Factor Analysis on Chemical Abundance Spaces

Summer 2022 ASC
u7281660 **Yangda Bei**

1st year

Abstract

Factor Analysis is a statistical machine learning model used to describe the variability of correlated variables that are observed, identifies a structure that underlies them, and transforms their correlation into underlying latent factors. Using Factor Analysis to approach the problem of reducing the dimensionality of stellar chemical space could provide a better understanding of how stars come into formation than Principal Component Analysis. We create an exploratory factor analysis model using a maximum likelihood SVD-based approach in Python. Applying this model to the high-dimensionality stellar chemical space of the GALactic Archaeology with Hermes survey for future work is a promising step towards providing evidence for the estimated number of dimensions of the space defined by the stellar chemical element abundances.

Contents

1 Introduction

Observational study of individual stars paired with probabilistic models provides detailed insight into the evolutionary history of galaxies. To reconstruct the order of events involved in the formation of the galactic disk of the Milky Way, the re-assembly of star-forming aggregates is crucial.

Many large-scale surveys of the Milky Way have allowed the tracing of stars through the galaxy's evolution to become increasingly possible (e.g. RAVE - Steinmetz et al. 2006; LAMOST - Zhao et al. 2012; GALAH - De Silva et al. 2015; *Gaia* - Gaia Collaboration et al. 2016; APOGEE - Majewski et al. 2017). However, due to disruptions of a star's path through the galaxy, e.g. by mass loss due to stellar evolution or gravitational interactions throughout its life, kinematic information alone is not reliable in extracting information. Although individual clusters formed from stars are chemically homogeneous (De Silva et al. 2006, 2007; Bovy 2016), most clusters dissipate and evolve dynamically resulting in the loss of dynamical information. However, the surface chemical composition of stars provides critical insight into their history as stars that have dispersed but were born in the same cluster will have similar chemical abundance patterns, therefore, reflecting their origin and chemical evolution of the gas from which they formed.

The stellar chemical space ($\mathcal{C}$-space) is the space of measurable abundances of chemical elements. We can employ a technique named chemical tagging, a process that groups stars based on their positions in the chemical space (Freeman & Bland-Hawthorn 2002), to attempt to reconstruct ancient star-forming aggregates involved in the formation of the Galactic disk. Successful chemical tagging to recover the dispersed aggregates requires precise spectroscopic data from a large number of stars.

Before applying chemical tagging to a spectral dataset obtained from the GALactic Archaeology with HERMES (GALAH) survey, we first apply our model on a mock data sample. We aim to mimic GALAH's measurement of a 25-dimensional chemical abundance space. However, because many parts of a star's spectra rely on the same element and will therefore be correlated due to underlying nucleosynthetic processes, it is estimated that around seven chemicals are independent and that the production of others strongly correlates with them. Ting et al. (2012) performed PCA analysis on [X/Fe] space to estimate the dimensionality of the $\mathcal{C}$-space available to HERMES, and found that the 17 elements measured can be represented in six dimensions both at high and low metallicity. It is estimated that the full 25-element HERMES space would give about eight or nine dimensions for the HERMES chemical space.

The bulk of this report will introduce factor analysis (FA), the model that aims to determine the driving factors of nucleosynthetic processes, how it differs from principal component analysis (PCA), and a comparison of our own factor model on a mock data sample against the `FactorAnalyzer` package created by `scikit-learn`.

This study served as an introduction to unsupervised machine learning algorithms, applying data analysis using Python, and placing the model into the context of determining the dimensionality of $\mathcal{C}$-space.

2 Stellar Processes

Metallicity is the abundance of elements that is higher than hydrogen and helium on the periodic table. The abundances of the heavy elements come from stellar nucleosynthesis and are formed in the cores of stars as they evolve over time. Metals are deposited via stellar winds and supernovae which enriches the interstellar medium to form new stars. It follows that older stars have lower metallicities than younger stars which formed in a more metal-rich universe.

The dimensionality of the $\mathcal{C}$-space is affected largely by the interplay of stellar processes. Although stars born from the same gas formations disperse, their element abundances would remain similar. The gas itself would have its own history of pollution from core-collapse supernovae, Type Ia supernovae (SNe Ia), stellar winds from asymptotic giant branch (AGB) stars, and neutron star mergers. The high dimensionality of $\mathcal{C}$-space relies on the interplay of these processes but we think that it is driven by 5-6 underlying factors which help form the chemical makeup of the space.

Whilst core-collapse releases metals into the interstellar medium, it does not account for every element. Neutron capture processes (n-capture) refer to the synthesis of heavier elements at lower temperatures due to their neutral charge so it does not get repulsed by nuclei. The majority of neutrons in supernovae do not participate in nucleosynthesis as the majority are trapped in the collapsing core (Thompson et al. 2001). N-capture processes are split up into two processes: the rapid (r-) and slow (s-) processes. The r-process produces highly unstable nuclei for the nucleosynthesis of elements heavier than zinc through the bombardment of neutrons that rapidly decays β-decays into more stable forms. Gravitational forces during neutron star/black hole or double-neutron star mergers can pull the neutron star apart, resulting in many neutron captures and ejection of processed material (Lattimer et al. 1977). However, neutron-star mergers are rare occurrences since two massive stars must explode as supernovae and their residuals form a sufficiently close binary system to merge within the age of the Universe (Abbott et al. 2017). In supernovae, the r-process is responsible for around half of all isotopes of elements heavier than iron (Qian et al. 1998).

Radioactive decay in the s-process on the other hand happens before another neutron is captured. This slow process produces nuclei close to the valley of stability which prefers almost-stable nuclei with small n-capture reaction rates as the neutron flux is not so intense. The production sites of the main component of the s-process that produces heavy elements beyond Sr and Y are believed to mainly in low-mass asymptotic giant branch stars (Boothroyd 2006).

Stellar winds are flows of gas ejected from the upper atmosphere of a star. Post-mainsequence stars lose nearing the end of their lifetime eject large amounts of mass, about 10^{-3} solar masses per year (Mattsson & Höfner 2011). For larger stars, shedding as much as 50% of its mass clearly has a significant impact on its later stages of evolution. Intermediate mass stars become white dwarfs rather than exploding as supernovae if enough mass has been lost in their stellar winds.

SNe Ia is the last nucleosynthesis event to contribute to the composition of the Galaxy. They require first the evolution into white dwarfs in a close binary system, followed by either spiraling together or mass transfer through the release of gravitational waves. If they can combine, once past the "critical mass", they reignite and trigger a supernova explosion in some cases to produce nuclei from silicon to iron peak. Its entire mass is ejected into the surrounding Galaxy and enriches the interstellar medium. We get that α-elements, such as O, Ne, Mg, Si, S and Ca are mainly produced by core-collapsing supernovae whereas SNe Ia produces iron peak elements, such as Cr, Mn, Ni and Fe.

These nucleosynthetic processes underpin the motivation for searching for the driving elements as we can then trace the formation of star aggregates. In the following sections, we introduce the model that will allow us to find the underlying factors. Interpreting the factors of real spectral data is left for future work.

3 PCA vs. FA

3.1 Principal Component Analysis

PCA is a machine learning technique used to reduce the dimension of a data-set. Most of the time, we cannot expect the training data to densely populate the space for high-dimensional problems, and so, whilst the data vectors may have a large dimension, they will typically lie close to a much lower dimensional "manifold", meaning that the distribution of the data is constrained heavily (Barber 2020).

From Chapter 15.2: *Principal Component Analysis* (Barber 2020), a datapoint $\mathbf{x}^n$ can be approximated to a lower dimensional coordinate system as

$$\mathbf{x}^n \approx \mathbf{c} + \sum_{j=1}^{M} y_j^n \mathbf{b}^j \equiv \tilde{\mathbf{x}}^n,$$

where $\mathbf{b}^j$ are the "basis" vectors that span the linear subspace that is fitted to the data (or "principal component coefficients") and y_i^n are the low dimensional coordinates of the data which forms a lower dimension $\mathbf{y}^n$ for each datapoint n for components $i = 1, \dots, M$. The optimal bias $\mathbf{c}$ is given to be the mean of the data and centres the coordinate system of the linear subspace.

To find the components, one way is to use *Singular Value Decomposition* (SVD) on a matrix

$\mathbf{X} \in \mathbb{R}^{D \times N}$, given by

$$\mathbf{X} = \mathbf{U}\mathbf{D}\mathbf{V}^T,$$

where $\mathbf{U} \in \mathbb{R}^{D \times D}$ satisfying $\mathbf{U}^T\mathbf{U} = \mathbf{I}$, $\mathbf{V} \in \mathbb{R}^{N \times N}$ satisfying $\mathbf{V}^T\mathbf{V} = \mathbf{I}$, and $\mathbf{D}$ is a $D \times N$ diagonal matrix of the positive singular values. $\mathbf{D}$ is assumed to have its singular values ordered with the upper left diagonal element to be the largest and so we can write the matrix $\mathbf{X}\mathbf{X}^T$ as

$$\mathbf{X}\mathbf{X}^T = \mathbf{U}\mathbf{D}\mathbf{V}^T\mathbf{V}\mathbf{D}^T\mathbf{U}^T = \mathbf{U}\tilde{\mathbf{D}}\mathbf{U}^T,$$

where $\tilde{\mathbf{D}} \equiv \mathbf{D}\mathbf{D}^T$ is a $D \times D$ diagonal matrix with N squared singular values on the main diagonal. We can see that $\mathbf{U}\tilde{\mathbf{D}}\mathbf{U}^T$ is in the form of an eigen-decomposition and so the solution is the same as the SVD decomposition of $\mathbf{X}$.

PCA components are orthogonal linear combinations that maximise total variance. However, interpreting the actual components is not so straightforward. PCA has been used to reduce the dimensionality of the $\mathcal{C}$-space and the respective components were used to interpret stellar chemistry evolution processes as explored by Ting et al. (2012) (PCA) and Price-Jones & Bovy (2017) (expectation maximised PCA).

Now, we present an alternative component recovery method to PCA: FA, a method that searches for hidden "factors" to explain the correlation between observed variables.

3.2 Factor Analysis

FA is a latent linear model that searches for influential underlying factors (or the latent variables) from a set of observed variables. FA fundamentally differs from PCA in the way that it approaches the problem of approximation to a low dimensional manifold. Instead of optimally creating components that form a larger set of variables through linear combinations, FA instead aims to express each variable as a weighted combination (factor loadings) of hypothesised latent factors with some added noise.

There are two types of factor models generally accepted (DeCoster 1998, Kline 2014): exploratory factor analysis and confirmatory factor analysis. The main difference between the two is exploratory factor analysis aims to discover the nature of constructs influencing observed variables and confirmatory factor analysis have *a priori* assumptions made about the underlying constructs and sets to test whether they behave in a predicted way.

From Chapter 21.1: *Factor Analysis* (Barber 2020), our factor model aims to find a lower dimensional probabilistic description of a dataset given by

$$\mathbf{V} = \{\mathbf{v}^1, \ldots, \mathbf{v}^N\}$$

where $\dim(\mathbf{v}) = D$.

If our dataset lies close to a H-dimensional linear subspace, we can project each datapoint to the subspace and accurately approximate them using the H-dimensional coordinate system, similar to PCA.

We attempt to explain a set of D observations in each of N individuals in $\mathbf{V}$ with a set of H common factors where there are fewer factors than observations ($H < D$).

The factor model will generate an observation based on the equation

$$\mathbf{V} - \mathbf{M} = \mathbf{F}\mathbf{h} + \boldsymbol{\varepsilon},$$

where $\mathbf{F} \in \mathbb{R}^{D \times H}$ is the factor matrix, $\mathbf{h} \in \mathbb{R}^{H \times N}$ is the *factor loading* matrix which contains the “weight” of each factor, and $\boldsymbol{\varepsilon}$ is the unobserved stochastic error with zero mean and covariance $\boldsymbol{\Psi}$. $\mathbf{M} \in \mathbb{R}^{D \times N}$ is the mean matrix that sets the origin of the coordinate system, where $\mathbf{m}^i = \bar{\mathbf{v}}^i$. We then set $\mathbf{X} \equiv \mathbf{V} - \mathbf{M}$.

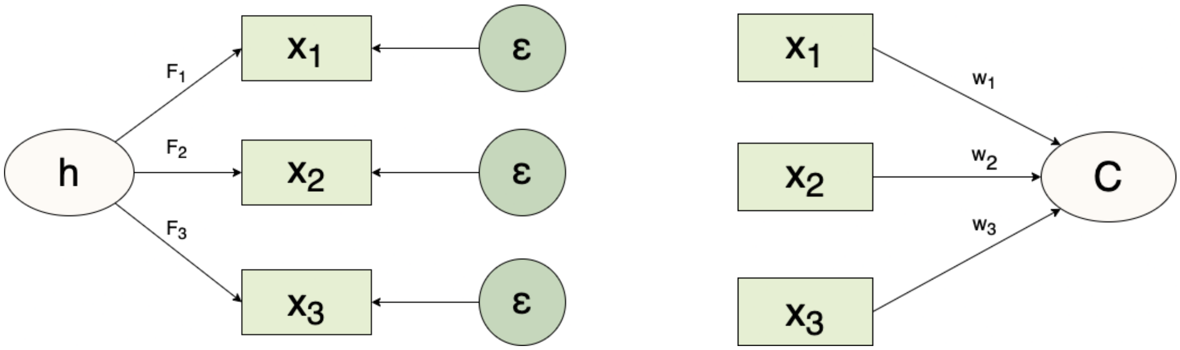
Figure 1 shows graphically how the direction of influence of the two techniques are reversed. We see that FA assumes that observable variables are based on underlying factors whereas components are defined as measured responses. Exploratory FA assumes the the variance in the variables can be decomposed into common and unique factors which account for it. The principal components contain both common and unique variance since they are just linear combinations of the measurements.

The choice of covariance also differs between PCA and FA. Both methods aim to provide a “low-rank” (limited number of principal components or latent factors used) approximation from a given covariance (or correlation) matrix. For a $D \times D$ covariance matrix $\boldsymbol{\Sigma}$, we have that

$$\begin{aligned} \text{PCA} : \boldsymbol{\Sigma} &\equiv \mathbf{W}\mathbf{W}^T \\ \text{PPCA} : \boldsymbol{\Sigma} &\equiv \mathbf{W}\mathbf{W}^T + \sigma^2\mathbf{I} \\ \text{FA} : \boldsymbol{\Sigma} &\equiv \mathbf{F}\mathbf{F}^T + \boldsymbol{\Psi} \end{aligned}$$

where $\boldsymbol{\Psi} = \text{diag}(\psi_1, \dots, \psi_D)$ and $\mathbf{W} \in \mathbb{R}^{D \times H}$ is the matrix representing H principal components. PPCA stands for *probabilistic PCA* which is an intermediate model complexity between PCA and FA. It takes the H principal eigenvalues and their corresponding eigenvectors of the sample covariance matrix. This serves as a useful initialisation for FA.

We see that FA differs from PCA through $\text{diag}(\boldsymbol{\Sigma})$. As the dimensionality D increases, the diagonal has a lesser impact since there are only D elements on the diagonal and $D(D-1)/2 = \mathcal{O}(D^2)$ elements off the diagonal. Therefore, FA models the “common part” of the matrix by taking into account all the off-diagonal elements and the common part of the diagonal, providing a richer description of the off-subspace noise $\boldsymbol{\Psi}$. PCA therefore explains variance but explains correlations imprecisely for a smaller number of observed dimensions. FA explains correlations but cannot account (using the common factors) as much data variation as PCA can.



(a) Factor model **(b)** Component model

Figure 1: *Factors are latent variables that explain the covariances (correlations) between observed variables. Components are a linear summation of variables and do not necessarily reveal the correlations between them. The correlations between the variables in the factor equation are expressed in terms of the factor loadings ($\mathbf{h}$) which is then multiplied by the factors ($\mathbf{F}_i$) to give the observed variables ($\mathbf{x}_i$). Noise (ε) is then added to give the final variable. The component equation shows components ($\mathbf{C}$) represented as a linear combination of the variables ($\mathbf{x}_i$) and their weights (w_i). Observed variables are represented by boxes and unobserved components are represented in ellipses. Graphs created using* `draw.io`

4 Implementation

We attempt to build from scratch an FA model as an exercise in data analysis and modelling in Python. The `FactorAnalyzer` package created by `scikit-learn` was used as a benchmark to compare the recovery of the mock latent factors. `FactorAnalyzer` performs a maximum likelihood estimate of the factor loading matrix using an SVD-based approach (Pedregosa et al. 2011) [1]. Our rudimentary model uses the same method following Algorithm 21.1 (Barber 2020).

The companion Jupyter Notebook to the report can be found at `https://github.com/yangdabei/factor-analysis`.

[1] `https://github.com/scikit-learn/scikit-learn/blob/main/sklearn/decomposition/_factor_analysis.py`

Algorithm 21.1

```
 1. Initialise the diagonal noise $\boldsymbol{\Psi}$.
 2. Find the mean $\bar{\mathbf{v}}$ of the data $\mathbf{v}^1, \ldots, \mathbf{v}^N$.
 3. Find the variance $\sigma_i^2$ for each component $i$ of the data $v_i^1, \ldots, v_i^N$.
 4. Compute the centred matrix $\mathbf{X} = [\mathbf{v}^1 - \bar{\mathbf{v}}, \ldots, \bar{\mathbf{v}}^N - \bar{\mathbf{v}}]$
 5. while Likelihood not converged or termination criterion not reached do
      6. Form the scaled data matrix $\tilde{\mathbf{X}} = \boldsymbol{\Psi}^{-\frac{1}{2}}\mathbf{X}/\sqrt{N}$.
      7. Perform SVD for $\tilde{\mathbf{X}} = \mathbf{U}\tilde{\boldsymbol{\Lambda}}\tilde{\mathbf{W}}^{\mathbf{T}}$ and set $\boldsymbol{\Lambda} = \tilde{\boldsymbol{\Lambda}}^2$
      8. Set $\mathbf{U}_H$ to the first $H$ columns of $\mathbf{U}$ and set $\boldsymbol{\Lambda}_H$ to contain the first $H$ diagonal
         entries of $\boldsymbol{\Lambda}$.
      9. $\mathbf{F} = \boldsymbol{\Psi}^{\frac{1}{2}}\mathbf{U}_H(\boldsymbol{\Lambda}_H - \mathbf{I}_H)^{\frac{1}{2}}$          ▷ factor update
     10. $L = \frac{N}{2}\left\{\sum_{i=1}^{H}\log\lambda_i + H + \sum_{i=H+1}^{D}\lambda_i + \log\det(2\pi\Psi)\right\}$          ▷ log likelihood
     11. $\boldsymbol{\Psi} = \text{diag}\left(\sigma^2\right) - \text{diag}\left(\mathbf{F}\mathbf{F}^{\mathbf{T}}\right)$          ▷ noise update
12. end while
```

First we set some initial constants:

```python
import numpy as np

# initial constants and variables
L_old = -np.infty
# error tolerance
tol = 1e-3
# max iterations before termination
max_iter = 100
```

The following code blocks correspond to the steps of the algorithm:

```python
# 1. Initialise diagonal noise
noise = 0.01*np.random.normal(size=(num_dimension,num_dimension))
psi = np.diag(np.diag(np.cov(noise)))
```

```python
# 2. Find the mean of the data
result = []
for vector in V:
    result.append([sum(vector)/len(vector)])

M = np.array(result)
```

```
#3. Find the variance
var = np.var(V, axis=0)
```

```
# 4. Compute the centred matrix
X = V - M
```

```
# 5. While loop runs if error is greater than tolerance or until max_iter
for i in range(max_iter):
    #----------------------------------------------------------------------------
    # 6. Form the scaled data matrix
    X_tilde = np.dot(np.linalg.inv(np.sqrt(psi)), X.T / (num_sample ** 0.5))
    #----------------------------------------------------------------------------
    # 7. Perform SVD for the scaled data matrix
    U,Lambda_tilde,VT = np.linalg.svd(X_tilde)
    Lambda_tilde = np.diag(Lambda_tilde)
    Lambda = Lambda_tilde ** 2
    #----------------------------------------------------------------------------
    # 8. Set U_H to the first H columns of U and set Lambda_H to contain the
    #    first H diagonal entries of Lambda
    H = num_latent
    U_H = U[:,:H]
    Lambda_H = Lambda_tilde[:H,:H]
    #----------------------------------------------------------------------------
    # 9. Factor update
    F = np.sqrt(psi) @ U_H @ np.sqrt(Lambda_H - np.identity(H))
    #----------------------------------------------------------------------------
    # 10. Log likelihood
    a = 0
    for j in range(H):
        a += np.log(Lambda[j,j])
    b = 0
    for k in range(H+1,num_dimension):
        b += Lambda[k,k]

    L_new = (num_dimension/2)*(a + H + b + np.log(np.linalg.det(2*np.pi*psi)))
    #----------------------------------------------------------------------------
    # 11. Psi update
    psi = np.maximum(np.diag(var)-np.diag(F@F.T),1e-12)
    #----------------------------------------------------------------------------
    if np.abs(L_new-L_old)<tol:
        break
    # log likelihood update
    L_old = L_new
```

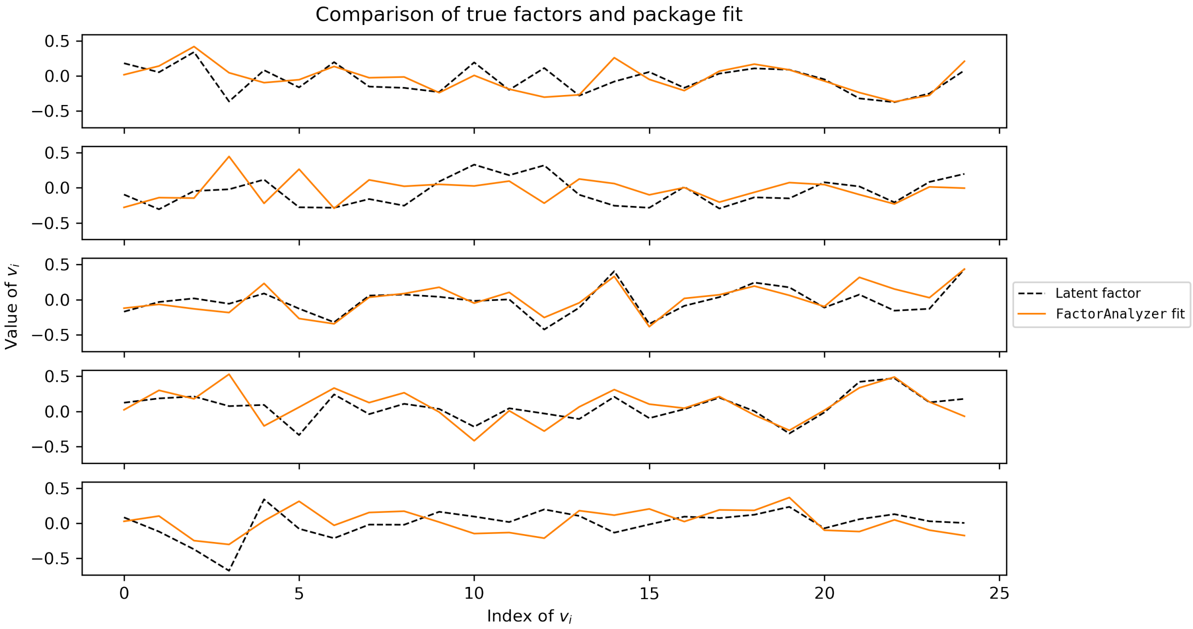


Figure 2: *True latent factors (dashed black line) compared to the factors derived from* `FactorAnalyzer` *(solid orange line).*

We apply the model outlined to a mock data sample and random latent factors. We first fit `FactorAnalyzer` to the mock data and call the method `.components_` to extract the factors. Then we apply our model to the sample. The number of factors were determined prior by finding the number of eigenvalues of the correlation matrix > 1. `mockdata.csv` contains the mock data and `latentfactors.csv` contain the mock factors. All files can be found at `https://github.com/yangdabei/factor-analysis`.

Note that some true factors are used more than once to compare the package fits. This is due to how closely the L_1-norm fits to the true factors. To show the comparison for the factors, Figure 2 contrasts the true factors with the components extracted from `FactorAnalyzer` and Figure 3 is the difference between the true factors and our created model fit.

5 Identifiability of Latent Factors up to a Rotation

Identifiability of factors is an issue in FA due to the nature of how the lower dimensional manifold is formed. Being able to uniquely identify the factor loading matrix and the latent factors ensures substantive interpretations to be made. Because reproduction of factors have an infinite amount of mathematically correct expressions, we need to have a way of choosing the solution which gives us the most substantive interpretations. This comes in the form of manipulating the reference axes of the spanning vectors of the subspace, also known as a rotation.

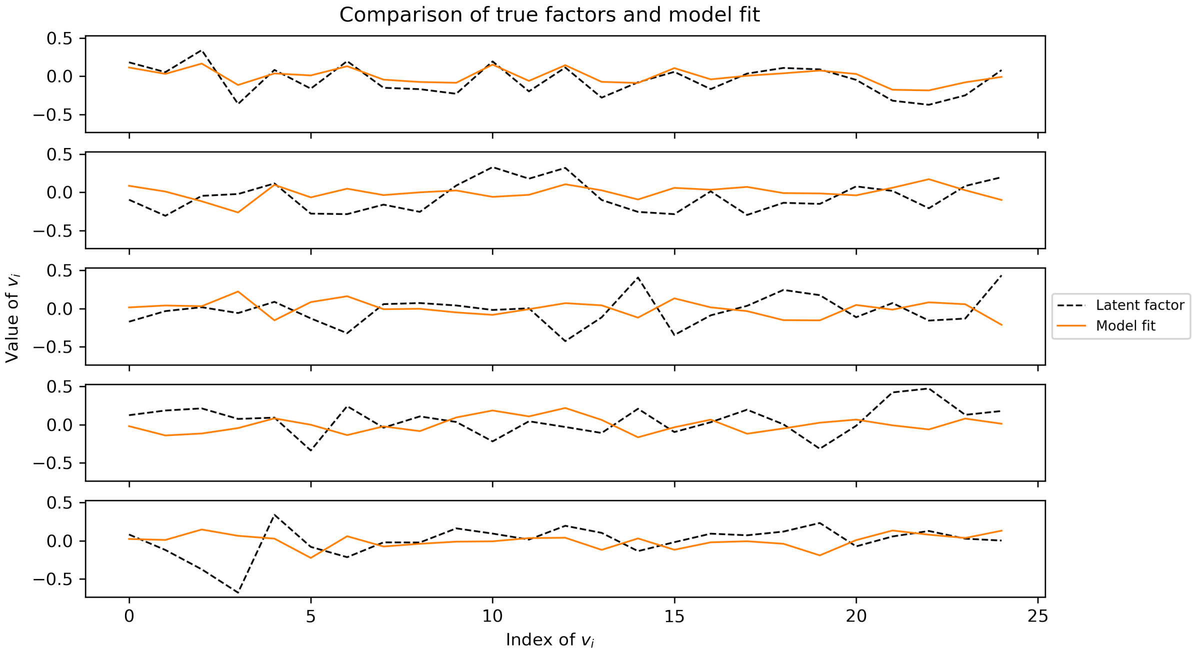


Figure 3: *True latent factors (dashed black line) compared to the factors derived from created model (solid orange line).*

Choosing the correct rotation can also make interpretation easier. Rotation methods are either orthogonal (assumes factors in analysis are uncorrelated) or oblique (assumes factors in analysis are correlated). `FactorAnalysis` currently has two rotation methods implemented, both of which are orthogonal: `varimax` and `quartimax`. Varimax is arguably the most popular method for orthogonal rotation where the sum of the variances of the squared loadings is maximised (Kaiser 1958). This still preserves the orthogonality and leaves the subspace invariant.

Consider an arbitrary $H \times H$ rotation matrix $\mathbf{R}$, and thus satisfying $\mathbf{R}\mathbf{R}^T = \mathbf{I}$. If we define $\tilde{\mathbf{F}} = \mathbf{F}\mathbf{R}^T$ and then rotate the latent factors as $\tilde{\mathbf{h}} = \mathbf{R}\mathbf{h}$, we have that

$$\tilde{\mathbf{F}}\tilde{\mathbf{h}} = \mathbf{F}\mathbf{R}^T\mathbf{R}\mathbf{h} = \mathbf{F}\mathbf{h}.$$

The covariance matrix for $\tilde{\mathbf{X}}$ is

$$\tilde{\boldsymbol{\Sigma}} = \tilde{\mathbf{F}}\tilde{\mathbf{F}}^T + \boldsymbol{\Psi} = \tilde{\mathbf{F}}\mathbf{R}\mathbf{R}^T\tilde{\mathbf{F}}^T = \mathbf{F}\mathbf{F}^T + \boldsymbol{\Psi} = \boldsymbol{\Sigma}.$$

Hence, datapoints x_i conditioned on latent factors h_i is not identifiable due to rotational indeterminacy and can simultaneously rotate the latent factors and the factor loading matrix without changing the distribution of the data.

Since solutions are not unique and can be affected by a rotation, we want to find individual vectors that span the plane of the coordinate system. Forcing the factor matrix to be orthonormal and ordering the columns in order of decreasing variance of the corresponding latent factors means that identifiability becomes possible due to more constraints being put onto $\mathbf{F}$. This is ultimately the strategy that PCA adopts.

Currently, the two parameters of our model ($\mathbf{F}$, $\mathbf{\Psi}$) are not initialised, resulting in some vectors having a skewed fit in Figure 3. To address the issue of rotation, if we initialise an $\mathbf{F}_0$ as the true factors with some added noise, it is expected that we can better the model. Then, whether the subspace is rotated will not play a part as the initialisation is close to the true value. At the moment of writing, the initialisation is being refined. We can initialise close to the true value for when we apply the model to spectral data.

6 Future Work

Although we offer some comparisons in Figures 2 and 3, it is beyond the scope of this report to interpret the factors of stellar spectra. For future observational research, we can begin to perform chemical tagging and apply our factor model to the high-resolution ($R \sim 28\,000$) stellar spectra gathered from the GALactic Archaeology with HERMES (GALAH) survey using the High Efficiency and Resolution Multi-Element Spectrograph (HERMES) instrument. From there, we can extract the latent factors of the multi-dimensional $\mathcal{C}$-space and interpret them as elements that drive stellar processes.

Further work can be done by altering the model so that it uses an "expectation maximisation" approach to retrieve the latent factors and to incorporate rotation methods to better fit the data.

7 Conclusion

The aim of this project was to be introduced to some preliminary machine learning models, FA and PCA. We created an exploratory factor model using a maximum likelihood SVD-based approach, inspired by the `FactorAnalyzer` package from the `scikit-learn` module. We then attempted to recover the latent factors from a mock data sample. There are many more features yet to be added to the model such as tackling the problem of rotations which can be solved by implementing a good initialisation $\mathbf{F}_0$ or adding a rotation optimiser. The ultimate goal is to apply the model to the spectral data gathered by the GALAH survey and to interpret the latent factors as nucleosynthetic processes.

8 Acknowledgements

This project was supported by Yuan-Sen Ting. Thanks is also to be given to Zechang Sun for inspiration with optimisation ideas from his factor model.

References

Abbott, B. P., Abbott, R., Abbott, T., Acernese, F., Ackley, K., Adams, C., Adams, T., Addesso, P., Adhikari, R., Adya, V. B. et al. (2017), 'Gw170817: observation of gravitational waves from a binary neutron star inspiral', *Physical review letters* **119**(16), 161101.

Barber, D. (2020), *Bayesian reasoning and machine learning*, Cambridge University Press.

Boothroyd, A. I. (2006), 'Heavy elements in stars', *Science* **314**(5806), 1690–1691.

Bovy, J. (2016), 'The chemical homogeneity of open clusters', *The Astrophysical Journal* **817**(1), 49.
URL: *http://dx.doi.org/10.3847/0004-637X/817/1/49*

De Silva, G. M., Freeman, K. C., Asplund, M., Bland-Hawthorn, J., Bessell, M. S. & Collet, R. (2007), 'Chemical homogeneity in collinder 261 and implications for chemical tagging', *NASA/ADS* .
URL: *https://ui.adsabs.harvard.edu/abs/2007AJ....133.1161D/abstract*

De Silva, G. M., Freeman, K. C., Bland-Hawthorn, J., Martell, S., de Boer, E. W., Asplund, M., Keller, S., Sharma, S., Zucker, D. B., Zwitter, T. & et al. (2015), 'The galah survey: scientific motivation', *Monthly Notices of the Royal Astronomical Society* **449**(3), 2604–2617.
URL: *http://dx.doi.org/10.1093/mnras/stv327*

De Silva, G. M., Sneden, C., Paulson, D. B., Asplund, M., Bland-Hawthorn, J., Bessell, M. S. & Freeman, K. C. (2006), 'Chemical homogeneity in the hyades', *The Astronomical Journal* **131**(1), 455–460.
URL: *http://dx.doi.org/10.1086/497968*

DeCoster, J. (1998), 'Overview of factor analysis'.

Freeman, K. & Bland-Hawthorn, J. (2002), 'The new galaxy: Signatures of its formation', *Annual Review of Astronomy and Astrophysics* **40**(1), 487–537.
URL: *http://dx.doi.org/10.1146/annurev.astro.40.060401.093840*

Gaia Collaboration et al. (2016), 'The gaia mission', *A&A* **595**, A1.
URL: *https://doi.org/10.1051/0004-6361/201629272*

Kaiser, H. F. (1958), 'The varimax criterion for analytic rotation in factor analysis', *Psychometrika* **23**(3), 187–200.

Kline, P. (2014), *An easy guide to factor analysis*, Routledge.

Lattimer, J. M., Mackie, F., Ravenhall, D. & Schramm, D. (1977), 'Decompression of cold neutron star matter'.

Majewski, S. R., Schiavon, R. P., Frinchaboy, P. M., Prieto, C. A., Barkhouser, R., Bizyaev, D., Blank, B., Brunner, S., Burton, A., Carrera, R. & et al. (2017), 'The apache point observatory galactic evolution experiment (apogee)', *The Astronomical Journal* **154**(3), 94.
URL: *http://dx.doi.org/10.3847/1538-3881/aa784d*

Mattsson, L. & Höfner, S. (2011), 'Dust-driven mass loss from carbon stars as a function of stellar parameters', *Astronomy Astrophysics* **533**, A42.
URL: *http://dx.doi.org/10.1051/0004-6361/201015572*

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M. & Duchesnay, E. (2011), 'Scikit-learn: Machine learning in Python', *Journal of Machine Learning Research* **12**, 2825–2830.

Price-Jones, N. & Bovy, J. (2017), 'The dimensionality of stellar chemical space using spectra from the apache point observatory galactic evolution experiment', *Monthly Notices of the Royal Astronomical Society* **475**(1), 1410–1425.
URL: *http://dx.doi.org/10.1093/mnras/stx3198*

Qian, Y., Vogel, P. & Wasserburg, G. J. (1998), 'Diverse supernova sources for ther-process', *The Astrophysical Journal* **494**(1), 285–296.
URL: *http://dx.doi.org/10.1086/305198*

Steinmetz, M., Zwitter, T., Siebert, A., Watson, F. G., Freeman, K. C., Munari, U., Campbell, R., Williams, M., Seabroke, G. M., Wyse, R. F. G. & et al. (2006), 'The radial velocity experiment (rave): First data release', *The Astronomical Journal* **132**(4), 1645–1668.
URL: *http://dx.doi.org/10.1086/506564*

Thompson, T. A., Burrows, A. & Meyer, B. S. (2001), 'The physics of proto-neutron star winds: implications for r-process nucleosynthesis', *The Astrophysical Journal* **562**(2), 887.

Ting, Y. S., Freeman, K. C., Kobayashi, C., De Silva, G. M. & Bland-Hawthorn, J. (2012), 'Principal component analysis on chemical abundances spaces', *Monthly Notices of the Royal Astronomical Society* **421**(2), 1231–1255.
URL: *http://dx.doi.org/10.1111/j.1365-2966.2011.20387.x*

Zhao, G., Zhao, Y., Chu, Y., Jing, Y. & Deng, L. (2012), 'Lamost spectral survey'.