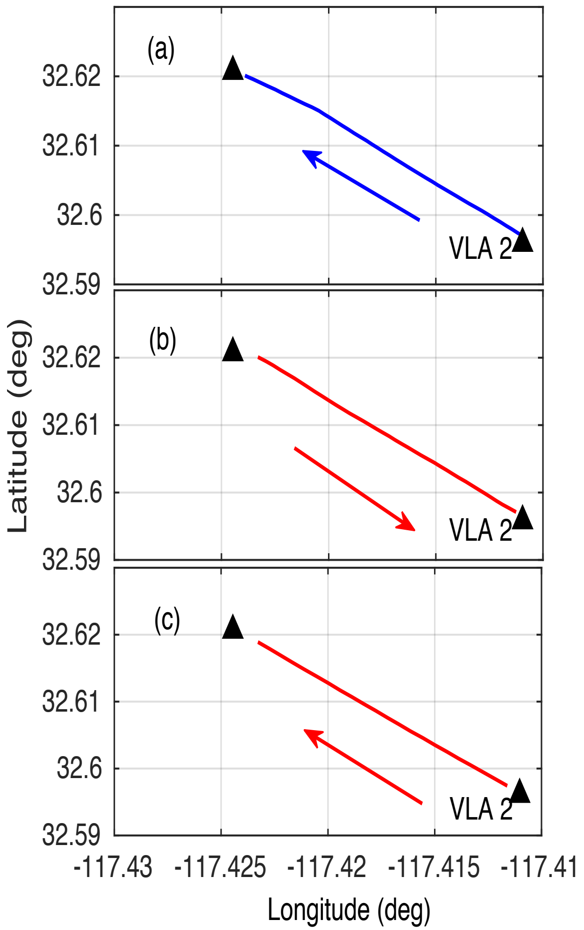
Underwater source localization using multi-frequency machine learning

Emma Ozanich
Haiqiang Niu and Peter Gerstoft
Scripps Institution of Oceanography, UCSD

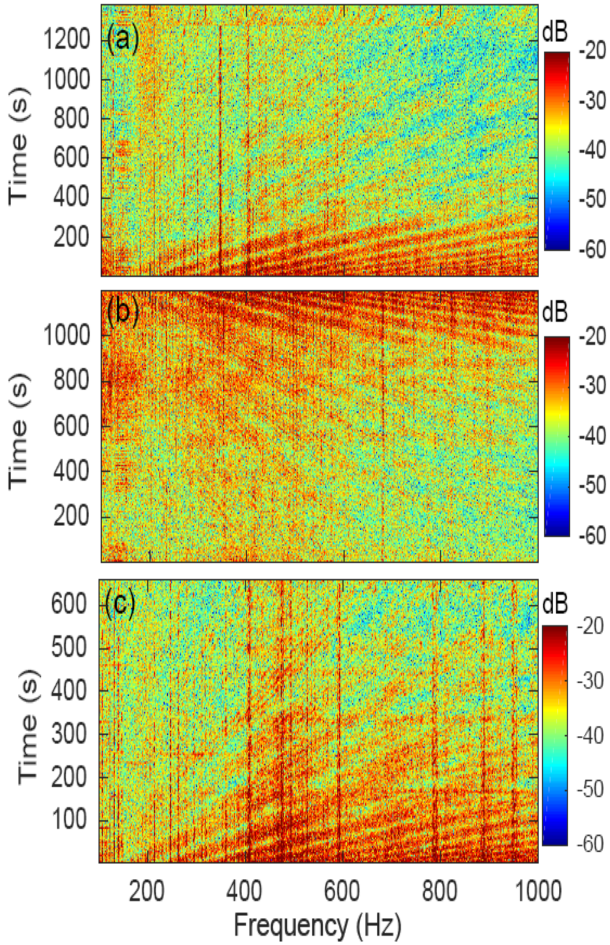
Example problem:
Ship Range for Noise 09 Experiment

- **Training data**
  - Jan. 31, 2009 01:43-2:05
  - **2 m/s**
- **Test-Data-1**
  - Jan. 31, 2009 01:01-01:24
  - **-2 m/s**
- **Test-Data-2**
  - Feb. 4, 2009 13:41-13:51
  - **4 m/s**



Example problem:
Ship Range for Noise 09 Experiment

- **Training data**
  - Jan. 31, 2009 01:43-2:05
  - 2 m/s
- **Test-Data-1**
  - Jan. 31, 2009 01:01-01:24
  - -2 m/s
- **Test-Data-2**
  - Feb. 4, 2009 13:41-13:51
  - 4 m/s



Example problem: Ship Range for Noise 09 Experiment

Comparison method: Matched-Field Processing (model-based)

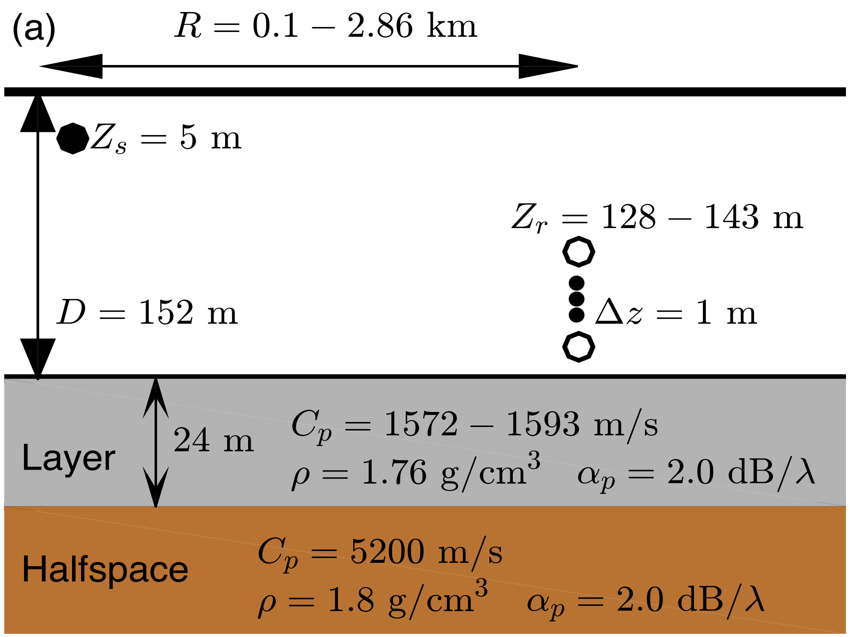
1. **Model the sound propagation.**
   Parameters: water depth, sound speed profile of water and sea floor, density, layers…

2. **Compare the modeled sound pressure (p_c) and observations (p_s) .**

maximizing $\boldsymbol{p}_c(\mathrm{r})^{\mathrm{H}}\mathbf{D}\boldsymbol{p}_c(\mathrm{r})$

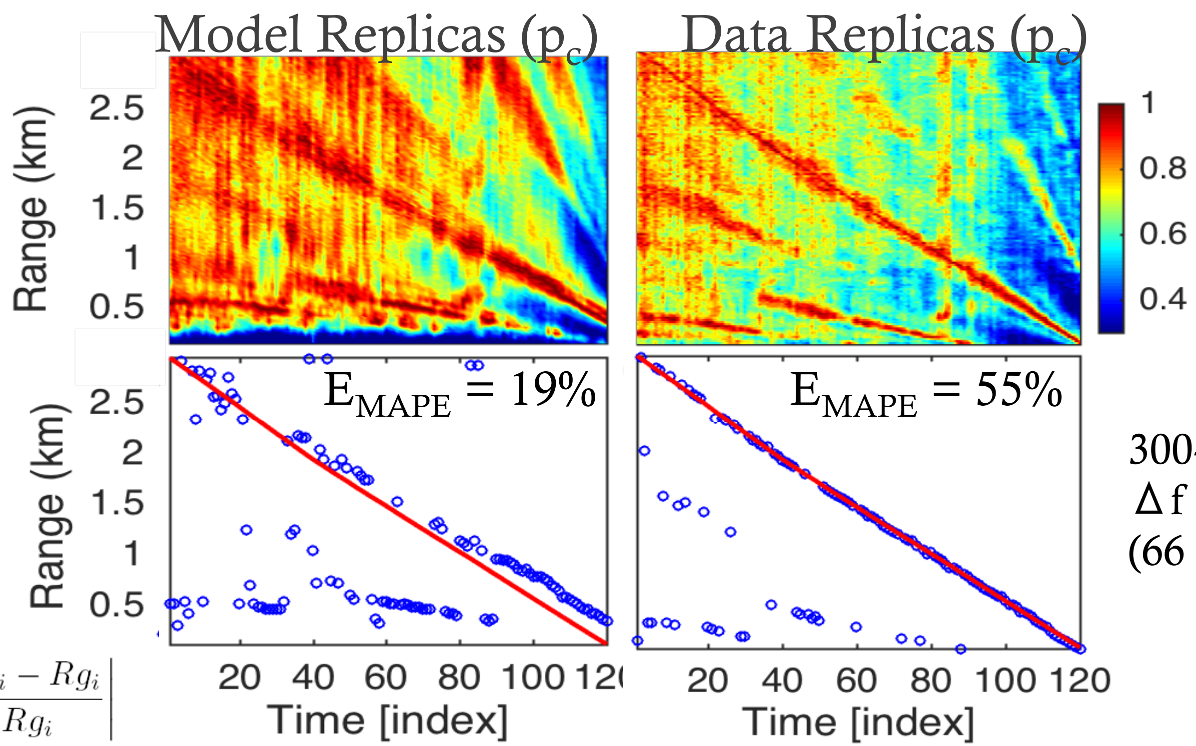
$$\mathrm{D}(f) = \frac{1}{N_s}\sum_{s=1}^{N_s}\tilde{\mathrm{p}}_s(f)\tilde{\mathrm{p}}_s^H(f)$$

16 sensors spanning from 128 – 143 m depth (15 m)



Example problem: Matched-Field Processing (MFP)

- "Ambiguity surfaces" show match at each modeled range
- MFP has challenges due to sidelobes



300-950 Hz,
Δf =10 Hz
(66 frequencies)

$$E_{\mathrm{MAPE}} = \frac{100}{N}\sum_{i=1}^{N}\left|\frac{Rp_i - Rg_i}{Rg_i}\right|$$

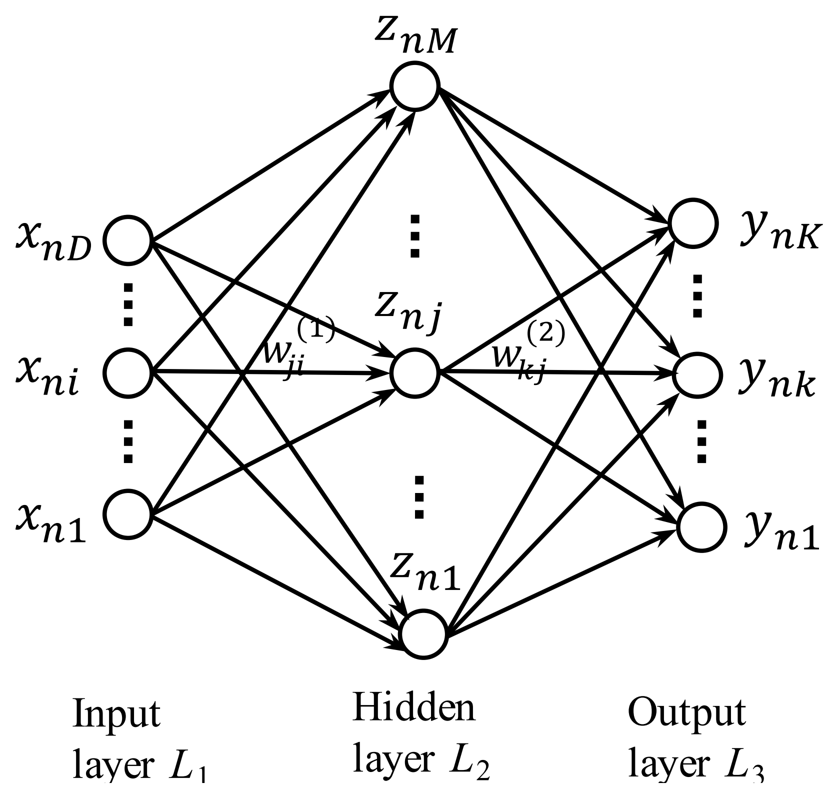
R_{pi} = predicted range, R_{gi} = ground truth range.

Example problem: Feed-Forward Neural Network (FNN)

- Feed-forward neural network, also called Multilayer Perceptron
- One hidden layer:
  - For inputs $\mathbf{x}_n$, $z_{nj} = \sigma((\mathbf{w}_j^{(1)})^T \cdot \mathbf{x}_n)$
  - $\sigma(\mathbf{x}) = \text{sigmoid}(\mathbf{x}) = (\exp(-x)+1)^{-1}$
  - Softmax output:

$$f(z_{nj}) = \frac{e^{z_{nj}}}{\sum_{k=1}^{M} e^{z_{nk}}}, j = 1, \ldots, M$$

  - Output is a probability, where maximum bin is model prediction



Example problem:
Feed-Forward Neural Network (FNN)

- Inputs for FNN (feature engineering)

Sound pressure

$$\mathbf{p}(f) = S(f)\mathbf{g}(f, \mathbf{r}) + \mathbf{n},$$

Normalize pressure to reduce the effect of $|S(f)|$

$$\tilde{\mathbf{p}}(f) = \frac{\mathbf{p}(f)}{\sqrt{\sum_{l=1}^{L} |p_l(f)|^2}} = \frac{\mathbf{p}(f)}{\|\mathbf{p}(f)\|_2}$$

Sample Covariance Matrix to reduce effect of source phase

$$\mathbf{C}(f) = \frac{1}{N_s} \sum_{s=1}^{N_s} \tilde{\mathbf{p}}_s(f) \tilde{\mathbf{p}}_s^H(f)$$

- Vectorize re{C} and imag{C}. Concatenate for multiple frequencies.

$S(f)$: Source term
L : Number of sensors
N_s : Number of snapshots

Example problem: Feed-Forward Neural Network (FNN)

- Classification:
  - Map range to bins
  - Use Kullbach-Liebler (KL, aka relative entropy) to compare softmax output with 'one-hot' vector

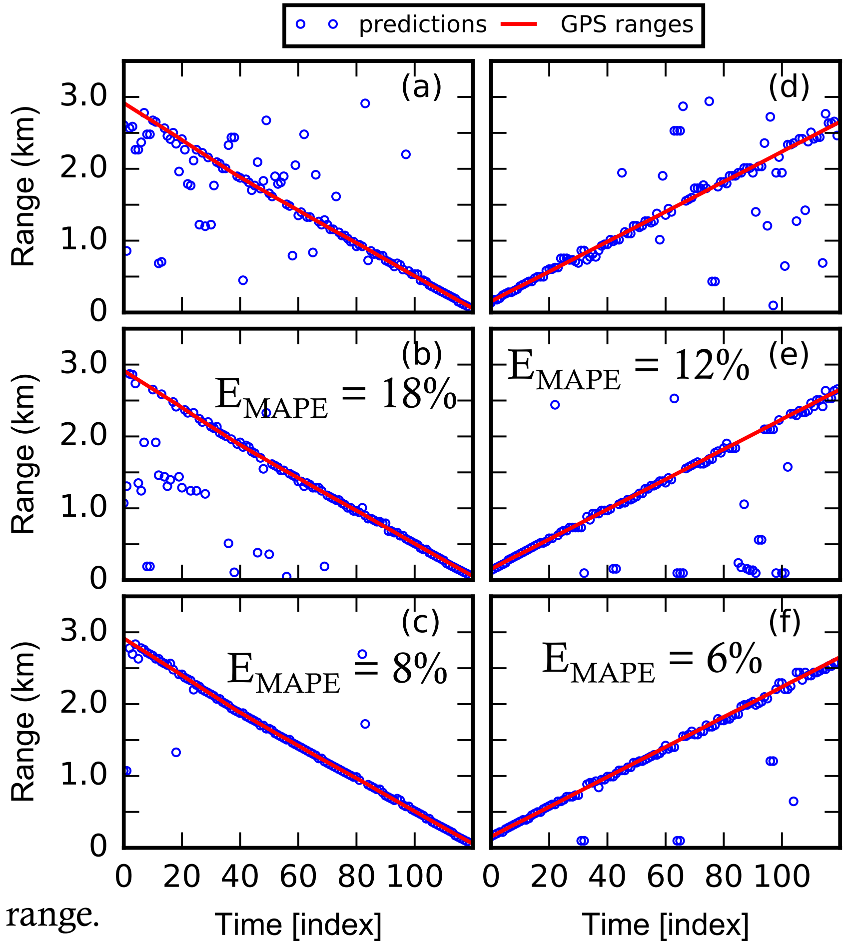
$$\begin{array}{lcccccccc}
r_1: & 1 & 0 & 0 & \ldots & 0 & 0 & 0 & 0 \\
r_2: & 0 & 1 & 0 & \ldots & 0 & 0 & 0 & 0 \\
r_3: & 0 & 0 & 1 & \ldots & 0 & 0 & 0 & 0 \\
\vdots & & & & & & & & \\
r_K: & 0 & 0 & 0 & \ldots & 0 & 0 & 0 & 1
\end{array}$$

$$\underset{\textbf{Input}}{\mathbf{x}_n} \longrightarrow \underset{\textbf{Label}}{\mathbf{t}_n}$$

$$t_{nk} = \begin{cases} 1 & \text{if } ||r - r_k|| \leq \Delta r/2, \\ 0 & \text{otherwise.} \end{cases}$$

Example problem:
Feed-Forward Neural Network (FNN)

- (a)-(c) Test-Data-1
- (d)-(f) Test-Data-2



550 Hz

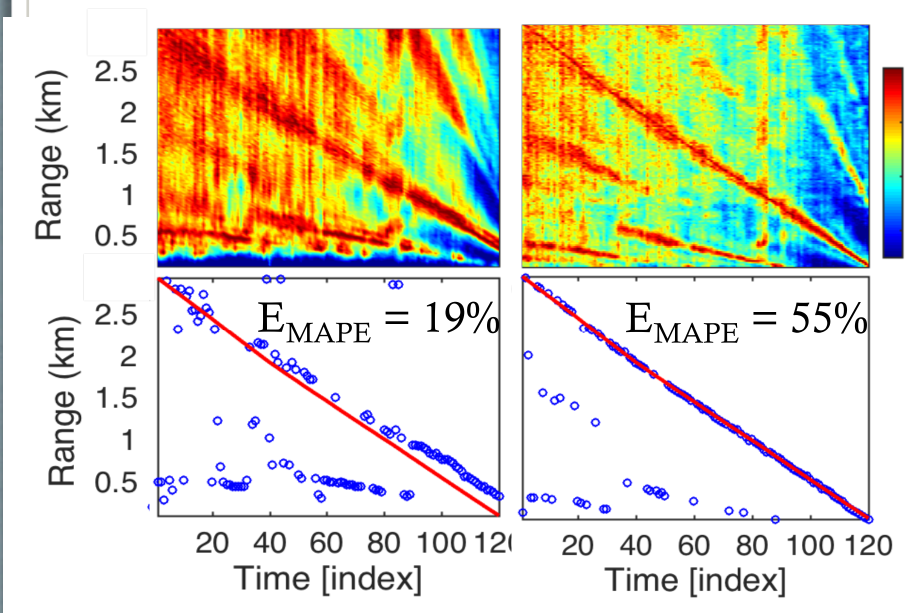
950 Hz

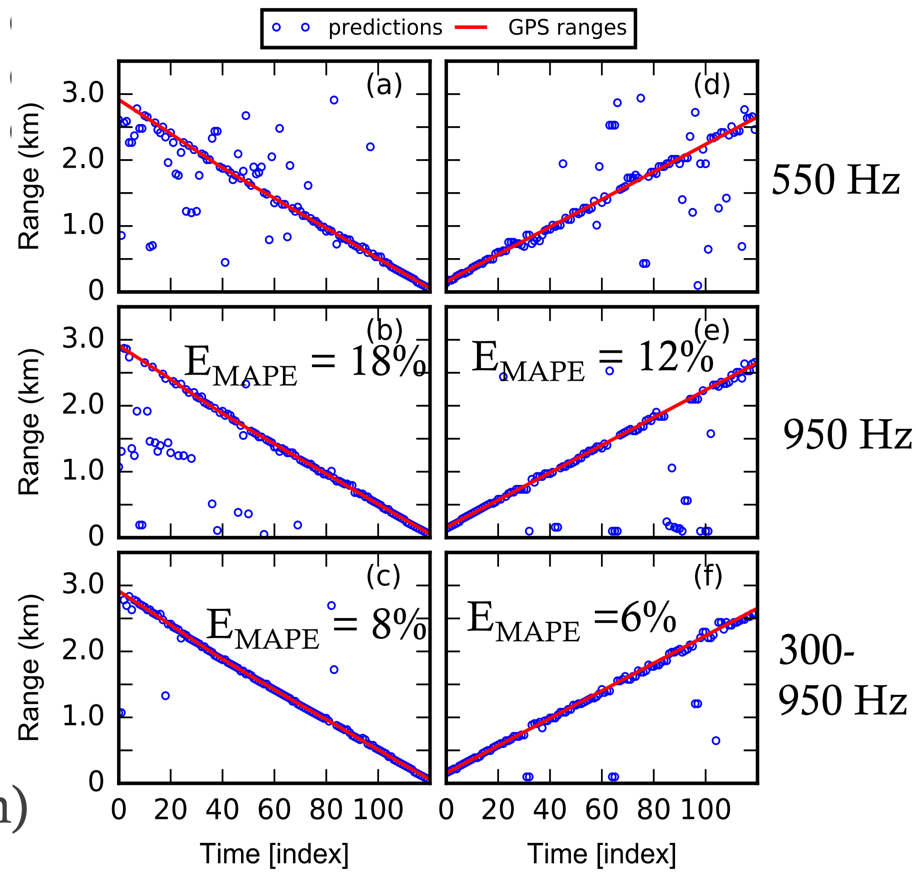
300-950 Hz,
Δ f =10 Hz
(66 frequencies)

$$E_{\mathrm{MAPE}} = \frac{100}{N} \sum_{i=1}^{N} \left| \frac{Rp_i - Rg_i}{Rg_i} \right|$$

R_{pi} = predicted range, R_{gi} = ground truth range.

Example problem: Ship Range for Noise 09 Experiment



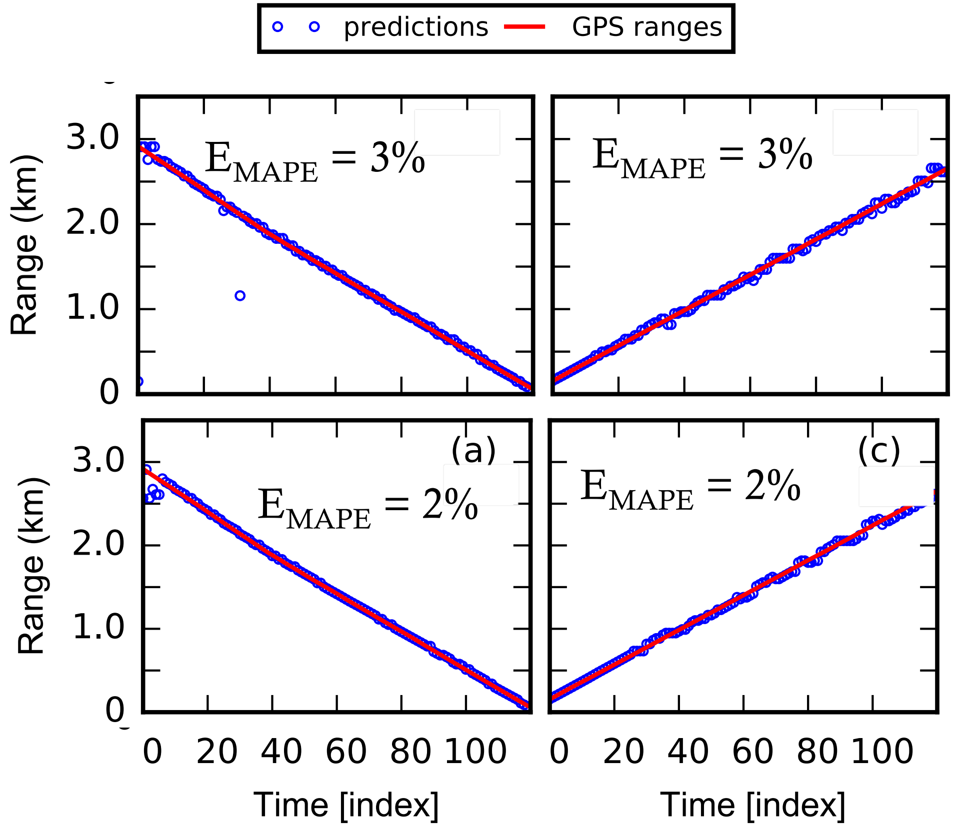


- (a)-(c) Test-Data-1
- (d)-(f) Test-Data-2
- Best FNN: E_{MAPE} = 3% (not shown)

Example problem: Support Vector and Random Forest

Random Forest (RF)

Support Vector Machine (SVM)



Example problem:
Ship Range for SBCEx16 Experiment

- Cargo shipping lanes, 3 passing ships

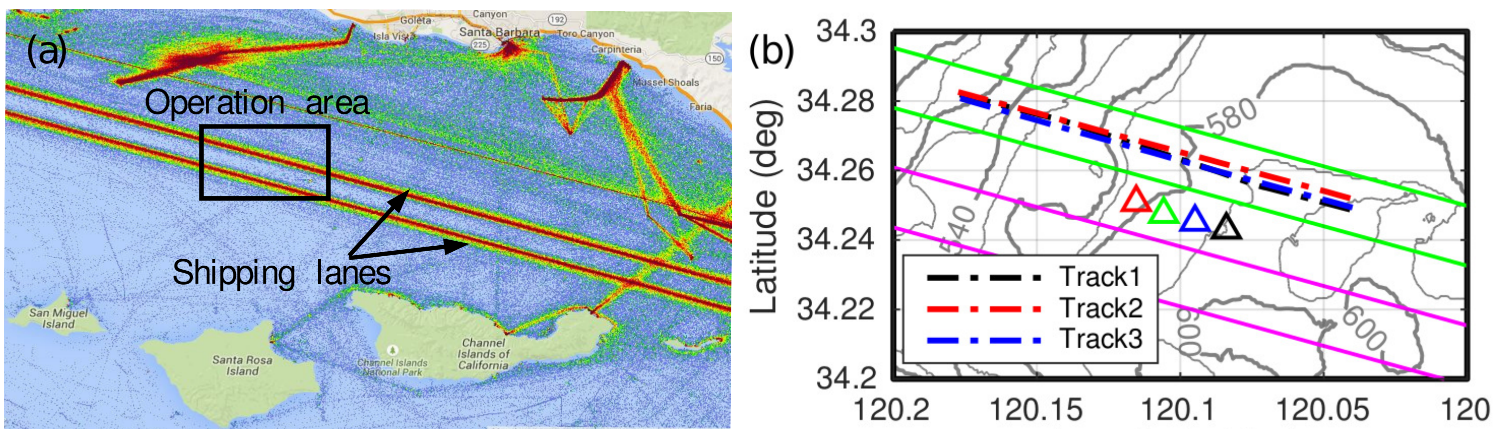
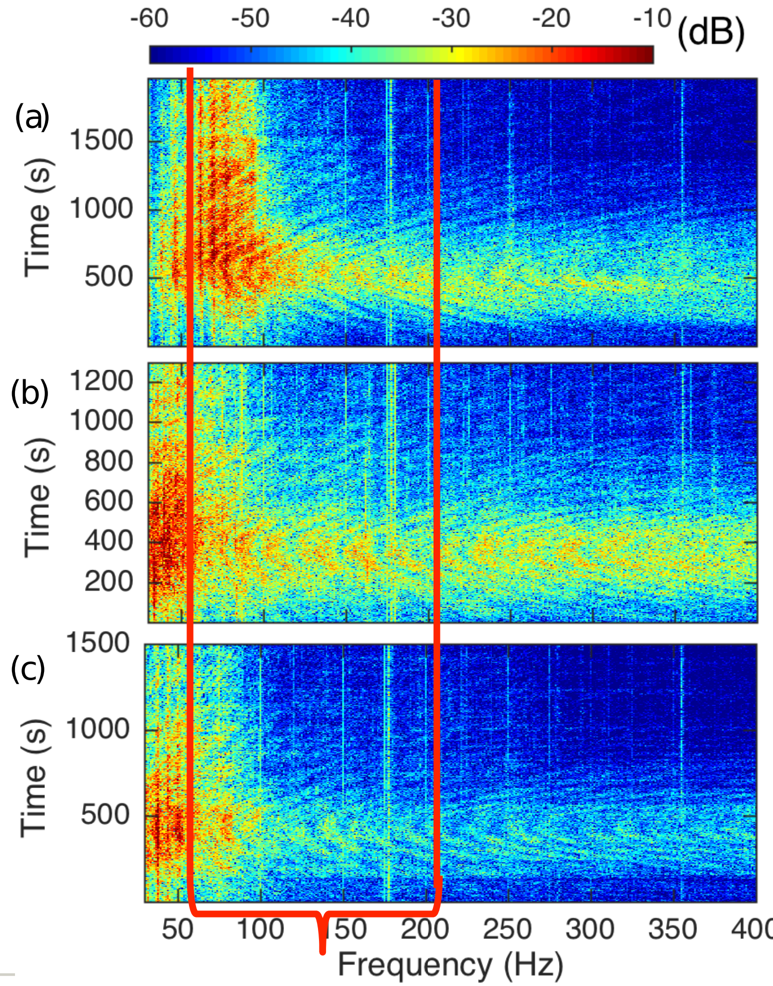


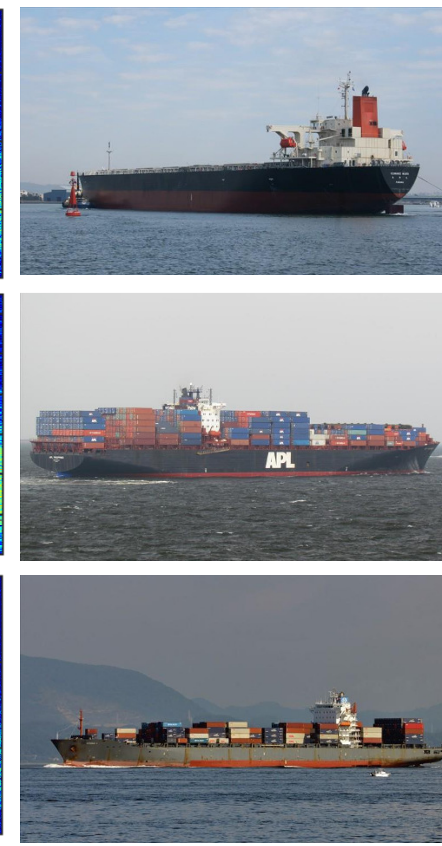
Table 1. Ship tracks.

Track No.	Data set	Time period	Ship name	Speed (m/s)
Track1	Training-Data	13:00–13:33 (9/15)	KUMANO MARU	6.7
Track2	Test-Data-1	19:11–19:33 (9/16)	APL PHILIPPINES	10
Track3	Test-Data-2	19:29–19:54 (9/17)	NORDSPRING	8.0

Example problem:
Ship Range for SBCEx16 Experiment

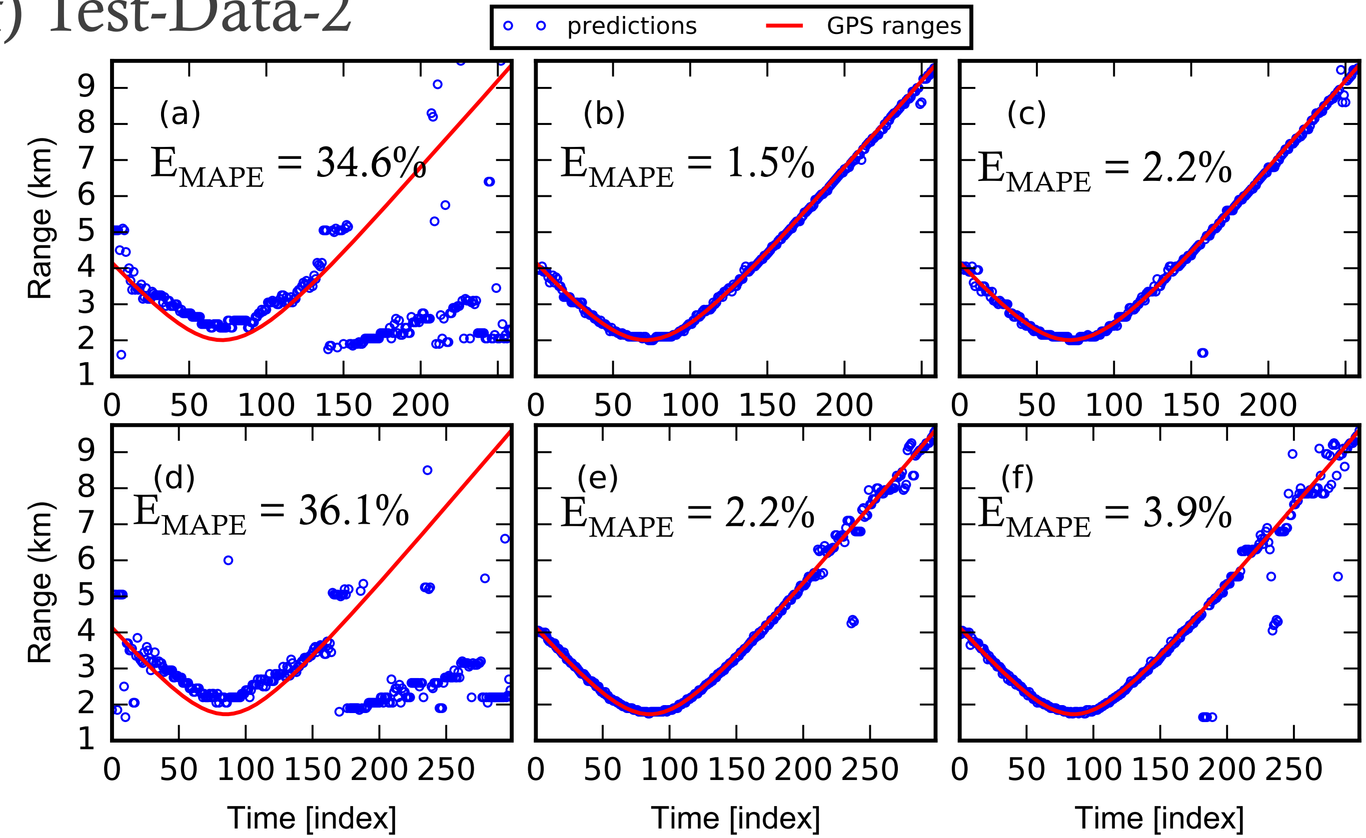
- Cargo ship spectral signatures
- Use 53-200 Hz, spacing of 3 Hz





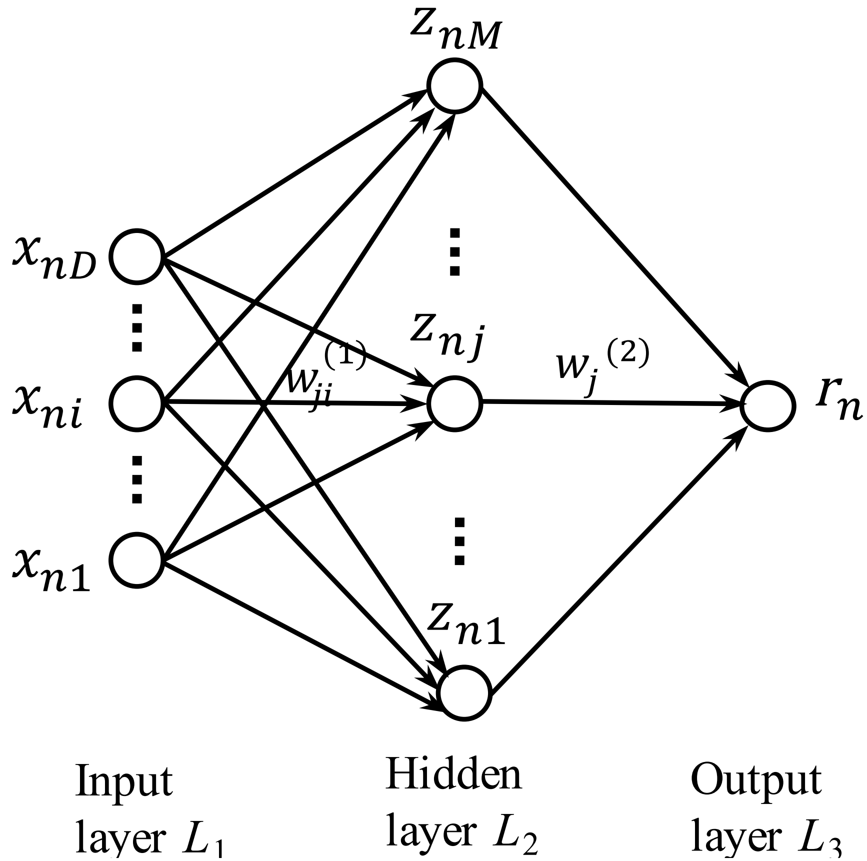
Example problem: Ship Range for SBCEx16 Experiment

- (a)-(c) Test-Data-1. (a),(d) MFP; (b),(d) SVM; (c),(f) FNN.
- (d)-(f) Test-Data-2



Feed-Forward Neural Network: Regression

- FNN with regression:
  - Single output is estimate of range
  - Classification maximally separates classes while regression minimizes target error
  - Mean squared error



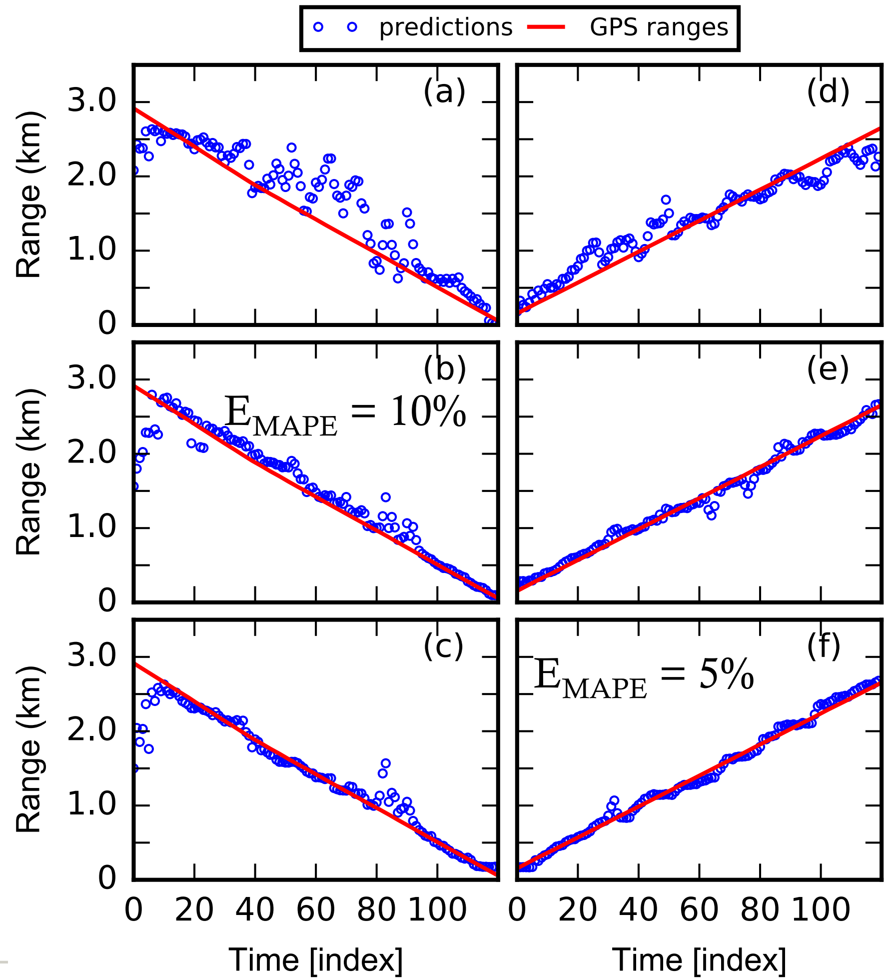
Noise 09 Results: FNN with Regression

- (a)-(c) Test-Data-1
- (d)-(f) Test-Data-2

One hidden layer

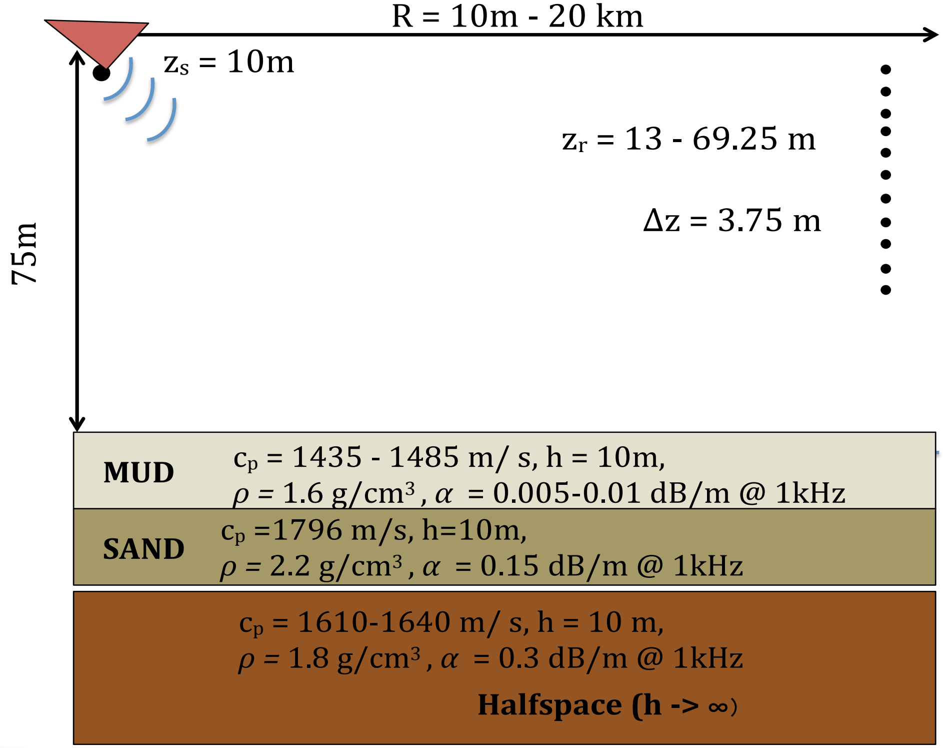
Two hidden layers

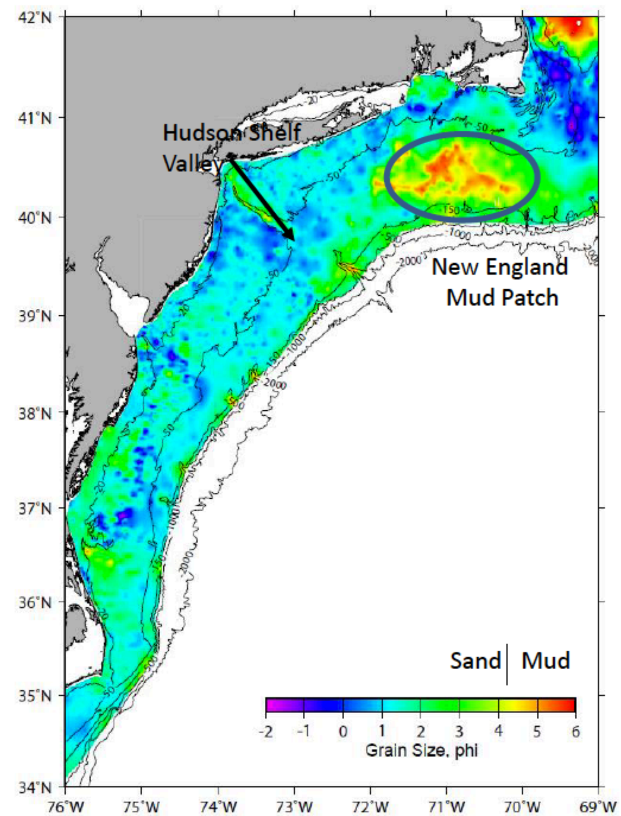
Three hidden layers



Example problem:
Multiple ships at SCE17

16 sensors spanning from 13m to 69.25m (56.25m)

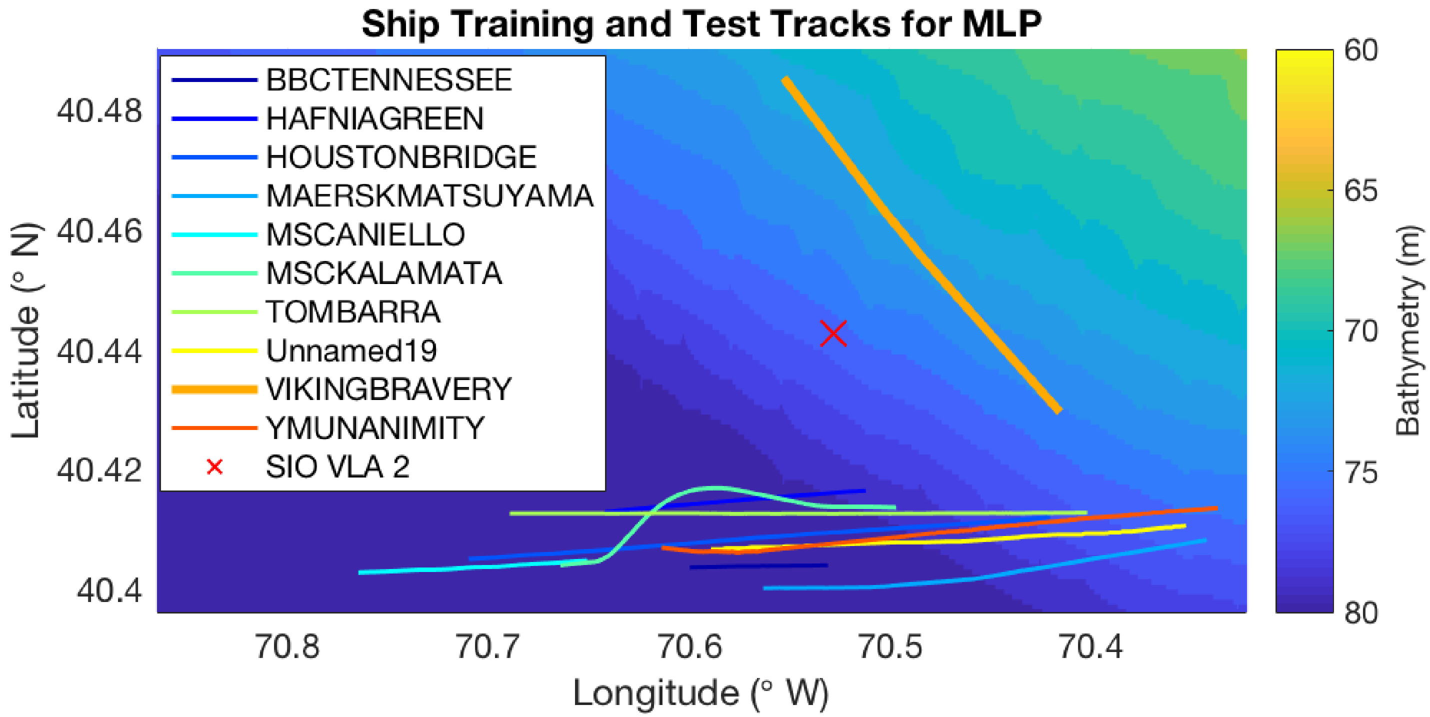




credit: John Goff

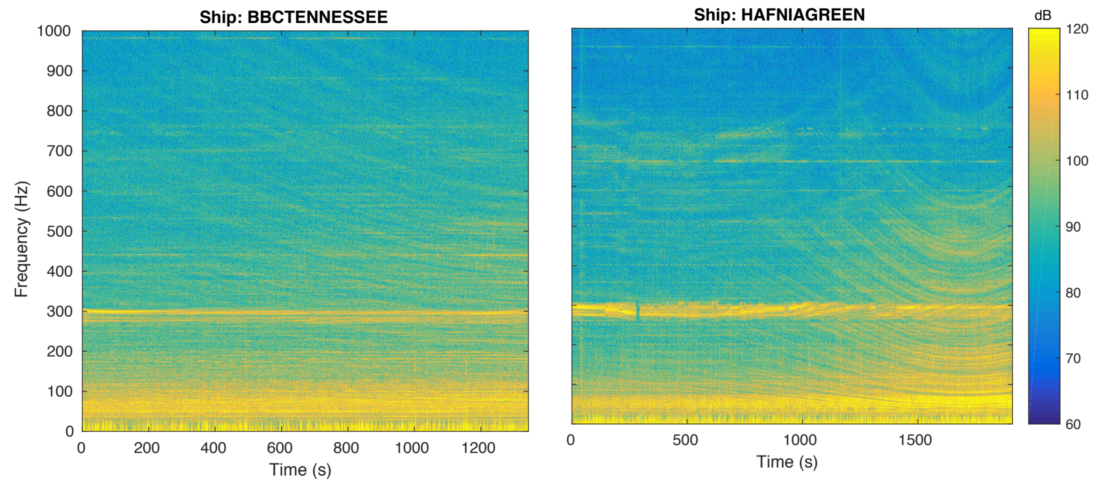
Example problem: Multiple ships at SCE17

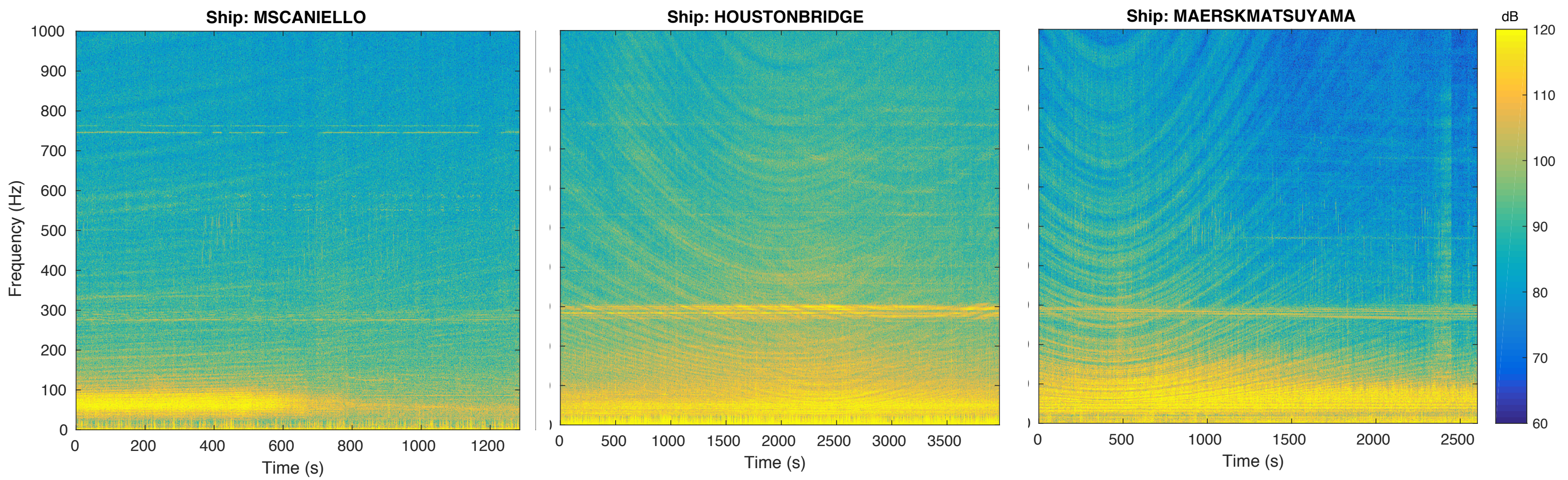
- 9 training tracks, 1 test track (bold, orange)
- Test track from different region



Example problem: Multiple ships at SCE17

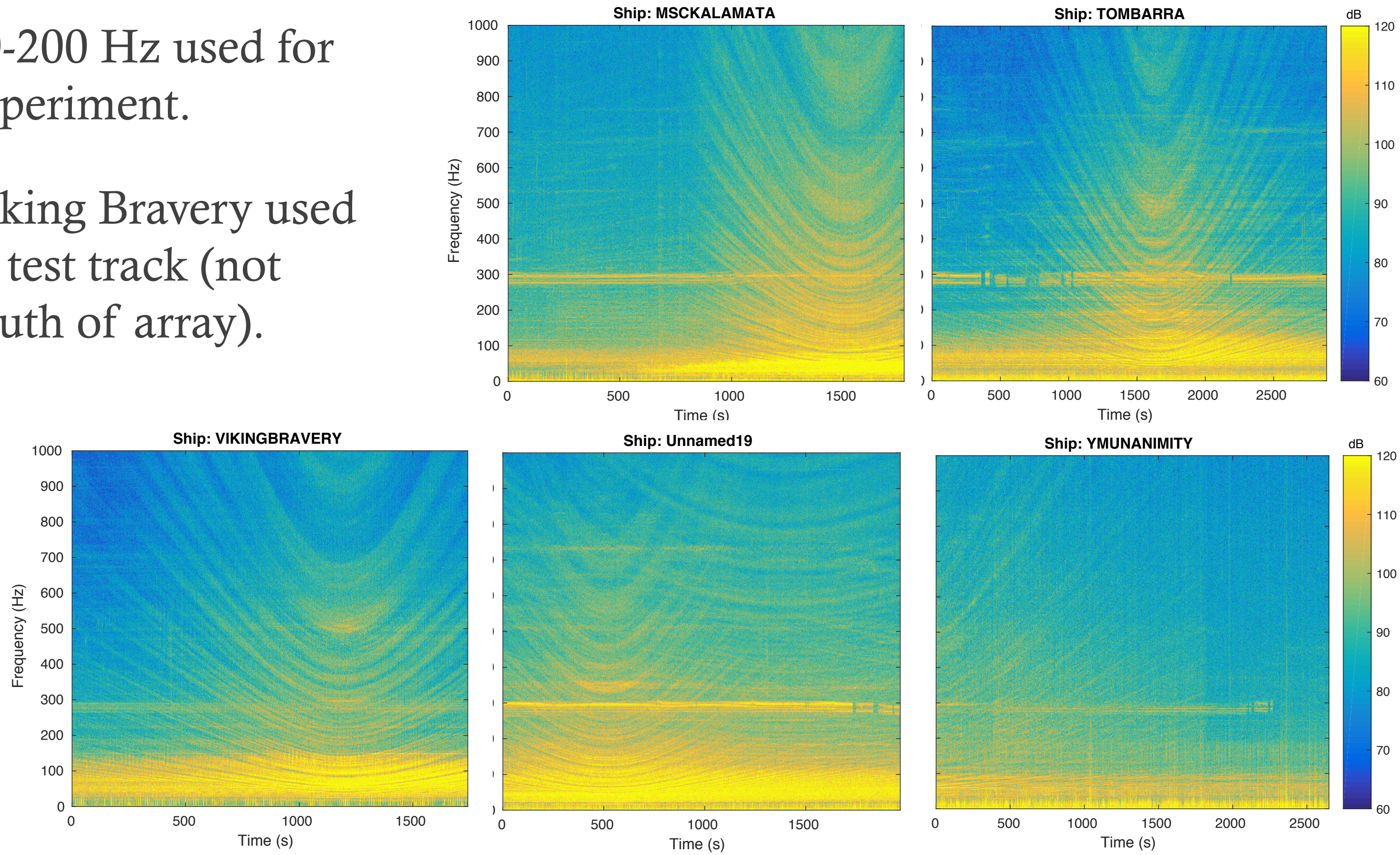
- Ship tracks from Mar. 23 - April 1, 2017.
- Tonals at ~300 Hz come from experiment ship R/V Endeavor.





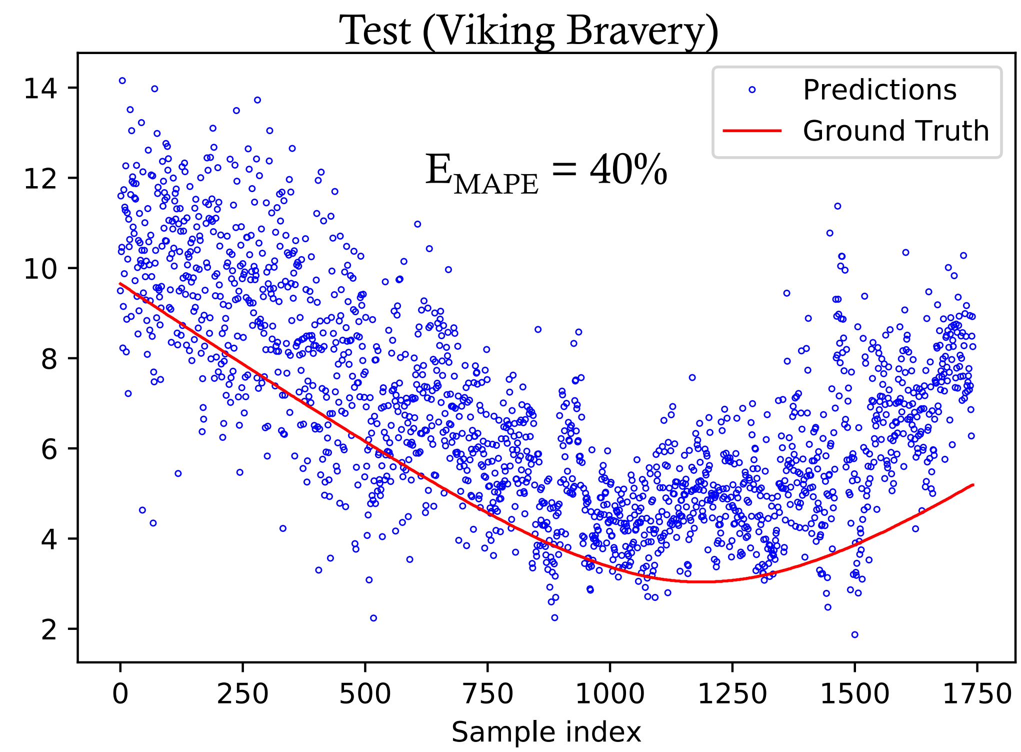
Example problem:
Multiple ships at SCE17

- 50-200 Hz used for experiment.
- Viking Bravery used as test track (not south of array).



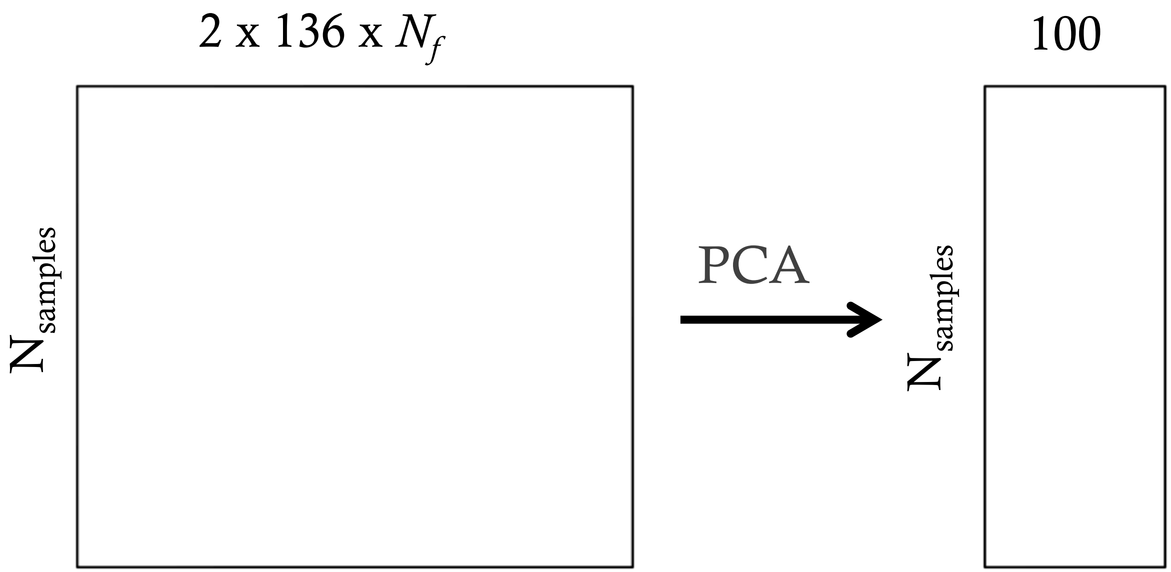
Example problem: Multiple ships at SCE17

- Training MAPE = 2.5%
- Validation MAPE = 20.2%



PCA for high-dimensional input data

- Problem: not enough data to train variation in high dimensions
- M is a real matrix
- M’ is a real matrix projected into a lower dimension



PCA for high-dimensional input data

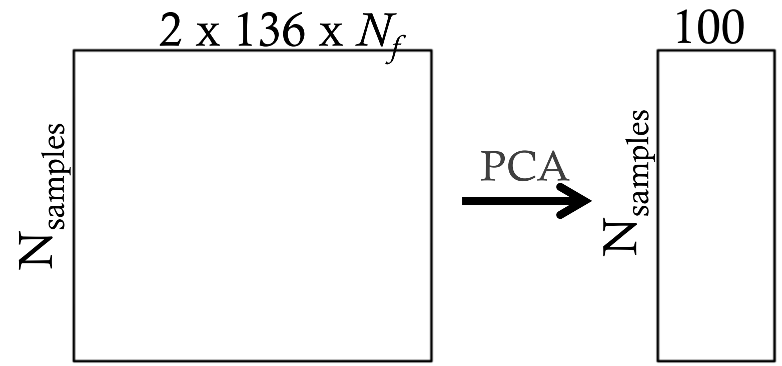
- Choose the top *k* singular values & project into the reduced data space

- *k* is chosen based on model performance

- U is a unitary matrix of size $N_{samples}$ x $N_{samples}$

- Σ is a diagonal matrix of size $N_{samples}$ x $N_{features}$

- V is size $N_{features}$ x $N_{features}$

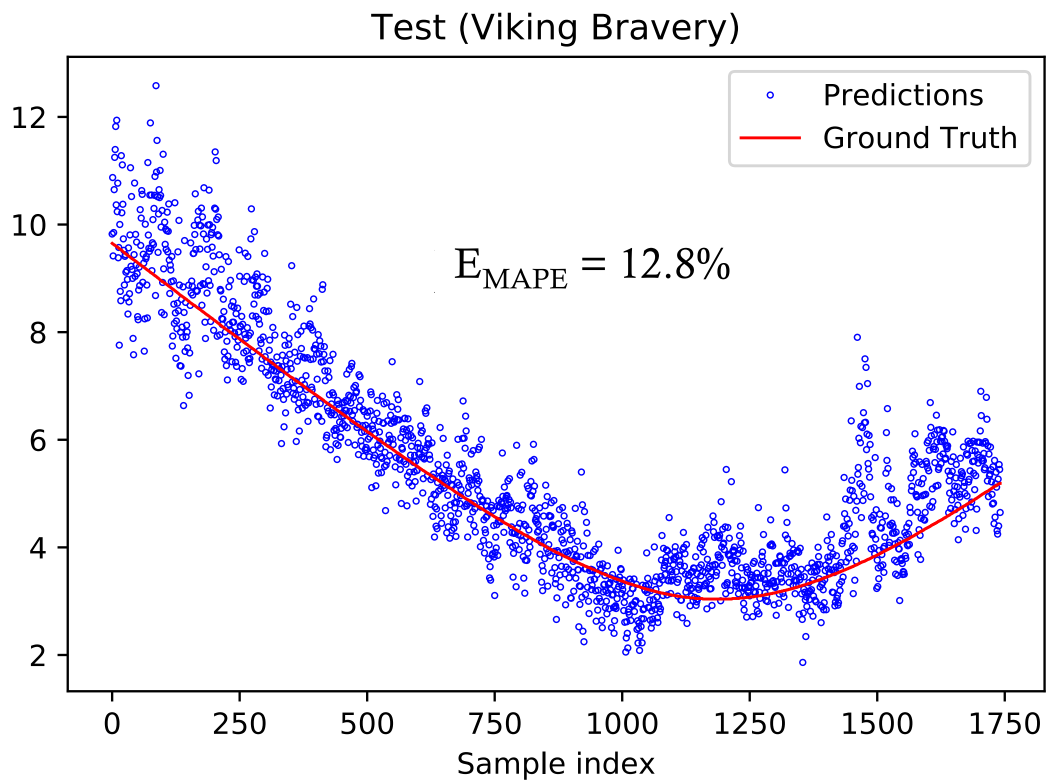


$$M = U\Sigma V^T$$

$$M' = MV[:,1:k]$$

Example problem: Multiple ships at SCE17

- 100 top components kept
- Explains ~20% of variance within data (sum of top *k* normalized singular values)



Conclusions

- Machine learning (ML) models were used to predict cargo ship ranges
  - Trained on similar previous paths
  - 3-layer FNN, SVM, or RF
- ML models achieve lower error than MFP for real data from 1. controlled ship paths, 2. cargo ships in lane
  - Typically one training track, similar test tracks
  - Tracks are close in location, range, and time
- Larger variation between ship tracks results in worse ML performance
  - PCA may help improve results when data is limited relative to variation within features

Appendix: SVM

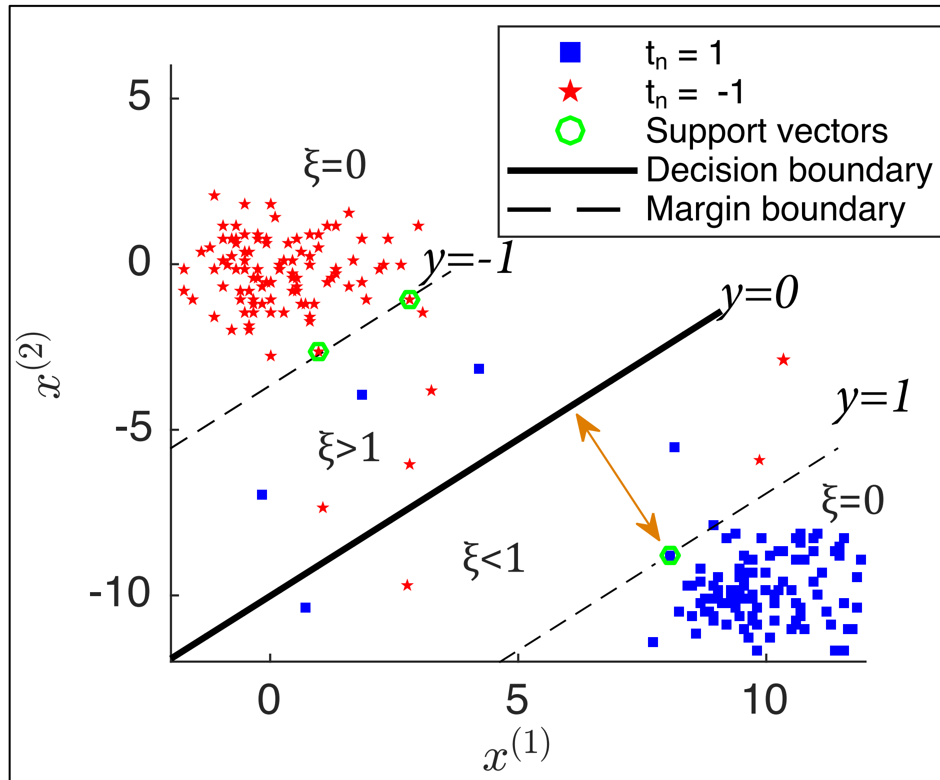
- SVM optimally divides the feature space by class (predicted label t_n, true label $y(\boldsymbol{x}_n)$):

$$\min_{\vec{w},b}\left[C\sum_{n=1}^{N}\xi_n + \frac{\|\vec{w}\|^2}{2}\right]$$

$$\xi_n = \begin{cases} 0 & t_n y(\vec{x}_n) \geq 1, \\ |t_n - y(\vec{x}_n)| & t_n y(\vec{x}_n) < 1 \end{cases}$$

$C > 0.$

Two-Class example of SVM in 2D space



- Multi-class problem solved by iterating 2-class problem

Appendix: RF

- Decision tree

For input $\boldsymbol{x}_n = [x_{n1}, x_{n2}]$:

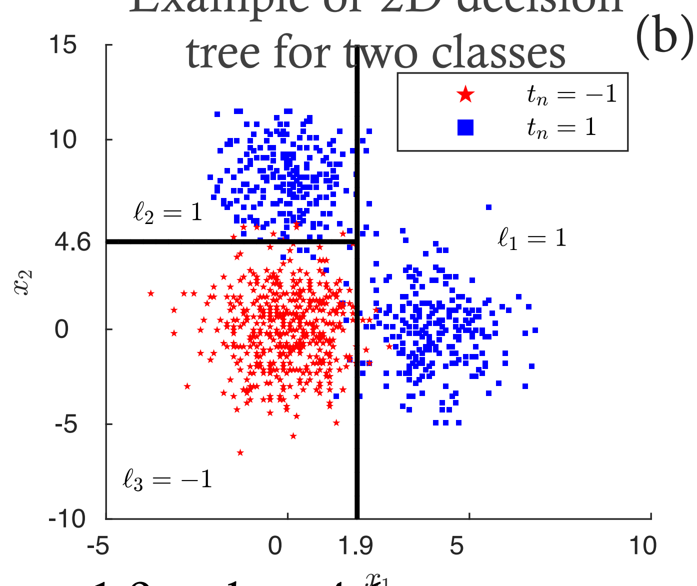
$$\vec{x}_n \in \vec{x}_{left} \qquad x_{ni} \le c$$

$$\vec{x}_n \in \vec{x}_{right} \qquad x_{ni} > c$$

Minimize the number of wrongly classified points per region by changing *c.*

- Random forest:
  - Generate hundreds of random trees
  - Keep most frequently occurring regions

Example of 2D decision tree for two classes



c_1=1.9 and c_2=4.6