## Week 1 – Bird Recognition in the City of Peacetopia

You are a famous researcher in the city of Peacetopia. The people of Peacetopia have a common characteristic: they are afraid of birds. To save them, you need to build an algorithm that will detect any bird flying over the city and alert the population.

The City Council has given you a dataset of 10,000,000 labelled images of the sky above Peacetopia, taken from the city’s security cameras. Your goal is to build an algorithm that can classify new images taken by security cameras.

There are many decisions to make, including:

* What is the evaluation metric?
* How are you going to divide the data into train, dev and test sets?

The City Council has told you that they want an algorithm that:

1. Has high accuracy.
2. Runs quickly and only takes a short time to classify a new image.
3. Can fit in a small amount of memory, so it can run in a small processor that the city will attach to many different security cameras.

**Metric of success**

1. You are delighted because this list of criteria will speed development and provide guidance on how to evaluate two different algorithms. True/False?
   1. False. (More than one metric expands the choices and trade-offs you need to decide for each with unknown effects on the other two.)
2. After further discussions, the city narrows down its criteria to:
   1. “We need an algorithm that can let us know a bird is flying over Peacetopia as accurately as possible.”
   2. We want the trained model to take no more than 10 seconds to classify a new image.”
   3. We want the model to fit in 10MB of memory.

Of the following models, which would you choose?

* 1. Test accuracy: 98%; Runtime: 9 seconds; Memory size: 9 MB. (This model has the highest test accuracy, the prominent criteria you are looking for, compared with other models, and has a runtime less than 10 seconds and memory size less than 10 MB.

1. The essential difference between an optimising and satisficing metric is the priority assigned by the stakeholders. True/False?
   1. False. (Satisficing metrics have thresholds for measurement and an optimising metric is unbounded.)

**Structuring your data**

1. Before implementing your algorithm, you need to split your data into train, dev and test sets. Which is the best choice?
   1. Train: 9,500,000; Dev: 250,000; Test: 250,000
2. After setting up your train, dev and test sets, the City Council comes across another 1,000,000 images, called the “citizens’ data”. Apparently, the citizens of Peacetopia are so scared of birds that they volunteered to take pictures of the sky and label them, thus contributing these additional 1,000,000 images. These images are different from the distribution of images the City Council had originally given you, but you think it could help your algorithm.

Notice that adding this additional data to the training set will make the distribution of the training set different from the distributions of the dev and test sets.

"You should not add the citizens' data to the training set, because if the training distribution is different from the dev and test sets, then this will not allow the model to perform well on the test set." True/False?

* 1. False. (Sometimes we need to train a model on the data that is available, and its distribution may not be the same as the data that will occur in production. Also, adding training data that differs from the dev set may still help the model improve performance on the dev set. What matters is that the dev and test sets have the same distribution.)

1. One member of the City Council knows a little about machine learning and thinks you should add the 1,000,000 citizen’s data images to the dev set. You object because:
   1. This would cause the dev and test set distributions to become different. This is a bad idea because you’re not aiming where you want to hit. (Adding a different distribution to the dev set will skew bias.)
   2. The dev set no longer reflects the distribution of data (security cameras) you most care about. (The performance of the model should be evaluated on the same distribution of images it will see in production.)
2. Human performance for identifying birds is less than 1%, training set error is 5.2% and dev set error is 7.3%. What is the best next step?
   1. Train a bigger network to drive down the more than 4% training error. (Avoidable bias is greater than 4.2%, which is larger than the 2.1% variance.)
3. You ask a few people to label the dataset to determine human-level performance. You find the following levels of accuracy:

|  |  |
| --- | --- |
| **Person** | **Error** |
| Bird watching expert #1 | 0.3% |
| Bird watching expert #2 | 0.5% |
| Normal person #1 | 1.0% |
| Normal person #2 | 1.2% |

If your goal is to have human-level performance be a proxy/estimate for Bayes error, how should you define human-level performance?

* 1. 0.3% (accuracy of expert #1)

1. Which statement do you agree with?
   1. A learning algorithm’s performance can be better than human-level performance, but it can never be better than Bayes error.
2. Which of the following best expresses how to evaluate the next steps in your project when results for human-level performance, train and dev set error are 0.1%, 2.0% and 2.1% respectively?
   1. Based on differences between the three levels of performance, prioritise actions to decrease bias and iterate.
3. You also evaluate your model on the test set and find the following:

|  |  |
| --- | --- |
| Human-level performance | 0.1% |
| Training set error | 2.0% |
| Dev set error | 2.1% |
| Test set error | 7.0% |

What does this mean?

* 1. You should try to get a larger dev set.
  2. You have overfit to the dev set.

1. After working on the project for a year, you achieve the following:

|  |  |
| --- | --- |
| Human-level performance | 0.1% |
| Training set error | 0.05% |
| Dev set error | 0.05% |

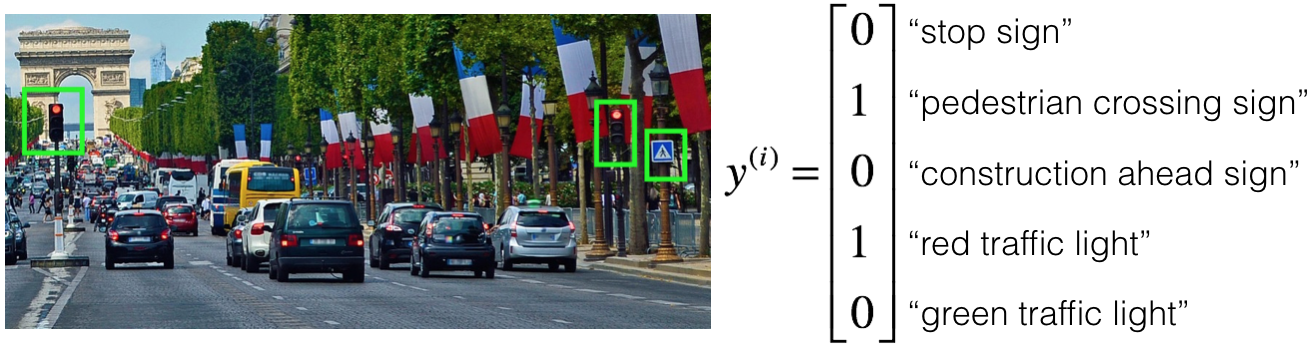
What can you conclude?

* 1. It is now harder to measure avoidable bias, which means progress will be slower going forward.
  2. If the test set is big enough for the 0.05% error estimate to be accurate, this implies Bayes error is less than or equal to 0.05.

1. It turns out Peacetopia has hired one of your competitors to build a system as well. Your system and your competitor both deliver systems with about the same running time and memory size, but your system is more accurate. However, when Peacetopia tries out your and your competitor’s systems, they like your competitor’s system better, because even though you have higher overall accuracy, you have more false negatives (failing to raise an alarm when a bird is in the air). What should you do?
   1. Rethink the appropriate metric for this task and ask your team to tune to the new metric.
2. You’ve handily beaten your competitor, and your system is now deployed in Peacetopia and is protecting the citizens from birds! But over the last few months, a new species of bird has been slowly migrating into the area, so the performance of your system is slowly degrading because your model is being tested on a new type of data. There are only 1,000 images of the new species and the city expects a better system from you within the next 3 months. What should you do first?
   1. Augment your data to increase the images of the new bird. (A sufficient number of images is necessary to account for the new species.)
3. The City Council thinks that having more cats in the city would help scare off birds. They are so happy with your work on the bird detector that they also hire you to build a cat detector. Because of years of working on cat detectors, you have such a large dataset of 100,000,000 cat images that training on this data takes about two weeks. Which of the statements do you agree with?
   1. Buying faster computes could speed up your teams’ iteration speed and thus their productivity.
   2. Needing two weeks to train will limit the speed at which you can iterate.
   3. If 100,000,000 examples is enough to build a quality cat detector, you might be better off training with just 10,000,000 images to gain an approximately 10 times improvement in how quickly you can run experiments, even if each model performs a little worse because it’s trained on less data.

## Week 2 – Autonomous Driving

You are employed by a startup building self-driving cars. You are in charge of detecting road signs (stop sign, pedestrian crossing sign, construction ahead sign) and traffic signals (red and green lights) in images. The goal is to recognize which of these objects appear in each image. As an example, the image below contains a pedestrian crossing sign and red traffic lights.



Your 100,000 labelled images are taken using the front-facing camera of your car. This is also the distribution of data you care most about doing well on. You think you might be able to get a much larger dataset off the internet, that could be helpful for training even if the distribution of internet data is not the same.

1. You are just getting started. What is the first thing you do assuming that all options would take about an equal amount of time.
   1. Spend a few days training a basic model and see what mistakes it makes (Applied ML is a highly iterative process. If you train a basic model and carry out error analysis (see what mistakes it makes), it will help point you in more promising directions.)
2. Your goal is to detect road signs (stop sign, pedestrian crossing sign, construction ahead sign) and traffic signals (red and green lights) in images. The goal is to recognize which of these objects appear in each image. You plan to use a deep neural network with ReLU units in the hidden layers.

Suppose that you use a sigmoid function for the output layer, and the output ​ has shape  
(5, 1). Which of the following best describes the cost function?

* 1. (Here we compare each component of the prediction with the respective component of the label , and sum over the individual losses.)

1. You are working out error analysis and counting up what errors the algorithm makes. Which of the following do you think you should manually go through and carefully examine, one image at a time?
   1. 500 images of the dev set, on which the algorithm made a mistake. (We focus on images that the algorithm got wrong from the dev set. That is the one we use to make choices between different iterations of the system.)
2. After working on the data for several weeks, your team ends up with the following data:
   1. 100,000 labeled images taken using the front-facing camera of your car.
   2. 900,000 labeled images of roads downloaded from the internet.
   3. Each image’s labels precisely indicate the presence of any specific road signs and traffic signals or combinations of them. For example, means the image contains a stop sign and a red traffic light.

When using a non fully-labelled image such as , which of the following strategies is most appropriate to calculate the loss function to train as a multi-task learning problem?

* 1. Calculate the loss as where the sum goes over all the known components of .

1. The distribution of data you care about contains images from your car’s front-facing camera, which comes from a different distribution than the images you were able to find and download off the internet. Which of the following are true about the train/dev/test split?
   1. The dev and test set must come from the front-facing camera. (This is the distribution we care about most, thus it should be used as a target).
   2. The dev and test sets must come from the same distribution. (This is required to aim the target where we want to be.)
2. Assuming you have the following data splits and algorithm errors:

|  |  |  |
| --- | --- | --- |
| **Dataset** | **Contains** | **Error** |
| Training | 940,000 images randomly picked from (900,000 internet images plus 60,000 front-facing camera images) | 8.8% |
| Training-Dev | 20,000 images randomly picked from (900,000 internet images plus 60,000 front-facing camera images) | 9.1% |
| Dev | 20,000 front-facing camera images | 14.3% |
| Test | 20,000 front-facing camera images | 14.8% |

You also know that human-level error on the road sign and traffic signals classification task is around 0.5%. Which of the following are true?

* 1. You have a large avoidable-bias problem because your training error is quite a bit higher than the human-level error.
  2. You have a large data-mismatch problem because your model does a lot better on the training-dev set than on the dev set.

1. Assuming you have the following data splits and algorithm errors:

|  |  |  |
| --- | --- | --- |
| **Dataset** | **Contains** | **Error** |
| Training | 940,000 images randomly picked from (900,000 internet images plus 60,000 front-facing camera images) | 2.0% |
| Training-Dev | 20,000 images randomly picked from (900,000 internet images plus 60,000 front-facing camera images) | 2.3% |
| Dev | 20,000 front-facing camera images | 1.3% |
| Test | 20,000 front-facing camera images | 1.1% |

You also know that human-level error on the road sign and traffic signals classification task is around 0.5%. A friend thinks that the training data distribution is much harder than the dev/test set distribution. What do you think?

* 1. Your friend is probably right (i.e. Bayes error for the dev/test distribution is probably lower than for the train distribution.) (Since the training-dev error is higher than the dev and test errors, the dev-test distribution is probably “easier” than the training distribution.)

1. You decide to focus on the dev set and check by hand what the errors are due to. The table summarises your findings.

|  |  |
| --- | --- |
| Overall dev set error | 15.3% |
| Errors due to incorrectly labelled data | 4.1% |
| Errors due to foggy pictures | 3.0% |
| Errors due to partially-occluded elements | 7.2% |
| Errors due to other causes | 1.0% |

In this table, 4.1%, 7.2%, etc. are a fraction of the total dev set (not just examples of your algorithm mislabelled). For example, about 7.2/15.3 = 47% of your errors are due to partially occluded elements.

You shouldn't invest all your efforts to get more images with partially occluded elements since 4.1 + 3.0 + 1.0 = 8.1 > 7.2. True/False?

* 1. False. (These kinds of arguments don't help us to decide on the strategy to follow. Other factors should be used, such as the trade-off between the cost of getting new images and the improvement of the system performance.)

1. You decide to focus on the dev set and check by hand what the errors are due to. Here is a table summarizing your discoveries:

|  |  |
| --- | --- |
| Overall dev set error | 15.3% |
| Errors due to incorrectly labelled data | 4.1% |
| Errors due to foggy pictures | 3.0% |
| Errors due to partially-occluded elements | 7.2% |
| Errors due to other causes | 1.0% |

In this table, 4.1%, 7.2%, etc. are a fraction of the total dev set (not just examples of your algorithm mislabelled). For example, about 7.2/15.3 = 47% of your errors are due to partially occluded elements.

From this table, we can conclude that if we fix the incorrectly labelled data, we will reduce the overall dev set error to 11.2%. True/False?

* 1. False. (The 4.1% only gives you an estimate of the ceiling of how much the error can be improved by fixing the labels.)

1. You decide to use data augmentation to address foggy images. You find 1,000 pictures of fog off the internet, and “add” them to clean images to synthesize foggy days, like this:



Which of the following statements do you agree with?

* 1. So long as the synthesised fog looks realistic to the human eye, you can be confident that the synthesised data is accurately capturing the distribution of real foggy images (or a subset of it), since human vision is very accurate for the problem you’re solving. (If the synthesised images look realistic, then the model will just see them as if you had added useful data to identify road signs and traffic signals in foggy weather. It will very likely help.)

1. After working further on the problem, you’ve decided to correct the incorrectly labelled data. Your team corrects the labels of the wrongly predicted images on the dev set. What’s the next necessary step in the process?
   1. Correct the labels of the test set. (Recall that the dev set and the test set must come from the same distribution.)
2. So far your algorithm only recognizes red and green traffic lights. One of your colleagues in the start-up is starting to work on recognizing a yellow traffic light. Images containing yellow lights are quite rare, and she doesn’t have enough data to build a good model. She hopes you can help her out using transfer learning.

What do you tell your colleague?

* 1. She should try using weights pre-trained on your dataset, and fine-tune further with the yellow light dataset. (You have trained your model on a huge dataset, and she has a small dataset. Although your labels are different, the parameters of your model have been trained to recognize many characteristics of road and traffic images which will be useful for her problem. This is a perfect case for transfer learning, she can start with a model with the same architecture as yours, change what is after the last hidden layer and initialize it with your trained parameters.)

1. One of your colleagues at the start-up is starting a project to classify stop signs in the road as speed limit signs or not. He has approximately 30,000 examples of each image and 30,000 images without a sign. He thought of using your model and applying transfer learning but then he noticed that you use multi-task learning, hence he can't use your model. True/False?
   1. False. (When using transfer learning we can remove the last layer. That is one of the aspects that is different from a binary classification problem.)
2. When building a system to detect cattle crossing a road from images taken with the front-facing camera of a truck, the designers had a large dataset of images. Which of the following might be a reason to use an end-to-end approach?
   1. There is a large dataset available. (To get good results when using an end-to-end approach, it is necessary to have a big dataset.)
3. An end-to-end approach doesn't require that we hand-design useful features, it only requires a large enough model. True/False?
   1. True. (This is one of the major characteristics of deep learning models, that we don't need to hand-design the features.)