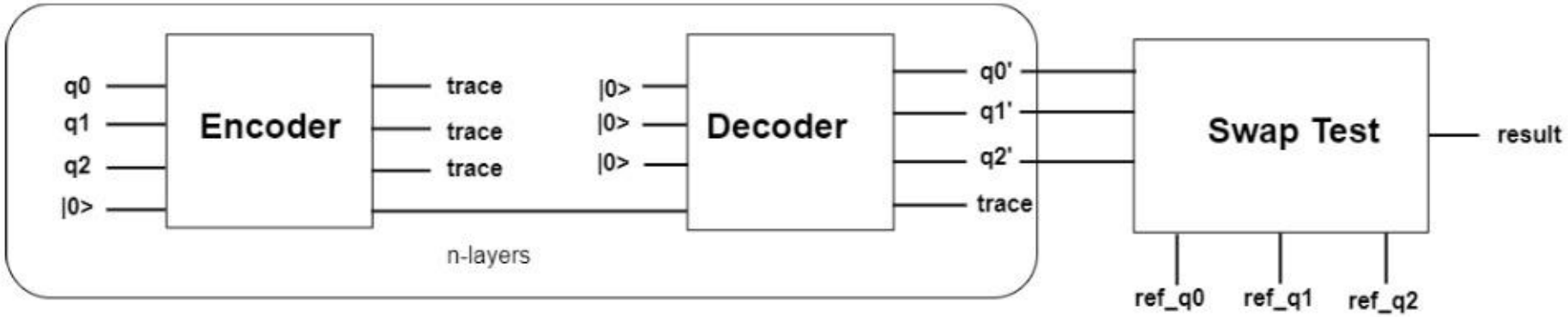
PROGRESS

Three Qubit QEC General Structure



Implementation

- Based from Owen Lockwood's Quantum Autoencoder implementation: https://github.com/lockwo/quantum_computation/tree/master/TFQ/QAE

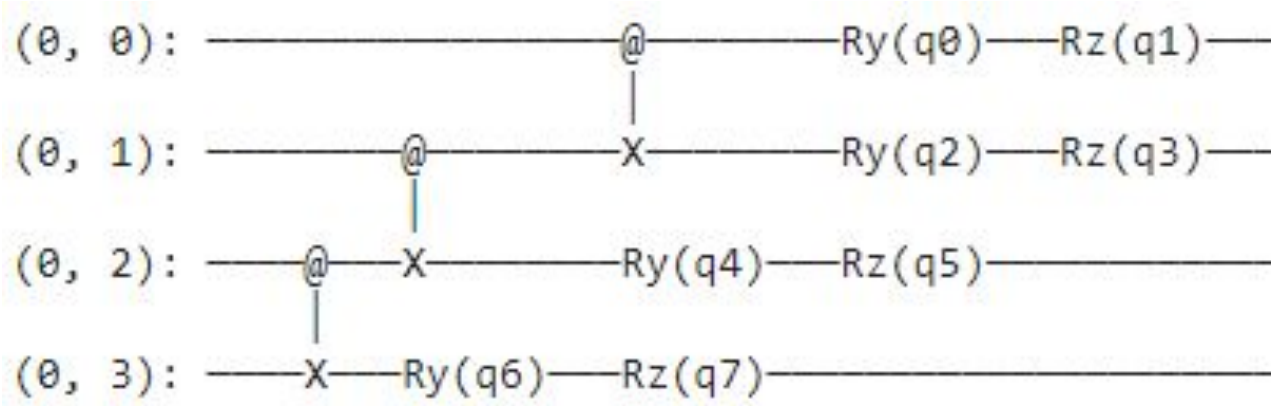
  https://www.youtube.com/watch?v=ju4B0t25Ky8

- Tried different ansatz, number of layers, optimizers, epochs and batch sizes
- Training and test is done with |0>, |1> and |+> (just one state) states for three qubits.
  - |0>: Errornous and legit statuses
    - **Legit:** |000>
    - **Errornous:** |001>, |010>, |100>
  - |1>: Errornous and legit statuses
    - **Legit:** |111>
    - **Errornous:** |110>, |101>, |011>
  - |+>: Just the legit status
    - (|000>+|111>)/sqrt(2)
- Loss function -> Mean Absolute Percentage Error

- Training and test outputs are done with results of Swap Test.
  - Swap Test is used to compare two quantum states. The probability of getting zero as the result of this test increases as two states get similar.
- We checked training and test loss results for validation.
- For tests with that uses just one layer, epoch size is set to 100 or 200. These runs aborted far more earlier than epoch sizes. Also, the history of them shows that values do not change much. Due to these reasons and seeing gradient calculation takes too much time, number of epochs kept as 5 for other number of layers. This value is enough for comparison.
- Size of training and test sets are 9000 and 1000. All of them are randomly selected from |0>, |1> and |+>.

SETUP 1

- **Ansatz:** this is what Owen Lockwood used



Results

Run Num	Number of Layers	Optimizers	Gradient Calculator	Epochs	Batch Size	Final Training Loss(%)	Final Test Loss(%)
1	1	Adam(learning_rate=0.1)	ForwardDifference	12	10	60.5842	64.2502
2	3	Adam(learning_rate=0.1)	ForwardDifference	5	10	19.2453	20.5269
3	7	Adam(learning_rate=0.1)	ForwardDifference	5	10	4.9924	5.0051

Results contd.

Run Num	Number of Layers	Optimizers	Gradient Calculator	Epochs	Batch Size	Final Training Loss(%)	Final Test Loss(%)
4	1	Adam(learning_rate=0.1)	ForwardDifference	13	3	62.0938	59.0006
5	3	Adam(learning_rate=0.1)	ForwardDifference	5	3	20.2496	18.6794
6	7	Adam(learning_rate=0.1)	ForwardDifference	5	3	6.4826	7.7014

Results contd.

Run Num	Number of Layers	Optimizers	Gradient Calculator	Epochs	Batch Size	Final Training Loss	Final Test Loss
7	1	Nadam(learning_rate=0.1, beta_1=0.9, beta_2=0.999)	ForwardDifference	11	3	67.1428	64.9002
8	3	Nadam(learning_rate=0.1, beta_1=0.9, beta_2=0.999)	ForwardDifference	5	3	19.9684	18.6347

Results contd.

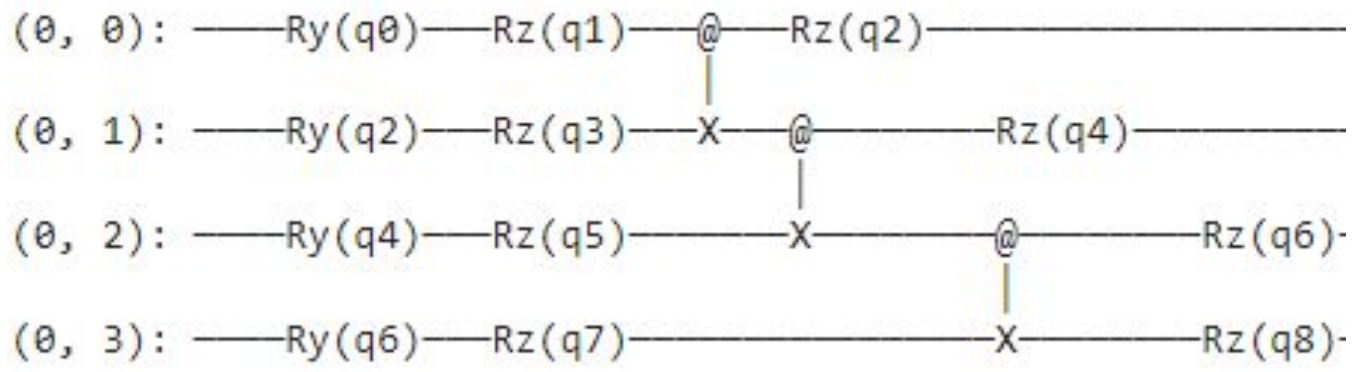
Run Num	Number of Layers	Optimizers	Gradient Calculator	Epochs	Batch Size	Final Training Loss	Final Test Loss
9	7	Nadam(learning_rate=0.1, beta_1=0.9, beta_2=0.999)	ForwardDifference	5	3	5.8952	5.2210
10	1	Nadam(learning_rate=0.1, beta_1=0.9, beta_2=0.999)	ForwardDifference	11	10	61.4717	61.0500

Results contd.

Run Num	Number of Layers	Optimizers	Gradient Calculator	Epochs	Batch Size	Final Training Loss	Final Test Loss
11	3	Nadam(learning_rate=0.1, beta_1=0.9, beta_2=0.999)	ForwardDifference	5	10	19.3736	18.8781
12	7	Nadam(learning_rate=0.1, beta_1=0.9, beta_2=0.999)	ForwardDifference	5	10	5.5875	4.9944

SETUP 2

- **Ansatz:** this is what authors of the paper pointed to us. We skipped some tests because from the first setup we learnt that they do not make too much difference.



Results

Run Num	Number of Layers	Optimizers	Gradient Calculator	Epochs	Batch Size	Final Training Loss	Final Test Loss
1	1	Adam(learning_rate=0.1)	ForwardDifference	15	10	27.4645	28.3505
2	3	Adam(learning_rate=0.1)	ForwardDifference	5	10	15.5248	14.4815
3	7	Adam(learning_rate=0.1)	ForwardDifference	5	10	5.5472	5.4367

Results contd.

Run Num	Number of Layers	Optimizers	Gradient Calculator	Epochs	Batch Size	Final Training Loss	Final Test Loss
4	1	Nadam(learning_rate=0.1, beta_1=0.9, beta_2=0.999)	ForwardDifference	16	3	27.8800	25.7071
5	3	Nadam(learning_rate=0.1, beta_1=0.9, beta_2=0.999)	ForwardDifference	5	3	20.0302	20.1113

Results contd.

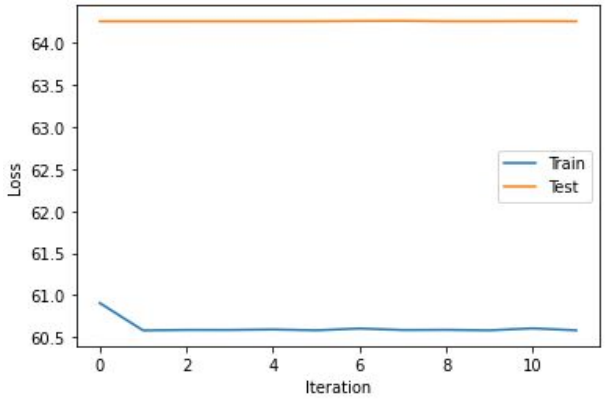
Run Num	Number of Layers	Optimizers	Gradient Calculator	Epochs	Batch Size	Final Training Loss	Final Test Loss
6	7	Nadam(learning_rate=0.1, beta_1=0.9, beta_2=0.999)	ForwardDifference	5	3	6.1238	5.6381

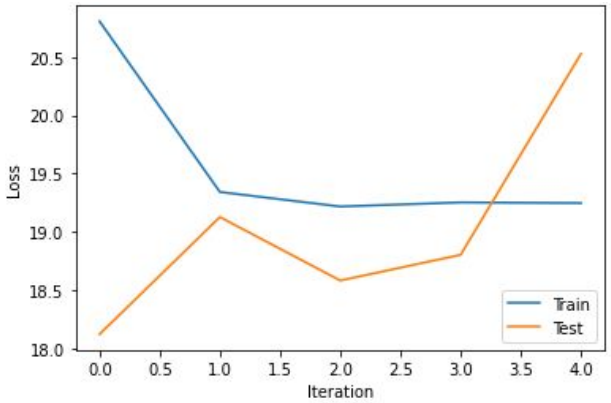
Interpretation

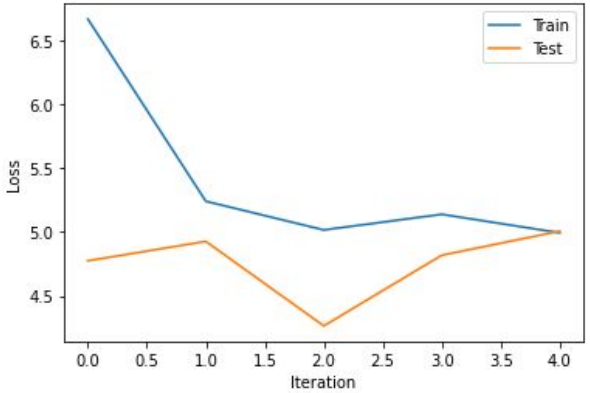
- For smaller number of layers, there is a huge difference between the first setup and the second one. This gap closes fastly as we increase number of layers. This result may also imply that there is a minimum loss that does not change no matter what setup and parameters are used.
- Results do not change much as other parameters except number of batches change within a setup.
- For each run, loss values do not change much after approximately 5 runs.

Appendix A - Loss History for the Setup 1

- Test Run 1, 2 and 3

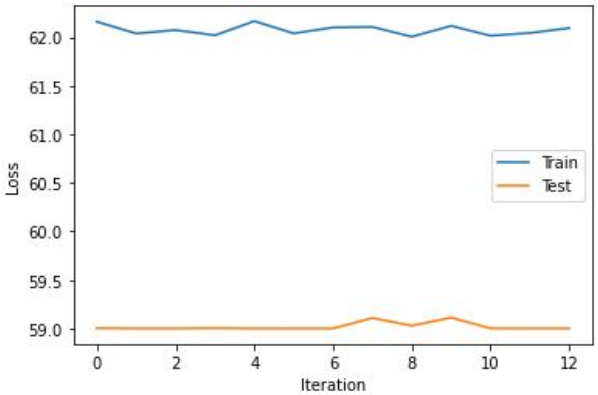


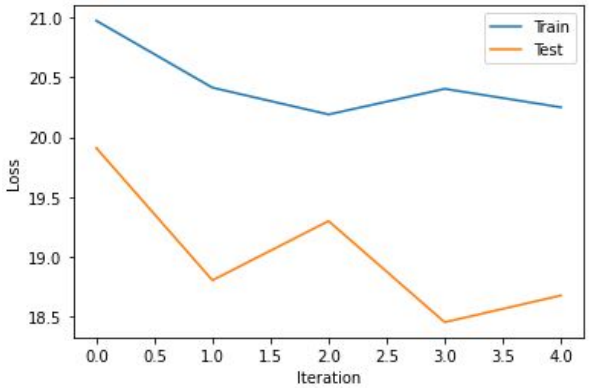


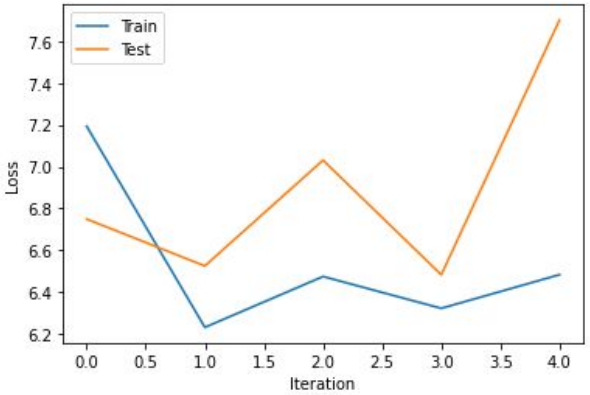


Appendix A - Loss History for the Setup 1 contd.

- Test Run 4, 5 and 6

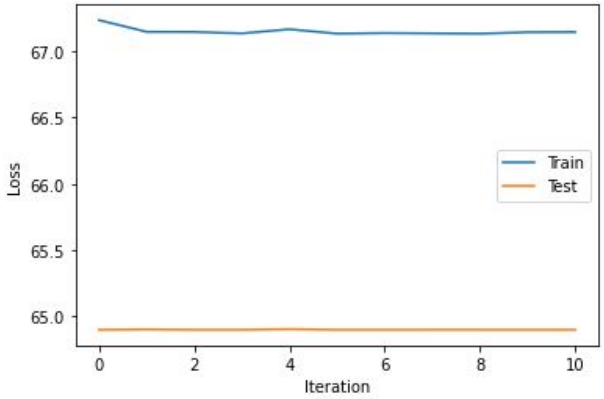


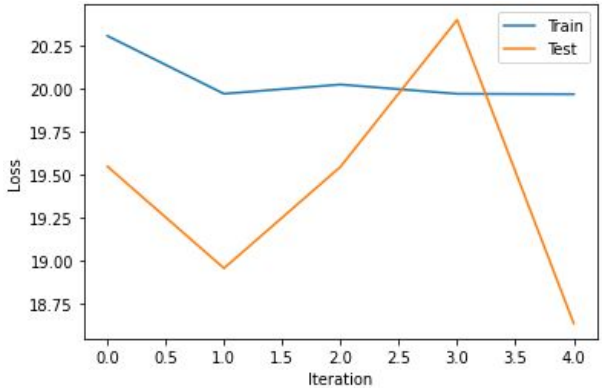


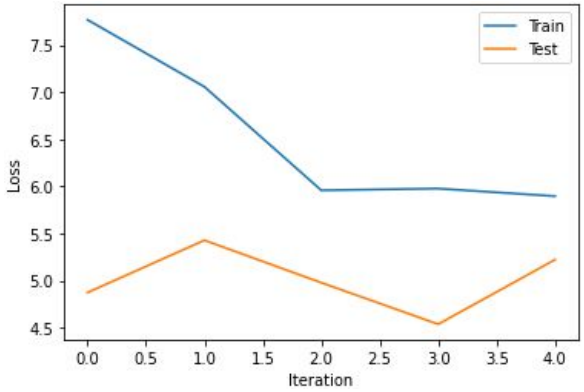


Appendix A - Loss History for the Setup 1 contd.

- Test Run 7, 8 and 9

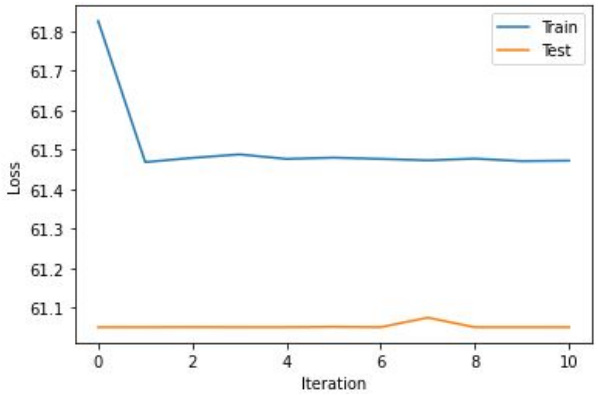


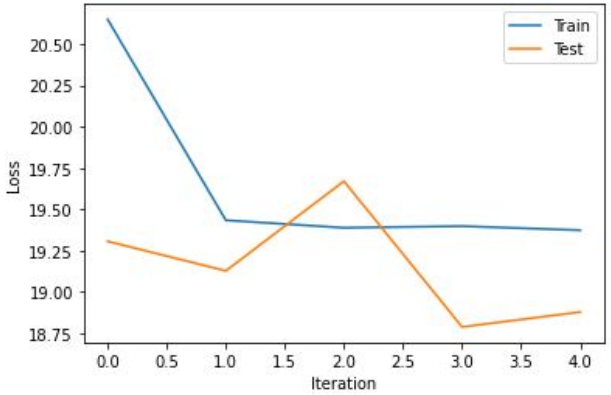


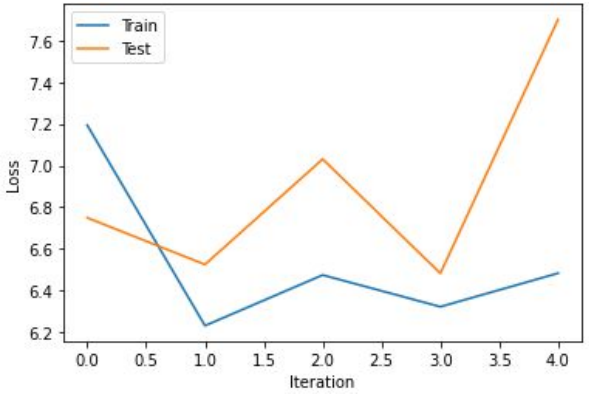


Appendix A - Loss History for the Setup 1 contd.

- Test Run 10, 11 and 12

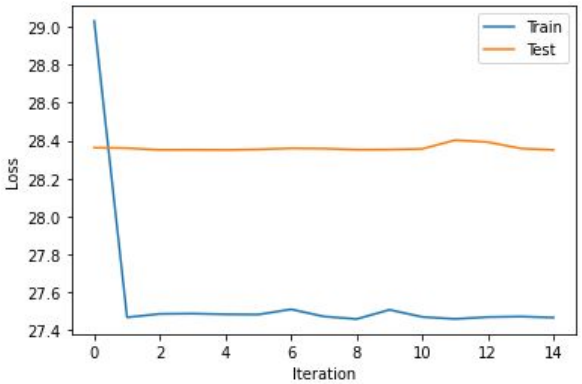


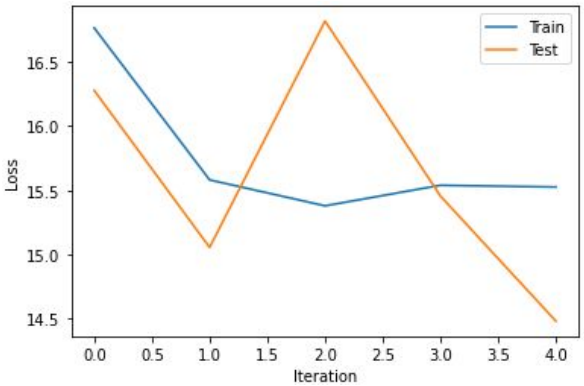


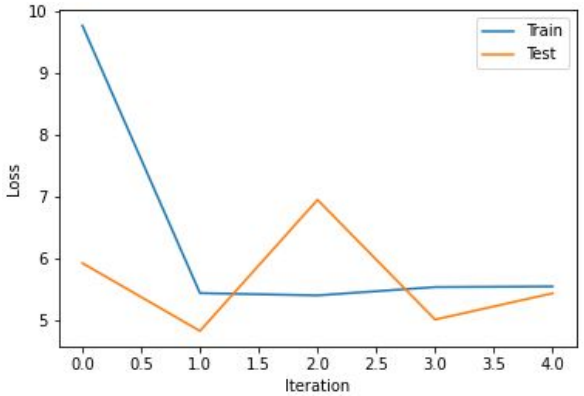


Appendix B - Loss History for the Setup 2

- Test Run 1, 2 and 3







Appendix B - Loss History for the Setup 2 contd.

- Test Run 4, 5 and 6