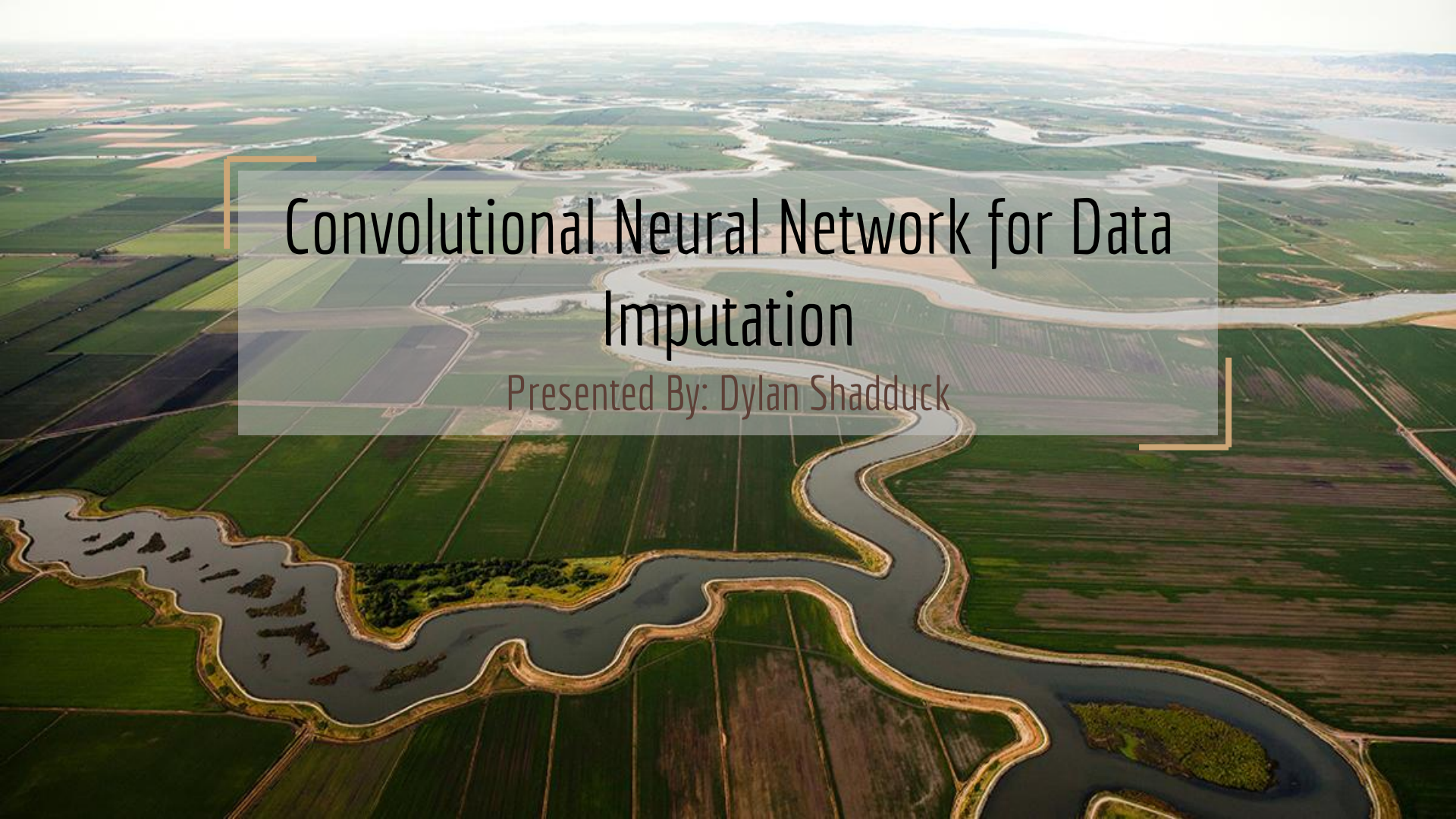


Convolutional Neural Network for Data Imputation

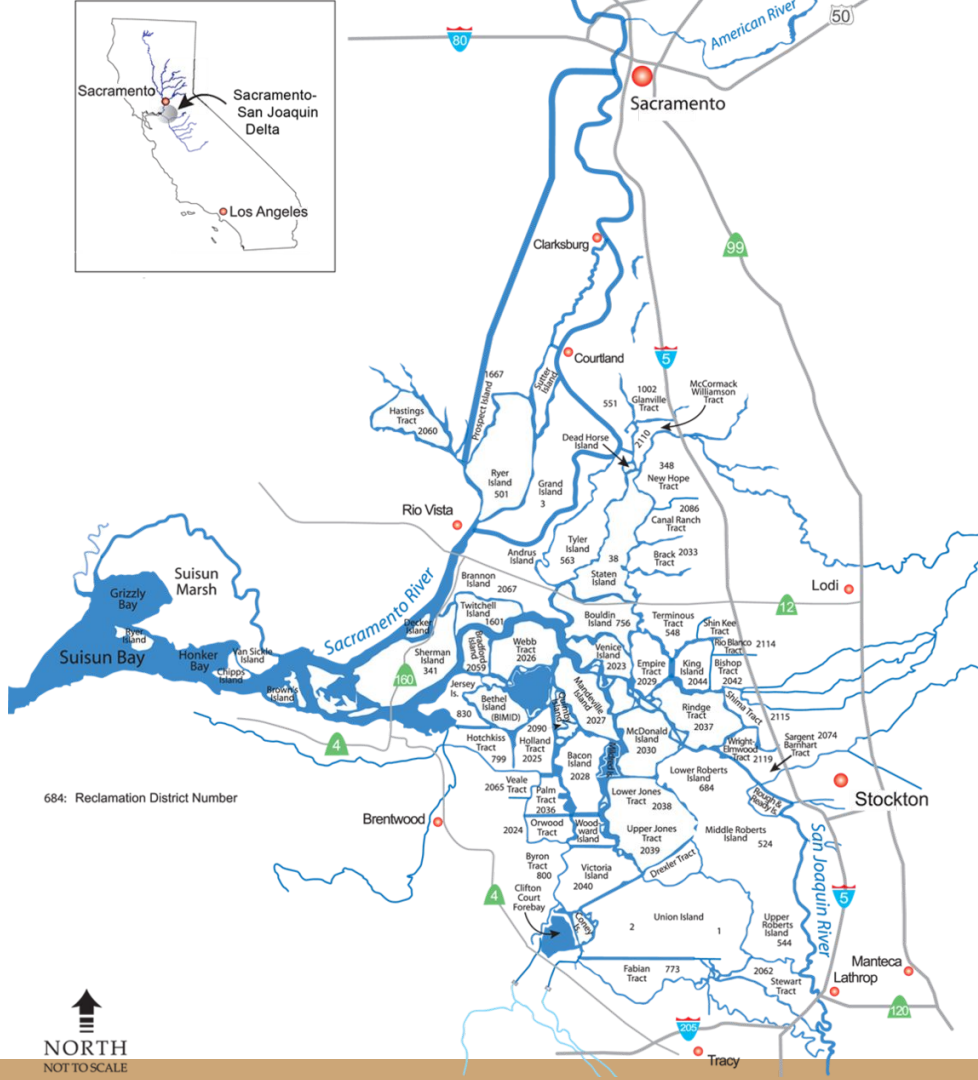
Presented By: Dylan Shadduck

Overview

- Dataset Introduction
- Problem Formulation
- Data Preparation
- Proposed Solution
- CNN Evaluation
- Remarks
- Future Work

Dataset Introduction

- The Delta is a complex system of waterways
- The Department of Water Resources (DWR) has field stations/salinity measurement tools at various locations
- This data, along with environmental factors, can be used to restrict or increase river flow rates



Dataset Introduction

- Historical data is used to model the delta salinity over time
- Accurately predicting salinity is crucial to agriculture and wildlife survival

Dataset Introduction

- Data is produced by Complex Delta Water Simulations designed by DWR
- Two DWR datasets were used throughout this project
  - Daily salinity measurements recorded from 12 stations
    - This data has been linearly interpolated by DWR from 1 sample per month
    - Dates from January 1940 - August 2019
  - Salinity measurements taken every 15 minutes from 26 stations
    - No interpolation
    - Dates from January 2000 - September 2019
- Salinity measurements are in μS/cm
  - A measurement of conductivity
  - Conductivity increases linearly with increase in salt concentration

Overview

- Dataset Introduction
- Problem Formulation
- Data Preparation
- Proposed Solution
- CNN Evaluation
- Remarks
- Future Work

Problem Formulation

- Delta Salinity data is used to train Artificial Neural Networks (ANN) to predict future salinity levels
- ANNs require lots of data to produce accurate models
- Without an effective method for reproducing missing values, large sections of data are not useable by ANNs

Overview

- Dataset Introduction
- Problem Formulation
- Data Preparation
- Proposed Solution
- CNN Evaluation
- Remarks
- Future Work

Data Preparation: Masking

- Daily dataset does not contain missing values
- Artificially masking data to simulate missing data
- Data is organized into sections of 10 days (across all 12 stations)
- Each 10 day section is given a random mask (percent missing remains same)

Data Preparation: Masking

- Masked values are filled in with linearly interpolated values

$7.5 = (8+7)/2$

$27 = 29 + (29-31)$

Days

4	2	10	3	5	17	32	1	5	27	9	6
5	1	8	4	7	21	29	2	4	29	11	4
6	1	7.5	5	8	23	28	3	4	31	12	2
8	3	7	6	11	20	25	5	2	30	15	3
7	4	5	7	14	18	26	6	3	29	17	4

Stations

$14 = 11 + (11-8)$

Data Preparation: Scaling

- Prior to training neural network, data is scaled
- New bounds for each **station** are [0, 1]

$$s = \frac{x - l_c}{r_c}$$

x = input data

l_c = minimum column value

r_c = column range

s = new scaled value.

if m_c is the maximum value of a column, then $r_c = m_c - l_c$

Data Preparation: Train/Test Split

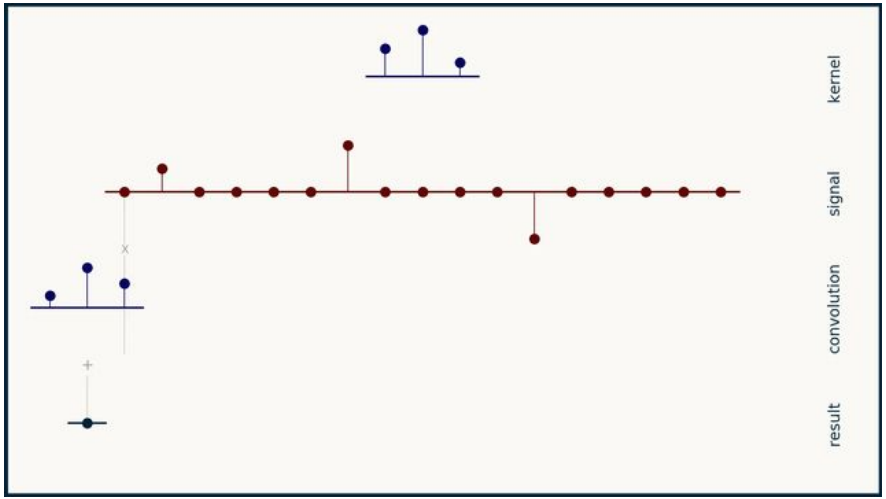
- Dataset is split into training (70%) and test (30%) sets
- Training dataset is further split into train (70%) and validation (30%)
- CNN uses validation dataset to tune parameters
- Test set is used to evaluate CNN performance

Overview

- Dataset Introduction
- Problem Formulation
- Data Preparation
- Proposed Solution
- CNN Evaluation
- Remarks
- Future Work

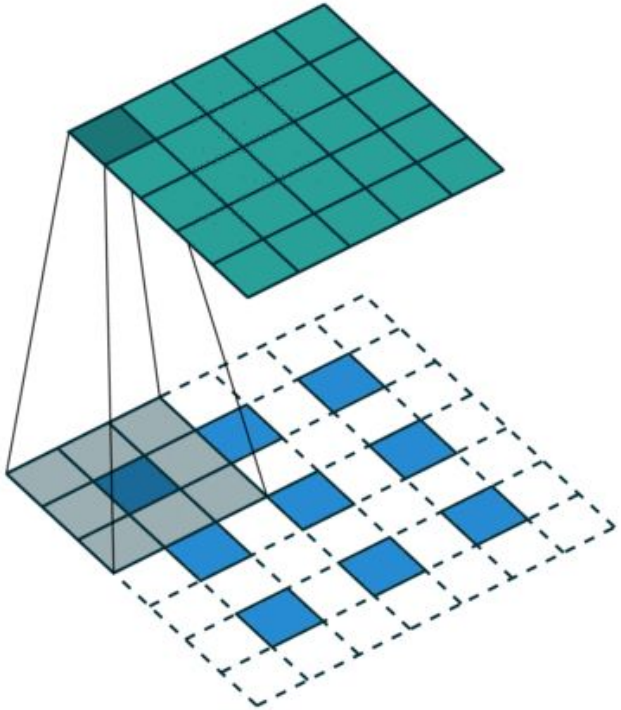
Proposed Solution: Convolutional Neural Network

- A kernel moves through a given vector to produce an output
- The values in the kernel are trainable by the network to produce the output desired
- Reduces dimensions of input vector



strides=1, kernel_size=3, zero_padding

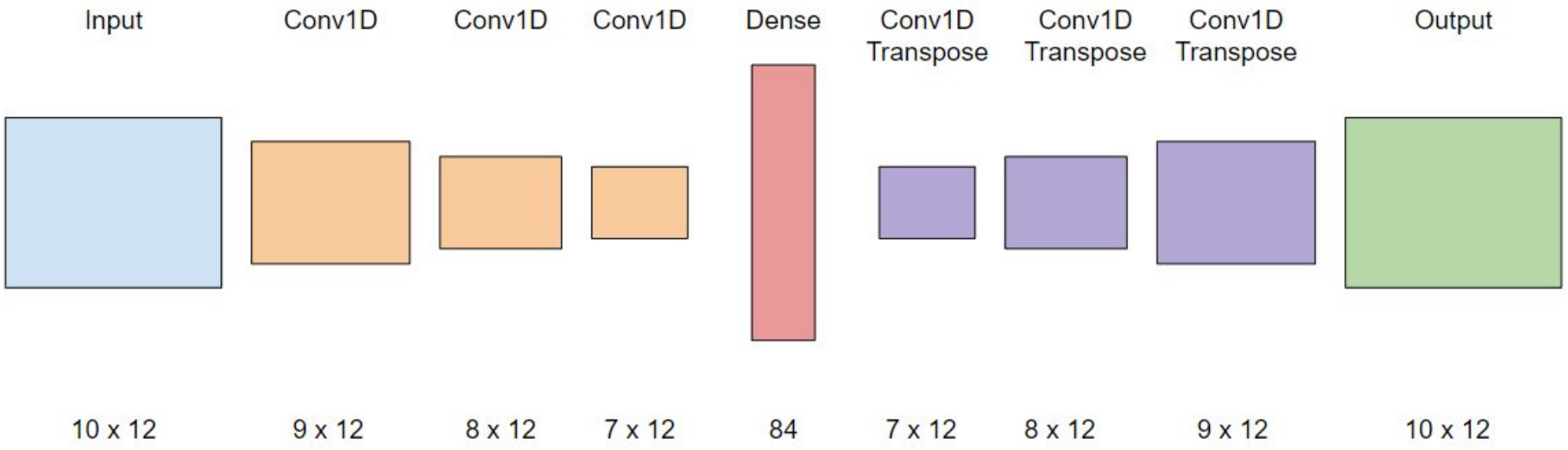
Proposed Solution: CNN (Transpose Convolution)



Strides=2, kernel_size=3x3, zero_padding

- Transpose convolution increases the dimensions of the output vector compared to input
- Again a kernel is used for the convolution
- Fractional strides/zero padding can be used to increase the dimension of the output vector

Proposed Solution: CNN



Proposed Solution: Training Parameter Definition

- Batchsize = 16
- Validation Split = 0.3
- Epochs = 500
- Optimizer = ADAM
- Loss = MSE
- Learning Rate = variable [4.8e-3, 1e-6]

Overview

- Dataset Introduction
- Problem Formulation
- Data Preparation
- Proposed Solution
- CNN Evaluation
- Remarks
- Future Work

CNN Evaluation: Error Functions

Mean Squared Error

$$MSE = \frac{1}{N}\sum_{n=1}^{N}(x_n - \hat{x}_n)^2$$

Mean Absolute Percentage Error

$$MAPE = \frac{1}{N}\sum_{n=1}^{N}\left|\frac{x_n - \hat{x}_n}{x_n}\right|$$

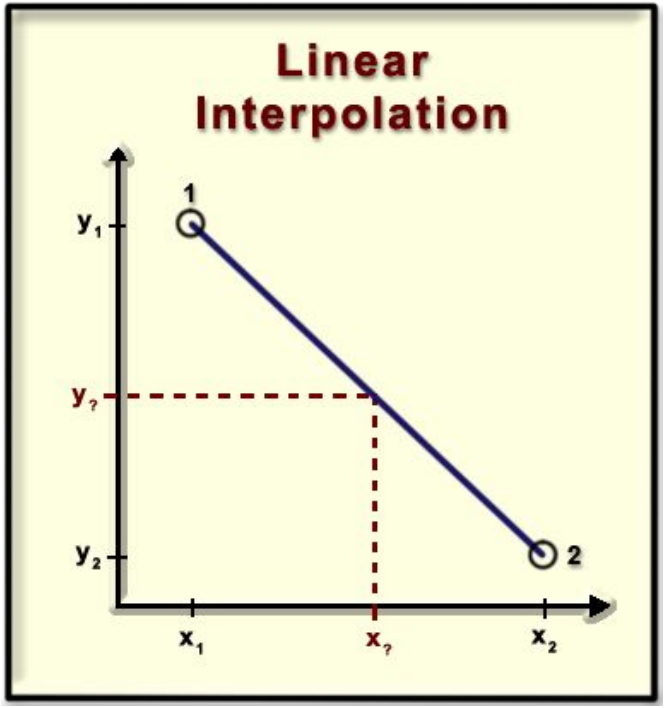
N = number of data points

x_n = true value

$\hat{x}_n$ = predicted value

CNN Evaluation: Baseline

- Linear Interpolation is used as our baseline method
- This is the same method used to fill in missing data during masking
- Also the same method used by DWR to interpolate data from 1 sample/month → 1 sample/day



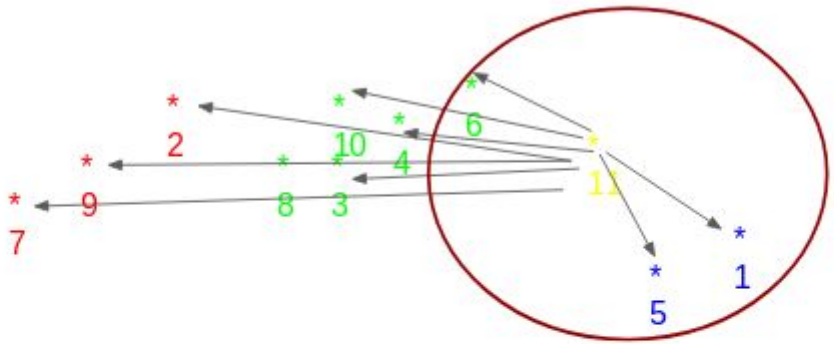
CNN Evaluation: Baseline Comparison

- Trained CNN on data with a missing rate of 20%
- Performed linear interpolation on the same data
- The following are the results of the error calculations

TABLE II
CNN VS LINEAR INTERPOLATION

	CNN		Lin Interp	
Station	**MSE**	**MAPE**	**MSE**	**MAPE**
Emmaton	4.84e3	0.048	4.28e3	0.022
Jersey Point	8.51e2	0.028	3.14e3	0.014
Collinsville	1.04e4	0.063	2.75e4	0.034
Rock Slough	4.40e1	0.0092	2.24e1	0.0039
Antioch	3.28e3	0.055	1.45e4	0.024
Mallard	2.54e4	0.073	6.18e4	0.034
Los Vaqueros	3.66e1	0.009	5.86e1	0.0066
Martinez	1.36e5	0.039	1.26e5	0.024
MiddleRiver	3.96e1	0.009	1.93e1	0.0044
Vict Intake	4.30e1	0.01	4.28e1	0.0099
CVP Intake	8.74e1	0.012	8.74e1	0.0099
CCFB OldR	1.07e2	0.014	9.43e1	0.0095
Average	1.50e4	0.031	1.98e4	0.016

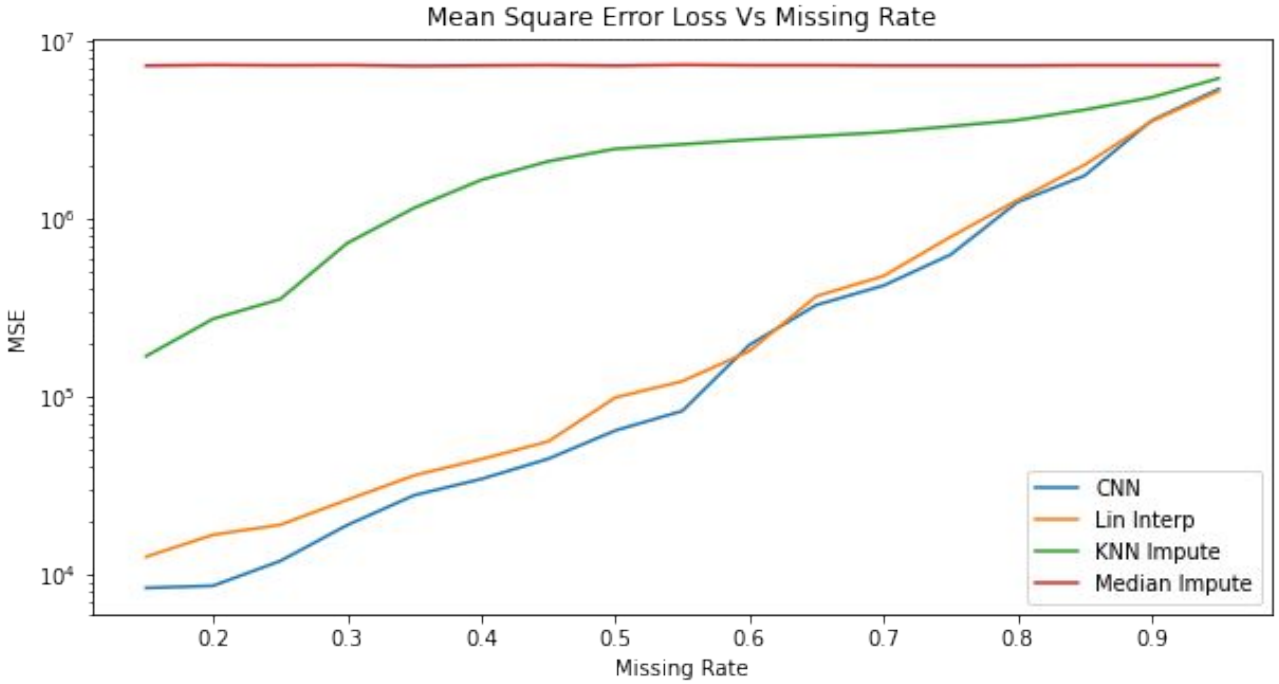
CNN Evaluation: Comprehensive Comparison



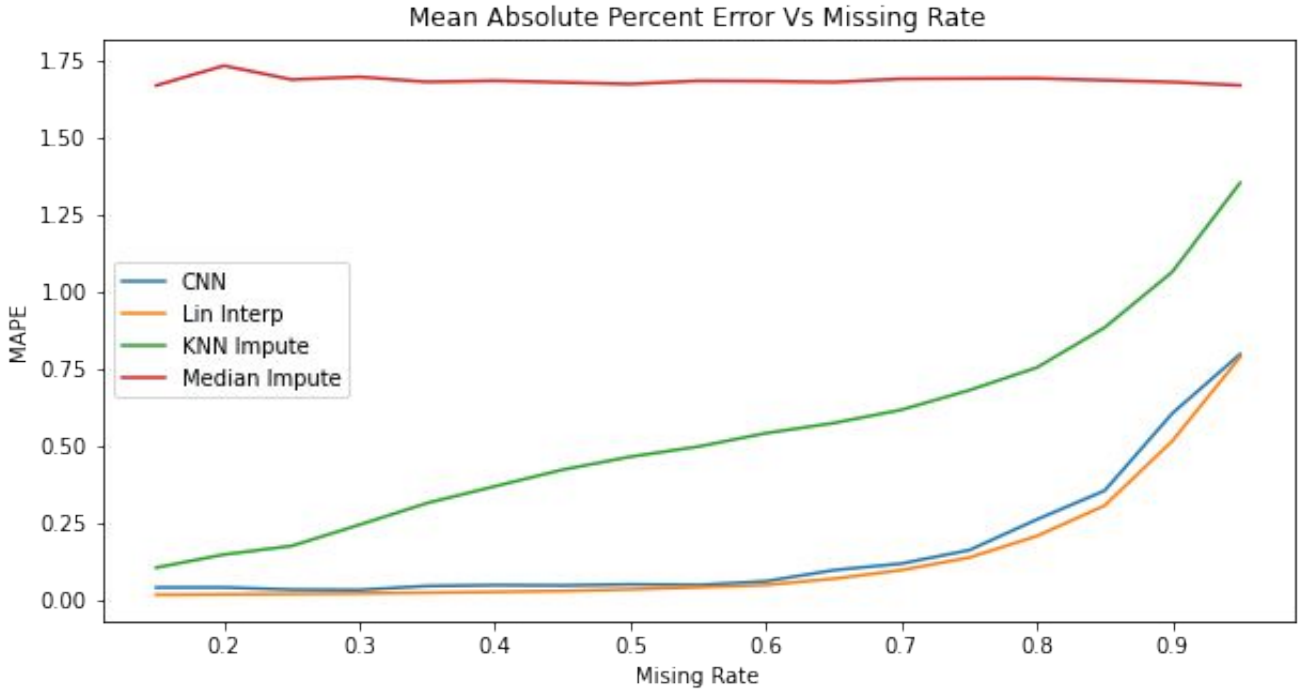
KNN Interpolation Example

- Compared CNN with two other interpolation methods
  - KNN and Median Replacement
- Trained CNN and KNN on variable missing rate data
  - Varied from 15 - 95 percent in 5 percent increments
  - Linear interpolation and median replacement were also done on each of these missing rates

CNN Evaluation: Comprehensive Comparison



CNN Evaluation: Comprehensive Comparison



Overview

- Dataset Introduction
- Problem Formulation
- Data Preparation
- Proposed Solution
- CNN Evaluation
- Remarks
- Future Work

Remarks

- Error evaluation results show that CNN does not outperform linear interpolation
- Possible reasons include:
  - Input data is linearly interpolated prior to training. This means that the data is biased to perform better with one method vs another
  - Loss function used to evaluate CNN is MSE loss. Across all evaluations, CNN performs better in MSE, but mostly worse in MAPE

Overview

- Dataset Introduction
- Problem Formulation
- Data Preparation
- Proposed Solution
- CNN Evaluation
- Remarks
- Future Work

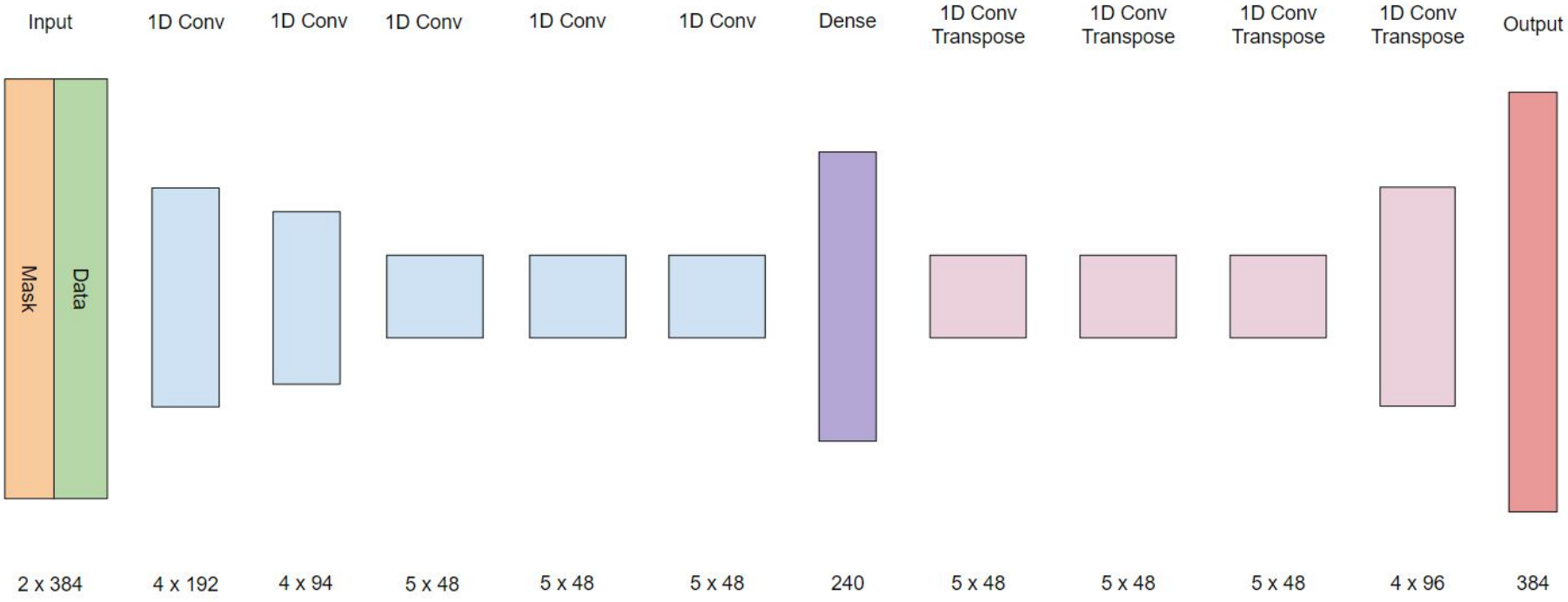
Future Work: Dataset Selection

- To remove some method bias, select 15 minute dataset mentioned earlier
- Focus on one station rather than all 26
- This requires two major modifications to the pipeline:
  - New method of data masking
  - CNN Model Redesign

Future Work: Data Masking

- 15 minute dataset contains large sections of missing data
- Mask will attempt to model this by grouping all masked values together
- This masking is meant to simulate a power outage at the station in question

Future Work: CNN Model Redesign



Future Work: CNN Model Redesign

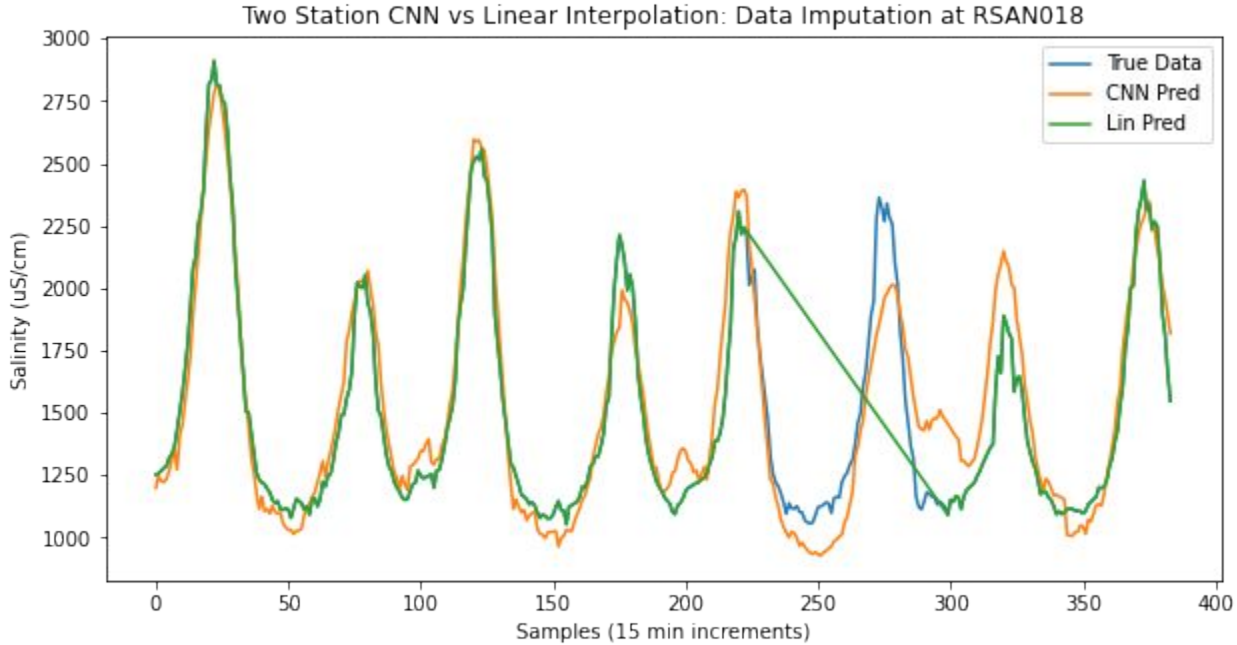
Layer Type	*Filters*	*Padding*	*Kernel Size*	*Strides*	Out Shape
Salinity (Input)					384
Mask (Input)					384
Concatenate					2x384
Conv1D	4	none	5	2	4x192
Batch Norm					4x192
Conv1D	4	none	4	2	4x96
Batch Norm					4x96
Conv1D	5	same	4	2	5x48
Batch Norm					5x48
Conv1D	5	same	4	2	5x48
Batch Norm					5x48
Flatten					240
Dense					240
Reshape					5x48
Conv1D	5	same	4	2	5x48
Batch Norm					5x48
Conv1D	5	same	4	2	5x48
Batch Norm					5x48
Conv1D Transpose	4	none	2	2	4x96
Batch Norm					4x96
Flatten					384
Output					384

Conv only parameter

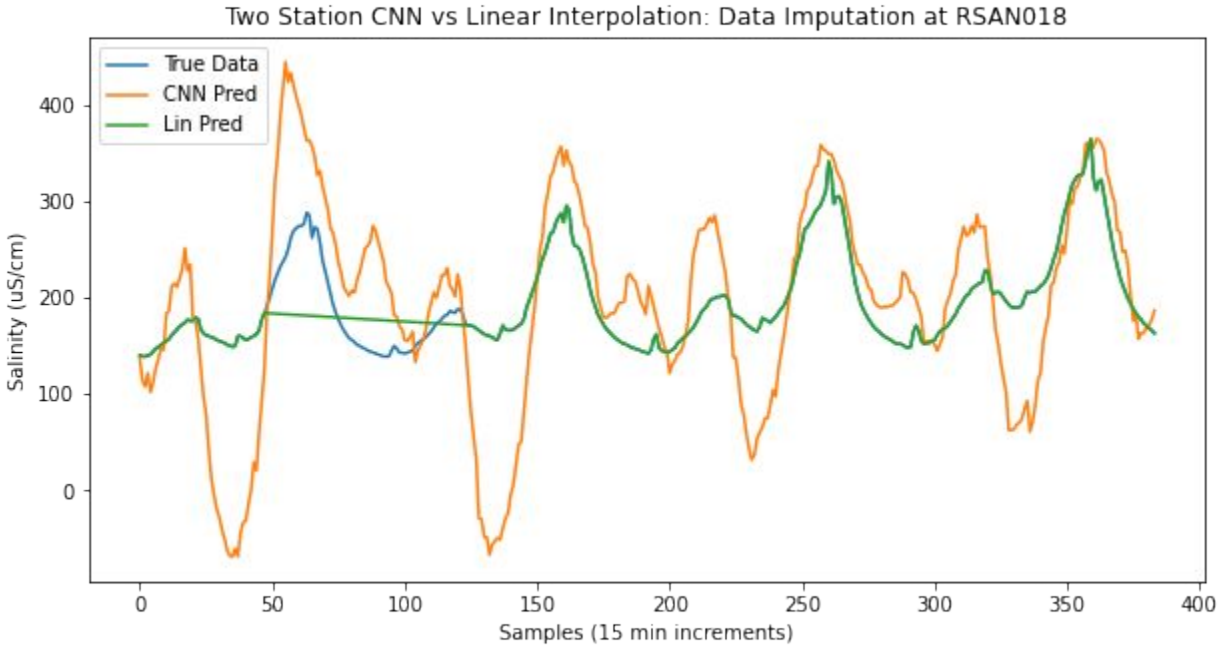
Future Work: Preliminary Evaluation

- Dataset is masked using the new method and a missing rate of 20 percent
- CNN is trained on this dataset with the following parameters:
  - Batch size: 16, Epochs: 500, Optimizer: ADAM, Learning Rate: Variable
  - Loss Function: MSE, Validation Split: 0.3
- Linear interpolation is also performed on this same dataset
- Average Errors:
  - CNN:
    - MSE: 2.84e4
    - MAPE: 0.178
  - Linear Interpolation:
    - MSE: 3.42e4
    - MAPE: 0.033

Future Work: Preliminary Evaluation



Future Work: Preliminary Evaluation



Future Work: Preliminary Evaluation

- CNN accurately predicts salinity in missing regions with large salinity range
  - Small range exposes the overfitting problem with this CNN

- Possible Solutions:
  - Instead of scaling all data prior to input to CNN, scale each 384 sample section individually.
    - Could remove bias towards large range sampling windows
  - Additional discriminator networks to determine if the CNN predicted values are “realistic”
    - Could produce smoother transitions between true values and predicted values


Questions?