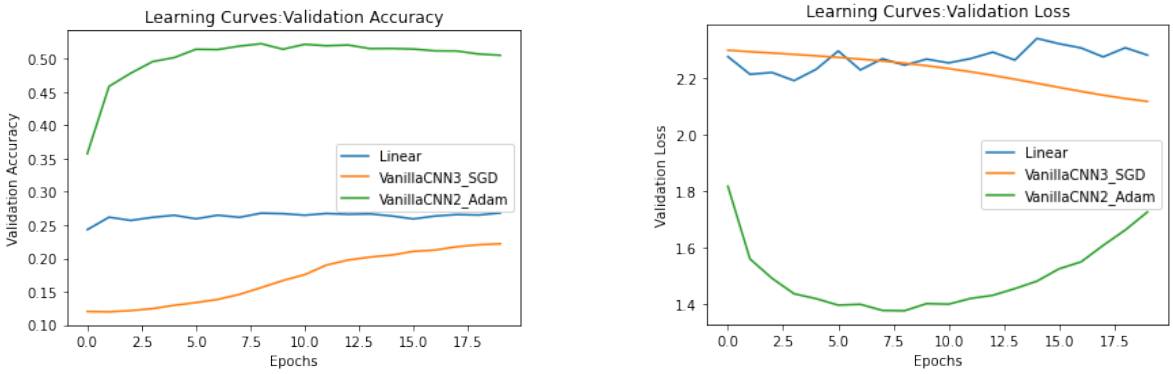
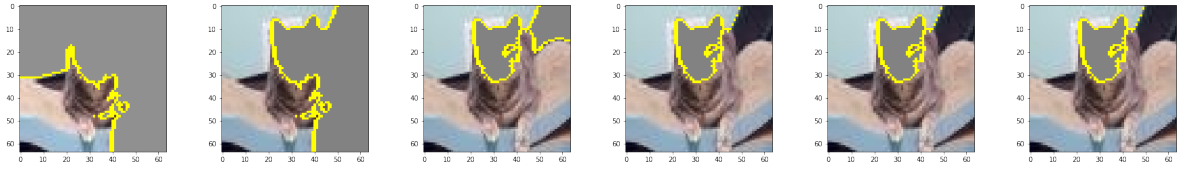
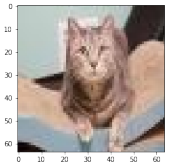


Figure 6: Validation loss and accuracy for optimizer Adam and SGD

images emphasized the body contour, while others focused on the shape of the head. Additionally, certain images highlighted the importance of the environment, such as tables or grass, where the cats were located.



n_features = 1 n_features = 2 n_features = 3 n_features = 4 n_features = 5 n_features= 10



original image

Figure 7: LIME using different number of superpixels

4 Conclusion

While working on this homework, I have three major feelings.

Firstly, tuning the models is indeed a crucial yet time-consuming task. Secondly, a faster computer system is so necessary to expedite the process. Lastly, "explainability" kind of like running too fast and then turning around–using a black box to capture the "law" of nature, and then use another (maybe simpler) model to capture the "law" of the black box lol.

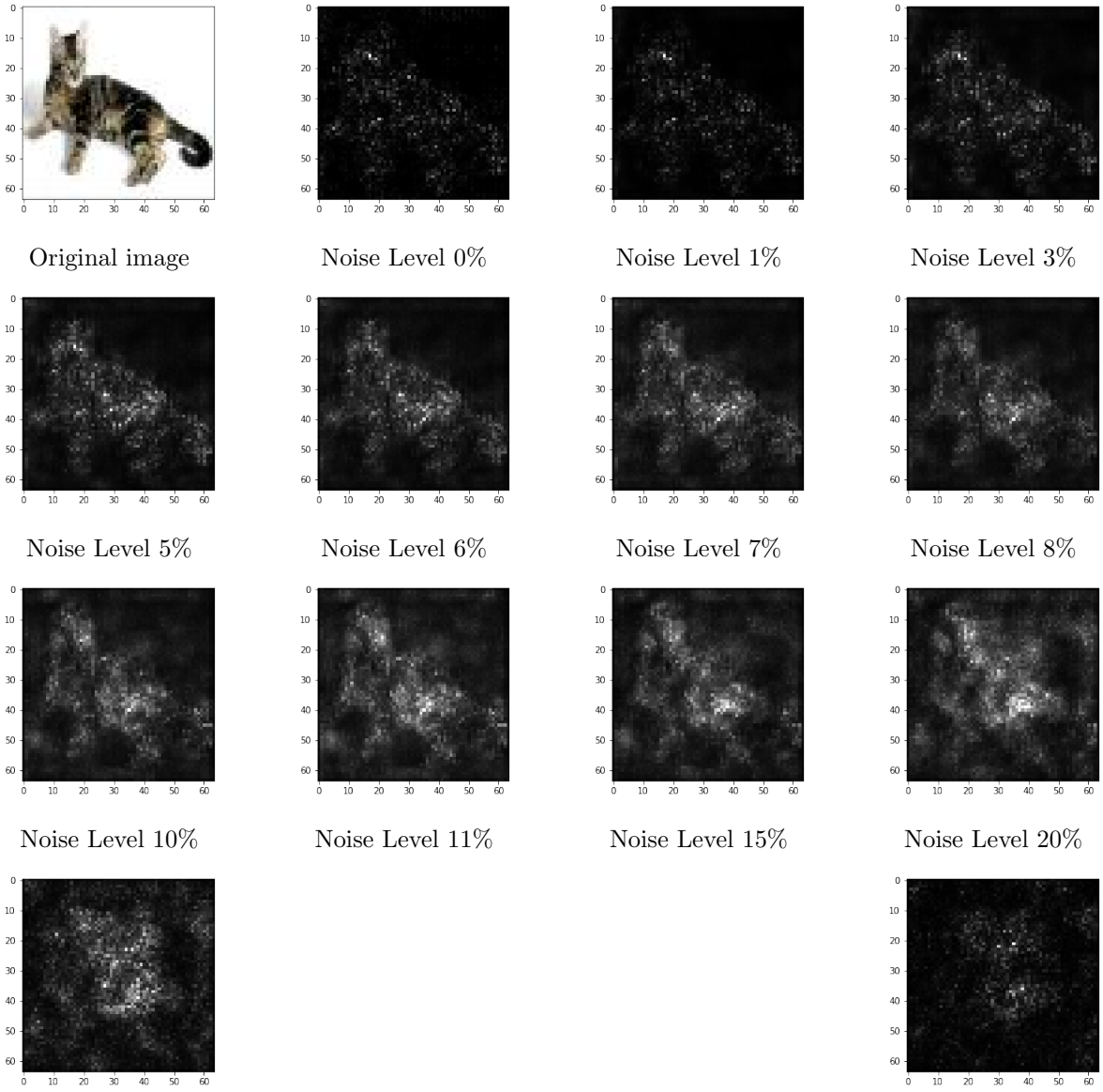


Figure 8: Gradient sensitivity maps with different noise levels