A novel approach for supernova photometric classification

Emille E. O. Ishida
In collaboration with Rafael S. de Souza

IAG – USP – September/2011
Overview

1. Principal Component Analysis (PCA)
   1.1 Definition
   1.2 General Applications
   1.3 Use in Astronomy/Cosmology: reconstructing functions

2. Our results in applying PCA reconstruction

3. Kernel Principal Component Analysis (kPCA)
   3.1 Limitations of PCA
   3.2 The use of kernels

4. The problem of supernova photometric classification
   4.1 How does kPCA perform?
   4.2 The SuperNova Classification Challenge (SNCC)
   4.3 kPCA preliminary results

5. Conclusions and Perspectives
1. Principal Component Analysis (PCA)

Goal:
Find a new basis, from a linear combination of the original one. In this new basis you should be able to describe most of the information in the data with a minimum number of parameters.
Example: ancient PCA history is in social sciences

N → number of individuals

P → number of things you know about them (math skills, social skills, height, weight, gender, concentration capacity, etc)

Analyze each pair of features and compute their correlation

Almost all tests showed correlations with the others, indicating that one unique variable could be capable of predicting the result of one person in all the tests.
Example: how do you decide the angle you should take a picture?

http://www.youtube.com/watch?v=BfTMmoDFXyE&feature=related
1. 1 Definition: how to find the PC?

Orthogonal projection of data onto lower-dimension linear space that...
- maximizes variance of projected data (purple line)
- minimizes mean squared distance between
  - data point and
  - projections (sum of blue lines)

Once the PCs are found, re-write the data (reconstruct the information) in the PCs basis, using a smaller number of variables.
1. 1 Definition: Mathematically

\[ C \rightarrow \text{covariance matrix} \]

\[ \text{cov}(x, y) = \sum_{i=1}^{n} \frac{(x_i - \bar{x})(y_i - \bar{y})}{n-1} \]

\[ P \rightarrow \text{vector of initial variables: } P=\{x,y\} \]

\[ \alpha \rightarrow \text{unit vector that maximizes the variance} \]

\[ \text{var}(\alpha) = \alpha' \ C \ \alpha \]

Using Lagrange multipliers, this means maximize

\[ \Lambda(\alpha, \lambda) = \alpha' \ C \ \alpha - \lambda \ (\alpha' \ \alpha - 1) \]

\[ C \ \alpha - \lambda \ \alpha = 0 \quad \text{or} \quad (C-\lambda \ I_p)\alpha = 0 \]

\[ \alpha' \ C \ \alpha = \alpha' \ \lambda \ \alpha = \lambda \]

Quantity to be maximize:

Larger eigenvalue gives larger variance
1.2 General Applications: image compression

- Main question: what features can be discarded if I want to keep the relevant information?

- Original image has 372 x 492 pixels;

- Divide the image in cells. Each cell contains 12 x 12 pixels

- Consider each cell a 144 dimensions vector

- Consider the contribution of each pixel a measurement

- Compute the mean image and the PCs
1.2 General Applications: image compression

- find the linear expansion coefficients that best describe each cell using the first 16 PCs
- reconstruct the image by putting the results side by side
1. 2 General Applications: image compression

60 most important eigenvectors
1.2 General Applications: image compression

1. Use in Astronomy/Cosmology: Reconstructing Functions

Information to be reconstructed:
A cosmological quantity (dark energy equation of state/luminosity distance/deceleration parameter/...)

Initial variables:
The value of the cosmological quantity in question in each redshift bin

Trick:
Determine the covariance matrix analytically (from the Fisher matrix)
For an unbiased estimator,

\[ F_{ij} = \left( C^{-1} \right)_{ij} = \left\langle -\frac{\partial^2 \ln L}{\partial \theta_i \partial \theta_j} \right\rangle \]

\[ \chi^2(\Omega_m, \Omega_\Lambda) = \sum_{i=1}^{n} \frac{\left( \mu_i - \mu \left( z_i; \Omega_m, \Omega_\Lambda \right) \right)^2}{2\sigma_i^2} \]
2. **Our results in applying PCA reconstruction**

\[
\mu(z) = 5 \log_{10} [d_L(z)] + \mu_0,
\]

\[
d_L(z) \equiv (1 + z) \int_0^z \frac{du}{H(u)}
\]

\[
H(z; \beta) = \sum_{i=1}^{N_{\text{bin}}} \beta_i c_i(z),
\]
Hubble parameter reconstruction from a principal component analysis: minimizing the bias

E. E. O. Ishida\textsuperscript{1,2} and R. S. de Souza\textsuperscript{1,2} A&A 527, A49 (2011)

Type Ia SN data

Not fiducial
Probing cosmic star formation up to $z = 9.4$ with GRBs

E. E. O. Ishida$^{1,2,3*}$, R. S. de Souza$^{1,3}$, A. Ferrara$^{4,1}$


GRB data
3. kernel PCA (kPCA)

3.1 Limitations of PCA

PCA doesn’t know labels!

PCA cannot capture NON-LINEAR structure!

Barnabas Pocksos, University of Alberta, 2009
3.2 The use of kernels: Addressing both problems at once

If we have 2 classes that cannot be separated in the original parameter space.

Going to a higher dimensional space might solve the problem:

\[
\Phi: \mathcal{X} = \mathbb{R}^2 \rightarrow \mathcal{H} = \mathbb{R}^3
\]

\[
(x_1, x_2) \mapsto (x_1, x_2, x_1^2 + x_2^2)
\]

http://cseweb.ucsd.edu/classes/fa01/cse291/kernelPCA_article.pdf
3.2 The use of kernels: 
Addressing both problems at once

Main problem of higher dimensional representation:

Computational cost rises exponentially!

Kernel trick

Sometimes, it is possible to compute dot products without explicitly mapping into the high dimensional feature space.

In what follows we shall use

\[ k(x, x') = \langle \Phi(x), \Phi(x') \rangle \]

http://cseweb.ucsd.edu/classes/fa01/cse291/kernelPCA_article.pdf
3.2 The use of kernels: The trick

\[ Cv = \frac{1}{m} \sum_{j=1}^{m} x_j x_j^T v = \lambda v \]

Taking all to another space:

\( \Phi : \mathcal{X} \to \mathcal{H}, \ x \mapsto \Phi(x) \)

Covariance matrix:

\[ C = \frac{1}{m} \sum_{j=1}^{m} \Phi(x_j) \Phi(x_j)^T \]

Eigenvalue equation:

\[ \lambda v = \lambda \sum_{i=1}^{m} \alpha_i \Phi(x_i) \]

Inner product kernel:

\[ K(x_i, x_j) = \Phi(x_i)^T \Phi(x_j) \]

Premultiply both sides by \( \Phi(x_k)^T \):

\[ \sum_{i=1}^{m} \sum_{j=1}^{m} \alpha_j \Phi(x_i) K(x_i, x_j) = m \lambda \sum_{j=1}^{m} \alpha_j \Phi(x_i) \]

Eigenvalue equation:

\[ Cv = \lambda v = \lambda \sum_{i=1}^{m} \alpha_i \Phi(x_i) \]

http://cseweb.ucsd.edu/classes/fa01/cse291/kernelPCA_article.pdf
3.2 The use of kernels: The trick

Choosing the kernel function properly we do not need to perform the mapping itself:

\[ k(\Phi(x_i), \Phi(x_j)) = \langle \Phi(x_i), \Phi(x_j) \rangle \]

The resulting set of eigenvectors are used to project a test point:

\[ \langle \mathbf{u}^k, \Phi(x) \rangle = \sum_{i=1}^{m} \alpha_i^n k(x_i, x), \quad n = 1, \ldots, p; \]
3.2 The use of kernels: Visualization

http://www.youtube.com/watch?v=3liCbRZPrZA&feature=player_embedded
4. The problem of SN photometric classification

<table>
<thead>
<tr>
<th>Type</th>
<th>Ia</th>
<th>Ib</th>
<th>Ic</th>
<th>II</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spectrum</td>
<td>Si</td>
<td>No Si</td>
<td>He</td>
<td>H</td>
</tr>
<tr>
<td>Physical mechanism</td>
<td>Nuclear explosion of low mass star</td>
<td>Core collapse of evolved massive star (may have lost its hydrogen or even helium envelope during red-giant evolution)</td>
<td>Large Variations</td>
<td></td>
</tr>
<tr>
<td>Light curve</td>
<td>Reproducible</td>
<td></td>
<td></td>
<td>~ 100 x Visible energy</td>
</tr>
<tr>
<td>Neutrinos</td>
<td>Insignificant</td>
<td></td>
<td></td>
<td>Neutron star (typically appears as pulsar) Sometimes black hole?</td>
</tr>
<tr>
<td>Compact Remnant</td>
<td>None</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rate/h^2SNu</td>
<td>0.36 ± 0.11</td>
<td>0.14 ± 0.07</td>
<td>0.71 ± 0.34</td>
<td></td>
</tr>
<tr>
<td>Observed</td>
<td>Total ~ 2000 as of today (nowadays ~200/year)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
4. The problem of SN photometric classification

Type Ia is considered to be a standardizable candles, they are the main focus of surveys focusing on cosmology.

Future surveys with discover thousands of SN per year → impossible to perform spectroscopic typification.

A glimpse of the problem: SN light curves generated with SNANA

[^1]: http://sdssdp62.fnal.gov/sdsssn/SNANA-PUBLIC/
4.1 How does kPCA perform?

We applied the kPCA method to SN light-curves simulated using SNANA.

Light-curve requirements:

- At least one observation before -5 days,
- At least one observation after +25 days,
- The two conditions above must be fulfilled in all available filters,
- Each individual observation epoch must have signal to noise ratio (SNR) > 5.

Gaussian kernel:

\[ k(x_i, x_j) = e^{-\frac{||x_i - x_j||^2}{\sigma_k^2}} \]
4.1 How does kPCA perform?

Results from SN in with $z < 0.1$: each SN type occupies a specific locus in the PCs space.

The SN types were generated in proportion to their expected rate.
4.1 How does kPCA perform? 

Classifying a new data point

Calculate projection

New point in PCs space: ●

Calculate geometrical distances between the new point and all the points belonging to the base

The new point will be classified as the SN type of its nearest neighbor
4.1 How does kPCA perform?

Caveat: These are extremely demanding selection cuts

Will the procedure perform as well in a realistic scenario?
4.2 The Supernova Classification Challenge (SNCC)

Blind mix of simulated SN:
- types: Ia, Ib, Ic, II
- total ~ 22,000 light-curves
- ~1100 labeled
- samples with/without host z

Groups should provide a type of each SN

Simulations used observation conditions and filters as expected for the Dark Energy Survey

Goals:
1. learn relative strength/weakness of different classification algorithms
2. use the results to improve future classification algorithms
3. understand what spectroscopically confirmed sample is needed to optimize the classification methods
4.2 The Supernova Classification Challenge (SNCC)

Replies to the SNCC:
- 10 groups
  - 13 entries for sample with host z
  - 9 entries for sample with no z information

Main results:
- different techniques give similar results
- No particular strategy was obviously superior
- results are much better for template sample

### 4.2 The Supernova Classification Challenge (SNCC)

#### TABLE 5
**List of Participants in the SNPhotCC.**

<table>
<thead>
<tr>
<th>Participants</th>
<th>Abbreviationa</th>
<th>Classified +Zb/noZc</th>
<th>SN zphd</th>
<th>CPUe</th>
<th>Description (strategy classf)</th>
</tr>
</thead>
<tbody>
<tr>
<td>P. Belov and S. Glazov</td>
<td>Belov &amp; Glazov</td>
<td>yes/no</td>
<td>no</td>
<td>90</td>
<td>light curve χ² test against Nugent templates (2)</td>
</tr>
<tr>
<td>S. Gonzalez</td>
<td>Gonzalez</td>
<td>yes/yes</td>
<td>no</td>
<td>120</td>
<td>cuts on SIFTO fit χ² and fit parameters (1)</td>
</tr>
<tr>
<td>J. Richards, Homrighausen, C. Schafer, P. Freeman</td>
<td>InCAg</td>
<td>no/yes</td>
<td>no</td>
<td>1</td>
<td>Spline fit &amp; nonlinear dimensionality reduction (4)</td>
</tr>
<tr>
<td></td>
<td>JEDI Boost</td>
<td>yes/yes</td>
<td>no</td>
<td>10</td>
<td>Boosted decision trees (4)</td>
</tr>
<tr>
<td></td>
<td>JEDI-Hubble</td>
<td>yes/no</td>
<td>no</td>
<td>10</td>
<td>Hubble diagram KDE (3)</td>
</tr>
<tr>
<td></td>
<td>JEDI Combo</td>
<td>yes/no</td>
<td>no</td>
<td>10</td>
<td>Boosted decision trees + Hubble KDE (3+4)</td>
</tr>
<tr>
<td>S. Philip, V. Bhatnagar, A. Singhal, A. Rai, A. Mahabal, K. Indulekha</td>
<td>MGU+DU-1i</td>
<td>no/yes</td>
<td>no</td>
<td>&lt; 1</td>
<td>light curve slopes &amp; Neural Network (2)</td>
</tr>
<tr>
<td></td>
<td>MGU+DU-2</td>
<td>no/yes</td>
<td>no</td>
<td>&lt; 1</td>
<td>light curve slopes &amp; Random Forests (2)</td>
</tr>
<tr>
<td>H. Campbell, B. Nichol, H. Lampietl, M. Smith</td>
<td>Portsmouth χ²</td>
<td>yes/no</td>
<td>no</td>
<td>1</td>
<td>SALT2-χ² &amp; False Discovery Rate Statistic (1)</td>
</tr>
<tr>
<td></td>
<td>Portsmouth-Hubble</td>
<td>yes/no</td>
<td>no</td>
<td>1</td>
<td>Deviation from parametrized Hubble diagram (3)</td>
</tr>
<tr>
<td>D. Poznanski</td>
<td>Poz2007 RAW</td>
<td>yes/no</td>
<td>yes</td>
<td>2</td>
<td>SN Automated Bayesian Classifier (SN–ABC) (2)</td>
</tr>
<tr>
<td></td>
<td>Poz2007 OPT</td>
<td>yes/no</td>
<td>yes</td>
<td>2</td>
<td>SN–ABC with cuts to optimize $C_{FOM−la}$ (2)</td>
</tr>
<tr>
<td>S. Rodney</td>
<td>Rodney</td>
<td>yes/yes</td>
<td>yes</td>
<td>230</td>
<td>SN Ontology with Fuzzy Templates (2)</td>
</tr>
<tr>
<td>M. Sako</td>
<td>Sako</td>
<td>yes/yes</td>
<td>yes</td>
<td>120</td>
<td>χ² test against grid of Ia/II/Ibc templates (2)</td>
</tr>
<tr>
<td>S. Kuhlmann, R. Kessler</td>
<td>SNANA cuts</td>
<td>yes/yes</td>
<td>yes</td>
<td>2</td>
<td>Cut on MLCS fit probability, S/N &amp; sampling (1)</td>
</tr>
</tbody>
</table>

---

79% purity
4.3 kPCA preliminary results

This is good... but only ~10% of the spectroscopic sample and less than 5% of the photometric sample survived our selection cuts.

We are currently working in the determination of selection cuts that allow us to classify the entire SNCC.
5. Conclusions and Perspectives

PCA is an useful dimensionality reduction technique, which is applied in many fields.... including Astronomy

PCA has its limitations, but in most cases, the use of kernels is able to solve classification problems were linear PCA cannot help.

The preliminary results for kPCA applied to the SNCC are promising, we are optimistic about the whole classification and results applied only to pre-maximum data

We do need to think of a clever way to introduce the redshift information

This method is quiet general, and can be used in basically any transient object. Our next goal is to include population III SN theoretical light-curves in SNANA in order to improve the chances of detecting such an object