**Summary**

- Machine learning, big data, data science, artificial intelligence are about the same.
- Data science has lots of opportunities in geophysics.
- Neural networks is one method. Similar are methods are Support Vector Machines (SVM) and Random Forest (RF). I recommend the latter for a first implementation.
- Unsupervised learning is more challenging than supervised learning.
- Coding: Matlab OK, Jupyter notebook is nice.
- Here I use graph signal processing methods, and dictionary learning.

---

**Machine learning versus knowledge based**

- **Data driven models**: Little or no physics
  - Hybrid model, combining machine learning and physics
  - 3D spectral elements

- **Fwma**
  - First order
  - Analytic physical models
  - Non-linear
  - Numerical physical models

---

**First principles vs Data driven**

<table>
<thead>
<tr>
<th>Feature</th>
<th>First principles</th>
<th>Data driven</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data</td>
<td>Small data</td>
<td>Big data to train</td>
</tr>
<tr>
<td>Domain expertise</td>
<td>High reliance on domain expertise</td>
<td>Results with little domain knowledge</td>
</tr>
<tr>
<td>Fidelity/Robustness</td>
<td>Universal link can handle non-linear complex relations</td>
<td>Limited by the range of values spanned by training data</td>
</tr>
<tr>
<td>Adaptability</td>
<td>Complex and time consuming derivation to use new relations</td>
<td>Rapidly adapt to new problems</td>
</tr>
<tr>
<td>Interpretability</td>
<td>Parameters are physical!</td>
<td>Physically agnostic, limited by the rigidity of the functional form</td>
</tr>
</tbody>
</table>

---

**We can't model everything…**

- Back scattering from fish school
- Reflection from complex geology
- Detection of mines. Navy uses dolphins to assist in this. Dolphins = real ML!
- Predict acoustic field in turbulence
- Weather prediction

---

**Noise Tracking of Cars/Trains/Airplanes**

March 7th, 6-7am, rush hour, Blue Line

Accelerating airplane on Long Beach airport runway, moving northwest and taking off at about 120 mi/h.

Riahi, Gerstoft, GRL 2015
**Compressed sensing**

Signal model: \( y = Ax + n \), \( x \) is sparse

- Data: \( y \), sensing matrix: \( A \), noise: \( n \), \( N < M \)
- Problem: Solve for sparse weights \( x \), sparsity \( K \ll M \)

**CS example: Challenging Nyquist sampling**

\[
y = Ax = \Psi \Phi x = \Psi z.
\]

<table>
<thead>
<tr>
<th>y(t)</th>
<th>z(t)</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \sum_{k=1}^{K} \alpha_k \sin(2\pi f_k t) )</td>
<td>( [z(t_1), \ldots, z(t_N)]^T )</td>
</tr>
</tbody>
</table>

- Fourier matrix \( \Phi \)
- \( K = 3 \)
- \( N = 1024 \)
- \( M = 190 \)

Blue: data \( y \)

**CS example: Fourier transform example**

\[
A_{mn} = e^{-j2\pi(n-1)(m-1)} / M.
\]

\( (f_0, f_1) = (100, 250, 400)/1024 = (0.097, 0.24, 0.39) \)

\( y(t) = \cos 2\pi f_0 t + \cos 2\pi f_1 t. \)

Recovery by FFT, does not work

Orthogonal Matching Pursuit

Sparse Bayesian Learning

**Sparse models and dictionaries**

- Sparse modeling assumes each signal model can be reconstructed from a few vectors from a large set of vectors, called a dictionary \( D \)
- Adds auxiliary sparse model to measurement model

\[
d = Am + n, \quad m \gg Dx \quad \text{such that} \quad |x| < Q
\]

Optimization changes from estimating \( m \) to estimating sparse coefficients \( x \)

- Sparse objective: \( \min \|ADx - d\|_2 \) subject to \( \|x\|_0 \leq T \)
Dictionary learning and sparsity

- Dictionary learning obtains "optimal" sparse modeling dictionaries directly from data.
- Dictionary learning was developed in neuroscience (a.k.a. sparse coding) to help understand mammalian visual cortex structure.
- Assumes (1) Redundancy in data: image patches are repetitions of a few elemental shapes; and (2) Sparsity: each patch is represented with few atoms from dictionary.

Learn dictionary $D$ describing $Y = [y_1, ..., y_T]$

- Each patch a signal $y_i$.
- Set of all patches $Y = [y_1, ..., y_T]$.

Sparse model for patch $y_i$ composed of few atoms from dictionary $D$:

$$s_i = \text{arg min} \|y_i - Ds_i\|_2 \text{ subject to } \|s_i\|_0 \leq T$$

$$Y = X + \epsilon$$

Dictionaries obtained via iterative algorithms

Dictionary learning objective (K-SVD):

$$\min_{D, X} \min \|y_i - Ds_i\|_2 \text{ subject to } \|s_i\|_0 \leq T \forall i$$

Can be solved with alternating minimization steps:
1. Solve for sparse coefficients $X$
2. Solve for dictionary $D$
3. Repeat until convergence.

2D random data, a 5 atom dictionary

$K$-means $K$-SVD ($T=1$) Error vs. iteration

Inverse problem: Travel time tomography

Travel time tomography with adaptive dictionaries

- Most travel time inversion regularize assuming smooth or discontinuous slownesses.
- Locally-sparse 2D travel time tomography (LST) has three main ingredients:
  - Sparsity constraint on slowness patches
  - Dictionary learning (unsupervised machine learning)
  - Damped least squares regularization on overall slowness map

Problem setup with 3 slowness fields

- Consider discrete slowness map $N \times W x W$ with $N \times W \times W$ overlapping $N \times W \times W$ pixel patches.
- $M$ straight-ray paths:
  - Linear travel time model $t = A \cdot s + \epsilon$
  - Local sparse model:
    $$x_i = \text{arg min}_{x_i} \|R_i x_i - D_i s_i\|_2 \text{ subject to } \|x_i\|_0 = T$$

Variables:

- $s_i \in \mathbb{R}^N$: global slowness
- $s_i \in \mathbb{R}^W$: local slowness
- $t \in \mathbb{R}^M$: travel time
- $A \in \mathbb{R}^{M \times N}$: tomography matrix
- $\epsilon \in \mathbb{R}^M$: Gaussian travel time error
- $R_i \in \mathbb{R}^{M \times N}$: sparse coefficients
- $C_i \in \{0, 1\}^N$: selection matrix
- $D_i \in \mathbb{R}^{W \times W}$: slowness dictionary
- $T_i$ number of non-zero coeffs.
- $q_i$ number of dictionary atoms.
From (1),

\[ p(s, X|t) = \frac{p(t|s, X)p(s, X)}{p(t|X)} \]

conditioned only on \( N \) and \( s \), \( s = (X/\gamma g_t)^T R \) and all patch slownesses from (2) independent, giving patch coefficients.

**Fig. 2: Discrete cosine transform (DCT) dictionary atoms. The checkerboard slowness in Section III, and discuss dictionaries for sparse modeling.**

In developing the LST method, we consider the case of straight-rays through the medium, similar to [23].

\[ p(s, X|t) = \prod_i p(R_{s,|i|}) \cdot \prod_i N(D_{x_i}, \Sigma_{x_i}) \]

\[ p(s|X) = \prod_i p(R_{s,|i|}) \cdot \prod_i N(D_{x_i}, \Sigma_{x_i}) \]

\[ \text{Taking MAP} \quad (\hat{s}_p, \hat{X}) = \arg\min_{s, X} \left\{ -\ln p(s, X|t) \right\} \]

**Generic dictionaries**

“Generic dictionaries”: Based on prior knowledge

Patches approximated by dictionary atoms: \( R_s \approx D_{x_i} \)

Solved by:

\[ \hat{s}_p = \arg\min_{s} \|D_{x_i} - R_s\|_2^2 \quad \text{subject to} \quad |s| = T \]

**Learned dictionaries**

“Generic dictionaries”: Based on prior knowledge

Patches approximated by dictionary atoms: \( R_s \approx D_{x_i} \)

Solved by:

\[ \hat{s}_p = \min_{s} \|D_{x_i} - R_s\|_2^2 \quad \text{subject to} \quad |s| = T \]

**Tests with 64 stations**

LST:

\[ (\hat{s}_p, \hat{X}) = \arg\min_{s, X} \left\{ \frac{1}{2} \|\hat{y} - A s\|_2^2 + \frac{1}{2\sigma^2} \|\hat{s}_p - s\|_2^2 + \frac{1}{2\lambda^2} \sum |D_{x_i} - R_{s,|i|}| \right\} \]

subject to \( |s| = T \)

Conventional tomography:

\[ \hat{s}_p = (A^T A + \lambda D_{x_i})^{-1} A^T \hat{y} \]

\[ \Xi_{(i,j)} = \exp(-D_{x_i}/L) \]

**LST vs. conventional method: synthetic inversions without noise**

Each example test < 1 sec on Intel Xeon.

**Bayesian based estimate**

**Bayesian formulation of LST and algorithm**

**Bayesian MAP objective**:

\[ (\hat{s}_p, \hat{X}) = \arg\min_{s, X} \left\{ \frac{1}{2} \|\hat{y} - A s\|_2^2 + \frac{1}{2\sigma^2} \|\hat{s}_p - s\|_2^2 + \frac{1}{2\lambda^2} \sum |D_{x_i} - R_{s,|i|}| \right\} \]

subject to \( |s| = T \)

**Solution via block-coordinate descent (decoupling global and local models)**

- **Global model**: the global slowness is solved as:

\[ \hat{s}_p = \arg\min_{s} \|\hat{y} - A s\|_2^2 + \lambda_1 |s| + \lambda_2 |s| \quad \text{subject to} \quad |s| = T \]


\[ \lambda_1 = \max D |D_{x_i} - R_{s,|i|}| \]

\[ \lambda_2 = \min D |D_{x_i} - R_{s,|i|}| \]

The sparse slowness is effectively averaged over patch solutions.

\[ \hat{s}_p = \frac{\lambda_1 s_{p,1} + \lambda_2 s_{p,2} + \lambda_3 s_{p,3}}{\lambda_1 + \lambda_2 + \lambda_3} \]
LST versus conventional tomography
Both use same travel times (from Fan-Chi Lin),

LST 3 mill rays

Fan-Chi Lin, Geophysics, 8 mill Rays

W=200, W=300 pixels
n=100, Q=200, T=1

Fast anomaly in LST inversion is Silverado aquifer

Depth to Silverado from gravity survey overlaid with LST phase speed

Simulated 1 Hz Rayleigh phase speed from density perturbation of Silverado

Imaging Long Beach, CA using LST
Bianco, Gerstoft, Lin, and Olsen (in prep)

* In March 2011, 5200 seismic stations were deployed in Long Beach, California over 70 km² area
* Ambient seismic noise cross-correlations were obtained for all unique virtual source-receiver pairs (~14 million ray paths) using 3 weeks of data
* We consider only the 1 Hz vertical component data, corresponding to Rayleigh surface waves
* After quality control there were ~3 million ray paths

High-resolution LST phase speed map from 3 mill traveltimes

We use a 300x300 pixel interference map with 3 million rays (tomography matrix A has dimensions MxN equal to 303x10,000,000).
104 transverse, 104 zips, 103 stations.
We are not imposing global correlations on pixels. A is sparse matrix, gel-based inversion for global model (which is bottleneck).
Newport-Inglewood fault network is black line.
Graph clustering for localization within a sensor array

Peter Gerstoft and Nima Riahi, noiselab.ucsd.edu

Based on: Riahi and Gerstoft, Signal Processing, 2017

March 5—12, 2011: 3TB, 5200 Stations in Long Beach, California

Summary

LST can solve for both smooth and dis-continuous slowness features in an image.

Using unsupervised learning for determining the dictionaries gives improved fit.

Convolutional dictionaries? 3D dictionaries?

Graph clustering for localization within a sensor array

=> Two sources in the network

Asymptotic case
Robust Coherence hypothesis test

Conventional coherence

$$C_{ij} = \frac{\sum_{m=1}^{M} x_i(m) x_j(m)}{\sigma_i \sigma_j}$$

Phase-only coherence

$$\tilde{C}_{ij} = \frac{1}{M} \sum_{m=1}^{M} -\frac{x_i(m)}{\sigma_i} \frac{x_j(m)}{\sigma_j}$$

Robust to heteroscedastic noise

Robust Coherence hypothesis test

"Noise-only" network

If α/2.375/N > 11 the network almost surely has a giant connected component, i.e., most sensors are linked (Frolov & Karpin, 1995).

We limit this by testing just the 8-nearest neighbors:

$$C_{ij} = 1 \text{ if } C_{ij} > C_{crit} \text{ and } i = N(j)$$

otherwise.

Example 2: Urban sounds... rotating machinery

Several peaks consistent with helicopter rotor harmonics (20-100 Hz).

Doppler shifts:

\begin{align*}
\text{O} & \quad \text{speed} \\
1 & \quad \text{200 km/h}
\end{align*}

\begin{align*}
\text{O} & \quad \text{speed over ground} \\
3 & \quad \text{70 km/h}, \text{200 km/h}
\end{align*}

\begin{align*}
\text{O} & \quad \text{50 km/h} \text{, not known}.
\end{align*}
Summary

• Machine learning, big data, data science, artificial intelligence are about the same.
• Data science has lots of opportunities in geophysics.
• Neural networks is just one method. Similar are methods are Support Vector Machines (SVM) and Random Forrest (RF). I recommend the latter for a first implementation.
• Unsupervised learning is more challenging than supervised learning
• Here I used graph signal processing methods, and dictionary learning