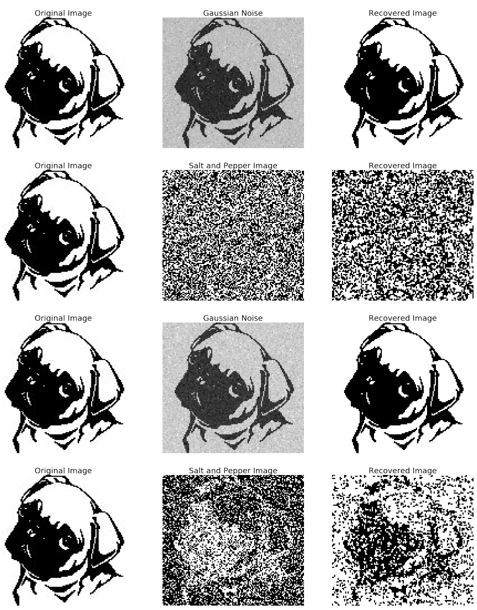
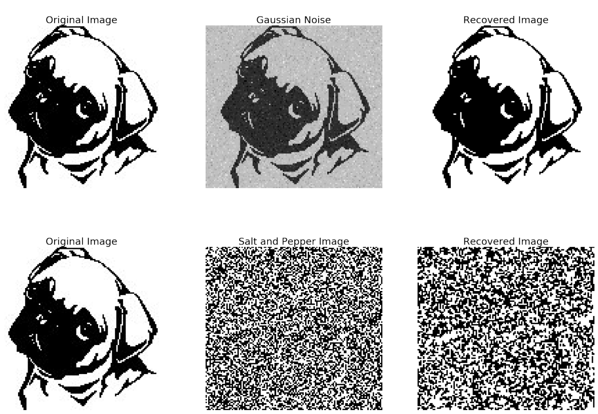
Inference

December 1, 2017

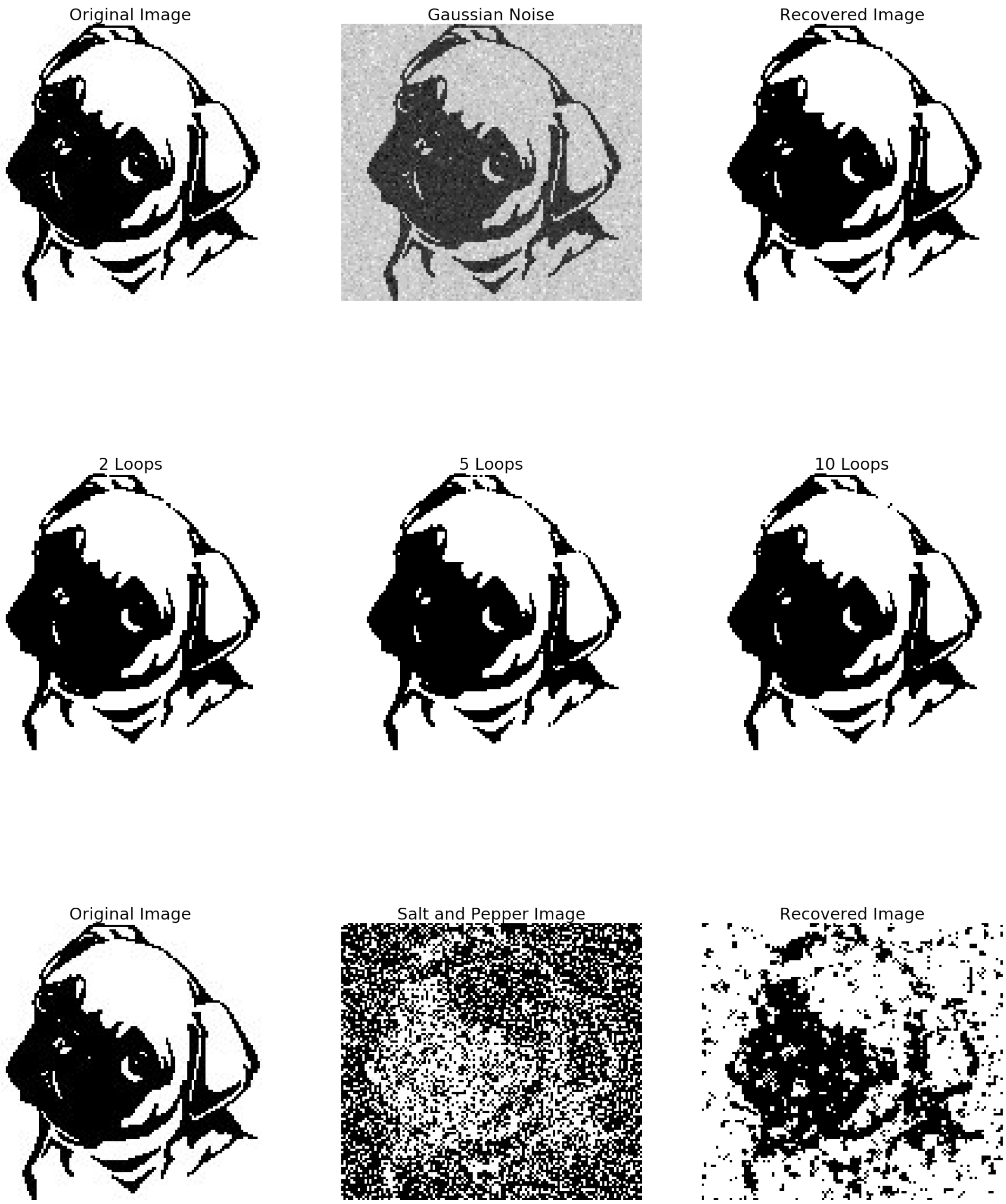
Q1



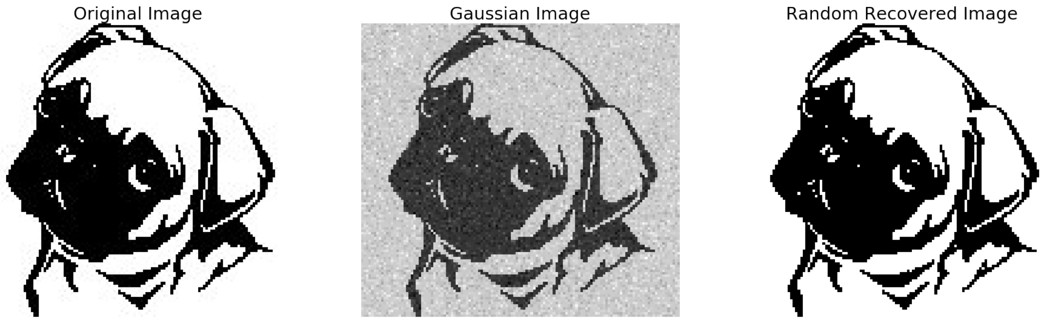


It takes 2 and 5 iterations respectively. For the salt and pepper image, this clearly still isn't a "decent" image, but for the gaussian, the image is very close to the original.

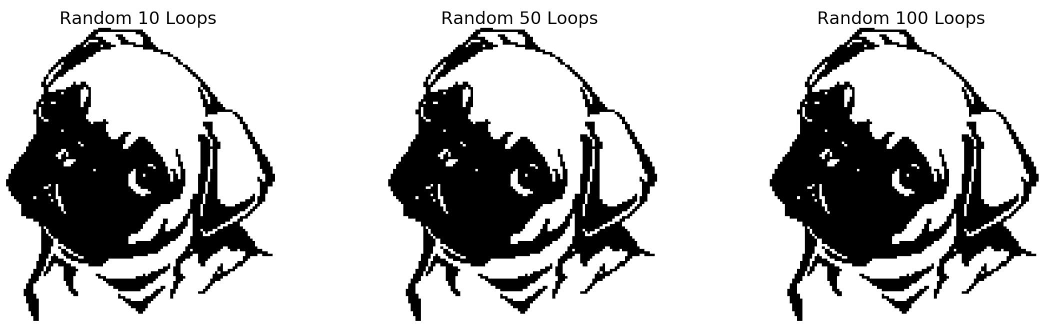
Q2



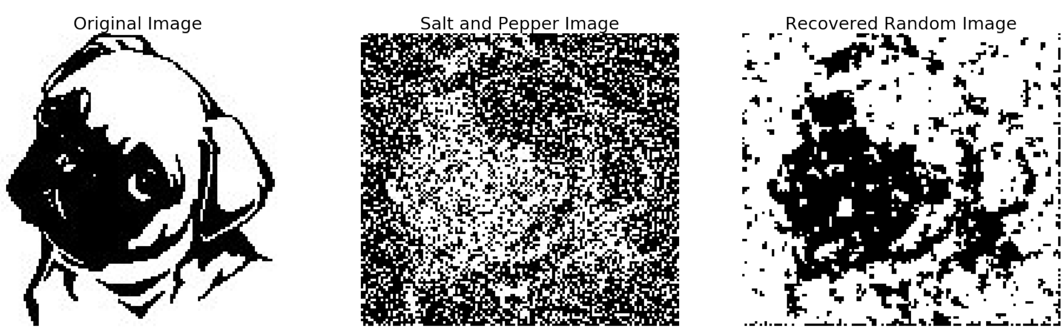

```
Number of iterations:  1
```




```
Number of iterations:  10
Number of iterations:  50
Number of iterations:  100
```




```
Number of iterations:  500000
```



Q3

When looping randomly, the gaussian noise image seems to near enough recover the image perfectly in what appears to be one iteration, whilst the S&P image seems to peak in quality at around the 500,000 iteration points. In this case, iterations are how many pixels are being updated.

Q4

As stated above, the image does not always improve, particularly with the salt and pepper image, after an optimal number of iterations, the image declines back into a less-recognisable mess. The gaussian example shows detrimental effects from looping more than a nice amount of times, as the image quality decreases the more loops are applied.

Q5

To reason about the difference between $KL(q(x)||p(x))$ and $KL(p(x)||q(x))$, we'll consider the specific case where we fix $p(x) = 0, q(x) > 0$. Using the definition of the KL Divergence, $KL(q(x)||p(x)) = \int q(x) log \frac{p(x)}{q(x)}$ we can compute that

$$\lim_{p(x) \to 0} q(x) log \frac{p(x)}{q(x)} = -\infty$$

This would mean that if there is some region where we assign probability mass in $q(x)$ that $p(x)$ assigns no probability mass to, then the KL divergence will be of magnitude ∞.

On the otherhand, if we fix $p(x), q(x)$ as before, and calculate $KL(p(x)||q(x))$ we get a very different result.

$$\lim_{p(x) \to 0} p(x) log \frac{q(x)}{p(x)} = 0$$

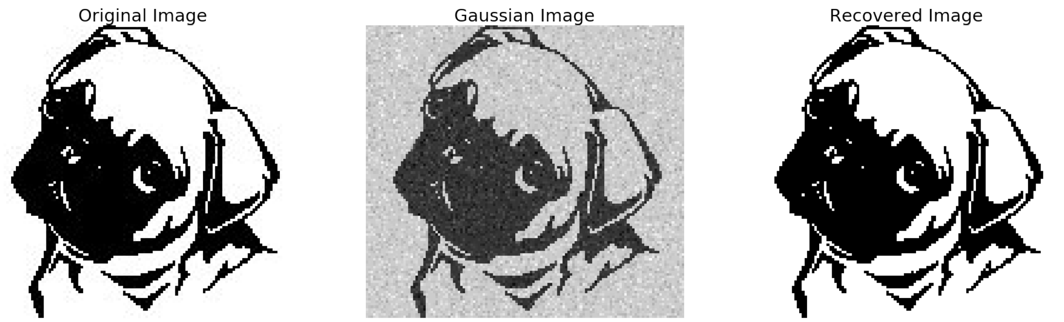
What this suggests is that if there is probability mass at q(x), but none at p(x), it doesn't make a difference to the disimilarity. Clearly we should only be interested in either $KL(q(x)||p(x))$ or $KL(p(x)||q(x))$ since the two (at least in this extreme case) tell us different information.

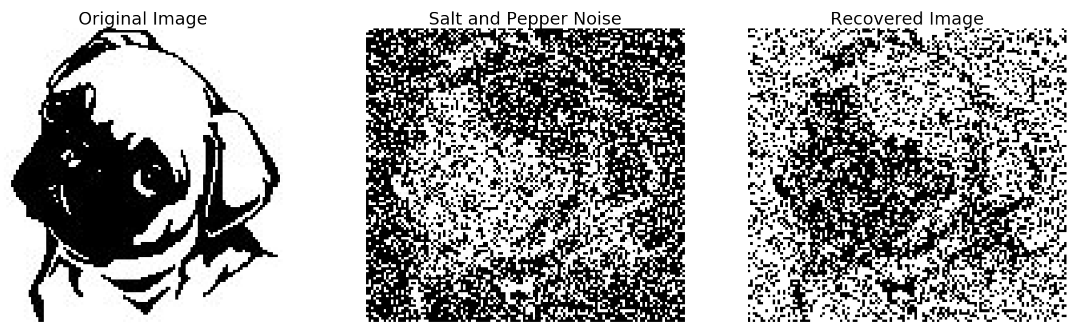
Let's say $q(x)$ is a distribution based off emperical evidence and $p(x)$ is our model and we fix the values as before, then $KL(q(x)||p(x)) = \infty$. Intuitively this makes sense, if a region in the model has 0 probability mass and yet the evidence places probability mass there, it seems sensible to say the model is disimilar to the evidence and that it should be rejected. The alternative, makes little sense when treating $p(x)$ as a model and q(x) as evidence. We saw before that the evidence $q(x)$ placing probability mass where there is none in the model p(x) doesn't increase (or change) the KL divergence which seems wrong.

Taking that $KL(q(x)||p(x))$ is more useful for our purposes, we can also look at the example where $p(x) > 0, q(x) = 0$. In that case, if we took that case then we have $\lim_{q(x) \to 0} q(x) log \frac{p(x)}{q(x)} = 0$. This also makes sense, just because none of our evidence resulted in probability mass at that point, it doesn't mean our model is wrong.

A way you can think of this is if we made the statement "all pugs have red collars", the observation of just one pug without a red collar refutes that statement (ie the case where $KL(q(x)||p(x)) = \infty$). On the otherhand, if we made the statement that "some pugs wear red collars" and didn't see any wearing red collars, we can't refute that statement (ie the case where $KL(q(x)||p(x)) = 0$).

Q6





Q7

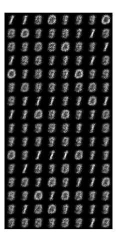
The variational bayes method, as with the others, produces a good quality retrieval of the gaussian image, but unlike the others, produces a fairly cleaner S&P image.

Q8

Q9

The VAE model has two parts, an encoding part and decoding part. The encoding step uses a neural network to approximate the function that can "compress" the data into a latent variable (z in the given code). The decoding step makes use of the parameters learned by the encoding to take an input that represents the desired image as its latent variable and returns the image. The special feature of a Variational Autoencoder is that the latent variable is constrained to be a Gaussian distribution. This factors into the loss function as it takes into account both the KL divergence of the latent variable distribution and a unit Gaussian, and the difference between the original image and generated image. It also makes use of the reparameterization trick and represents the latent variable as the mean and standard deviation of it's distribution instead of real values so that it can optimise the model using gradient descent. This is useful in image denoising as we could potentially use the encoding step to get the latent variable representation of the noisy image and then use the decoder to generate a denoised image. The difference between VAE and Variational Bayes is that VAE can learn to generate different images based on the different latent variables.

Q10



Images are generated by the VAE after 0, 50 and 100 iterations respectively