7500 Workbook Activity 1
January 12, 2026
Typeset using LaTeX in AASTeX631

Principal Component Analysis & Regression

Reading: Francis et al 1992 ApJ 398, 476, Ivezic 7.3-7.6

1 Data Activity: PCA of Quasar spectra

In this activity, we will learn to query SDSS for bright quasar spectra, find the principal components, and try to interpret them (thus encountering some limitations to PCA). Then, we will use the principal components to do some linear regression.

(1) Go to https://skyserver.sdss.org/dr19, click on "SQL search", then "Spectroscopic Search".

(2) Select "None" for positional constraints, limit the redshift range of QSOs to 0.45 to 0.48, r mag range $15-18$ (we want the highest S/N QSOs) and select 0 for max entries. You should get about 120 spectra in total.

(3) Upload list of Plate, MJD, and Fiber to SAW.

(4) Browse a few spectra, and notice some identifications of the broad and narrow lines. Then download spectra through Rsync or wget.

(4a) use the `find . -name ‘‘*fits" > file.lst` Unix command to pipe all the downloaded .fits files to a file called file.lst.

(5) Shift all spectra to the rest frame wavelength using the "Z" header keyword in the 2nd extension of each FITS file. In the first extension you'll find loglam, flux, and ivar (inverse variance). Interpolate the fluxes and uncertainties to a common rest-frame wavelength range (I used $\approx$ 2635 Å to 6218 Å). Normalize the spectra by $\sqrt{\sum_\lambda f_{\lambda_i}^2}$ (see Connolly et al 1995 ApJ 110, 3). Are there any especially strange spectra you don't want to include because PCA is sensitive to outliers? When flipping through, compare to the mean spectrum.

(6) When flipping through the spectra to remove (a few only!) strange or crummy ones, compare to the mean and notice wher a lot of the variance occurs (by eye).

(7) Find eignvalues and eigenspectra, e', from the algorithm above. To start, create your data matrix, X with spectra along rows. You can use `np.linalg.eig` to find the eigenvectors, and I use `np.matmul` to multiply matrices. Plot the first 5 eigenspectra, including the mean. Write a one or two sentence explanation describing what you're seeing in each eigenspectrum (it might help to overplot the mean spectrum on a different axis). Note: with PCA you often have a sign ambiguity (e.g., the eigensectrum

might need to be multiplied by -1). It might also help to refer to Yip et al. 2004 AJ 128, 2603 and Francis et al 1992 ApJ 398, 476. It might also help to skip ahead to Part 3 to gain some insight into what the eigenspectra mean in terms of linear regression (you can come back to this part).

(8) How much of the variance is contained in the first 2 eigenspectra, not counting the mean spectrum? First 4 not counting the mean?

(9) At what eigenspectrum does the variance start to really die out?

More on following pages...

2 Data Activity: PCA Regression

Armed with your first several (5, including the mean) principal component eigenspectra, you can approximately reconstruct any one of the original quasar spectra. When the eigenspectra form basic functions for linear least squares, this is called PCA regression. In this case, you are fitting for coefficients θ which can be done with the matrix equation:

$$\hat{\vec{\theta}}_{\rm ML} = (J^{\rm T} C_{\vec{y}}^{-1} J)^{-1} (J^{\rm T} C_{\vec{y}}^{-1} \vec{y})$$

where J is the design matrix with basis funtions (eigenspectra) along columns, $\vec{y}$ is the spectrum of one of the AGN (here the data matrix is a single column vector), and $C_{\vec{y}}$ is the covariance (uncertainties) of the data.

(1) Take one Quasar spectrum. Your favorite one. Using the variance (see "IVAR" in FITS data) on each data point, find the maximum likelihood coefficients, $\hat{\vec{\theta}}_{\rm ML}$, with the first five eigenvectors (including the mean). Plot up the spectrum and the best fit superposition of eigenspectra.

(2) Given the process we just performed (*linear* regression) what is a limitation to the PCA analysis? (recall some discussion on this point in Ivezic textbook and in the Francis+1992 reading)

(3) Use the F-test to determine the if adding one more eigenvector (a 6th, including the mean) to the fit results in a significantly improved value of χ^2. To find the PTE (PTE=probability to exceed; recall that PTE gives the probability that the null hypothesis could have generated your value or larger) values for the F distribution, I would use the Appendix C.5 of Bevington (see Google Drive in "Other Reading Materials"). Does a BIC give the same conclusion? Recall that the Bayesian Information Criterion penalizes χ^2 by adding to it m log N where m is the number of model components. The F-test statistic is

$$\frac{\Delta\chi^2}{\chi^2_{\nu,m+1}} \tag{1}$$

where the denominator is the *reduced* χ^2 value for the model with $m+1$ components, whereas the numerator is the difference in total χ^2 (non-reduced) for the models with m and $m+1$ components.

Note: scikit-learn has a powerful PCA tool, `fit_transform()`, but you now have your own!

So, begin creating your "arsenal" of Python routines by adding a generic PCA routine and a generic linear least squares solver with a design matrix J that takes arbitrary basic functions. For the latter, you can use this for many more purposes other than PCA regression (I recommend to also create a separate routine that has polynomials as basis functions since they have a functional form that can be easily

coded up). For your python coding, you can stick to using basic functions or if you're fancy, use classes.

Reading for next time: Protassov et al 2002 ApJ 571, 545 "STATISTICS, HANDLE WITH CARE: DETECTING MULTIPLE MODEL COMPONENTS WITH THE LIKELIHOOD RATIO TEST".

Extra space...