Note 1: The results shown and cited may vary slightly after every run of the program because of the stochastic nature of the whole process.

Note 2: The appendix contains the classification report as well as the four plots.

Choice of Task and Dataset

Conceptually, at least at first glance, image classification is a simple task. Given an image, it must be classified as belonging to one of a given number of classes. Thus, the input is the image, and the output is the class to which the object depicted in the image is predicted to belong. It is rather easy to see where uncertainty can arise – if the image is of poor resolution, if the object looks like it could belong to more than one class, if the object looks like it might not belong to any of the defined classes, etc. This was the motivation behind using image classification as the task on which uncertainty quantification is explored.

CIFAR-10 was used as the dataset for this task because aside from being able to access the dataset easily through common libraries, it has a good number of classes – ten – for this task, and the images are of low resolution. The low resolution of the images introduces a certain level of uncertainty, which, for this particular project, actually becomes a desirable feature. Also, the pictures themselves depict real-life objects, which makes this task addressed in this project at least somewhat realistic.

Choice of Model Architecture

Convolutional Neural Networks, or CNNs, are now the near-unanimous choice for image-related machine-learning tasks, including image classification. However, there are truly an infinite number of choices of CNN model architecture; exploring them all is literally impossible.

To provide a good direction in which to start, the architecture was constructed in the style of Visual Geometry Group, or VGG. VGG-style architecture made sense for this model because it is a proven and high-performing network for image recognition and classification and also because the reason for the architecture’s efficacy is relatively easily understood. This kind of architecture consists of multiple blocks that each contain two convolutional layers and one max-pooling layer, with convolutional layers in each successive block containing double the number of filters as in the previous one. Meanwhile, dropout layers were added not only to prevent overfitting, but to help quantify uncertainty later in this project.

Models with different numbers of convolutional blocks, dropout layers, and fully connected layers were trained and tested, with the final architecture shown in the code being the one that yielded the best testing accuracy after achieving an impressive training accuracy. Regarding the dropout layers in the final architecture, only 20% of units will be dropped by each dropout layer, but because there are multiple such dropout layers, overfitting is still mitigated.

In general, overfitting was a common problem, especially when the models did not contain a sufficient quantity of dropout layers. Training accuracies regularly exceeded 90%, but testing accuracies languished between 60% to 70%. This project is a good example of the importance of dropout and methods to combat overfitting in general.

Testing Results

Accuracy is the most common metric cited for model performance on CIFAR-10. The model achieved ~78% testing accuracy. The author of https://franky07724-57962.medium.com/once-upon-a-time-in-cifar-10-c26bb056b4ce# states that “a simple convolutional neural network with a proper training strategy can achieve accuracy between 80% and 90%”. Because the accuracy achieved by the subject model comes quite near that range, the model’s performance, based on accuracy, is quite decent.

The classification report generated on the testing predictions, shown in Figure 1 of the Appendix, provides further insights into the model’s performance, specifically the precision, recall, and f1-score. These demonstrate the model’s performance on a class-by-class basis. Classes 3 and 5, which, respectively, are cat and dog, are clearly detected more poorly than the other classes, based on the f1-score. Classes 1 and 8, which, respectively, are automobile and ship, are the classes that are detected the best, based on the f1-score. The report shows that six of the ten classes had f1-scores greater than 80%, while the remaining four had f1-scores less than 75%, with two of them having f1-scores less than 70%. This shows that the model clearly struggled with a few classes somewhat significantly more than the others. Perhaps more convolutional blocks coupled with a greater number of filters would improve the model’s performance on these classes.

Choice of Uncertainty Quantification

For this project, uncertainty was quantified using the dropout layers. Dropout was be enabled during prediction, multiple predictions were made, and the mean and variance of those predictions were calculated to provide quantitative insight into the model's uncertainty.

This method is highlighted as Monte-Carlo Dropout in this article: https://medium.com/uncertainty-quantification-for-neural-networks/uncertainty-quantification-for-neural-networks-a2c5f3c1836d

It is also described here: https://www.cs.ox.ac.uk/people/yarin.gal/website/blog_3d801aa532c1ce.html

This particular method was chosen for uncertainty quantification because it is the one method that is described by Gal as having been applied to image classification. It is also relatively easy to understand the motivation behind the algorithm and perform its implementation.

Choice of Number of Predictions with Uncertainty

Once dropout was enabled for testing, the model could have been made to predict the labels on the testing data any number of times. A choice had to be made such that a reasonable snapshot of the theoretical distribution of outputs could be expected to be captured while limiting computation time from becoming excessive. Thus, the number of predictions with uncertainty was chosen to be ten.

Method of Uncertainty Quantification

The mean probabilities and variance of the probabilities over the ten sets of predictions were obtained. Then, the class predictions were made based on the mean probabilities. The index of the greatest mean probability within each mean prediction was designated as the predicted class, and the greatest mean probability was called the mean predictive probability. Finally, the variance that corresponds to the mean predictive probability was identified.

Uncertainty Quantification Results

Mean Predictive Probability

The mean predictive probability is plotted as a function of the sample index. This plot is shown in Figure 2 of the Appendix. Because the indices are ordered such that their respective mean predictive probabilities are in ascending order, there is an increasing curve. This plot may help in observing which predicted classes tend to have higher – and lower – mean predictive probabilities.

The part of the curve corresponding to the highest mean predictive probabilities has a "purplish" hue, which means that horse and automobile may consistently be more confidently predicted than other classes. Meanwhile, the parts of the curve corresponding to the medium and low mean predictive probabilities have a decent infusion of yellow, which means that deer and bird may be more consistently predicted with medium or low confidence than other classes.

Predictive Variance

Similarly, the predictive variance is plotted as a function of the sample index. This plot is shown in Figure 3 of the Appendix. This plot may help in observing which predicted classes tend to have higher – and lower – predictive variances, especially considering that the sample indices are ordered in ascending order of mean predictive probability.

This plot may not at first glance be very suggestive, but the part of the plot with the lowest predictive variances – the "tail" on the right side of the plot – also has a "purplish" hue, which is consistent with the observation on the previous plot that the part of the curve with the highest mean predictive probabilities also has a "purplish" hue. Consistently high mean predictive probabilities also often result in consistently low predictive variances. Meanwhile, the middle and left side of the predictive-variance plot has a decent infusion of yellow, which coincides well with the observation on the previous plot that the parts of the curve corresponding to the medium and low mean predictive probabilities also have a decent infusion of yellow. Consistently medium and low mean predictive probabilities also often result in consistently medium and high predictive variances.

Average Mean Predictive Probability

The average mean predictive probability for each predicted class is obtained and plotted as a function of predicted class. This plot is shown in Figure 4 of the Appendix. This plot, with much fewer points, will help in clearly seeing which classes were, on average, predicted with higher – and lower – probabilities.

The automobile class has the highest average mean predictive probability, which means that it is the class that is predicted on average with the highest confidence. The horse class is within the top five classes that are predicted on average with the highest confidence. Thus, the previous observation that the horse and automobile classes may consistently be more confidently predicted than other classes has been verified. Other classes in the top five are ship, truck, and frog. The cat class has the lowest mean average predictive probability, which means that it is the class that is predicted on average with the least confidence. Finally, the previous observation that deer and bird may be more consistently predicted with medium or low confidence than other classes has also been verified because deer and bird are in the bottom four classes that are predicted on average with the highest confidence.

Average Predictive Variance

Similarly, the average predictive variance for each class is obtained and plotted as a function of predicted class. This plot is shown in Figure 5 of the Appendix. This plot, also with much fewer points, will help in clearly seeing which classes were, on average, predicted with higher - and lower - variances. There may be a correspondence between the average mean predictive probability and the average predictive variance.

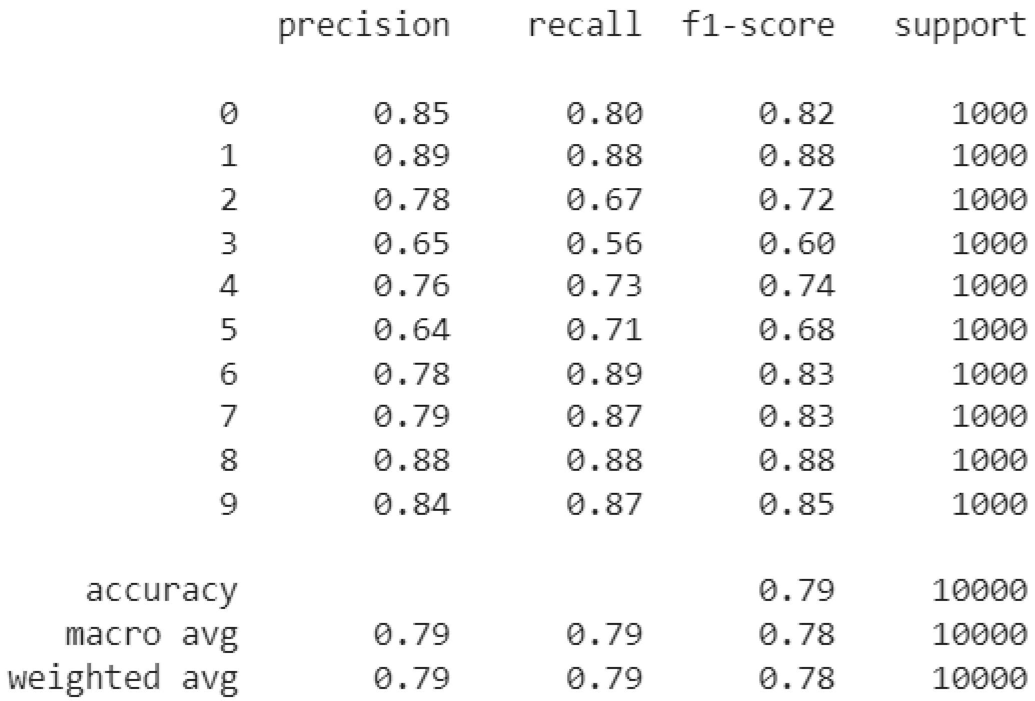
This plot is largely consistent with the previous plot of average mean predictive probability. The classes that have the lowest average mean predictive probabilities generally also have the highest average predictive variance. In the same vein, the classes that have the highest average mean predictive probabilities generally also have the lowest average predictive variance.

Future Improvements and Expansions

Several improvements and expansions can be made to this project:

- More layers in model: Generally, the greater the number of layers the model contains, the better its performance (assuming that potential overfitting is mitigated). If the computation time required for training large models is not a concern, the model may be enlarged with even more layers.
- Data augmentation: The given training and testing sets may be augmented using techniques like rotation such that there are more data available. Care should be taken such that the augmentation techniques do not distort the image such that no longer look realistic because one of the benefits of using CIFAR-10 is the fact that it contains real images, albeit with low resolution.
- Other datasets:
  - CIFAR-100 would be a natural choice of extension of this project, though because of the number of classes in that dataset (100), some of the code used in this project is not easily extensible without tedious extra work.
  - Another dataset that could be used is one that contains a combination of custom images and images from the Cornell Grasp Dataset, which can be found at https://www.kaggle.com/datasets/oneoneliu/cornell-grasp. Because of the nature of the task of robotic grasping, the objects shown in the images may be classified as circular and non-circular. These images and classes can then be used for image classification in much the same way as CIFAR-10 is used in this project.
- More sets of predictions: For uncertainty quantification, generating more sets of predictions is undoubtedly better, but higher computation time would be the tradeoff. If the higher computation time is not a concern, the number of sets of predictions generated may be set to much higher than ten.
- Other methods of uncertainty quantification: An alternative to the Monte-Carlo dropout used in this project is the method of deep ensembles, which is mentioned in the article https://medium.com/uncertainty-quantification-for-neural-networks/uncertainty-quantification-for-neural-networks-a2c5f3c1836d. In this method, multiple models with the same architecture are initialized with different weights and trained separately. Then, the testing predictions made by each of those models are averaged to obtain a single set of predictions. As the article states, the mean of the different sets of predictions and the corresponding variance are used for uncertainty quantification. Of course, this method will consume more computation time than Monte-Carlo dropout, so it may be a good technique to attempt if high computation times are not a concern.

Appendix



```
              precision    recall  f1-score   support

           0       0.85      0.80      0.82      1000
           1       0.89      0.88      0.88      1000
           2       0.78      0.67      0.72      1000
           3       0.65      0.56      0.60      1000
           4       0.76      0.73      0.74      1000
           5       0.64      0.71      0.68      1000
           6       0.78      0.89      0.83      1000
           7       0.79      0.87      0.83      1000
           8       0.88      0.88      0.88      1000
           9       0.84      0.87      0.85      1000

    accuracy                           0.79     10000
   macro avg       0.79      0.79      0.78     10000
weighted avg       0.79      0.79      0.78     10000
```

Figure 1: Classification Report

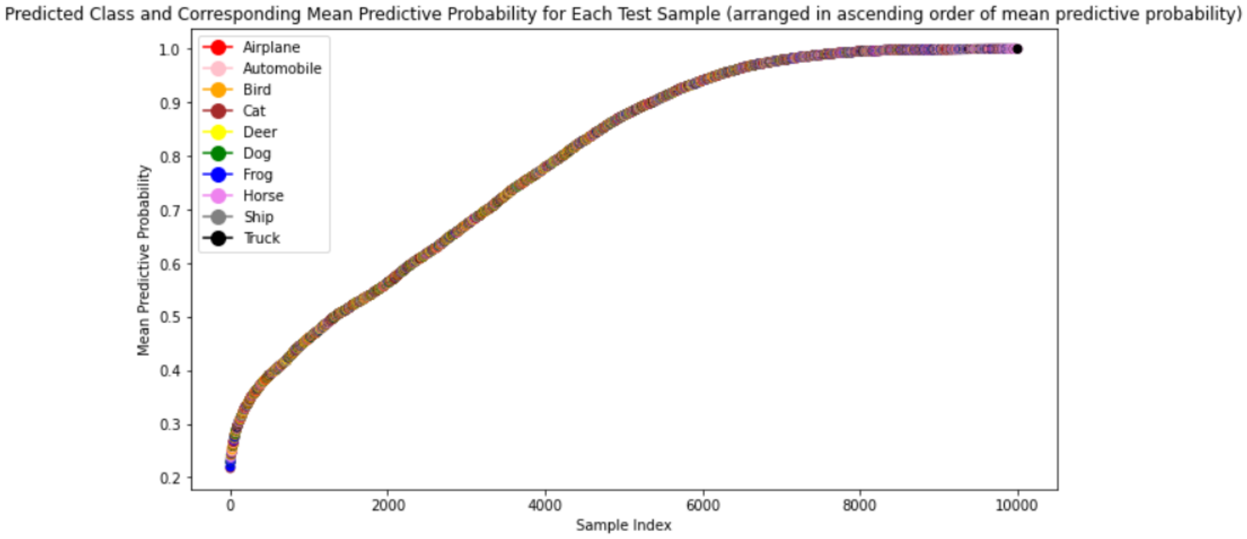


Figure 2: Plot of Mean Predictive Probability vs. Sample Index

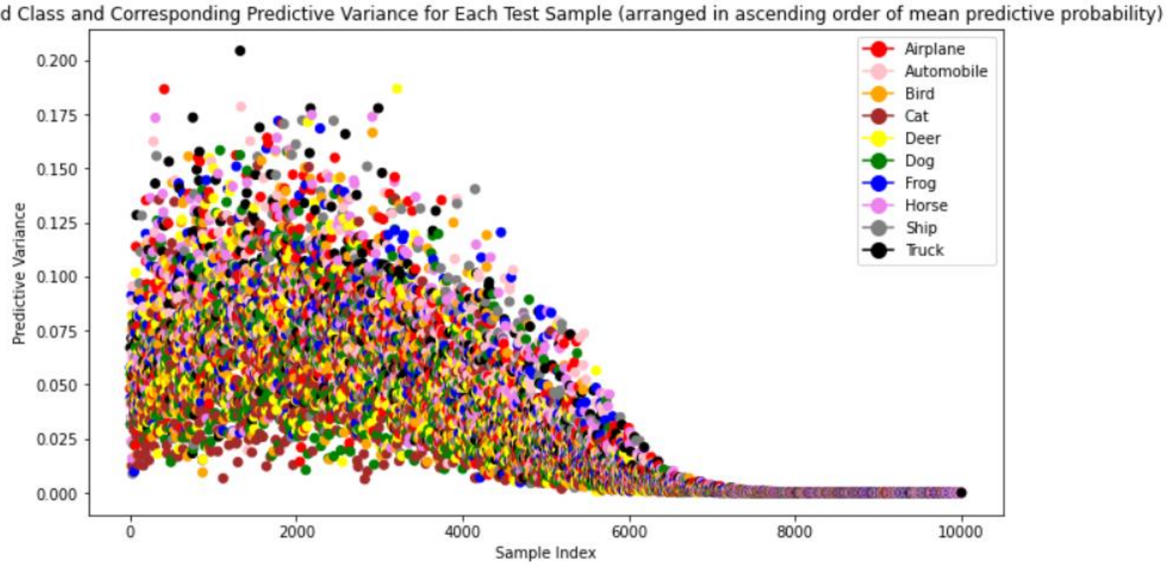


Figure 3: Plot of Predictive Variance vs. Sample Index

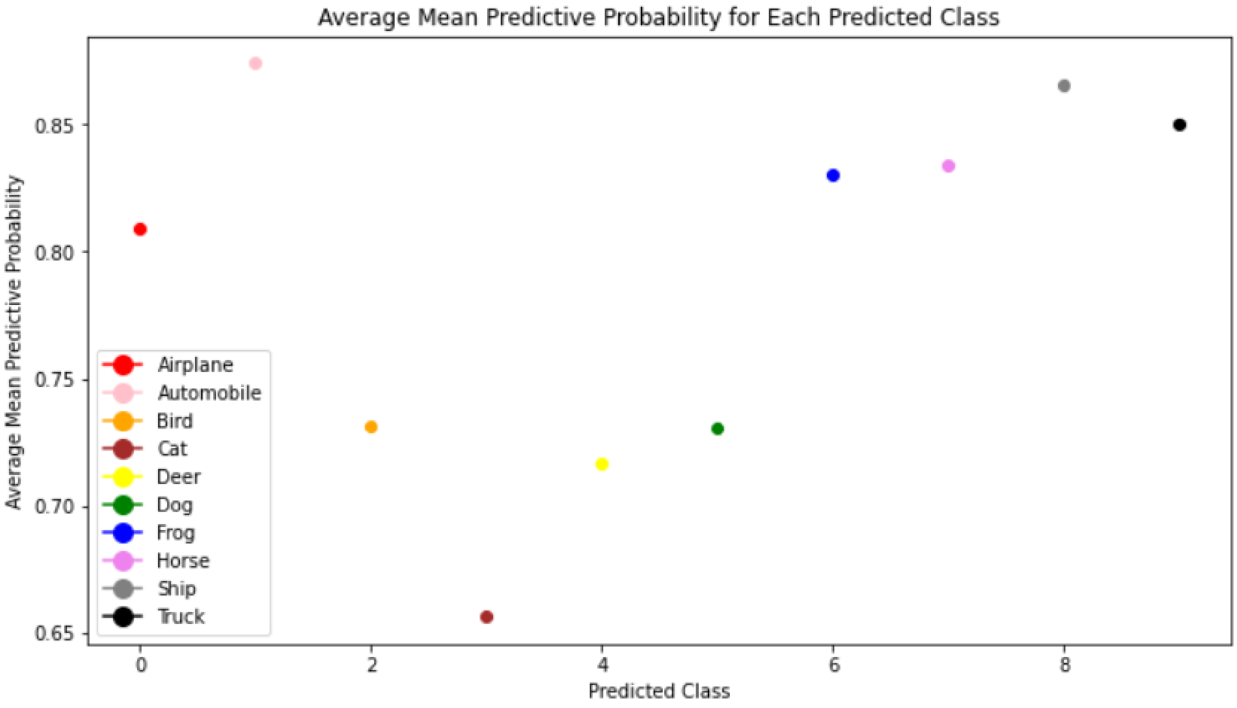


Figure 4: Plot of Average Mean Predictive Probability vs. Predicted Class

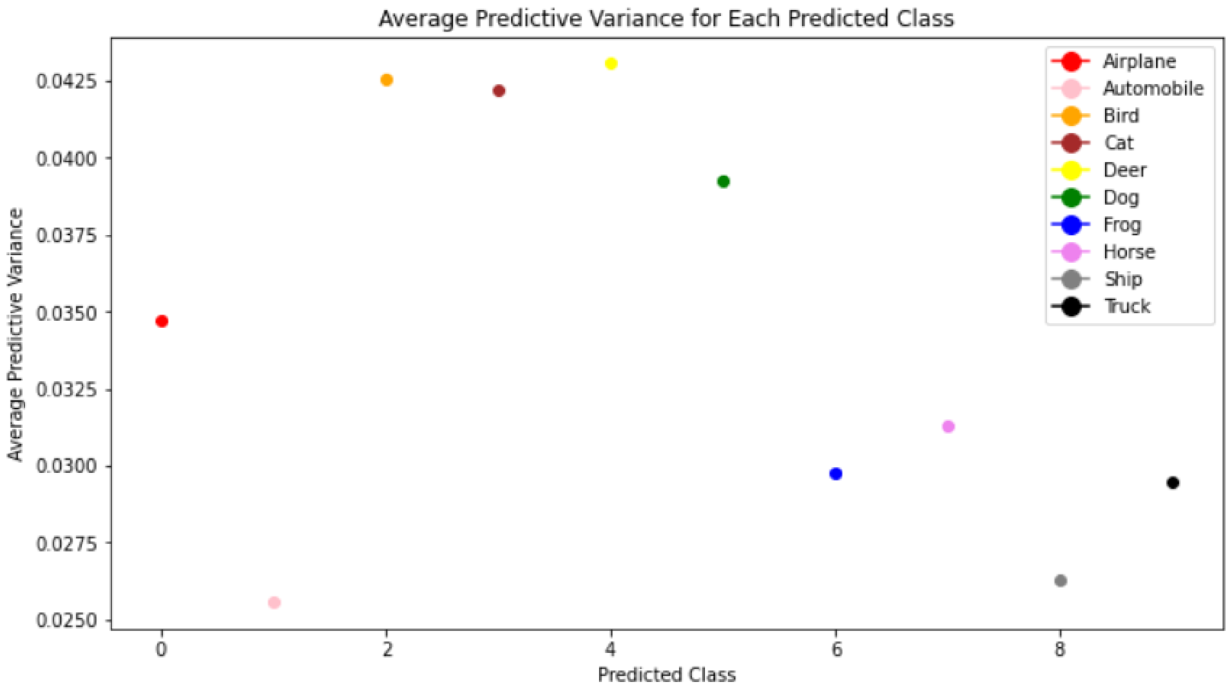


Figure 5: Plot of Average Predictive Variance vs. Predicted Class