**Variational Auto Encoder**

For this project, we have decided to use Variational Auto Encoders for the generation of novel molecules.

**Inputs**

The Inputs to the network consists of the Vector representations of molecules smiles, generated by the ChemBERTa model, as described previously.

**Encoder**

The encoder currently consists of dense layers followed by Batch Normalisation, with each dense layer decreasing in size to compress the input. The batch normalisation improves training stability and accelerates convergence. It addresses the covariate shift problem. The covariate shift problem is the change in distribution of input values to intermediate layers during training. In later stages of experimentation, I will test different depths, types and lengths of hidden layers.

**Latent Dimension**

The encoder produces a compressed representation of the input, this is called the latent dimension. This is also the stage where we input the conditions that effect the generated molecule. During the training process the conditions are just the current states of the molecule. When I have access to the full dataset I will be able to give a full description of the conditions used and their chemical significance. The conditions are concatenated with the latent dimension and passed into the decoder.

**Decoder**

The decoder consists of 1 Dense layer followed by multiple Convolutional Transposing Layers upscaling the image to generate an output vector.

**Output**

This vector can then be converted into an image and the loss can be calculated, allowing the model weights to be updated.

**Hyperparameters/Other information**

These values are subject to change for experimentation purposes.

Loss Functions : Adam Optimiser (Adaptive Moment Estimation) with Mean Squared error and KL Loss

Input Size : 768

Output Size : 400x400 pixels (160000)

Batch Size : 128

Condition Size : 10

Latent Dimension : 128

Learning Rate : 0.0001

After training the outputs are displayed to the user.

The next stage of the development process is to create a description profile of these molecules and their difference to the starting molecules. Then, following this, we need to simulate the molecules and rank them on their performance for there given targets. For this we may use QSAR – Quantitative Structure-Activity Relationships.