Zoo of Principal Component Analysis (PCA)

Ziming Liu
Peking University
July 3, 2019
Different PCA suitable for different physics problems

- High-order PCA
- High-dimensional
- Non-linear
- Robustness
- Independent
- Generalized
- Robust PCA
- ICA
- Kernel PCA (auto-encoder)
- Poisson PCA, Laplacian PCA

Different PCA suitable for different physics problems
1. Kernel PCA (auto-encoder)

—— Learn the mapping from particle distribution to initial geometry
The wildest dream

Final Particle Tracks

Going back in time?
Can learn from Monte Carlo data
Change v2 a bit, initial?

In the dataset:

Hydro

V2 = 0.1

V2 = 0.11

V2 = 0.12
Ill-defined?

Different fluctuations size, but same v2!

Not a one-to-one mapping problem?!

Add constraint:
• The initial profile can be generated from the initial model – lie on "initial manifold"
• As little change as possible – "gradient"
When you climb a hill ......

Fly ?

Normal way

Wander around
Auto encoder

\[ \Phi_{\text{extr}} : \mathcal{X} \rightarrow \mathcal{Z} \]

\[ y_2 = f(y_1) \]

\[ y_1 = W_1 x \]

\[ z_1 = W_2 y_2 \]

\[ y_3 = W_3 z_1 \]

\[ \hat{x} = W_4 y_4 \]

\[ \Phi_{\text{gen}} : \mathcal{Z} \rightarrow \hat{\mathcal{X}} \]

\[ y_4 = f(y_3) \]

\[ x = W_4 y_4 \]

extraction

generation

\[ f \] linear: PCA

\[ f \] non-linear: kernel PCA
Step 1: unsupervised learning, train the autoencoder. “Code” represents the most important information.

Step 3: Feed the gradient to the decoder, get the new profile.

Step 2: supervised learning. we have the “Code”, and an observable. we train the neural network to fit the observable. After training, the “gradient” can be computed efficiently.
V2 Results

- Hydro data: Trento + Vishnu + iss

\[ \text{dif} = 0.0 \]
\[ \nu_2 = 0.0777 \]
V3 Results

- Hydro data: Trento + Vishnu + iss

$$\text{dif}=0.0$$

$$\nu_3=0.0431$$
2. Robust PCA
—— automated Learning to separate flow and non-flow
<table>
<thead>
<tr>
<th>QGP</th>
<th>Jet</th>
<th>Background</th>
<th>Foreground</th>
</tr>
</thead>
<tbody>
<tr>
<td>Thermalized</td>
<td>Not thermalized</td>
<td>static</td>
<td>moving</td>
</tr>
<tr>
<td>Long range correlation</td>
<td>Short range correlation</td>
<td>Long correlation</td>
<td>Short correlation (sparse) in time</td>
</tr>
</tbody>
</table>

Learn Flow/Non-flow with many events of heavy-ion collisions

Learn Background/foreground with many snapshots of a single video
What’s wrong with traditional PCA?

Blue: Robust PCA; Green: Traditional PCA

Samples (red) from a one-dimensional subspace (blue) corrupted by small Gaussian noise. The output of classical PCA (green) is very close to the true subspace despite all samples being noisy.

Samples (red) from a one-dimensional subspace (blue) corrupted by sparse, large errors. The principal component (green) is quite far from the true subspace even when over three-fourths of the samples are uncorrupted.
Why can we do this?

① Flow

② Non-Flow

……

Prob density

……

Prob density

coefficient

coefficient
No free parameter!

X: input; D: low rank; X-D: sparse matrix

\[
\min_D \quad \text{rank}(D) + \|X - D\|_0
\]

\[
\min_{\tilde{D}} \quad \|\tilde{D}\|_* + \lambda \|X - D\|_1
\]

sum of singular values \quad sum of all elements (absolute value)

A rather remarkable fact is that there is no tuning parameter in our algorithm. Under the assumption of the theorem, minimizing

\[
\|L\|_* + \frac{1}{\sqrt{n(1)}} \|S\|_1, \quad n(1) = \max(n_1, n_2)
\]

always returns the correct answer. This is surprising because one might have expected that one would have to choose the right scalar \(\lambda\) to balance the two terms in \(\|L\|_* + \lambda \|S\|_1\) appropriately (perhaps depending on their relative size). This is, however, clearly not the case. In this sense, the choice \(\lambda = 1/\sqrt{n(1)}\) is universal. Further, it is not a priori very clear why \(\lambda = 1/\sqrt{n(1)}\) is a correct choice no matter what \(L_0\) and \(S_0\) are. It is the mathematical analysis which reveals the correctness of this value. In fact, the proof of the theorem gives a whole range of correct values, and we have selected a sufficiently simple value in that range.

https://statweb.stanford.edu/~candes/papers/RobustPCA.pdf
Toy model

Flow: elliptic $\sin(2\phi), \cos(2\phi)$
Nonflow: Guassian, randomized angle
Observed: Flow+NFlow

Robust PCA can separate flow/non-flow without any training (unsupervised)!
Comment: No physics, but very good technique :P
3. ICA

Independent Component Analysis
—— conquer limitations of PCA to study sub-leading flow
Sub-leading flow?

$V_3(p_T)$

Leading flow $\leftrightarrow$ Sub-leading flow

Orthogonal? Not necessary! (PCA fails)
But independent for linear response! (ICA works)
Cocktail Party problem

\[
\begin{bmatrix}
    a_{11} & a_{12} \\
    a_{21} & a_{22}
\end{bmatrix}
\]

We only have observation \( x \), How to determine \( A \) and \( s \)?

\[ x = As \]
ICA


\[ x = As \]

We only have observation \( x \), how to determine \( A \) and \( s \)?

Assumption:
1. Sources are independent with each other
2. Sources should be as non-gaussian as possible

Why non-gaussian?
Central limit theorem: gaussian implies mixing!

Irwin–Hall distribution

\( z_1: \) uniform distribution
\( z_2: \) sum of two uniform distribution
ICA prefers \( z_1 \) over \( z_2 \)
Fake data

- Sub-leading ecc
- Leading ecc
- Sub-leading flow
- Leading flow

Indepant-variant basis

Observation basis

PCA axes
ICA axes

Nicely separated

mixing

PCA
ICA
4. Traditional PCA

— Probing number of initial sources
Nucleonic fluctuation

Sub-nucleonic fluctuation

Different number of initial sources?
A naïve argument:

\[ \text{Au} + \text{Au} \]

197 + 197 = 394 nucleons

Each nucleon has two-dim coordinates

So we have 394*2 = 788 parameters

Data should lie on a 788-dim manifold

Hydrodynamics evolution is finite time

So final particle distribution is a 788-dim manifold, too

The manifold could be non-linear,

**BUT**

Let’s try PCA first!
PCA to two-particle data

\[
\frac{d^2 N_{pair}}{d\Delta\eta d\Delta\varphi}
\]
data, 32*32=1024 bins, 2000 events

10%-20% centrality

For all models, the singular values show an abrupt drop at 768<~788

Guess: 768 corresponds to the mean number of participants*2

Comment:
1. Independent of collision energy? Number of events? Number of bins?
   Need further check
2. May serve as an observable to measure number of sources
   Can we observe such drop in experiments? Statistical errors ......
5. High-order PCA  
— For multi-particle correlation data

\[
\begin{align*}
X &= U S V^T \\
(I_1, I_2, I_3) &= (R_1, R_2, R_3)
\end{align*}
\]

Multi-particle correlation data

6. Generalized PCA  
— Suitable for different data types

\[
\min \left\| A - \tilde{A} \right\|_F^2 \iff A = \tilde{A} + \epsilon, \epsilon \sim N(0, \sigma^2)
\]

gaussian

\[
\begin{align*}
? \iff A &= \tilde{A} + \epsilon, \epsilon \sim p(\epsilon) \\
&\text{Bernoulli, Poisson, Laplacian} \ldots
\end{align*}
\]
Conclusion

• PCA still has a lot to explore, at least in the application of heavy-ion physics.
• Our ultimate goal is to make PCA useful in experiments.
• But first we should publish a few method papers to elucidate the performance of these methods and arouse interests of experimentalists.
Thank you!
Backup
Singular values

ampt_39

ampt_200

hijing

karpenko

drop!
Eigenvectors in Region I

ampt_39

ampt_200

hijing

karpenko

$\delta \phi$

$\delta \eta$

$-0.5 \pi$

$1$

$1.5 \pi$

$2$
Eigenvectors in Region I

ampt_39

ampt_200

hijing

karpenko
Eigenvectors in Region II and III

The bright area is “expanding” towards outside. So in Region II, each eigenmode focuses on one local area. (In region I, the eigenmode is global)