Deep Learning for Signal and Image Processing with Limited Data

Jacqueline Alvarez

Department of Applied Mathematics, UC Merced

Advisor: Dr. Roummel F. Marcia
Committee Members: Dr. Omar DeGuchy (LLNL)
Dr. Arnold Kim (Applied Math)
Dr. Chrysoula Tsogka (Applied Math)

Qualifying Examination for Advancement to Candidacy

Wednesday, March 2, 2022

Overview

Overview

Machine learning algorithms typically require a large quantity of high-quality data. In applications, e.g. biomedical imaging, machine learning algorithms may have difficulties learning their intended task due to data limitations. To address this issue,

- Common Method: increase the number of training examples
- Proposed Method: improve the optimization framework used during training

Overview

Machine learning algorithms typically require a large quantity of high-quality data. In applications, e.g. biomedical imaging, machine learning algorithms may have difficulties learning their intended task due to data limitations. To address this issue,

- Common Method: increase the number of training examples
- Proposed Method: improve the optimization framework used during training

My work focuses addressing **data limitations** through:

1. Machine learning for image processing with limited data
2. Low-resolution reconstructions for synthetic aperture radar (SAR) images
3. Structural variant detection with low quality data
4. Improved optimization methods for machine learning

I. Previous Work

I. Image Denoising Using Recurrent Neural Network

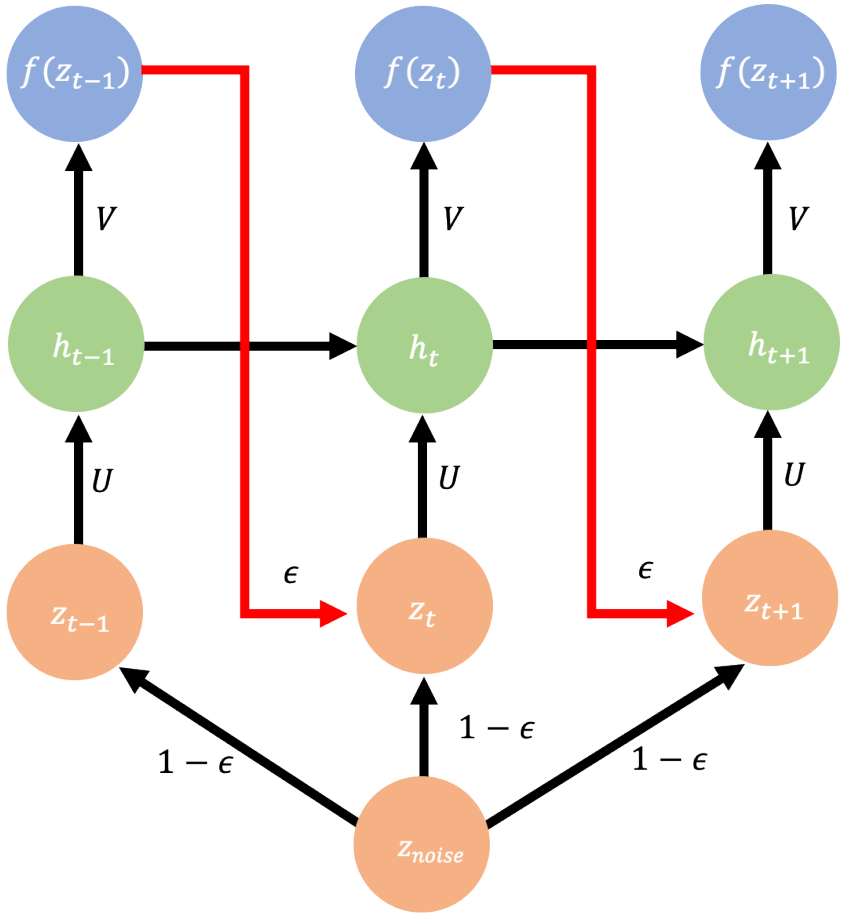
Goal: Denoise image using a recurrent neural network (RNN) **without training data**[1].

[1] A. Ho, J. Alvarez and R. F. Marcia, "Convolution Padding in Recurrent Neural Networks for Image Denoising with Limited Data," Submitted to 2021 Asilomar Conference on Signals, Systems, and Computers.

I. Image Denoising Using Recurrent Neural Network

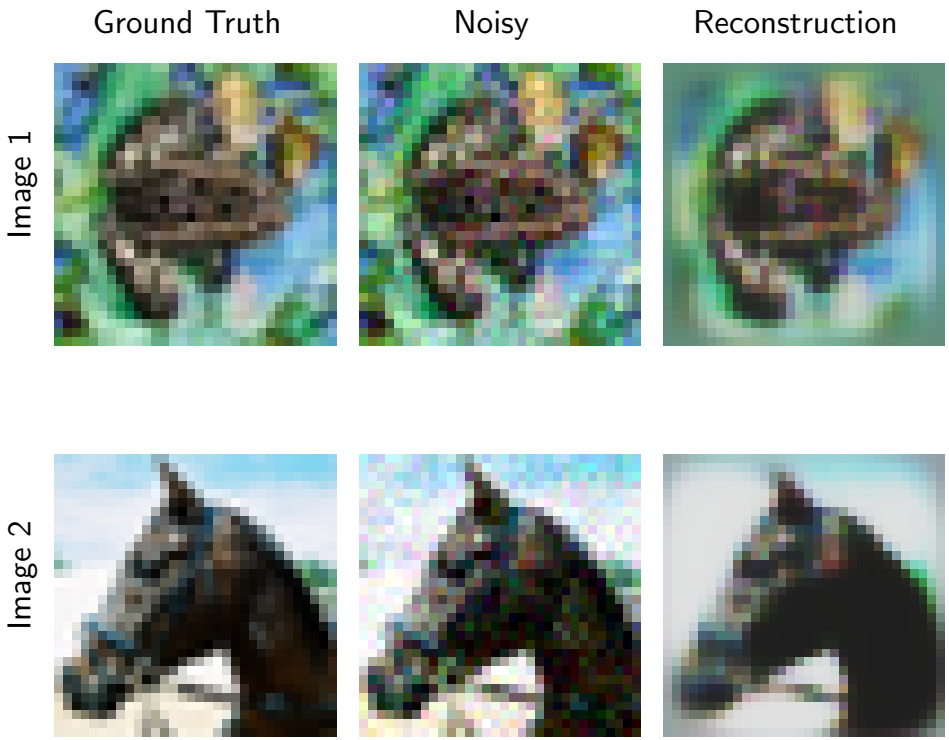
Goal: Denoise image using a recurrent neural network (RNN) **without training data**[2].

z_{noise}	=	Gaussian noise
z_t	=	Current input
ϵ	=	weighted sum parameter
U_ϕ	=	Encoder with parameters ϕ
V_θ	=	Decoder with parameters θ
$f(z_t)$	=	Reconstruction at time t



[2] D. Ulyanov, A. Vedaldi, and V. Lempitsky, "Deep Image Prior," 2018.

I. Results

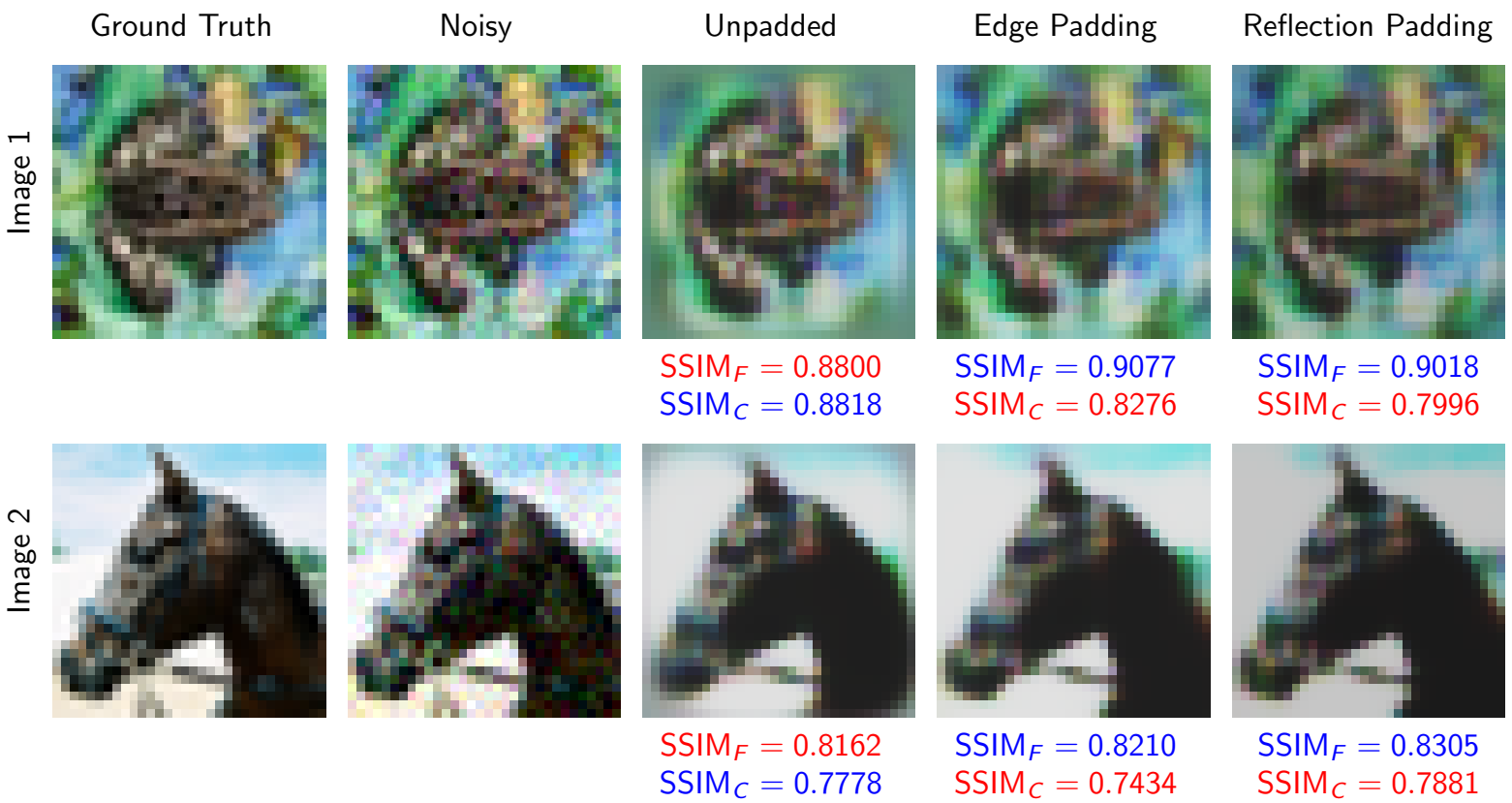


Conclusions:

- Utilized RNN architecture for image denoising without any information from the true image.

[1] SSIM: Structural Similarity Index Measure

I. Results



Conclusions:

- Utilized RNN architecture for image denoising without any information from the true image.
- The overall restoration of the image was improved by adding padding.

[1] SSIM: Structural Similarity Index Measure

II. Ongoing Work

II. Synthetic Aperture Radar (SAR)

Setup: In SAR typically a single transmitter/receiver is used to collect the scattered electromagnetic field over a synthetic aperture that is created by a moving platform[1].

$\mathbf{f}(t)$ = broadband pulse

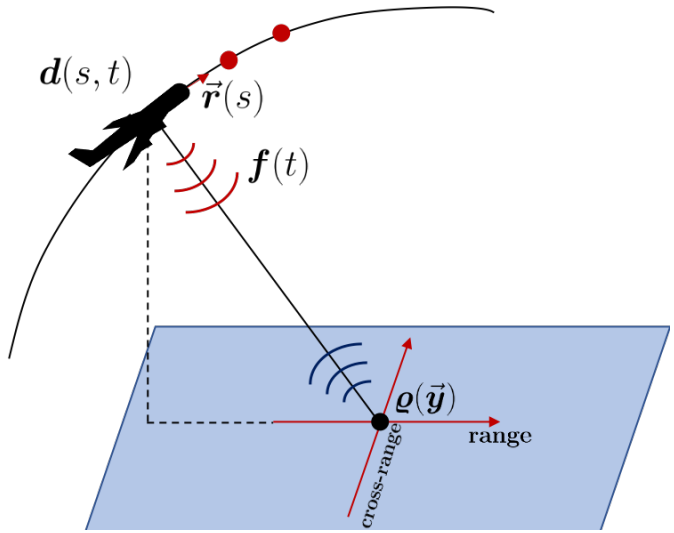
t = fast time parameter

$\vec{\boldsymbol{r}}(s)$ = location of the platform

s = slow time parameter

$\boldsymbol{\varrho}(\vec{\boldsymbol{y}})$ = unknown reflectivity

$\boldsymbol{d}(s,t)$ = SAR observation data



[1] M. Cheney, "A mathematical tutorial on synthetic aperture radar," 2001.

II. Direct Scattering Problem

The SAR process can be modeled using a scalar wave equation, which is discritized and reduces the imaging problem as

$$\boldsymbol{A}\underline{\boldsymbol{\varrho}} = \boldsymbol{d},$$

where

$\boldsymbol{A} \in \mathbb{C}^{N\times K}$ is the sensing matrix

$\underline{\boldsymbol{\varrho}} \in \mathbb{C}^{K}$ is the reflectivity vector

$\boldsymbol{d} \in \mathbb{C}^{N}$ is the SAR observation data.

II. Direct Scattering Problem

The SAR process can be modeled using a scalar wave equation, which is discritized and reduces the imaging problem as

$$\boldsymbol{A}\underline{\boldsymbol{\varrho}} = \boldsymbol{d},$$

where

$\boldsymbol{A} \in \mathbb{C}^{N\times K}$ is the sensing matrix

$\underline{\boldsymbol{\varrho}} \in \mathbb{C}^{K}$ is the reflectivity vector

$\boldsymbol{d} \in \mathbb{C}^{N}$ is the SAR observation data.

We assume that $N \gg K$ and $\boldsymbol{A}$ has full-column rank.

II. Inverse Scattering Problem

Approximating ϱ: The least-squares solution is given by

$$\hat{\underline{\varrho}}_{\ell_2} = (\boldsymbol{A}^H \boldsymbol{A})^{-1} \boldsymbol{A}^H \boldsymbol{d},$$

where $\boldsymbol{A}$ is full column rank and $\boldsymbol{A}^H$ is the complex conjugate-transpose of $\boldsymbol{A}$.

[1] L.Borcea et al, "Synthetic aperture imaging of directional and frequency dependent reflectivity," 2016.

II. Inverse Scattering Problem

Approximating ϱ: The least-squares solution is given by

$$\hat{\boldsymbol{\varrho}}_{\ell_2} = (\boldsymbol{A}^H \boldsymbol{A})^{-1} \boldsymbol{A}^H \boldsymbol{d},$$

where $\boldsymbol{A}$ is full column rank and $\boldsymbol{A}^H$ is the complex conjugate-transpose of $\boldsymbol{A}$.

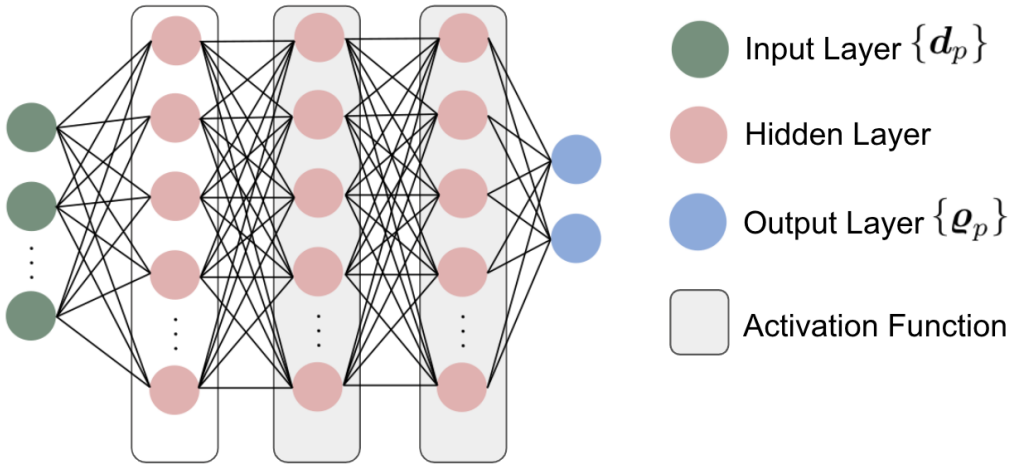
A common choice in discretizing the imaging window leads to the matrix $\boldsymbol{A}^H \boldsymbol{A}$ being close to diagonal[1], which justifies the SAR inversion formula,

$$\hat{\boldsymbol{\varrho}}_{\text{SAR}} = \boldsymbol{A}^H \boldsymbol{d}.$$

[1] L.Borcea et al, "Synthetic aperture imaging of directional and frequency dependent reflectivity," 2016.

II.A: Machine Learning for SAR

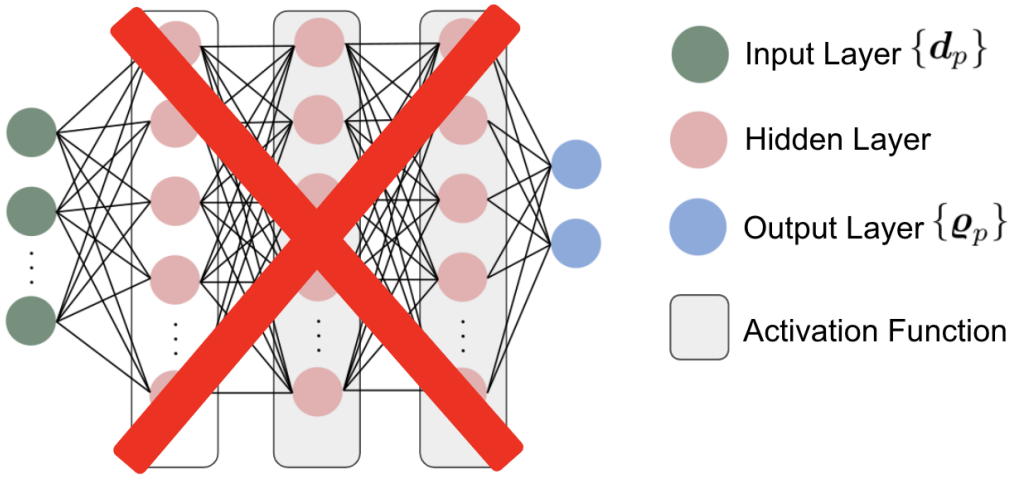
Our method[1] is based on the use of a **neural network** in order to discover a mapping between a set **reflectivities** $\{\varrho_p\}$ and a set of **measurements** $\{d_p\}$.



[1] O. DeGuchy, J. Alvarez, A. D. Kim, R. F. Marcia and C. Tsogka, "Forward and inverse scattering in synthetic aperture radar using machine learning," Proceedings of SPIE 2020, Applications of Machine Learning.

II.A: Machine Learning for SAR

Our method[1] is based on the use of a **neural network** in order to discover a mapping between a set **reflectivities** $\{\varrho_p\}$ and a set of **measurements** $\{d_p\}$.



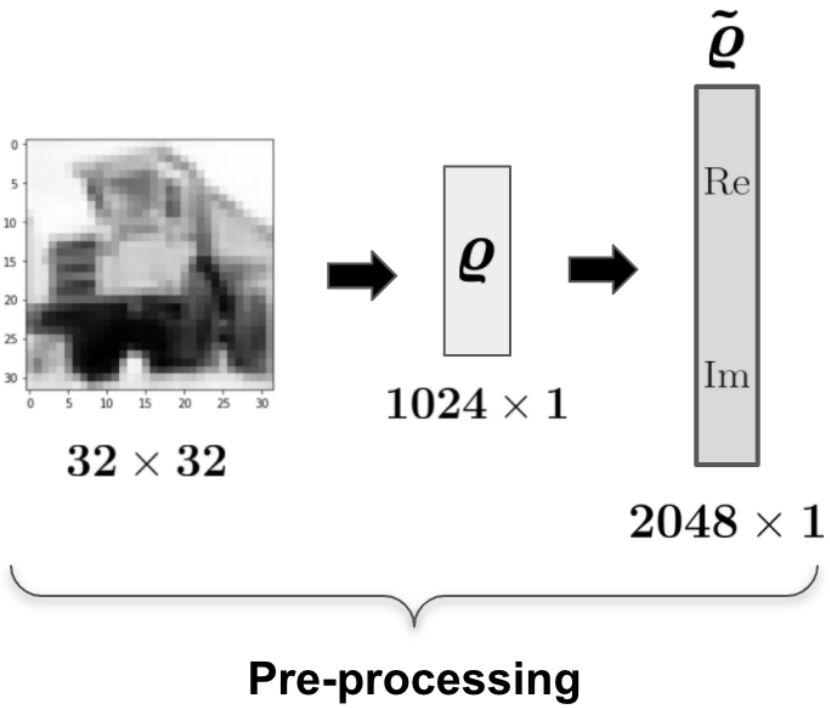
Our Approach:

- One fully connected layer is used.
- No activation function is used.
- The focus is on the recovery of the transformation.

[1] O. DeGuchy, J. Alvarez, A. D. Kim, R. F. Marcia and C. Tsogka, "Forward and inverse scattering in synthetic aperture radar using machine learning," Proceedings of SPIE 2020, Applications of Machine Learning.

II.A: Training Process - Direct Scattering

Recovering forward mapping: Given a set of **reflectivities** $\{\varrho_p\}$ and their corresponding **measurements** $\{\boldsymbol{d}_p\}$ we recover the sensing matrix $\boldsymbol{A}$ by



1. **Reshape** the reflectivity into its vector format and separate the real and imaginary values.

II.A: Training Process - Direct Scattering

Recovering forward mapping: Given a set of **reflectivities** $\{\varrho_p\}$ and their corresponding **measurements** $\{\boldsymbol{d}_p\}$ we recover the sensing matrix $\boldsymbol{A}$ by

$$\min_{\tilde{\boldsymbol{A}}} \frac{1}{k} \sum_{\tilde{\boldsymbol{\varrho}}} \left\| \begin{bmatrix} \text{Re} & \text{-Im} \\ \text{Im} & \text{Re} \end{bmatrix} \begin{bmatrix} \text{Re} \\ \text{Im} \end{bmatrix} - \begin{bmatrix} \text{Re} \\ \text{Im} \end{bmatrix} \right\|_2^2$$

$\tilde{\boldsymbol{A}}$: $2M \times 2N$; $\tilde{\boldsymbol{\varrho}}$: $2N \times 1$; $\tilde{\boldsymbol{d}}$: $2M \times 1$

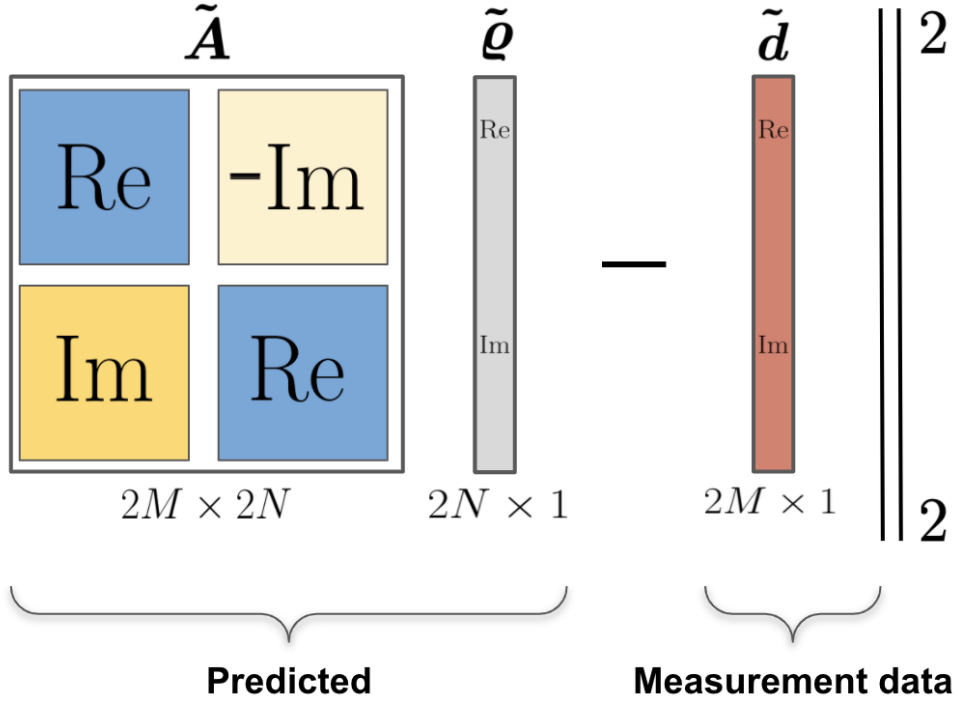
Predicted — Measurement data

1. **Reshape** the reflectivity into its vector format and separate the real and imaginary values.
2. Perform a **left multiply** by the sensing matrix initialized with **random values**.

II.A: Training Process - Direct Scattering

Recovering forward mapping: Given a set of **reflectivities** $\{\varrho_p\}$ and their corresponding **measurements** $\{\boldsymbol{d}_p\}$ we recover the sensing matrix $\boldsymbol{A}$ by

$$\min_{\tilde{\boldsymbol{A}}} \frac{1}{k} \sum_{\tilde{\boldsymbol{\varrho}}} \left\| \tilde{\boldsymbol{A}} \tilde{\boldsymbol{\varrho}} - \tilde{\boldsymbol{d}} \right\|_2^2$$



1. **Reshape** the reflectivity into its vector format and separate the real and imaginary values.
2. Perform a **left multiply** by the sensing matrix initialized with **random values**.
3. **Compare** the output measurement with the target measurements using the **Mean Squared Error (MSE)** and minimize using a variant of **gradient descent**.

II.A): Training Process - Inverse Scattering

Recovering inverse mapping: Given a set of **measurements** $\{d_p\}$ and their corresponding **reflectivities** $\{\varrho_p\}$ we recover the pseudo-inverse $\boldsymbol{B}$ by

$$\min_{\tilde{B}} \frac{1}{k} \sum_{\tilde{d}} \left\| \begin{bmatrix} \text{Re} & \text{-Im} \\ \text{Im} & \text{Re} \end{bmatrix} \begin{bmatrix} \text{Re} \\ \text{Im} \end{bmatrix} - \begin{bmatrix} \text{Re} \\ \text{Im} \end{bmatrix} \right\|_2^2$$

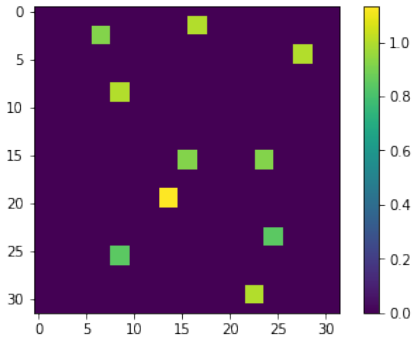
$\tilde{B}$: $2M \times 2N$; $\tilde{d}$: $2N \times 1$; $\tilde{\varrho}$: $2M \times 1$

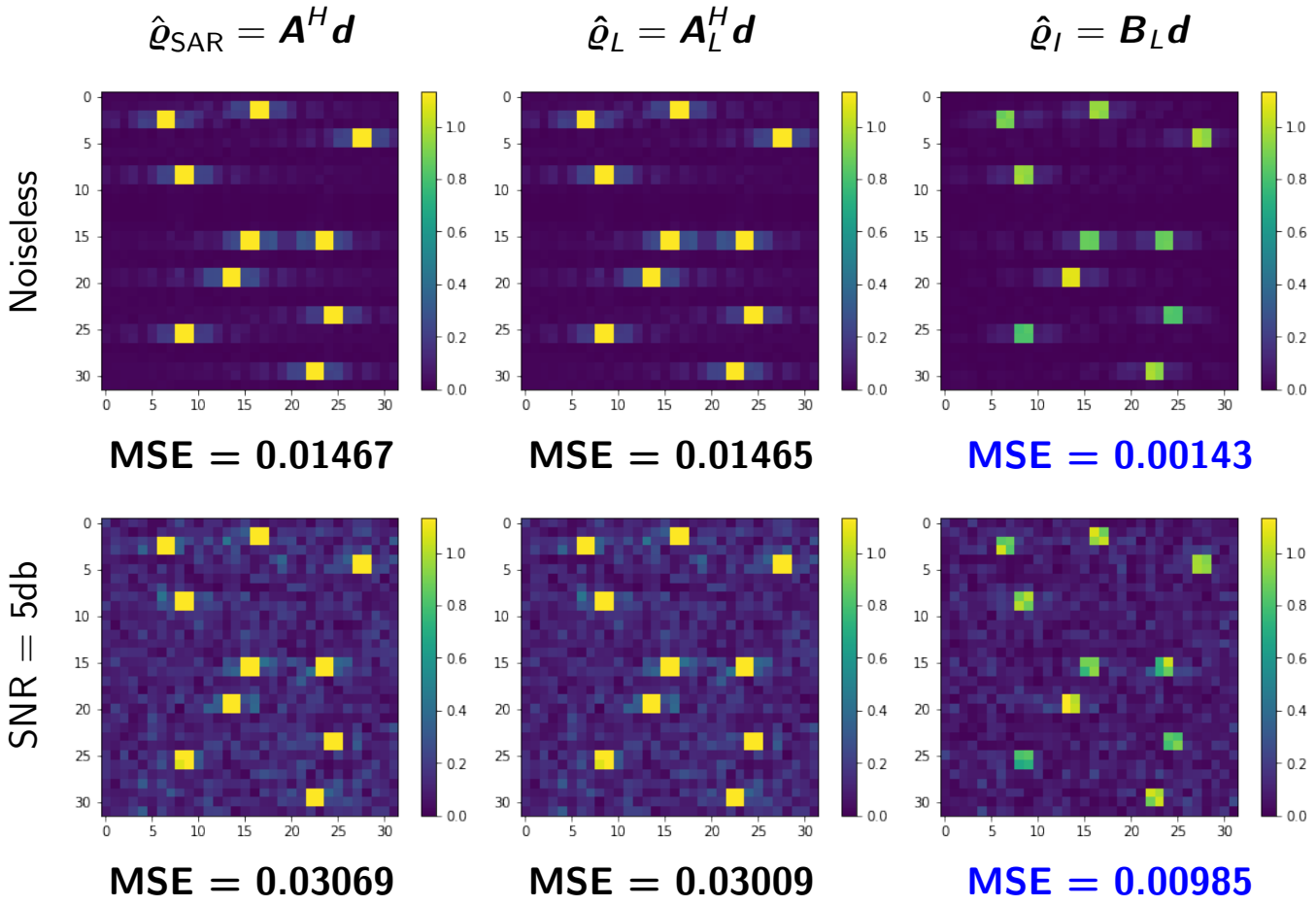
Predicted — Reflectivity

1. **Reshape** the reflectivity into its vector format and separate the real and imaginary values.
2. Perform a **left multiply** by the sensing matrix initialized with **random values**.
3. **Compare** the output measurement with the target measurements using the **Mean Squared Error (MSE)** and minimize using a variant of **gradient descent**.

II.A: Experiment 1

Low-resolution training data for low-resolution reconstructions.

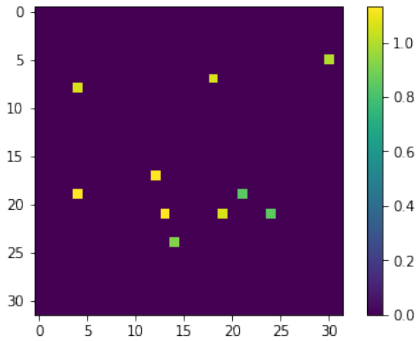


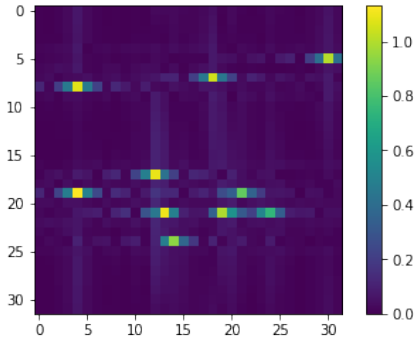
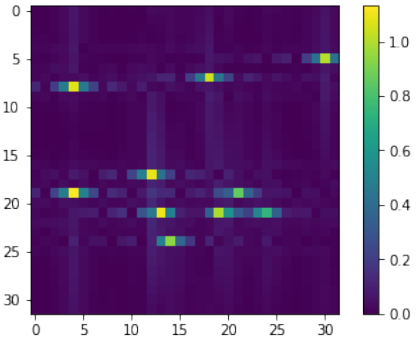
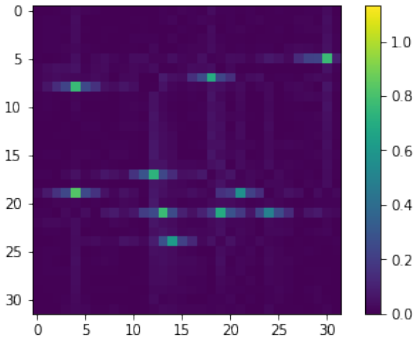
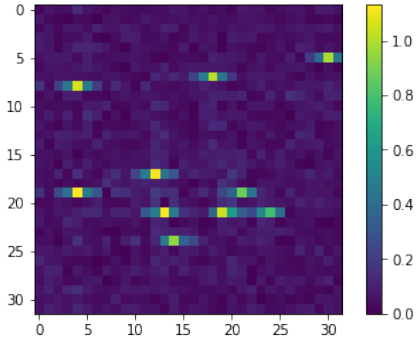
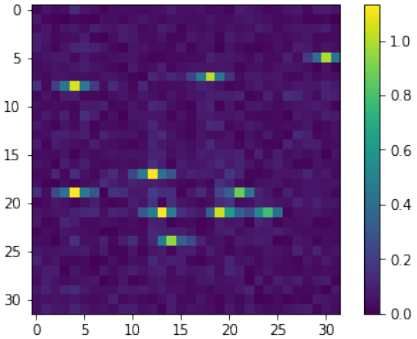
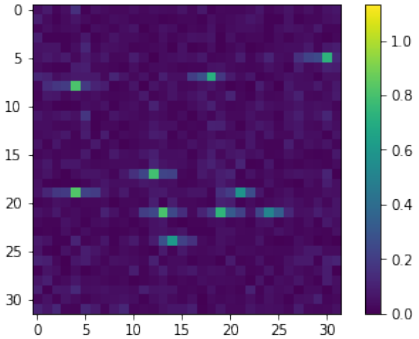


II.A: Experiment 2

Low-resolution training data for high-resolution reconstructions.

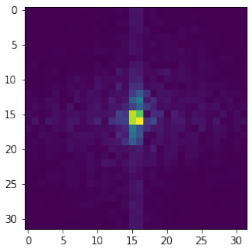
Target Image



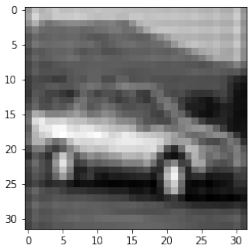
	$\hat{\varrho}_{\text{SAR}} = \boldsymbol{A}^H \boldsymbol{d}$	$\hat{\varrho}_L = \boldsymbol{A}_L^H \boldsymbol{d}$	$\hat{\varrho}_I = \boldsymbol{B}\boldsymbol{d}$
Noiseless			
	MSE = 0.00614	**MSE = 0.00604**	**MSE = 0.00309**
SNR - 5db			
	MSE = 0.00930	**MSE = 0.00910**	**MSE = 0.00479**

II.B: Image Classification in SAR

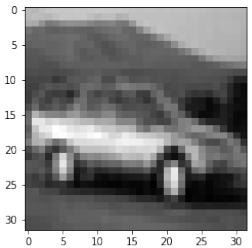
Goal: Given SAR observation data $\boldsymbol{d}$, **classify** the original unknown reflectivities $\boldsymbol{\varrho}$ using three different reconstruction methods[1].



Method I: Classify using $\boldsymbol{d}$ (raw SAR data).



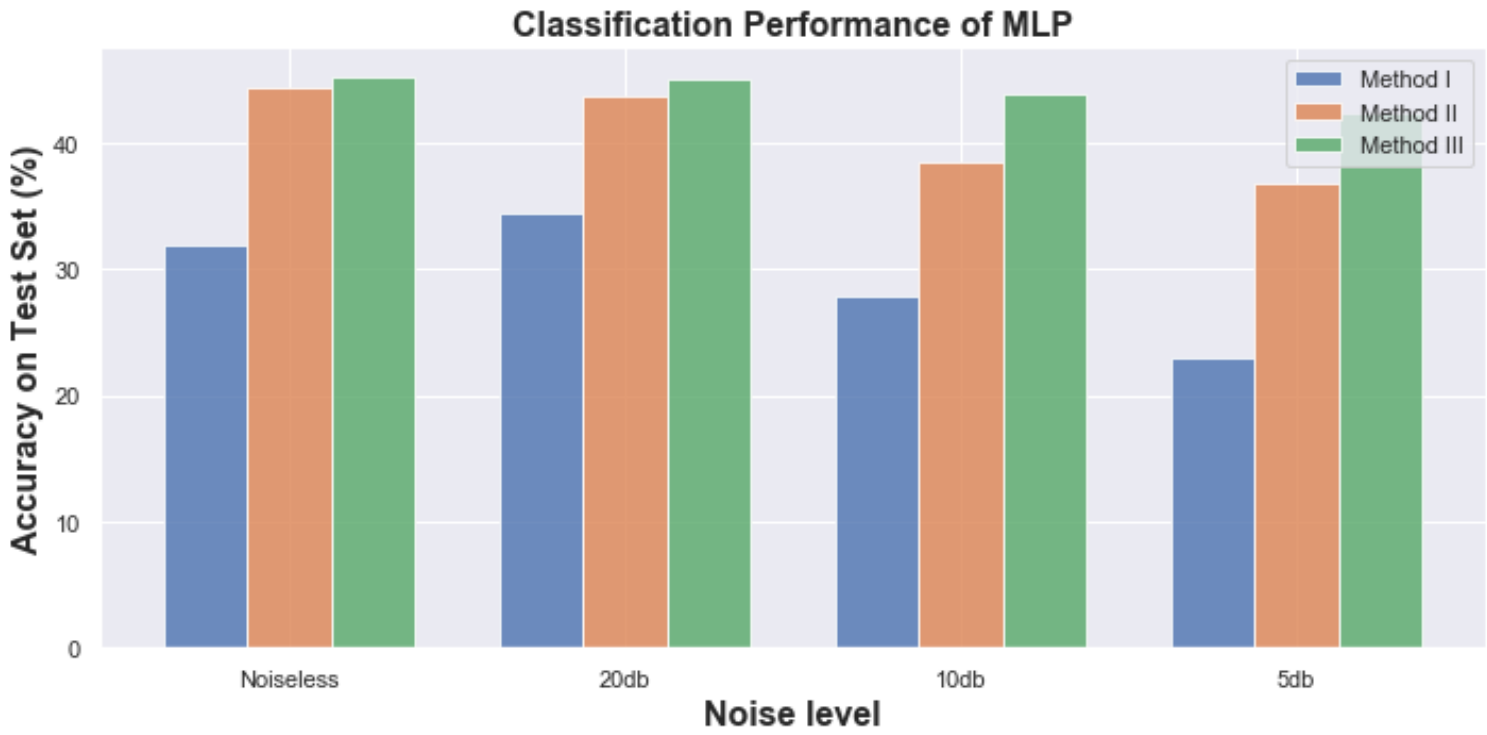
Method II: Classify using reconstructions, $\hat{\boldsymbol{\varrho}}_{\text{SAR}} = \boldsymbol{A}^H \boldsymbol{d}$.



Method III: Classify using reconstructions from the learned approximate inverse, $\hat{\boldsymbol{\varrho}}_I = \boldsymbol{B}_L \boldsymbol{d}$.

[1] J. Alvarez, O. DeGuchy and R. F. Marcia, "Image Classification In Synthetic Aperture Radar Using Reconstruction From Learned Inverse Scattering," 2020 IEEE Geoscience and Remote Sensing Symposium (IGARSS).

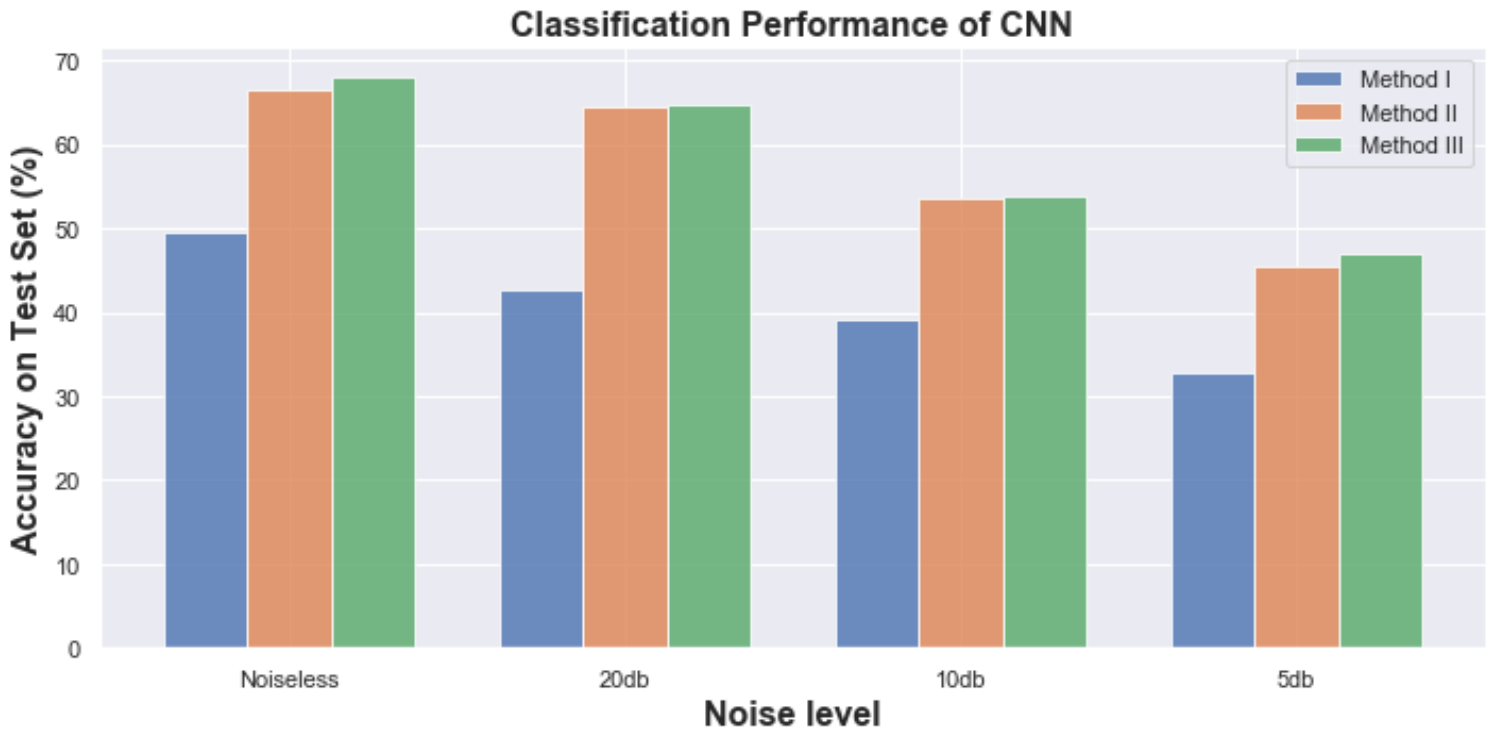
II.B: Experiment 1: Multi-Layer Perceptron (MLP)



Summary:

- Method III (learned inverse) has the highest classification accuracy.
- As noise levels increase, Method III continues to outperform the other two methods.

II.B: Experiment 2: Convolutional Neural Network (CNN)



Summary:

- Method III has the highest classification accuracy.
- CNN achieves a higher classification accuracy over MLP.

II.C: SAR Extension (ongoing work)

Goal: Improve reconstruction of SAR images using information from the learned sensing matrix and convolution neural networks.

Approach:

1. Learn $\boldsymbol{A}_L$ and compute $\hat{\boldsymbol{\varrho}}_L = \boldsymbol{A}_L^H \boldsymbol{d}$.
2. Denoise reconstruction using CNN.

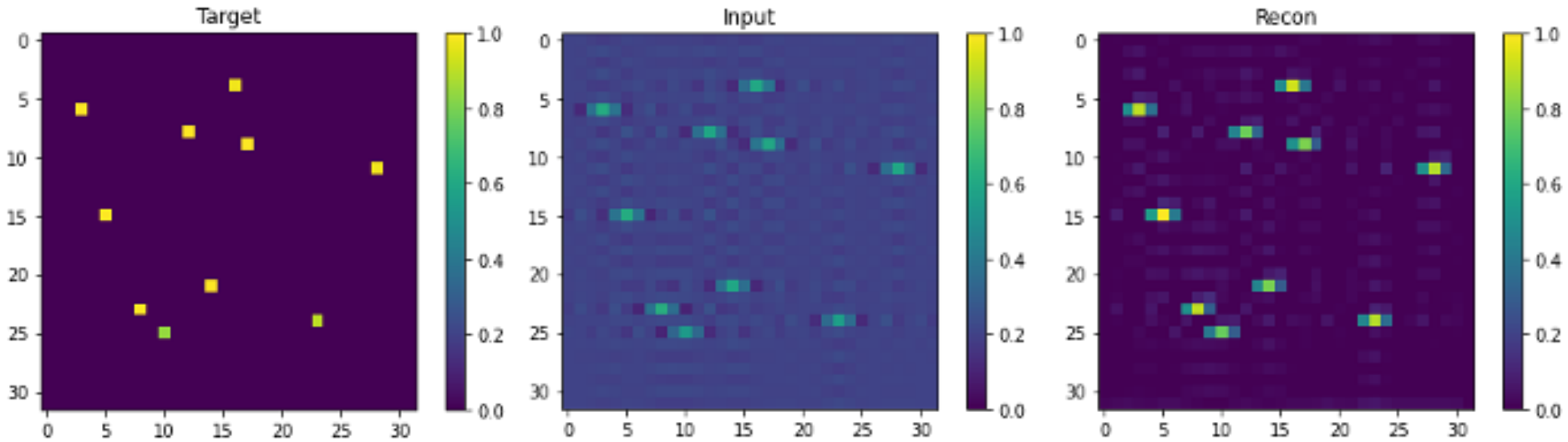
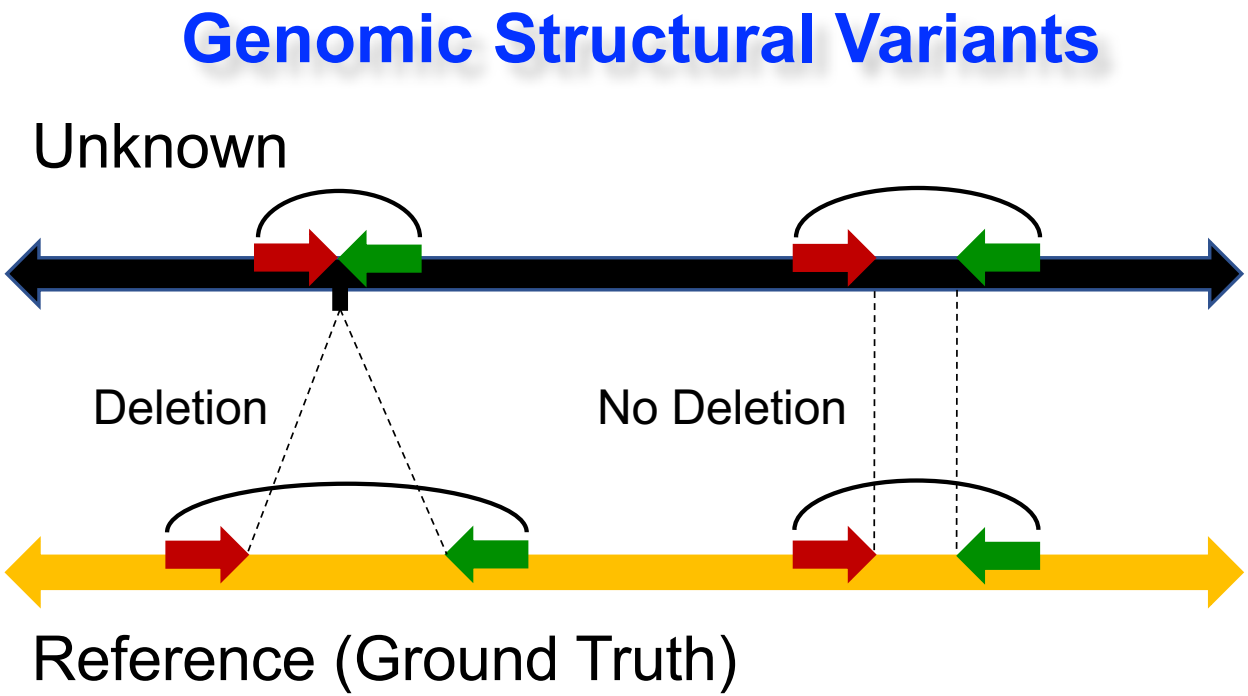


Figure: Preliminary Results

[1] Submitted abstract to 2022 SPIE Optics and Photonics.

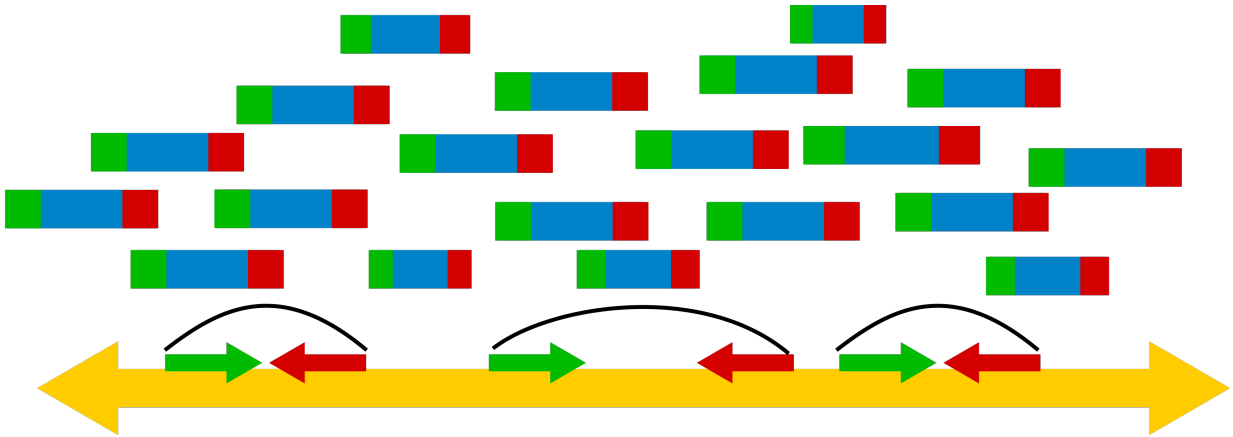
II.D: Structural Variant Genome Detection



- Structural variants (SVs) are deletions, inversions, duplications, and transpositions in genomic sequences
- Structural variants are rare occurrences
- **Goal:** Improve structural variant predictions

[1] M. Spence et al, "Detecting novel structural variants in genomes by leveraging parent-child relatedness," 2018.

II.D: Sequencing data

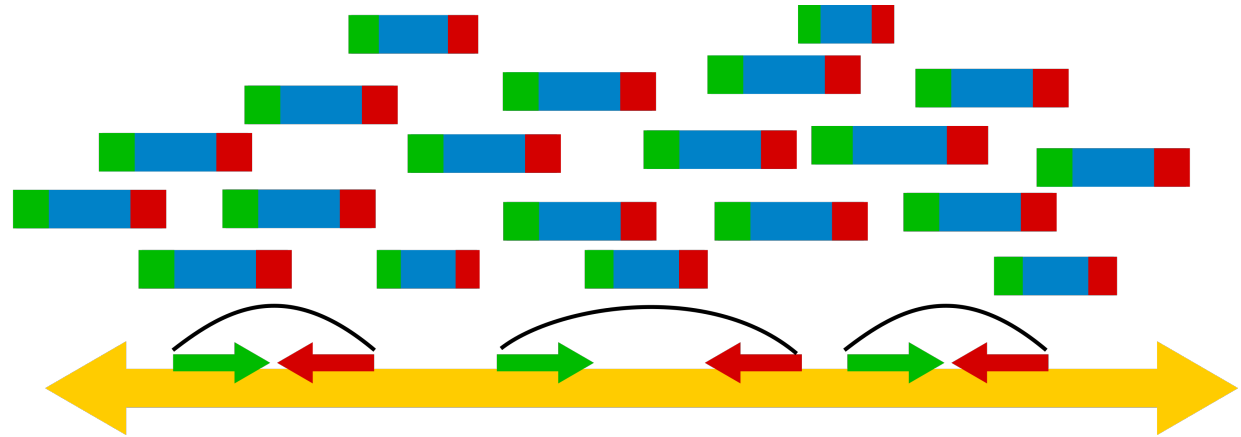


Reference (ground truth)

High coverage	Low coverage
Less susceptible to error Expensive	Noisier data Cheaper

[1] M. Spence et al, "Detecting novel structural variants in genomes by leveraging parent-child relatedness," 2018.

II.D: Sequencing data



Reference (ground truth)

High coverage	Low coverage
Less susceptible to error Expensive	Noisier data Cheaper

Our regime: Low-coverage sequencing data from related individuals.

[1] M. Spence et al, "Detecting novel structural variants in genomes by leveraging parent-child relatedness," 2018.

II.D: Detection Framework

True SV signal for each individual is a binary vector, $\vec{f}$:

$$\begin{aligned} f_i &= 1 \quad \text{if SV is present at location } i \\ f_i &= 0 \quad \text{otherwise} \end{aligned}$$

[1] M. Spence et al, "Detecting novel structural variants in genomes by leveraging parent-child relatedness," 2018.

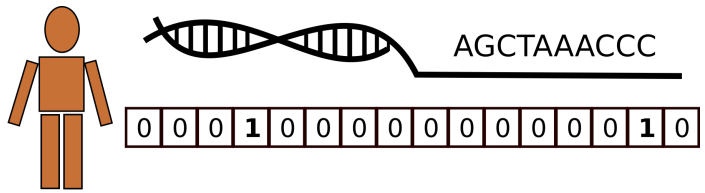
II.D: Detection Framework

True SV signal for each individual is a binary vector, $\vec{f}$:

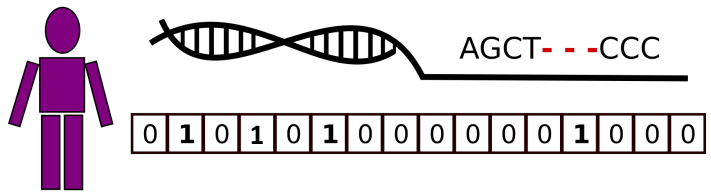
$$f_i = 1 \text{ if SV is present at location } i$$
$$f_i = 0 \text{ otherwise}$$

Assume we have sequencing data for two parents and one child:

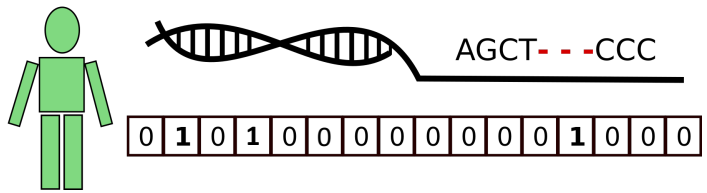
Parent 1



Parent 2



Child



[1] M. Spence et al, "Detecting novel structural variants in genomes by leveraging parent-child relatedness," 2018.

II.D: Relatedness

Assumptions:

- If both parents have an SV, the child must have the SV.
- If neither parents have an SV, the child cannot have the SV.
- If one parent has an SV, the child may have the SV.

[1] M. Spence et al, "Detecting novel structural variants in genomes by leveraging parent-child relatedness," 2018.

II.D: Relatedness

Assumptions:

- If both parents have an SV, the child must have the SV.
- If neither parents have an SV, the child cannot have the SV.
- If one parent has an SV, the child may have the SV.

Mathematically, we write these as constraints:

$$\begin{array}{ccccc} \vec{f}_{p_1} + \vec{f}_{p_2} - \vec{1} & \leq & \vec{f}_c & \leq & \vec{f}_{p_1} + \vec{f}_{p_2} \\ \vec{0} & \leq & \vec{f}_{p_1} & \leq & \vec{1} \\ \vec{0} & \leq & \vec{f}_{p_2} & \leq & \vec{1} \\ \vec{0} & \leq & \vec{f}_c & \leq & \vec{1} \end{array}$$

[1] M. Spence et al, "Detecting novel structural variants in genomes by leveraging parent-child relatedness," 2018.

II.D: Optimization framework

The optimization framework for the SV detection can be written as the following:

$$\begin{aligned} \underset{\vec{f} \in \mathbb{R}^{3n}}{\text{minimize}} \quad & F(\vec{f}) + \tau \|\vec{f}\|_1 \\ \text{subject to} \quad & \vec{f}_{p_1} + \vec{f}_{p_2} - \vec{1} \leq \vec{f}_c \leq \vec{f}_{p_1} + \vec{f}_{p_2} \\ & \vec{0} \leq \vec{f}_{p_1}, \vec{f}_{p_2}, \vec{f}_c \leq \vec{1} \end{aligned}$$

where

- $F(\vec{f})$: statistical model for genome mapping
- $\tau \|\vec{f}\|_1$: ℓ_1 penalty term where $\tau > 0$ is a regularization parameter

[1] M. Spence et al, "Detecting novel structural variants in genomes by leveraging parent-child relatedness," 2018.

II.D: Optimization framework

The optimization framework for the SV detection can be written as the following:

$$\begin{aligned} \underset{\vec{f} \in \mathbb{R}^{3n}}{\text{minimize}} \quad & F(\vec{f}) + \tau \|\vec{f}\|_1 \\ \text{subject to} \quad & \vec{f}_{p_1} + \vec{f}_{p_2} - \vec{1} \leq \vec{f}_c \leq \vec{f}_{p_1} + \vec{f}_{p_2} \\ & \vec{0} \leq \vec{f}_{p_1}, \vec{f}_{p_2}, \vec{f}_c \leq \vec{1} \end{aligned}$$

where

- $F(\vec{f})$: statistical model for genome mapping
- $\tau \|\vec{f}\|_1$: ℓ_1 penalty term where $\tau > 0$ is a regularization parameter

Approach:

1. Use second-order Taylor series to approximate $F(\vec{f})$
2. Write as a sequence of quadratic minimization problems with linear constraints

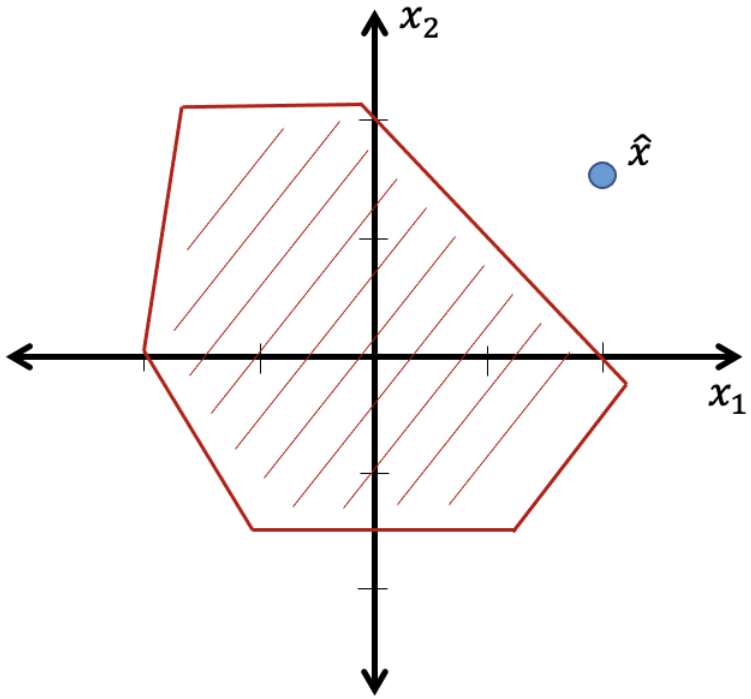
[1] M. Spence et al, "Detecting novel structural variants in genomes by leveraging parent-child relatedness," 2018.

II.D: Projection onto Affine Constraints

The sequence of quadratic minimization problems take on the following form:

$$\begin{aligned} \underset{x \in \mathbb{R}^n}{\text{minimize}} \quad & \frac{1}{2}||x - \hat{x}||_2^2 \\ \text{subject to} \quad & Ax \geq b \end{aligned} \tag{1}$$

where $\hat{x}$ might not be feasible.



II.D: Projection onto Affine Constraints

The sequence of quadratic minimization problems take on the following form:

$$\begin{aligned} \underset{x \in \mathbb{R}^n}{\text{minimize}} \quad & \frac{1}{2}||x - \hat{x}||_2^2 \\ \text{subject to} \quad & Ax \geq b \end{aligned} \tag{1}$$

where $\hat{x}$ might not be feasible.

Goal: Solve large-scale problems efficiently.

- Common approaches include interior point and active set methods.
- Alternative: Apply simpler method.

II.D: Projection onto Affine Constraints

The sequence of quadratic minimization problems take on the following form:

$$\begin{aligned} \underset{x \in \mathbb{R}^n}{\text{minimize}} \quad & \frac{1}{2}||x - \hat{x}||_2^2 \\ \text{subject to} \quad & Ax \geq b \end{aligned} \tag{1}$$

where $\hat{x}$ might not be feasible.

Goal: Solve large-scale problems efficiently.

- Common approaches include interior point and active set methods.
- Alternative: Apply simpler method.

Approach:

1. Transform primal problem (Eq. 1) into a different problem with simpler constraints.
2. Formulate Lagrangian function.

II.D: Lagrangian Function

The **Lagrangian function** $L : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$ associated with the primal problem is given by

$$L(x, \lambda) = \underbrace{\frac{1}{2}||x - \hat{x}||_2^2}_{f(x)} - \lambda^T \underbrace{(Ax - b)}_{c(x)},$$

with Lagrange multipliers $\lambda \in \mathbb{R}^m$ where $\lambda \geq 0$.

II.D: Lagrangian Function

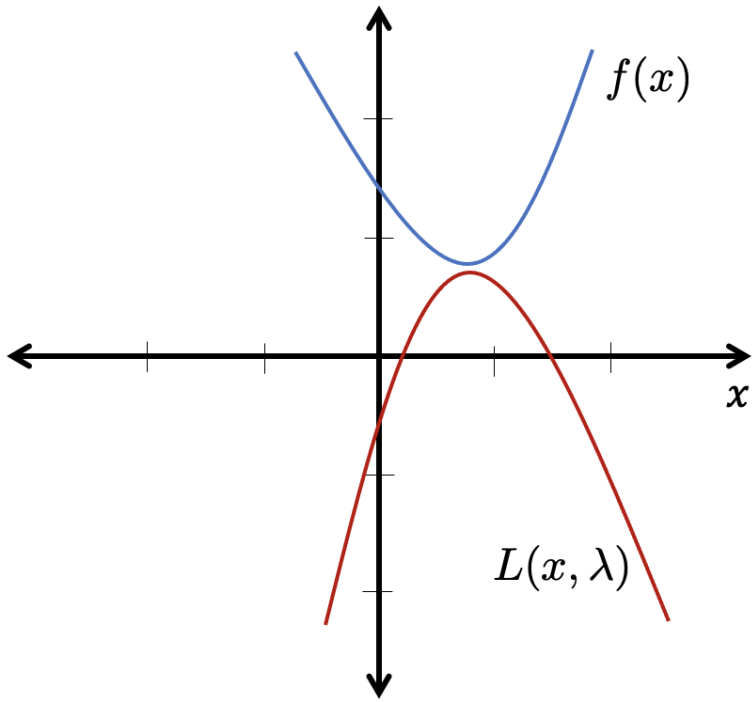
The **Lagrangian function** $L : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$ associated with the primal problem is given by

$$L(x, \lambda) = \underbrace{\frac{1}{2}||x - \hat{x}||_2^2}_{f(x)} - \lambda^T \underbrace{(Ax - b)}_{c(x)},$$

with Lagrange multipliers $\lambda \in \mathbb{R}^m$ where $\lambda \geq 0$.

Intuition:

- For $\lambda \geq 0$ and $c(x) \geq 0$: $\lambda^T c(x) \geq 0$.
- Thus, $L(x, \lambda) \leq f(x)$.
- $\underset{x,\lambda}{\text{Maximize}}\ L(x, \lambda) \iff \underset{x}{\text{Minimize}}\ f(x)$.



II.D: Lagrangian Dual Function

The **Lagrangian dual function** is defined as $g : \mathbb{R}^m \to \mathbb{R}$ where

$$g(\lambda) = \inf_{x \in \mathbb{R}^n} L(x, \lambda) = \inf_{x \in \mathbb{R}^n} \left(f(x) - \lambda^T c(x) \right)$$

Defining $g(\lambda)$: Setting $\nabla L_x(x, \lambda) = 0$, we obtain

$$x^* = \hat{x} + A^T \lambda.$$

Then substituting x^* into $L(x, \lambda)$, we find

$$g(\lambda) = \inf_{x \in \mathbb{R}^n} L(x, \lambda) = -\frac{1}{2} \lambda^T A A^T \lambda - \lambda^T (A\hat{x} - b).$$

II.(d): Dual Problem

Maximizing $g(\lambda)$ is equilavent to minimizing $-g(\lambda)$. Thus, the **dual problem** associated with our primal problem is

$$\begin{aligned} \underset{\lambda \in \mathbb{R}^m}{\text{minimize}} \quad & \tilde{g}(\lambda) = \frac{1}{2}\lambda^T A A^T \lambda + \lambda^T (A\hat{x} - b) \\ \text{subject to} \quad & \lambda \geq 0. \end{aligned}$$

Then using the solution λ^* we can define x^* from $x^* = \hat{x} + A^T \lambda$.

The **duality gap** is the difference between the primal and dual solutions,

$$f(x^*) - g(\lambda^*)$$

where x^* and λ^* are the minimizers of the primal and dual problems, respectively.

II.(d): Dual Problem

Maximizing $g(\lambda)$ is equilavent to minimizing $-g(\lambda)$. Thus, the **dual problem** associated with our primal problem is

$$\begin{aligned} \underset{\lambda \in \mathbb{R}^m}{\text{minimize}} \quad & \tilde{g}(\lambda) = \frac{1}{2}\lambda^T A A^T \lambda + \lambda^T (A\hat{x} - b) \\ \text{subject to} \quad & \lambda \geq 0. \end{aligned}$$

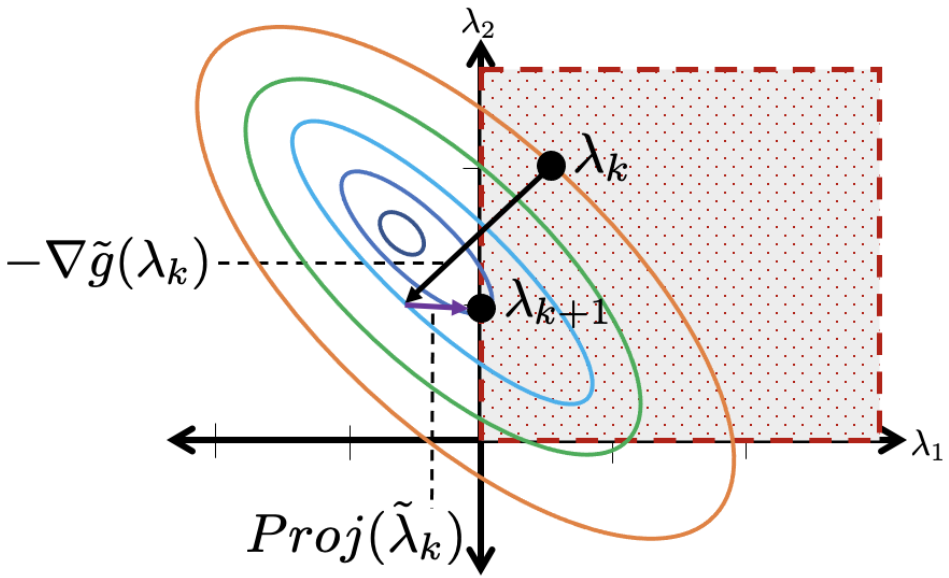
Then using the solution λ^* we can define x^* from $x^* = \hat{x} + A^T \lambda$.

Approach: Use projected gradient method where

1. Perform gradient descent on the current solution
2. Project it back onto the constraint set

II.D: Proposed Method

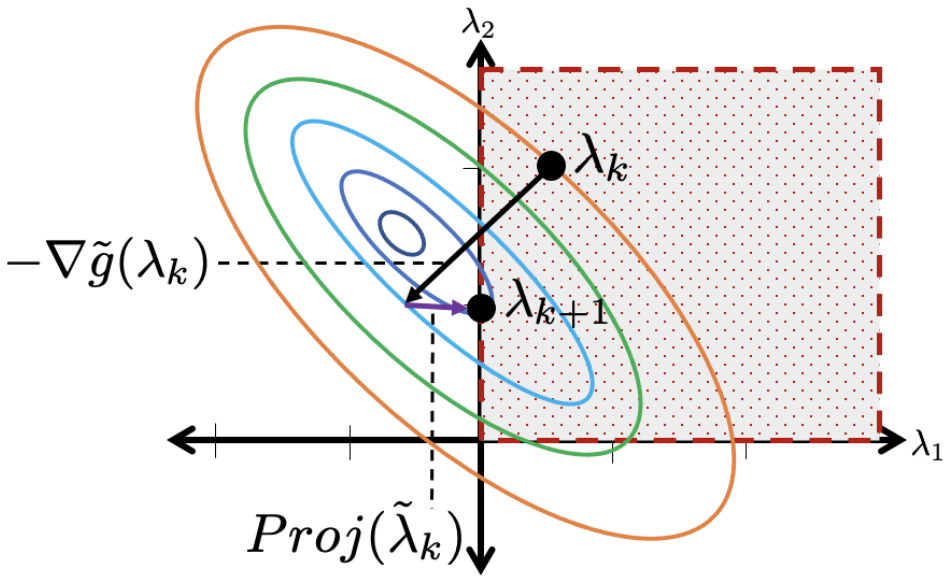
Projected Gradient Method[1]:



[0] J. Nocedal and S. Wright. Numerical optimization.

II.D: Proposed Method

Projected Gradient Method[1]:



The update on λ is given by:

$$\lambda_{k+1} = [\lambda_k - \alpha\nabla\tilde{g}(\lambda_k)]_+$$

where

- α is the step size and calculated using a backtracking line search
- $[\,\cdot\,]_+ = \max\{0, \cdot\}$

[0] J. Nocedal and S. Wright. Numerical optimization.

II.D: Numerical Experiment

We construct the set of constraints $Ax \geq b$, where $A \in \mathbb{R}^{m \times n}$ and $b \in \mathbb{R}^m$ where

- $m = 200$
- $n = 60$

Stopping criterion:

- $||x_k - x_{k-1}||_2 < \epsilon$ where $\epsilon = 10^{-5}$
- Max number of iterations is $k = 100$

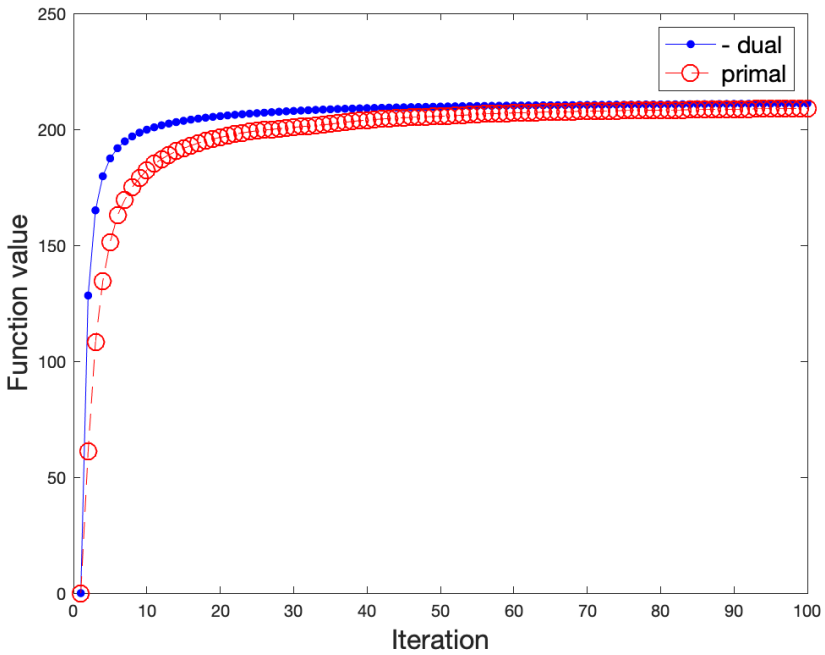
Metrics:

- Calculate duality gap: $f(x) - g(\lambda)$.
- Check infeasibility: Calculate the residual, given by $Ax - b$. Specifically,

$$\sum_{i \in \{i:(Ax-b)_i<0\}}^{m} (Ax - b)_i.$$

II.D: Results

Duality Gap



Infeasibility

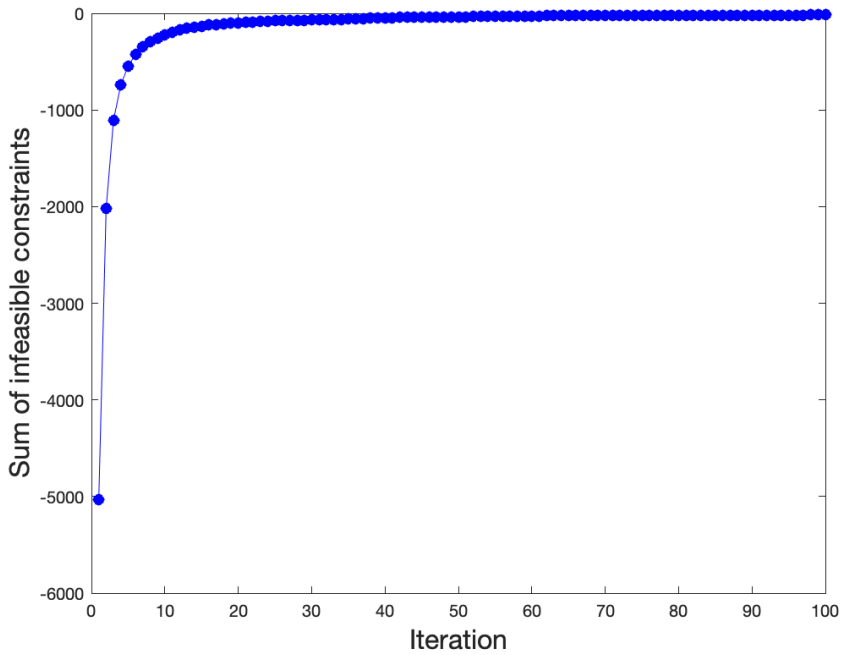


Figure: $m = 200$ and $n = 60$

Summary:

- Duality gap and infeasibility converged to zero.
- Proposed method was able to find a solution for high dimensional problems.

II.D: Extensions

Goal 1: Solve sparse reconstruction problem[1] given by

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & \frac{1}{2}\|y - Ax\|_2^2 + \tau \|x\|_1 \\ \text{subject to} \quad & Ex \geq d \end{aligned}$$

with linear constraints.

Goal 2: Solve the constrained sparse Poisson inverse problem[2] given by

$$\begin{aligned} \underset{f}{\text{minimize}} \quad & 1^T Af - \sum_{i=1}^{m} y_i \log(e_i^T Af + \beta) + \tau \|x\|_1 \\ \text{subject to} \quad & f \geq 0. \end{aligned}$$

Goal 3: Apply developed method to structural variant detection framework.

---

[1] M.A.T. Figueiredo et al, "Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems," 2007.

[2] Z.T. Harmany et al, "This is spiral-tap: Sparse poisson intensity reconstruction algorithms—theory and practice," 2011.

III. Future Work

III. Trust-region for Machine Learning

Machine learning problems, such as text or image classification, are often nonlinear and nonconvex unconstrained optimization problems given by

$$\underset{w}{\text{minimize}} \quad f(w) = \frac{1}{n}\sum_{i=1}^{n} \mathcal{L}(h(x_i, w), y_i)$$

where

n : number of training samples

w : parameter of the mapping function

x_i : feature vector of the i^{th} training sample

y_i : corresponding real-valued observation

$\mathcal{L}$: loss function

Goal: Improve the optimization method and apply to machine learning problem in limited data regimes.

III. Common Methods

The most common approaches in machine learning include:

1. Stochastic gradient descent (SGD)
   - Approximates the gradient using a **random subset** of the data.
   - **Advantage:** Computationally inexpensive.
   - **Disadvantage:** Manual hyperparameter tuning.

III. Common Methods

The most common approaches in machine learning include:

1. Stochastic gradient descent (SGD)
   - Approximates the gradient using a **random subset** of the data.
   - **Advantage:** Computationally inexpensive.
   - **Disadvantage:** Manual hyperparameter tuning.

2. Quasi-Newton approach using the limited-memory Broyden-Fletcher-Goldfarb-Shannon (L-BFGS) update
   - Quasi-Newton methods use first-order information to **approximate the second derivative**.
   - The L-BFGS updates generate a sequence of **positive definite matrices**.
   - **Advantage:** Does not require manual parameter tuning.
   - **Disadvantage:** Approximate an indefinite matrix as positive definite.

III. Common Methods

The most common approaches in machine learning include:

1. Stochastic gradient descent (SGD)
   - Approximates the gradient using a **random subset** of the data.
   - **Advantage:** Computationally inexpensive.
   - **Disadvantage:** Manual hyperparameter tuning.

2. Quasi-Newton approach using the limited-memory Broyden-Fletcher-Goldfarb-Shannon (L-BFGS) update
   - Quasi-Newton methods use first-order information to **approximate the second derivative**.
   - The L-BFGS updates generate a sequence of **positive definite matrices**.
   - **Advantage:** Does not require manual parameter tuning.
   - **Disadvantage:** Approximate an indefinite matrix as positive definite.

Proposed Approach:

- Symmetric Rank-1 Update: Not guaranteed to be positive definite.
- Trust-region: Suitable for indefinite approximations.

III. Symmetric Rank-1 (SR1) Update

A typical update of a quasi-Newton method is as follows:

$$w_{k+1} = w_k + \eta_k p_k \quad \text{where} \quad p_k = -B_k^{-1} \nabla f(w_k)$$

III. Symmetric Rank-1 (SR1) Update

A typical update of a quasi-Newton method is as follows:

$$w_{k+1} = w_k + \eta_k p_k \quad \text{where} \quad p_k = -B_k^{-1} \nabla f(w_k)$$

The symmetric rank-1 (SR1) update given by

$$B_{k+1} = B_k + \frac{(y_k - B_k s_k)(y_k - B_k s_k)^T}{(y_k - B_k s_k)^T s_k}$$

where $s_k = w_{k+1} - w_k$ and $y_k = \nabla f(w_{k+1}) - \nabla f(w_k)$.

- Unlike BFGS, the update is not guaranteed to be positive definite.
- This allows for more accurate approximations of the true Hessian matrix.
- Due to the indefinite Hessian approximation, we are not guaranteed a descent direction.
- Thus, we utilize a trust-region framework over a line search framework.

III. Line search vs. Trust-Region

Two major techniques for unconstrained optimization.

Line search methods:

1. Generate a search direction
2. Find an appropriate step size to move in search direction

Trust-region methods:

1. Define a region in which the quadratic model approximates the objective function well
2. Choose direction which minimizes the model in this region

III. Trust-Region Subproblem

At each iteration of the trust-region method, we solve what is know as the trust-region subproblem given by

$$\begin{aligned} \underset{p}{\text{minimize}} \quad & Q(p) = g^T p + \frac{1}{2} p^T B_k p \\ \text{subject to} \quad & \|p\| \leq \delta. \end{aligned}$$

where g is the gradient of f and B is an approximation of the Hessian.

[1] D.M. Gay, "Computing optimal locally constrained steps. SIAM Journal on Scientific and Statistical Computing," 1981.

[2] J.J. Moré et al, "Newton's method," 1984.

III. Trust-Region Subproblem

At each iteration of the trust-region method, we solve what is know as the trust-region subproblem given by

$$\begin{aligned} \underset{p}{\text{minimize}} \quad & Q(p) = g^T p + \frac{1}{2} p^T B_k p \\ \text{subject to} \quad & \|p\| \leq \delta. \end{aligned}$$

where g is the gradient of f and B is an approximation of the Hessian.

Formulation using 2-norm:

- Global minimizer can be characterized [1][2].
- There is no closed form solution to this subproblem.

[1] D.M. Gay, "Computing optimal locally constrained steps. SIAM Journal on Scientific and Statistical Computing," 1981.

[2] J.J. Moré et al, "Newton's method," 1984.

III. SR1 Trust-Region Approach

The SR1 trust-region approach has shown to improve performance on MNIST classification[1].

[1] J.B. Erway, et al, "Trust-region algorithms for training responses: machine learning methods using indefinite hessian approximations," 2020.

III. SR1 Trust-Region Approach

The SR1 trust-region approach has shown to improve performance on MNIST classification[1].

Motivation:

1. Improving choice of initial B_0.
2. Solving trust-region subproblem efficiently.

[1] J.B. Erway, et al, "Trust-region algorithms for training responses: machine learning methods using indefinite hessian approximations," 2020.

III. SR1 Trust-Region Approach

The SR1 trust-region approach has shown to improve performance on MNIST classification[1].

Motivation:

1. Improving choice of initial B_0.
2. Solving trust-region subproblem efficiently.

Proposed Approaches:

1. Alternative choice for B_0.
2. Use of a particular norm to obtain closed form solutions to the trust-region subproblem.

[1] J.B. Erway, et al, "Trust-region algorithms for training responses: machine learning methods using indefinite hessian approximations," 2020.

Improving L-SR1 Trust-Region Framework

Dense Initialization

The SR1 update can be written as the following:

$$B_k = B_{k-1} + \underbrace{\frac{(y_{k-1} - B_{k-1}s_{k-1})(y_{k-1} - B_{k-1}s_{k-1})^T}{(y_{k-1} - B_{k-1}s_{k-1})^T s_{k-1}}}_{\text{rank 1}}$$

Improving L-SR1 Trust-Region Framework

Dense Initialization

The SR1 update can be written as the following:

$$B_k = \underbrace{B_0}_{\gamma_k I} + \underbrace{\sum_{j=0}^{k-1} \frac{(y_j - B_j s_j)(y_j - B_j s_j)^T}{(y_j - B_j s_j)^T s_j}}_{\text{rank } k}$$

Improving L-SR1 Trust-Region Framework

Dense Initialization

The SR1 update can be written as the following[1]:

$$B_k = \underbrace{B_0}_{\gamma_k I} + \underbrace{\begin{bmatrix} \Psi_k \end{bmatrix} [\ M_k\ ] [\quad \Psi_k^T \quad]}_{\text{rank } k}$$

where

$$\Psi_k = Y_k - B_0 S_k \in \mathbb{R}^{n \times k}$$
$$Y_k = [\ y_0 \quad y_1 \quad \cdots \quad y_{k-1}\ ] \in \mathbb{R}^{n \times k}$$
$$S_k = [\ s_0 \quad s_1 \quad \cdots \quad s_{k-1}\ ] \in \mathbb{R}^{n \times k}$$
$$M_k = k \times k \text{ matrix}$$

[1] R.H. Byrd et al, "Representations of quasi-Newton matrices and their use in limited-memory methods", 1994.

Improving L-SR1 Trust-Region Framework

Dense Initialization

The SR1 update can be written as the following[1]:

$$B_k = \underbrace{B_0}_{\gamma_k I} + \underbrace{\begin{bmatrix} \Psi_k \end{bmatrix} \begin{bmatrix} M_k \end{bmatrix} \begin{bmatrix} \quad \Psi_k^T \quad \end{bmatrix}}_{\text{rank } k}$$

where

$$\begin{aligned} \Psi_k &= Y_k - B_0 S_k \in \mathbb{R}^{n \times k} \\ Y_k &= \begin{bmatrix} y_0 & y_1 & \cdots & y_{k-1} \end{bmatrix} \in \mathbb{R}^{n \times k} \\ S_k &= \begin{bmatrix} s_0 & s_1 & \cdots & s_{k-1} \end{bmatrix} \in \mathbb{R}^{n \times k} \\ M_k &= k \times k \text{ matrix} \end{aligned}$$

We compute $\Psi_k = QR$ and $M_k = W \hat{\Lambda} W^T$.

[1] R.H. Byrd et al, "Representations of quasi-Newton matrices and their use in limited-memory methods", 1994.

Improving L-SR1 Trust-Region Framework

Dense Initialization

The SR1 update can be written as the following[1]:

$$B_k = \underbrace{B_0}_{\gamma_k I} + \underbrace{\begin{bmatrix} \Psi_k \end{bmatrix} [\ R^{-1}\ ][\ W\ ][\ \hat{\Lambda}\ ]}_{P_{\|}} \underbrace{[\ W^T\ ][\ R^{-T}\ ][\quad \Psi_k^T\quad]}_{P_{\|}^T}$$

[1] R.H. Byrd et al, "Representations of quasi-Newton matrices and their use in limited-memory methods", 1994.

Improving L-SR1 Trust-Region Framework

Dense Initialization

The SR1 update can be written as the following[1]:

$$B_k = \underbrace{B_0}_{\gamma_k I} + \begin{bmatrix} P_{\|} \end{bmatrix} [\ \hat{\Lambda}\ ] [\quad P_{\|}^T \quad]$$

[1] R.H. Byrd et al, "Representations of quasi-Newton matrices and their use in limited-memory methods", 1994.

Improving L-SR1 Trust-Region Framework

Dense Initialization

The SR1 update can be written as the following[1]:

$$B_k = \underbrace{B_0}_{\gamma_k I} + \begin{bmatrix} P_{\parallel} & P_{\perp} \end{bmatrix} \begin{bmatrix} \hat{\Lambda} & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} P_{\parallel}^T \\ P_{\perp}^T \end{bmatrix}$$

where $P_{\perp}$ is the orthogonal complement of $P_{\parallel}$.

[1] R.H. Byrd et al, "Representations of quasi-Newton matrices and their use in limited-memory methods", 1994.

Improving L-SR1 Trust-Region Framework

Dense Initialization

The SR1 update can be written as the following[1]:

$$B_k = \begin{bmatrix} P_{\|} & P_{\perp} \end{bmatrix} \begin{bmatrix} \hat{\Lambda} + \gamma_k I & 0 \\ 0 & \gamma_k I \end{bmatrix} \begin{bmatrix} P_{\|}^T \\ P_{\perp}^T \end{bmatrix}$$

where $P_{\perp}$ is the orthogonal complement of $P_{\|}$.

[1] R.H. Byrd et al, "Representations of quasi-Newton matrices and their use in limited-memory methods", 1994.

Improving L-SR1 Trust-Region Framework

Dense Initialization

The SR1 update can be written as the following[1]:

$$B_k = \begin{bmatrix} P_{\|} & P_{\perp} \end{bmatrix} \begin{bmatrix} \hat{\Lambda} + \gamma_k I & 0 \\ 0 & \gamma_k I \end{bmatrix} \begin{bmatrix} P_{\|}^T \\ P_{\perp}^T \end{bmatrix}$$

where $P_{\perp}$ is the orthogonal complement of $P_{\|}$.

- Since $PP^T = I$, we write $B_0 = \gamma_k I = \gamma_k PP^T = \gamma_k P_{\|} P_{\|}^T + \gamma_k P_{\perp} P_{\perp}^T$.
- Define two different γ parameters[2] such that

$$\hat{B}_0 = \gamma_k P_{\|} P_{\|}^T + \gamma_k^{\perp} P_{\perp} P_{\perp}^T.$$

- **Approach:** Apply to L-SR1 and investigate alternative approaches to choosing $\gamma_k^{\perp}$.

[1] R.H. Byrd et al, "Representations of quasi-Newton matrices and their use in limited-memory methods", 1994.

[2] J. Brust et al, "A dense initialization for limited-memory quasi-newton methods," 2019.

III. Improving L-SR1 Trust-Region Framework

Shape-changing norm

Solve the trust-region subproblem defined by the shape-changing norm[1]:

$$\begin{aligned} \underset{p}{\text{minimize}} \quad & Q(p) = g^T p + \frac{1}{2} p^T B_k p \\ \text{subject to} \quad & \|p\|_{P,\star} \leq \delta_k \end{aligned}$$

where

$$\|p\|_{P,\star} \triangleq \max\left(\|P_{\|}^T p\|_\star, \|P_{\perp}^T p\|_2 \right)$$

- Decouples into two separate problems, each with a closed form solution.
- **Approach:** Apply to L-SR1 trust-region framework.

[1] O. Burdakov, et al, "On efficiently combining limited-memory and trust-region techniques," 2017.

III. Trust-Region: Example Application

Goal: Classify MRI images of brains to aid in prevention of Alzheimer's disease (AD) using deep learning techniques and apply proposed optimization methodology[1].

- The images focus on the identification of mild cognitive impairment (MCI).
- MCI is divided into early and late MCI (EMCI and LMCI), which are both intermediate stage before the diagnosis of AD.
- We want to classify between normal brains and those with EMCI and LMCI.
- Moreover, this classification was studied under data limitations.

[1] A. De Luna, "Classification of Mild Cognitive Impairment Under Data Limitations Using Deep Learning Techniques," 2020.

Summary

1. **Imaging Denoising Using Recurrent Neural Networks** (complete)
2. **Synthetic Aperture Radar**
   - Machine Learning for SAR (complete)
   - SAR Image Classication (complete)
   - Extension (ongoing)
3. **Projected Gradients**
   - Sparse Reconstructions (Goal 1)
   - Poisson Reconstructions (Goal 2)
   - SV Detection (Goal 3)
4. **Proposed Research**
   - Dense Initialization
   - Shape-Changing Norm
   - Applications

IV. Accomplishments

IV. Course and Special Requirements

Special Requirements	Details	Status
Preliminary Exam Requirement	Calculus, ODEs, Linear Algebra	Completed
Course Requirements	MATH 221, 222, 223, 224, 231, 232, 233, 201, 291	Completed
Special Topics Requirement	Eight units	Completed
Teaching Requirement	Two semesters	Completed
Presentation Requirement	At least one open technical oral presentation	Completed
Residency Requirement	Two semesters	Completed
Submit Written Proposal	Submit one month prior to qualifying exam	Completed
Qualifying Exam	Oral presentation and exam	In progress
Write Dissertation	Submit one month prior to defense	Needed
Defend Dissertation	Oral presentation and exam	Needed

IV. Publications

1. O. DeGuchy, J. Alvarez, A. D. Kim, R. F. Marcia and C. Tsogka, "Forward and inverse scattering in synthetic aperture radar using machine learning," Proceedings of SPIE 2020, Applications of Machine Learning.

2. J. Alvarez, O. DeGuchy and R. F. Marcia, "Image Classification In Synthetic Aperture Radar Using Reconstruction From Learned Inverse Scattering," 2020 IEEE Geoscience and Remote Sensing Symposium (IGARSS).

3. A. Ho, J. Alvarez and R. F. Marcia, "Convolution Padding in Recurrent Neural Networks for Image Denoising with Limited Data," Submitted to 2021 Asilomar Conference on Signals, Systems, and Computers.

4. J. Alvarez, A. D. Kim, R. F. Marcia and C. Tsogka, "Convolution Padding in Recurrent Neural Networks for Image Denoising with Limited Data," Abstract submitted to 2022 SPIE Optics and Phonotics.

IV. Presentations

1. "Forward and inverse scattering in synthetic aperture radar using machine learning"
   - SPIE Optics + Photonics Conference (August 2020)

2. "Image Classification In Synthetic Aperture Radar Using Reconstruction From Learned Inverse Scattering"
   - SIAM Annual - Workshop Celebrating Diversity (July 2020)
   - IEEE Geoscience and Remote Sensing Symposium (September 2020)
   - ACM Richard Tapia Conference (September 2020)
   - SACNAS National Diversity in STEM Conference (October 2020)
   - SIAM CSE - Broader Engagements Program (March 2021)

3. "Convolution Padding in Recurrent Neural Networks for Image Denoising with Limited Data"
   - ACM Richard Tapia Conference (September 2021)
   - SACNAS National Diversity in STEM Conference (October 2021)
   - Asilomar Conference on Signals, Systems, and Computers (November 2021)

IV. Accomplishments

Summer Programs/Internships:

- 2019 ATR Center Summer Internship
- 2020 ATR Center Summer Internship
- 2020 Data Science Challenge
- 2021 LLNL Data Science Summer Institute Internship

Conference Scholarships:

- 2020 SACNAS Travel Scholarship
- 2020 Tapia Travel Scholarship
- 2020 Grace Hopper Celebration Scholar
- 2021 SIAM CSE Broader Engagement Program
- 2021 SACNAS Travel Scholarship
- 2021 Tapia Travel Scholarship

IV. Accomplishments and Service

2019-2020: NSF-RTG Fellowship
UC Merced SIAM Student Chapter Secretary
Applied Mathematics Excellence in Service Award
Applied Mathematics Summer Travel Fellowship
Cal State Fresno SIAM Grad School Panelist

2020-2021: UC Merced SIAM Student Chapter Officer Vice President
Applied Mathematics Excellence in Service Award
Led Machine Learning Mini Course
Applied Mathematics Internship Recognition Award

2021-2022: NRT-IAS Fellowship
UC Merced SIAM Student Chapter Officer President
Cal State Fresno SIAM Grad School Panelist

V. Timeline

V. Timeline

Time	Tasks
Spring 2022	Complete Qualifying Exam Write manuscript for SAR Extension - Submit manuscript to SPIE (July 2022) Write manuscript for Goal 1 of projected gradients - Submit manuscript to ICASSP (October 2022)
Summer 2022	Summer Internship - Lawrence Livermore National Laboratory

V. Timeline

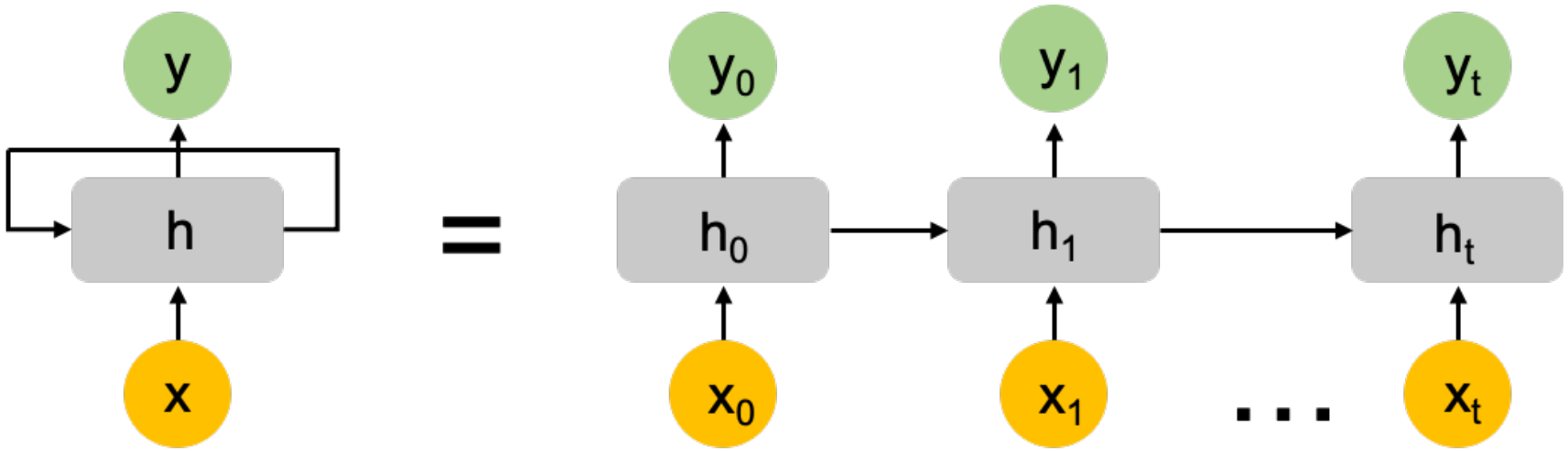
Time	Tasks
Fall 2022	Complete Goal 2 and Goal 3 for projected gradients - Write and submit manuscripts for Goals 2 and 3
Spring 2023	Implement SR1 trust-region approach in Python - Complete initial draft of manuscript
Summer 2023	Summer Internship

V. Timeline

Time	Tasks
Fall 2023	Finalize draft of trust-region and submit manuscript Apply proposed trust-region method to applications Begin initial draft of dissertation manuscript
Spring 2024	Complete dissertation manuscript Defend dissertation

Thank you!

Recurrent Neural Networks



- Use feedback loops
- Contain internal states (serves as memory)
- Used for tasks with sequential data such as speech recognition, target tracking, etc.
- Typically not used for image processing tasks

Recurrent Neural Networks

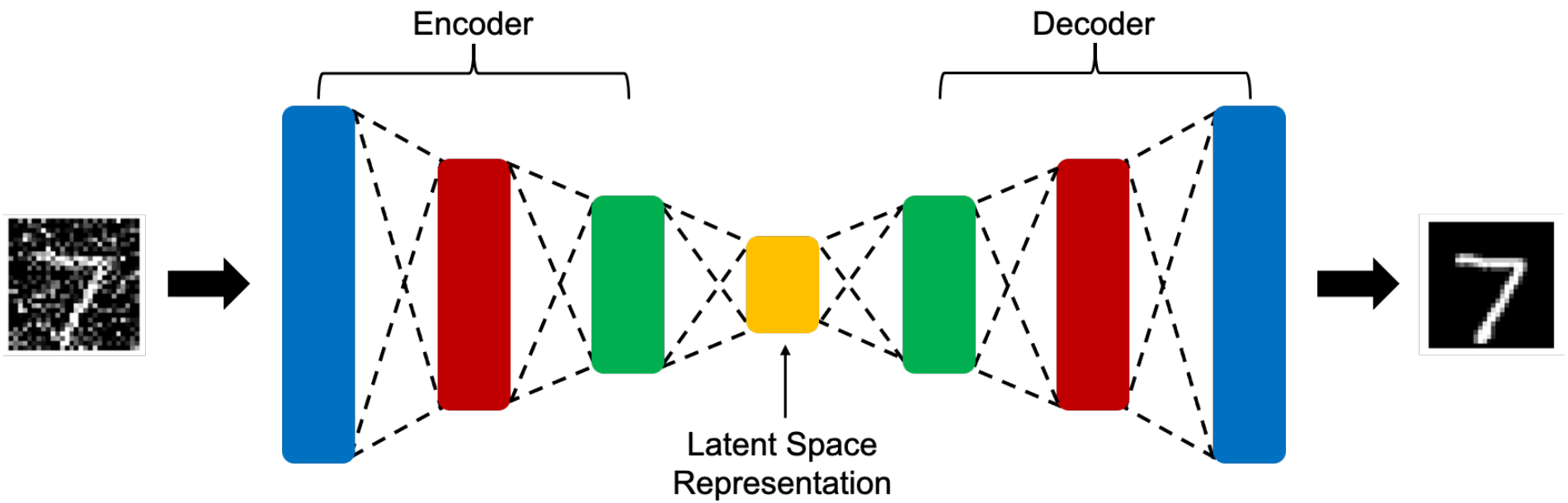
A typical RNN structure can be described by the following:

$$\mathbf{h}_t = \tanh(\mathbf{W}\mathbf{h}_{t-1} + \mathbf{U}\mathbf{x}_t + \mathbf{b})$$
$$\hat{\mathbf{y}}_t = \text{softmax}(\mathbf{V}\mathbf{h}_t + \mathbf{c})$$

where $\mathbf{W}, \mathbf{U}$ and $\mathbf{V}$ are weight matrices, $\mathbf{b}$ and $\mathbf{c}$ are bias vectors, $\mathbf{x}_t$ is the input vector and $\mathbf{h}_t$ is the current state used to find the predicted output vector $\hat{\mathbf{y}}_t$ where time goes from $t = 0$ to $t = T$.

Autoencoders

A common neural network architecture used for denoising is an autoencoder, which is composed of three parts: encoder, decoder and latent space representation.



Evaluation Criterion

After recovering the restored image we evaluate the denoising approach using the **structural similarity index (SSIM)** between the recovered image and the true image, which is given by

$$\text{SSIM}(x, y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}$$

where μ_x and μ_y are the averages of each input respectively, σ_x and σ_y are the associated variances, and c_1 and c_2 are variables used to stabilize the denominator.

Numerical Experiments: Padding

- Preliminary results suggested that we did not have enough information along the edges and corners of an image
- We investigated various types of padding to introduce more information along the edges

Original

0-Padding

Edge

Reflection

Optimization framework

Solve

$$\begin{aligned} \underset{\vec{f}\in\mathbb{R}^{3n}}{\text{minimize}} \quad & F_{\text{NegBin}}(\vec{f}) + \tau\|\vec{f}\|_1 \\ \text{subject to} \quad & \vec{f}_{p_1} + \vec{f}_{p_2} - \vec{1} \leq \vec{f}_c \leq \vec{f}_{p_1} + \vec{f}_{p_2} \\ & \vec{0} \leq \vec{f}_{p_1}, \vec{f}_{p_2}, \vec{f}_c \leq \vec{1} \end{aligned}$$

Use a second-order Taylor series expansion to approximate $F_{\text{NegBin}}(\vec{f})$ at iterate $\vec{f}^k$:
$F^k(\vec{f}) \approx F(\vec{f}^k) + (\vec{f} - \vec{f}^k)^T \nabla F(\vec{f}^k) + \frac{\alpha_k}{2}\|\vec{f} - \vec{f}^k\|_2^2$

Solve the quadratic subproblem,

$$\begin{aligned} \vec{f}^{k+1} = \underset{\vec{f}\in\mathbb{R}^{3n}}{\arg\min} \quad & \frac{1}{2}\|\vec{f} - \vec{s}^k\|_2^2 + \frac{\tau}{\alpha_k}\|\vec{f}\|_1 \\ \text{subject to} \quad & \vec{f}_{p_1} + \vec{f}_{p_2} - \vec{1} \leq \vec{f}_c \leq \vec{f}_{p_1} + \vec{f}_{p_2} \\ & \vec{0} \leq \vec{f}_{p_1}, \vec{f}_{p_2}, \vec{f}_c \leq \vec{1} \end{aligned}$$

where $\vec{s}^k = \vec{f}^k - \frac{1}{\alpha_k}\nabla F_{\text{NegBin}}(\vec{f}^k)$ and α_k is the learning rate.

(Banuelos et al, 2018)

Optimization framework

Solve

$$\begin{aligned} \underset{\vec{f}\in\mathbb{R}^{3n}}{\arg\min} \quad & \frac{1}{2}\|\vec{f}-\vec{s}^{k}\|_2^2+\frac{\tau}{\alpha_k}\|\vec{f}\|_1 \\ \text{subject to} \quad & \vec{f}_{p_1}+\vec{f}_{p_2}-\vec{1}\leq\vec{f}_c\leq\vec{f}_{p_1}+\vec{f}_{p_2} \\ & \vec{0}\leq\vec{f}_{p_1},\vec{f}_{p_2},\vec{f}_c\leq\vec{1} \end{aligned}$$

Separating the objective function,

$$\begin{aligned} \underset{f_{p_1},f_{p_2},f_c\in\mathbb{R}}{\text{minimize}} \quad & \frac{1}{2}(f_{p_1}-s_{p_1})^2+\frac{1}{2}(f_{p_2}-s_{p_2})^2+\frac{1}{2}(f_c-s_c)^2 \\ & +\frac{\tau}{\alpha_k}|f_{p_1}|+\frac{\tau}{\alpha_k}|f_{p_2}|+\frac{\tau}{\alpha_k}|f_c| \end{aligned}$$

Completing the squares,

$$\underset{f_{p_1},f_{p_2},f_c\in\mathbb{R}}{\text{minimize}} \quad \frac{1}{2}(f_{p_1}-a)^2+\frac{1}{2}(f_{p_2}-b)^2+\frac{1}{2}(f_c-d)^2$$

where $a=s_{p_1}-\frac{\tau}{\alpha_k}$, $b=s_{p_2}-\frac{\tau}{\alpha_k}$ and $d=s_c-\frac{\tau}{\alpha_k}$

(Banuelos et al, 2018)

Strong Duality

Theorem:
If the objective function is quadratic convex, and the constraints are all affine, then the duality gap is always zero, provided that one of the primal or dual problems is feasible.

Our problem:

$$\begin{aligned} \underset{x \in \mathbb{R}^n}{\text{minimize}} \quad & \frac{1}{2}||x - \hat{x}||_2^2 \\ \text{subject to} \quad & Ax \geq b \end{aligned}$$

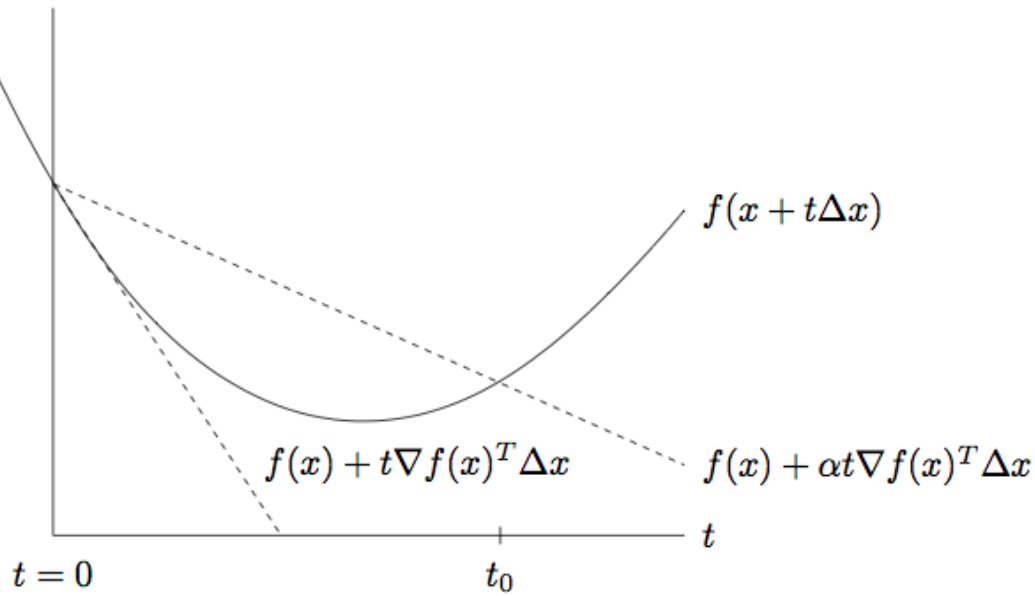
We meet the two conditions that guarantee the duality gap is zero. Therefore, finding a solution to the dual problems ensures we have found the solution to the primal problem.

Backtracking Linesearch

Let $\alpha \in (0, 1/2), \beta \in (0, 1)$. Starting at $t = 1$, repeat with $t = \beta t$ until

$$f(x + t\Delta x) < f(x) + \alpha t \nabla f(x)^T \Delta x$$

Graphical Representation:



Proposed Method

Algorithm 1: Projected Gradinet Method with Backtracking Linesearch

Initialize: $\lambda_0; \mu = 0.1; \beta = 0.5; k = 0$

while *not converged* **do**
  $\nabla \tilde{g} = AA^T\lambda + r;$
  $\alpha = \frac{||\nabla \tilde{g}(\lambda_k)||_2^2}{||A\nabla \tilde{g}(\lambda_k)||_2^2}$; /* Initialze step size */
  while *true* **do** /* Backtracking Linesearch */
    $\tilde{\lambda} = [\lambda_k - \alpha \nabla \tilde{g}(\lambda_k)]_+;$
    if $\tilde{g}(\tilde{\lambda}) \leq \tilde{g}(\lambda_k) - \mu \nabla \tilde{g}(\lambda_k - \tilde{\lambda})$ **then**
      break
    end
    $\alpha = \beta * \alpha$
  end
  $\lambda_{k+1} = \tilde{\lambda};$ /* Accept step */
  $x_{k+1} = \hat{x} + A^T\lambda_{k+1};$
end

Compact Representation

Recall the SR1 update, given by

$$B_{k+1} = B_k + \frac{(y_k - B_k s_k)(y_k - B_k s_k)^T}{(y_k - B_k s_k)^T s_k}.$$

The SR1 update can be written as using the following the compact representation:

$$B_k = B_0 + \Psi_k M_k \Psi_k^T.$$

Then, given $B_0 = \gamma_k I$ and the eigendecomposition of $\Psi_k M_k \Psi_k^T$, we can write

$$B_k = \gamma_k I + \Psi_k M_k \Psi_k^T = P \Lambda P^T$$

where

$$P \triangleq \begin{bmatrix} P_{\parallel} & P_{\perp} \end{bmatrix}, \quad \hat{\Lambda} \triangleq \begin{bmatrix} \hat{\Lambda} + \gamma_k I_r & \\ & \gamma_k I_{n-r} \end{bmatrix}$$

Shape Changing Norm

Given the compact representation, we now consider the trust-region subproblem defined by the shape-changing infinity norm:

$$\|p\|_{P,\infty} \triangleq \max\left(\|P_{\|}^T p\|_\infty, \|P_{\perp}^T p\|_2\right).$$

(Burdakov et al, 2017)

- This norm depends on the eigenvectors of B_k, thus the shape of the trust region changes each time the quasi-Newton matrix is updated.
- Defining the subproblem with this norm decouples into two separate problems, each with a closed form solution.
- Recall, when the trust region subproblem is formulated using the 2-norm, we do not have a closed form solution.

Dense Initialization

We introduced the compact representation of SR1, where the recursive formulation depends on an initialization B_0.

A common choice for B_0 is the solution to a generalized eigenvalue problem.

In (Brust et al, 2019), an, one that spans the columns of $P_{\|}$ and the other which spans the columns of $P_{\perp}$, i.e.,

$$\hat{B}_0 = \gamma_k P P^T = \gamma_k P_{\|} P_{\|}^T + \gamma_k^{\perp} P_{\perp} P_{\perp}^T.$$

The choices for γ_k are the same as before, whereas the choice of $\gamma_k^{\perp}$ is given by

$$\gamma_k^{\perp}(c, \lambda) = \lambda c \gamma_k^{\max} + (1 - \lambda)\gamma_k$$

where $c \geq 1, \lambda \in [0, 1]$, and

$$\gamma_k^{\max} \triangleq \max_{1 \leq i \leq k} \gamma_i.$$