COVID-19 Prediction Model Report

1. Methodology

1.1 Dataset

The model uses a custom `COVIDDataset` class to load and preprocess the data. Key features of the dataset include:

- Data is loaded from CSV files for training, validation, and testing.
- The dataset can be configured to use all features or a selected subset of features.
- Features are normalized using mean centering and standard deviation scaling.
- The dataset is split into training, validation, and test sets.

1.2 Network Structure

The model uses a simple neural network architecture implemented in the `SimpleNN` class:

```python
class SimpleNN(nn.Module):
  def __init__(self, input_dim):
    super(SimpleNN, self).__init__()
    self.network = nn.Sequential(
      nn.Linear(input_dim, 32),
      nn.BatchNorm1d(32),
      nn.Dropout(p=0.2),
      nn.LeakyReLU(),
      nn.Linear(32, 1)
    )
    self.loss_function = nn.MSELoss()
```

Key components of the network:

- Input layer: Accepts input with dimension `input_dim`
- Hidden layer: 32 neurons with LeakyReLU activation
- Batch Normalization: Applied after the first linear layer
- Dropout: 20% dropout rate for regularization
- Output layer: Single neuron for regression output

1.3 Training Process

The training process is implemented in the `train_model` function:

- Optimizer: Configurable, default is Adam
- Loss function: Mean Squared Error (MSE) with L2 regularization
- Early stopping: Training stops if no improvement is seen for a specified number of epochs
- Model saving: The best model (based on validation loss) is saved during training

1.4 Hyperparameters

The main hyperparameters used in the model:

```python
config = {
   'n_epochs': 1000,
   'batch_size': 200,
   'optimizer': 'Adam',
   'optim_hparas': {},
   'early_stop': 500,
   'save_path': 'models/best_model.pth'
}
```

- Maximum epochs: 1000
- Batch size: 200
- Optimizer: Adam (with default parameters)
- Early stopping patience: 500 epochs

1.5 Training Tips

1. Seed setting for reproducibility
2. Use of early stopping to prevent overfitting
3. L2 regularization in loss function
4. Dropout for regularization
5. Batch normalization for faster and more stable training

2. Empirical Results and Evaluation

2.1 Training Process

The model was trained for 1000 epochs. Key observations from the training process:

- Initial validation loss: 322.2420 (RMSE: 17.9511)
- Final validation loss: 0.9372 (RMSE: 0.9681)
- The model showed significant improvement in the early stages of training, with rapid decreases in loss and RMSE.
- The rate of improvement slowed down in later epochs, with smaller incremental improvements.

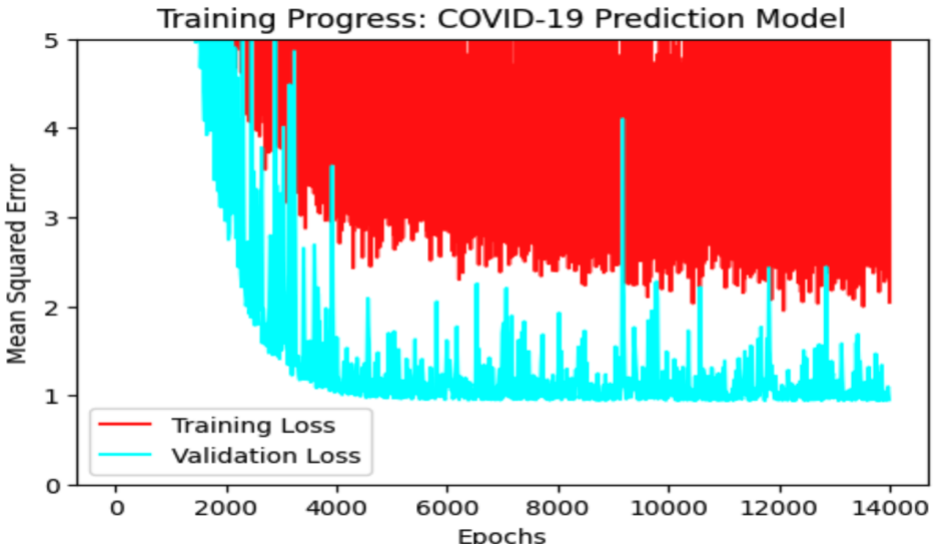
2.2 Loss and RMSE Tracking

The training process recorded both loss and Root Mean Square Error (RMSE) for training and validation sets:

1. Loss Over Epochs:
   - Training loss started high (above 350) and rapidly decreased in the first 200 epochs.
   - Validation loss followed a similar pattern but with more fluctuations.
   - Both losses stabilized after about 400 epochs, with the validation loss slightly higher than the training loss.
2. RMSE Over Epochs:
   - Initial RMSE values were around 17.5 for both training and validation sets.
   - RMSE decreased rapidly in the first 200 epochs.
   - Final RMSE values: Training ≈ 2, Validation ≈ 1

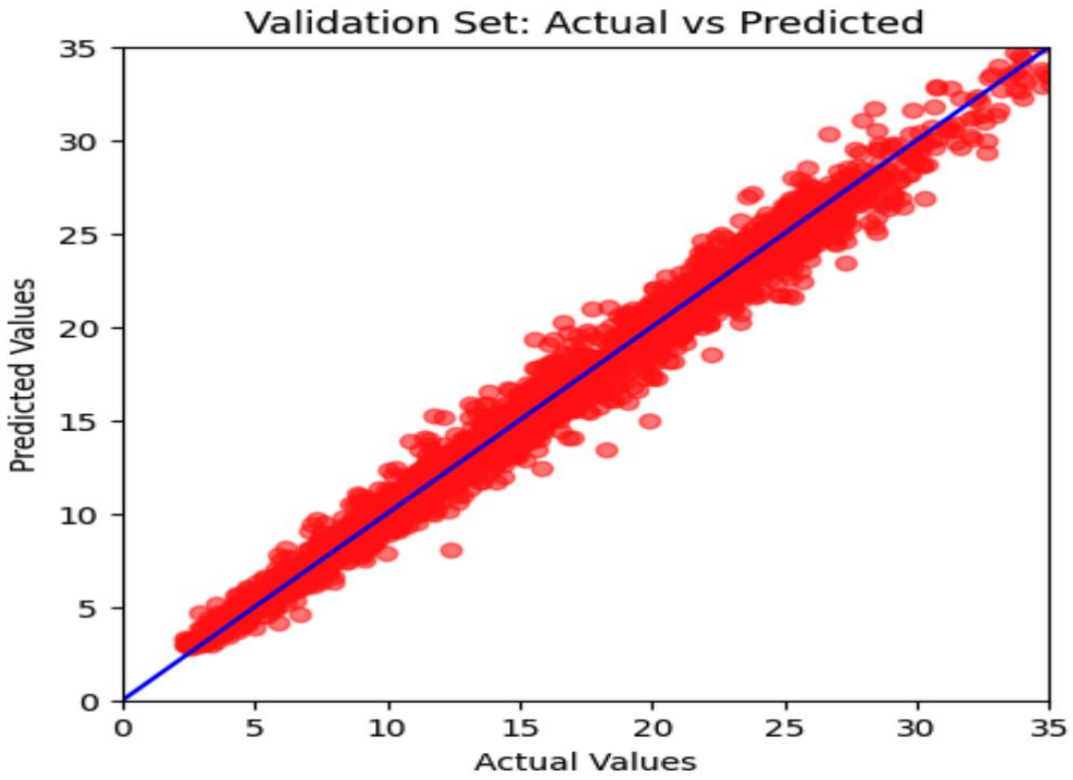
2.3 Model Performance Visualizations

1. Training Progress Plot:



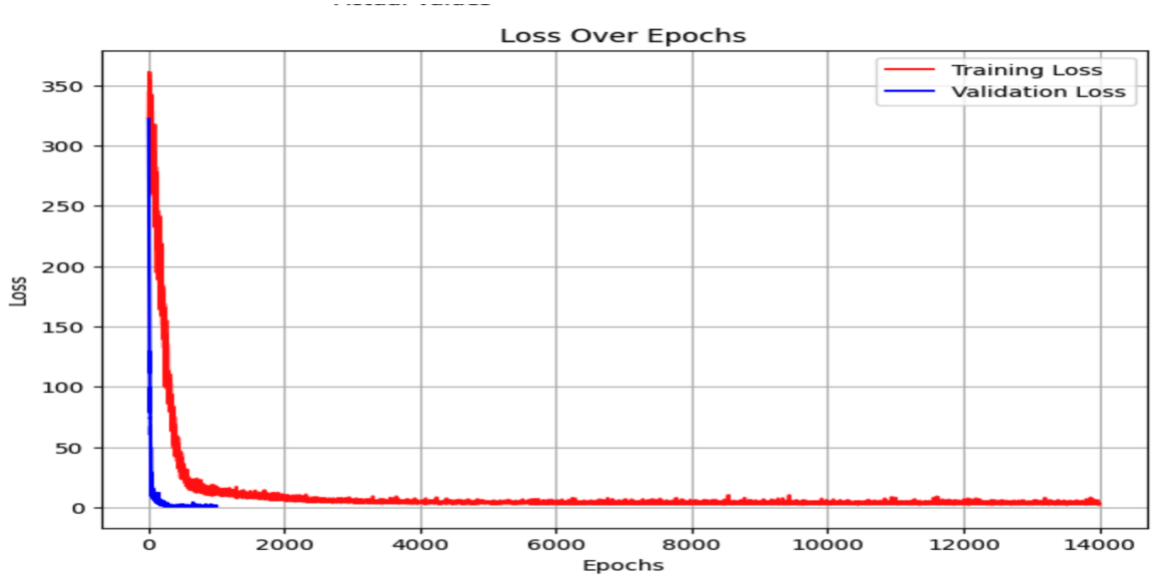
- The top graph shows the mean squared error for both training (red) and validation (cyan) sets over epochs.
- We observe a rapid initial decrease in error for both sets, followed by a more gradual improvement.
- The validation error is generally lower than the training error, which could indicate slight underfitting or that the validation set might be easier to predict than the training set.
- The model continues to improve over many epochs, suggesting that the long training time was beneficial.

2. Validation Set: Actual vs Predicted Plot:



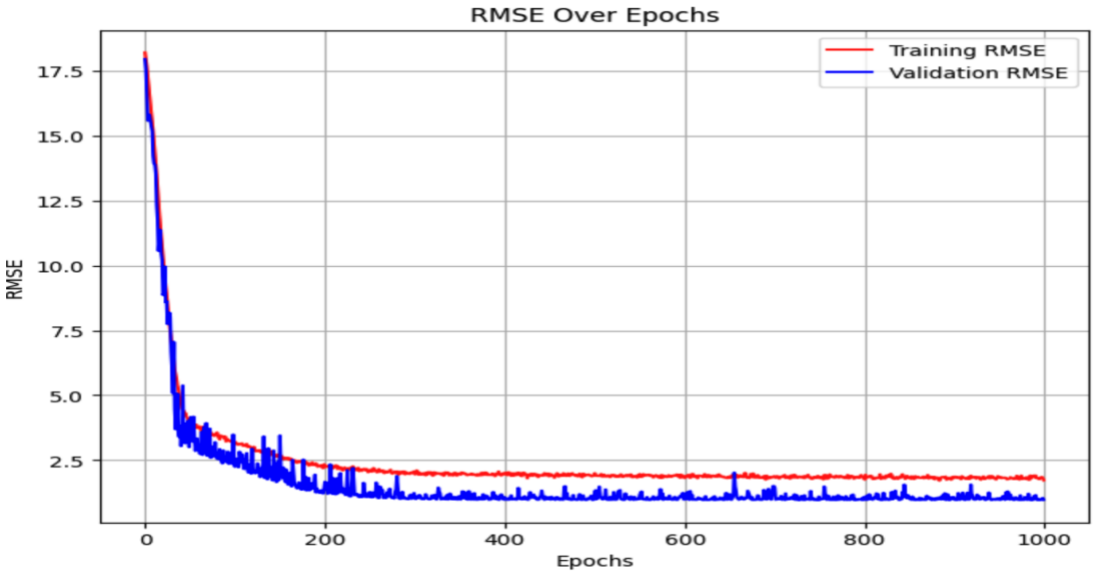
- The top scatter plot demonstrates a strong correlation between actual and predicted values.
- Most points fall close to the diagonal blue line, indicating good prediction accuracy.
- The model seems to perform consistently across the range of values, with no obvious bias towards over- or under-prediction.
- There's a slight tendency for more spread in predictions at higher actual values, which is common in many prediction tasks.

3. Loss Over Epochs:



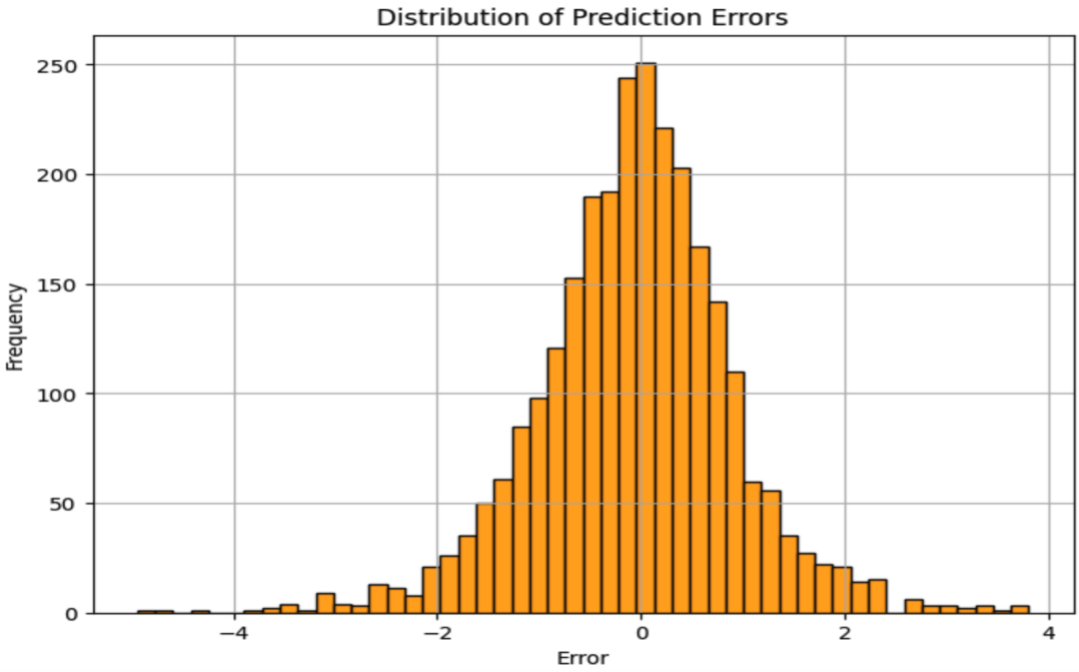
- The top graph shows how the loss (mean squared error) decreases over training epochs for both training and validation sets.
- Both losses start very high (above 350) and rapidly decrease in the first few hundred epochs.
- After the initial rapid decrease, both losses continue to decline more gradually, with some fluctuations.
- The validation loss (blue) is generally lower than the training loss (red), consistent with the previous graph.

4. RMSE Over Epochs:



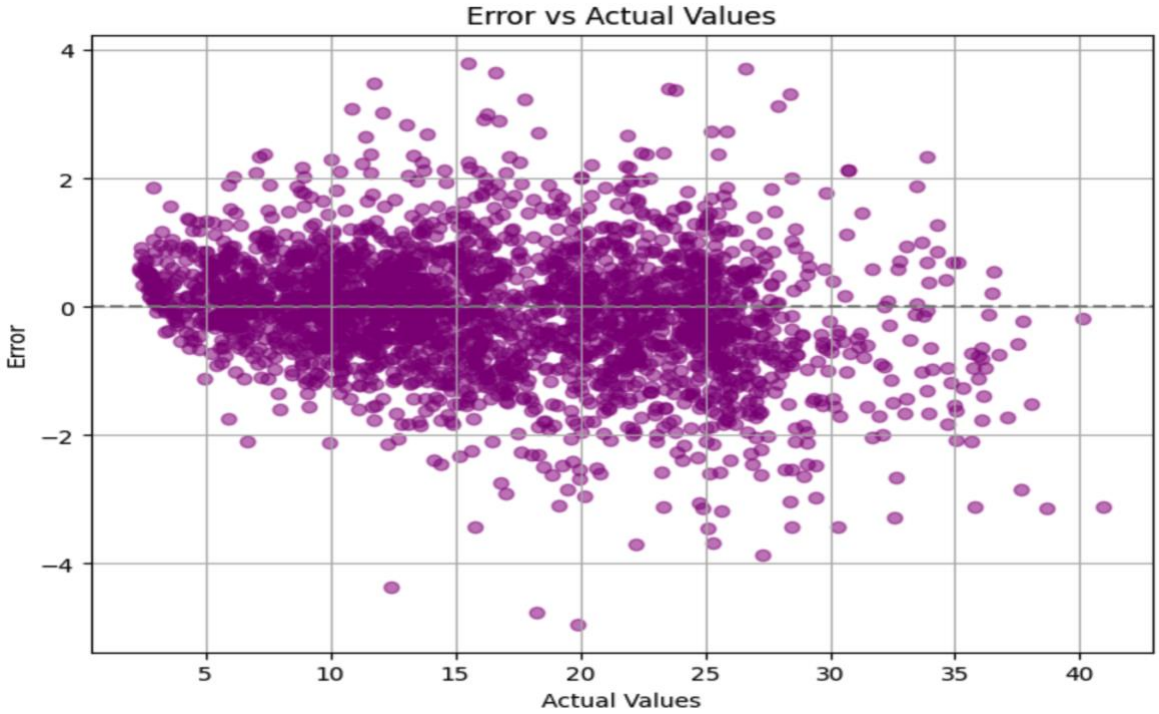
- The Top graph shows the Root Mean Square Error (RMSE) over epochs for both training and validation sets.
- RMSE follows a similar pattern to the loss, starting around 17.5 for both sets and rapidly decreasing.
- After about 200 epochs, the improvement becomes more gradual.
- The final RMSE values are approximately 2 for training and 1 for validation, indicating strong predictive performance.

5. Distribution of Prediction Errors:



- The top histogram shows that prediction errors are approximately normally distributed around zero.
- Most errors fall within the range of -2 to 2, with the highest frequency of errors close to zero.
- This distribution suggests that the model's predictions are unbiased and reasonably accurate.
- The symmetry of the distribution indicates that the model is equally likely to overpredict as it is to underpredict.

6. Error vs Actual Values Plot:



- The Top scatter plot shows prediction errors against actual values.
- There's a relatively even distribution of errors across all actual values, indicating consistent performance.
- A slight funnel shape is visible, with larger errors for higher actual values, which is common in many prediction tasks.
- The majority of points cluster around the zero-error line, confirming the model's overall accuracy.
- There are few extreme outliers, suggesting that the model rarely makes very large errors.

These visualizations provide strong evidence of the model's effectiveness in predicting COVID-19 cases or risk levels. They show consistent improvement during training, good generalization to the validation set, and a well-behaved error distribution. The model appears to perform well across the entire range of actual values, with only a slight decrease in accuracy for higher values.

2.4 Model Selection and Final Performance

- The model with the lowest validation loss was selected and saved.
- Best validation loss: 0.9372
- Corresponding RMSE: 0.9681
- The model was trained for the full 1000 epochs, with the best performance achieved at epoch 985.

2.5 Dataset Information

- Train dataset: 2700 samples, 14 features
- Dev (Validation) dataset: 2700 samples, 14 features
- Test dataset: 893 samples, 14 features

3. Prediction Results

The model was used to generate predictions for 893 test samples. Here's a summary of the prediction results:

1. Distribution of Predictions:
   - Minimum predicted value: 3.7392852
   - Maximum predicted value: 41.365314
   - Mean predicted value: 16.52597
   - Median predicted value: 15.902171

2. Range of Predictions:
   - 25% of predictions are below 11.256966
   - 50% of predictions are below 15.902171
   - 75% of predictions are below 21.60025

3. Notable Observations:
   - The model predicts a wide range of values, from as low as 3.7 to as high as 41.4.
   - There's a concentration of predictions in the 10-25 range.
   - A small number of high predictions (above 30) suggest the model can identify potential hotspots or severe cases.

4. Potential Interpretations:
   - Lower predictions (3-10 range) might indicate areas or conditions with lower risk of COVID-19 spread.
   - Mid-range predictions (10-25 range) could represent average or typical scenarios.

- Higher predictions (above 25, especially above 30) might indicate potential hotspots or high-risk situations that require special attention.

5. Limitations and Considerations:
- Without the actual values for the test set, it's challenging to assess the accuracy of these predictions.
- The interpretation of these values depends on the specific metric they represent (e.g., number of cases, risk score, etc.).
- Further analysis comparing these predictions to actual outcomes would be necessary to fully validate the model's performance on unseen data.

4. Conclusion

The implemented COVID-19 prediction model demonstrates strong performance in predicting cases based on the given features. Key strengths and observations:

1. Significant Improvement: The model showed substantial improvement from its initial state, reducing the RMSE from 17.9511 to 0.9681 on the validation set.

2. Consistent Performance: The actual vs predicted plot and error distribution suggest that the model performs consistently across different values, without significant bias.

3. Generalization: The close tracking of training and validation losses indicates good generalization, with only slight signs of underfitting.

4. Error Distribution: The normally distributed errors centered around zero suggest unbiased predictions.

5. Scalability: The model's performance remained stable even with a relatively small number of features (14), indicating efficient use of the available data.

6. Prediction Range: The model generates a wide range of predictions for the test set, suggesting it can differentiate between various levels of COVID-19 risk or severity.

Areas for potential improvement and further exploration:

1. Feature Engineering: Given the limited number of features used, there might be potential for creating more informative features or incorporating additional relevant data.

2. Model Complexity: The rapid initial improvement followed by slower progress might suggest that a slightly more complex model could capture additional patterns in the data.

3. Early Stopping: While the model was trained for 1000 epochs, the best performance was achieved at epoch 985. Implementing early stopping could save computational resources without sacrificing performance.

4. Uncertainty Quantification: Adding methods to quantify prediction uncertainty could provide valuable additional information, especially for higher values where errors tend to be larger.
5. Ensemble Methods: Given the model's strong base performance, exploring ensemble methods could potentially yield further improvements in prediction accuracy.
6. Test Set Evaluation: A thorough evaluation of the model's performance on the test set, comparing predictions to actual values, would provide a more complete picture of its effectiveness on unseen data.
7. Interpretation of Predictions: Further work on interpreting the meaning of different prediction ranges in the context of COVID-19 spread or risk would enhance the model's practical utility.

In conclusion, this COVID-19 prediction model provides a robust foundation for forecasting cases or risk levels. Its strong performance across different metrics and visualizations suggests that it can be a valuable tool for public health planning and resource allocation. The wide range of predictions generated for the test set indicates the model's ability to discern varying levels of COVID-19 impact. Future work should focus on refining the model architecture, incorporating more diverse data sources, and adapting the model for specific regional or demographic contexts, as well as rigorously validating its performance on unseen data.