Unit 6-2: Dimensionality Reduction

Text Processing & Information Retrieval

Geraldine Gray

Overview

- To date we have looked at preprocessing techniques applicable to text & web content mining, i.e. applicable to semi- or un- structured text.
- These techniques tend to produce a dataset with a large number of attributes, i.e. dimensionality tends to be high
- Apart from Support Vector Machines, classification and clustering algorithms work better if dimensionality is NOT high. Therefore, this unit looks at techniques to reduce the dimensionality of the dataset

Overview

- These slides will focus on two categories of attributes reduction techniques:

1. Attribute Weighting: identifies the most <u>predictive</u> attributes in a dataset (classification)
2. Attribute reduction: combines attributes using compression techniques to minimise loss of information content (classification and clustering)

Attribute Weighting

- Attribute weighting techniques rank attributes based on how useful they will be to a classification algorithm
- Weights are generally in the range [0,1], with more predictive attributes getting a higher weighting.

Question: Will all algorithms find the same attributes useful?

There are many classification algorithms, and so there are many ways to weight attributes . . .

Attribute Weighting

Using **Information Theory** to weight attributes:

- Ranks attributes in the same way as a Decision Tree ranks attributes, using information gain, gini index or chi-squared
- An attribute is 'useful' if it can provide a clean split between classes of data, as defined by the class label

Weight by Relief: based on Nearest Neighbour, an attribute is evaluated based on two criteria:

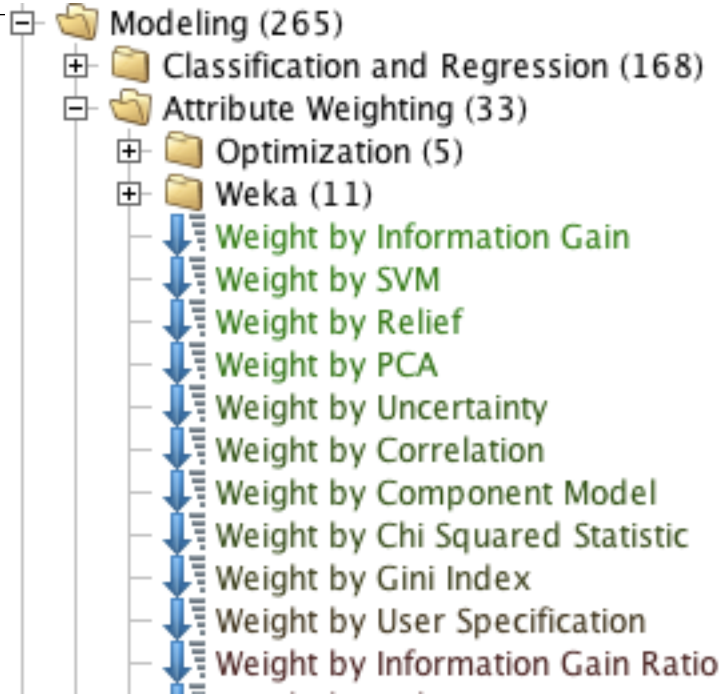
1. Does it have the same value as it's nearest neighbours in the same class?
2. Does it have a different value from it's nearest neighbours in different classes?

Attribute Weighting

Weight by Uncertainty weights attributes based on the probability of a term belonging to just one class, rather than being equally represented in all classes, and so matches Naïve Bayes.

Weight by SVM weights attributes based on whether are not they were influential in defining the class boundary, and so matches Support Vector Machines.

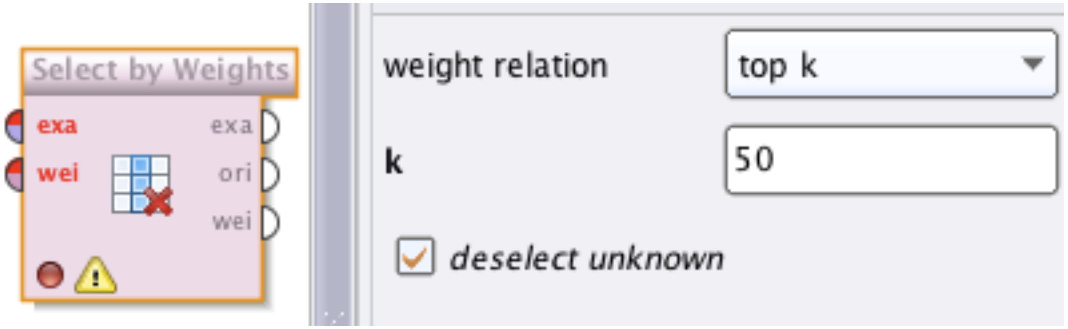
Attribute Weighting



- Modeling (265)
  - Classification and Regression (168)
  - Attribute Weighting (33)
    - Optimization (5)
    - Weka (11)
    - Weight by Information Gain
    - Weight by SVM
    - Weight by Relief
    - Weight by PCA
    - Weight by Uncertainty
    - Weight by Correlation
    - Weight by Component Model
    - Weight by Chi Squared Statistic
    - Weight by Gini Index
    - Weight by User Specification
    - Weight by Information Gain Ratio

- Rapidminer supports a number of weighting operators, some of which are shown here. They are listed under modeling/feature weights.
  - Operator names start with 'Weight by . . .'

- Once weighted, attributes can be selected using '<u>Select By Weight</u>' as show below:



weight relation: top k

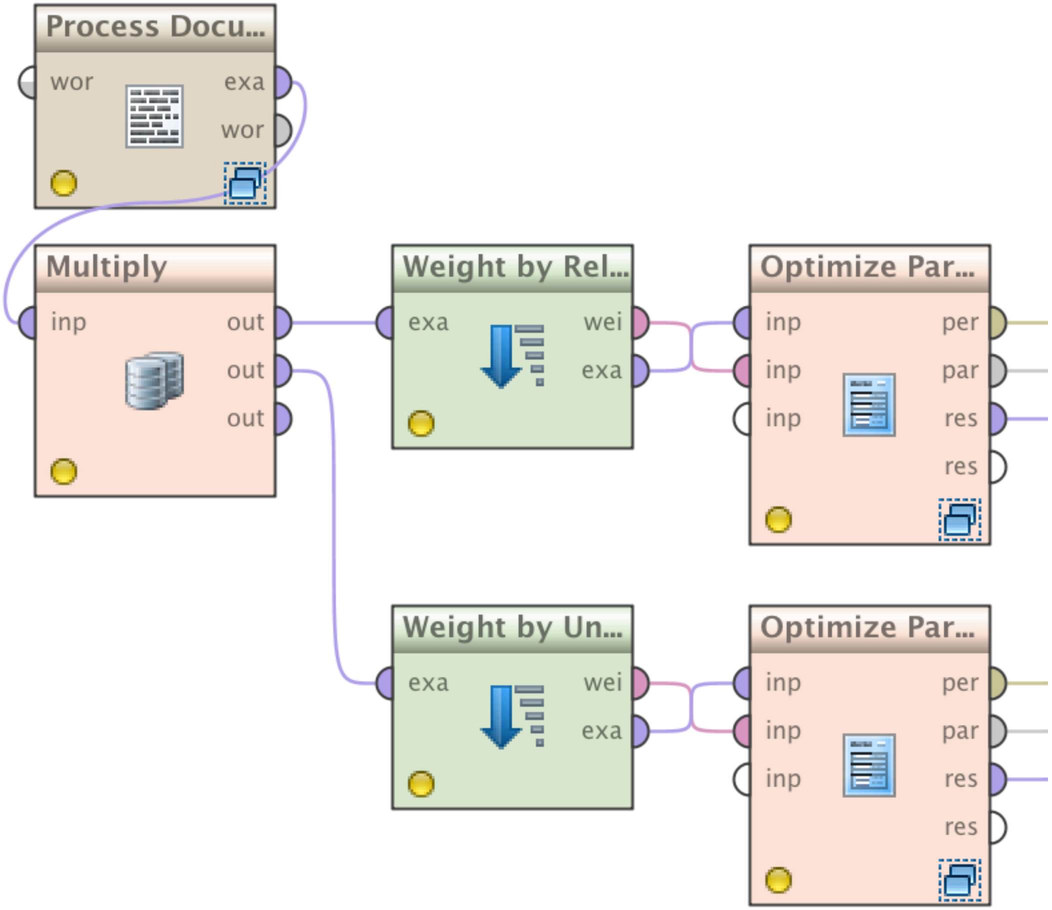
k: 50

☑ *deselect unknown*

Selects top 50 attributes based on weight.

Comparing weighting algorithms

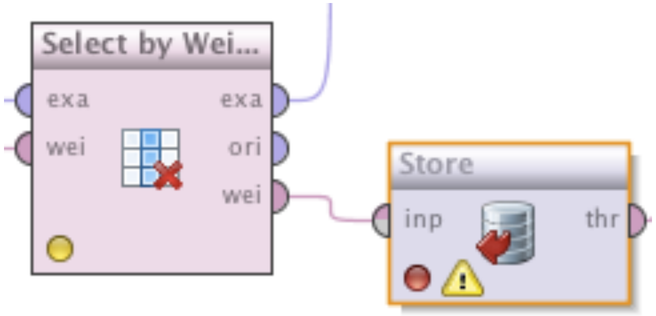
- Take a look at lect6-attributeWeighting.rmp on moodle



Attribute Weighting

What is the effect of using attribute weighting on generating a document vector for unlabeled data?

- Weighting has the effect of reducing the number of attributes available to the learner.
- The same weightings need to be used when preprocessing new document vectors (unlabeled).
- This can be done by outputting/storing the attribute weights, and using this list of weights for pre-selecting attributes in future processes.

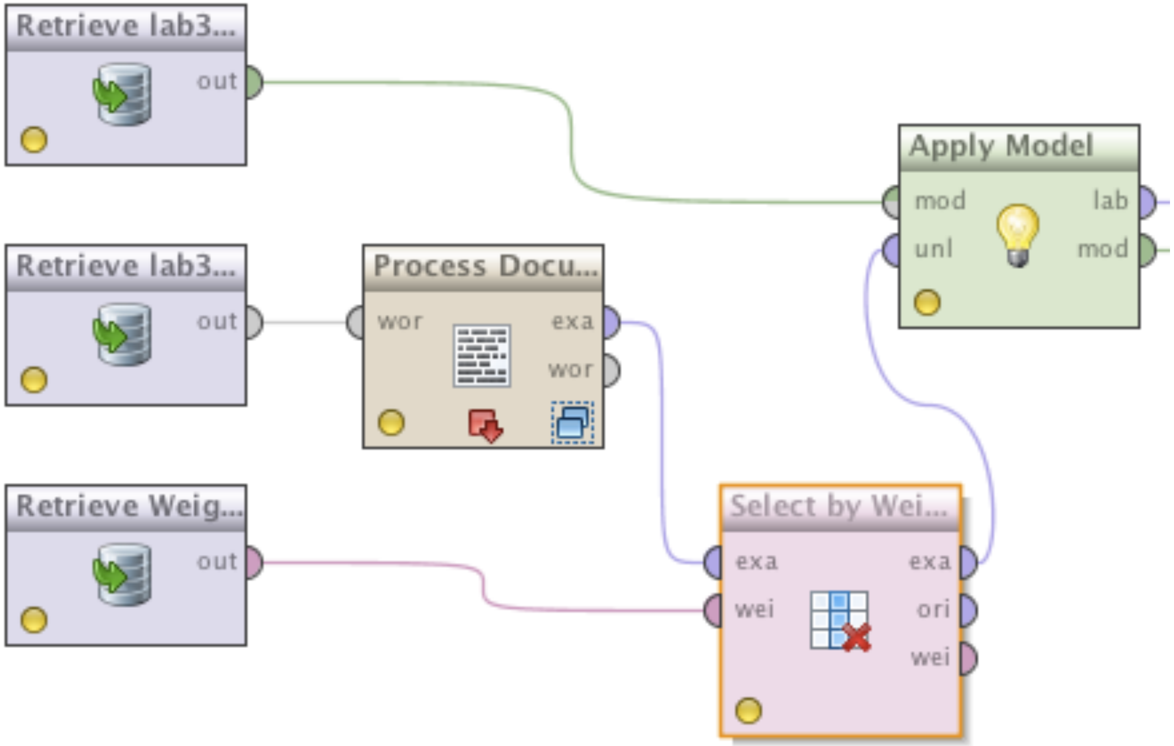


- So ‘mineunseen.rmp’ from lab 3 would become:

Retrieve the model

Retrieve the wordlist

Retrieve the weights



Dimensionality reduction

Combining attributes without losing information using

PRINCIPAL COMPONENT ANALYSIS

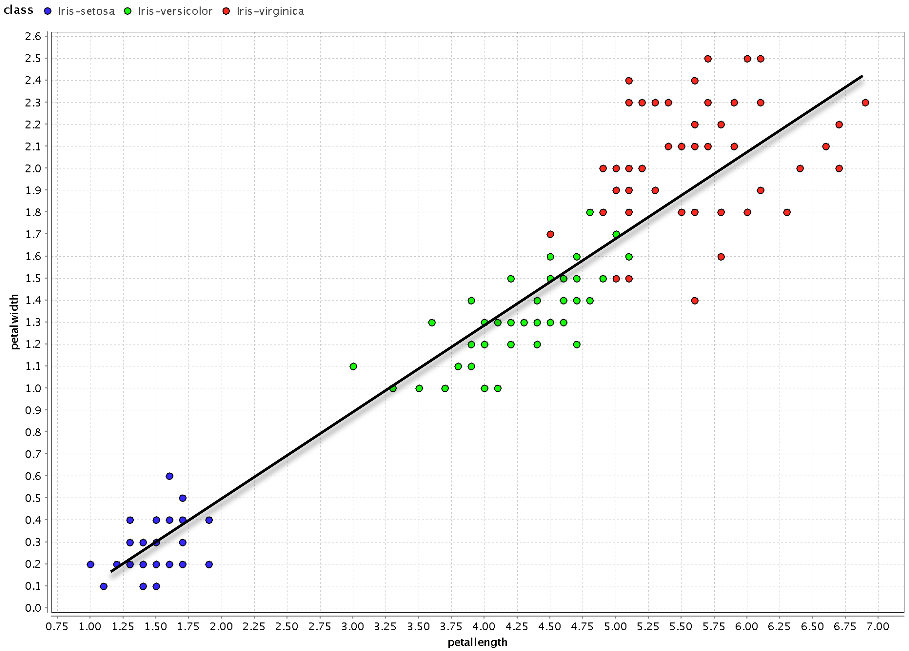
Racap from Data Analysis last semester.
Not needed for Text Analysis exam 2018

Principal Component Analysis (PCA)

- The objective of PCA is to:
  - Use the least number of attributes to capture all the relevant information in the dataset. These attributes are generated combinations of exiting attributes in the dataset.
  - Eliminate noise or random variation in the dataset

Idea behind PCA

- Take the following scatter plot of Petal Length versus Petal Width from the Iris data. Clearly they are correlated, so a single variable would capture most of the variation (changes in value) of these two attributes.



PCA attempts to capture underlying variability of a dataset using less attributes. The new attributes are called Principal Components, and they represent combinations of the original attributes in the dataset.

Principal Component Analysis

- PCA attempts to capture the pattern of the dataset using less attributes than the original dataset
  - The first principal component captures the main trend in the dataset
  - The second principal components attempts to capture the majority of the remaining pattern not covered in the first principal component
  - A third principal component would attempt to capture the remaining pattern not covered by the first two
  - etc. . . .
- When running PCA, you can determine what percentage of the original pattern you want to capture.

PCA in Rapidminer

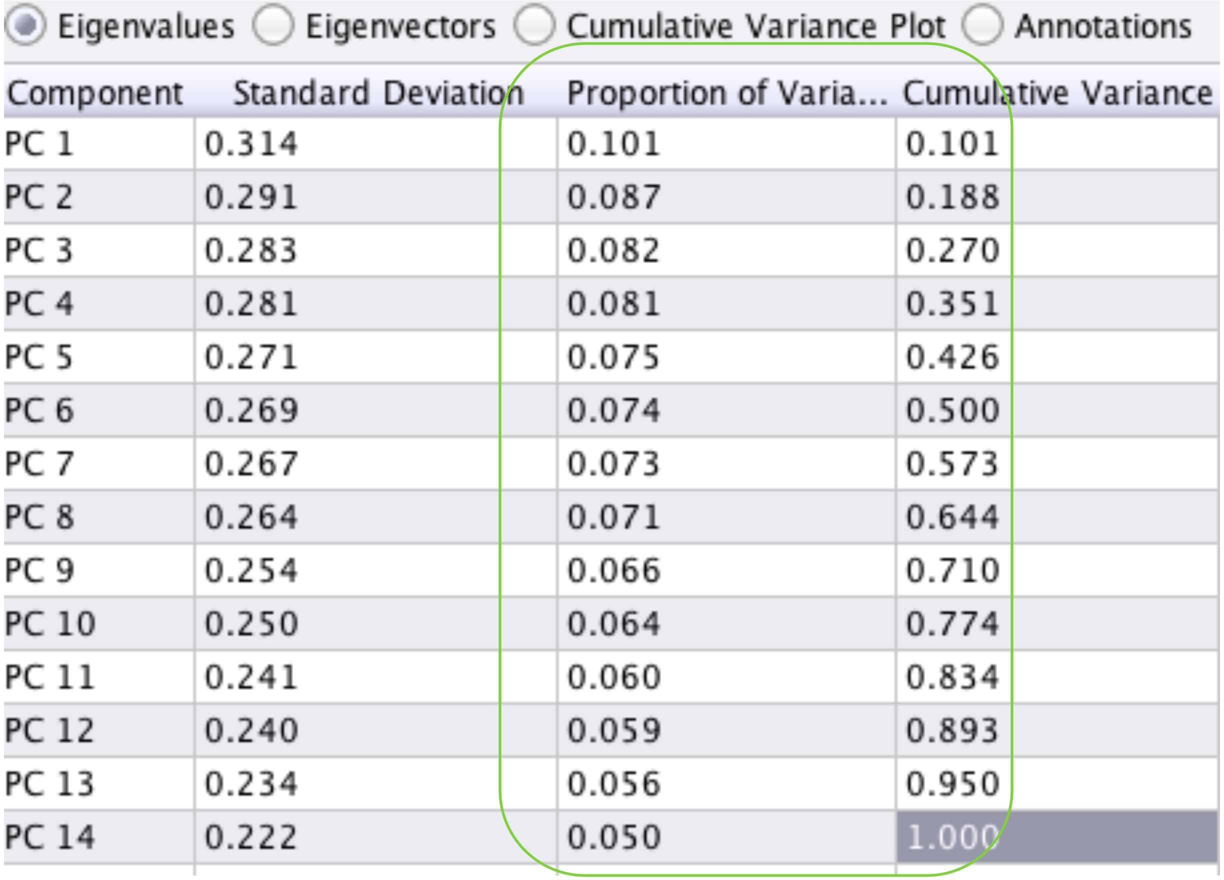
See PCAexample.rmp on moodle. You will need to redirect 'Process documents from Data' to the relevant folders on your machine.

- Applying PCA to the dataset from the fifteen news articles on Kenya, Crime and Healthcare, which had 204 attributes, generates a dataset of 14 attributes which capture ALL of the variability in the original dataset:

The first component (PC1) captured 10% of the pattern in the dataset (0.101). The second principal component captured another 8% (0.087) giving a total of 18.8% captured by PC1 & PC2 . . . By PC13, 95% of the original pattern had been captured.

So the original 204 attributes can now be replaced by 14 principal components (attributes) for modeling.

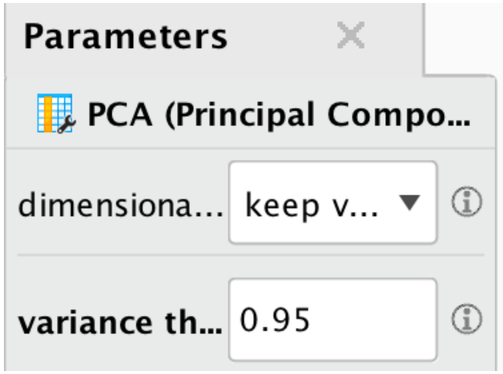
PCA model



Component	Standard Deviation	Proportion of Varia...	Cumulative Variance
PC 1	0.314	0.101	0.101
PC 2	0.291	0.087	0.188
PC 3	0.283	0.082	0.270
PC 4	0.281	0.081	0.351
PC 5	0.271	0.075	0.426
PC 6	0.269	0.074	0.500
PC 7	0.267	0.073	0.573
PC 8	0.264	0.071	0.644
PC 9	0.254	0.066	0.710
PC 10	0.250	0.064	0.774
PC 11	0.241	0.060	0.834
PC 12	0.240	0.059	0.893
PC 13	0.234	0.056	0.950
PC 14	0.222	0.050	1.000

PCA – keep all the variance?

- Every dataset has some noise / incorrect or unusual values
- As noise is not a dominant pattern in the dataset, noise tends to be captured by the final few principal components.
- Removing these can clean up a dataset.
- The default in Rapidminer is to include enough principal components to catpure 95% of the patterns of the original dataset:



Parameters

PCA (Principal Compo...

dimensiona... keep v...

variance th... 0.95

From the last slide: that would return a dataset with 13 attributes.

PCA - warning

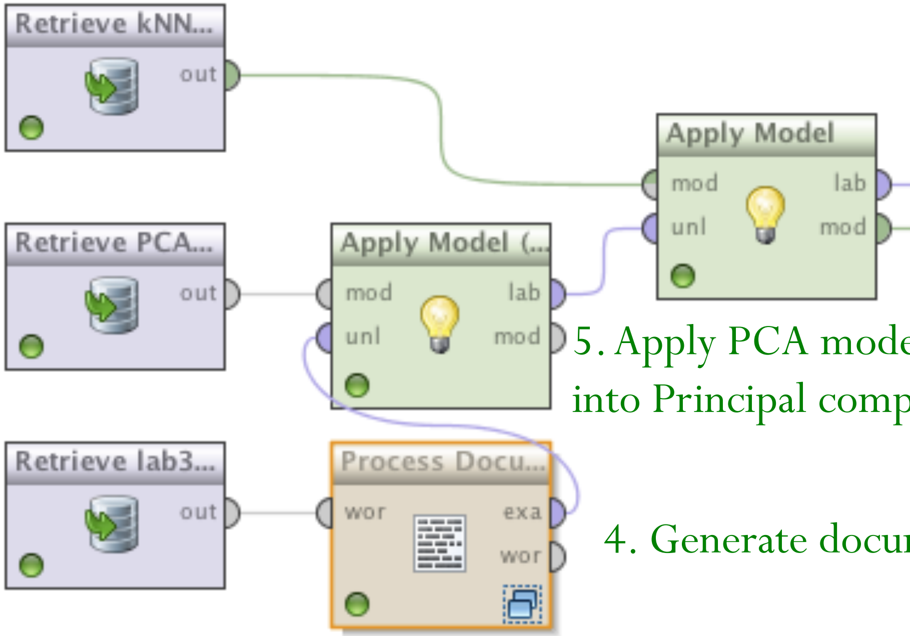
- PCA does not need a class label. Principal components are decided without considering the class label.

- If an attribute is responsible for a lot of the variance in the dataset, but is not predictive of the class label, it will still feature in a number of the principal components.
- Attributes that are irrelevent to the prediction task should to be removed before running PCA.
  - Weighting algrithms can help identify irrelevent attributes

PCA in Rapidminer

mineUnseenDataWithPCA.rmp on moodle.

- As with attribute weighting, if using PCA when training a model, the same PCA model must be applied to any future document vectors the model is applied to. A PCA model can be stored as per any other model, and then applied to future datasets to generate comparable principle components.
- So ‘mineunseen.rmp’ adapted for PCA would become:



1. Retrieve classification model (k-NN in this case)

2. Retrieve the PCA model

3. Retrieve the word list

4. Generate document vector

5. Apply PCA model to convert the dataset into Principal components

6. Apply k-NN model (which was trained on Principal components) to the new dataset.

Summary

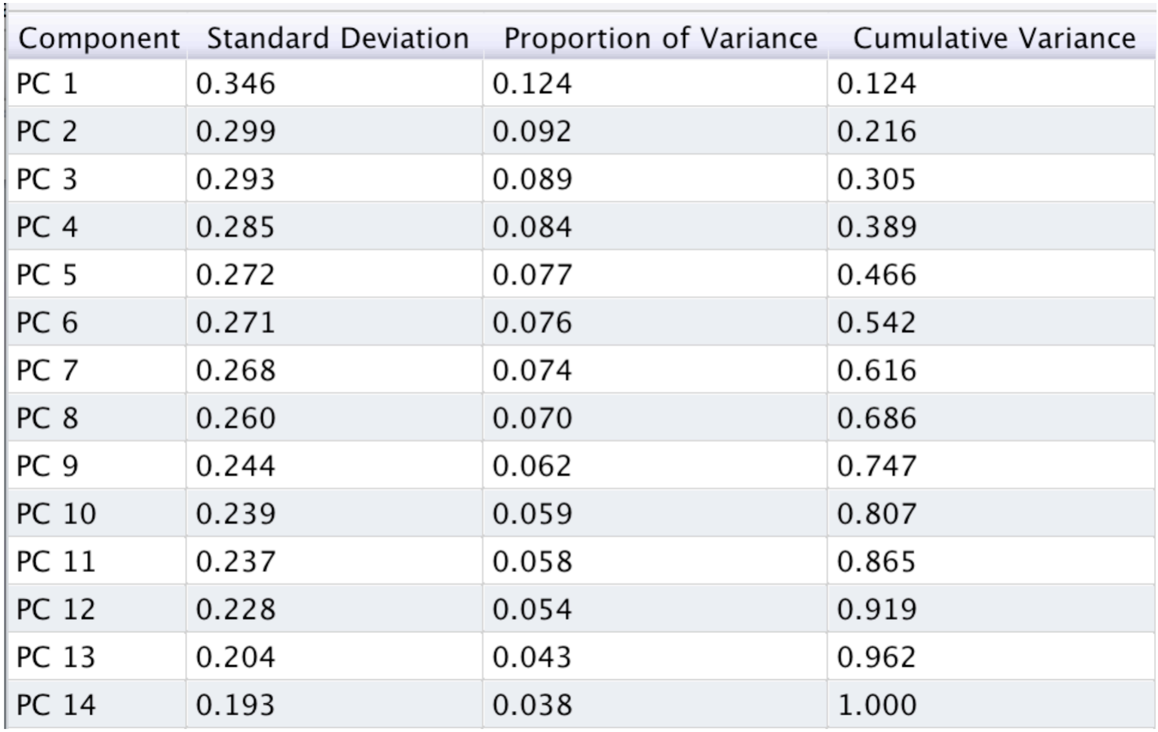
- To reduce the number of attributes you can:
    - Apply a weighting algorithm to rank the attributes, and select those with the highest weighting.
        - Remember to match the weighting algorithm with the learner you are using.
    - Extract the pattern from the dataset and represent that pattern using less attributes (PCA)

- For either approach, you will need to save the weights / PCA model to apply to any future documents that will be classified.

Past exam questions

May 2014, Q3a

- Give an overview of the effect of applying *Principal Component Analysis* (PCA) to a dataset of document vectors with 200 terms. In your answer, explain how to interpret the *Proportion of Variance* and *Cumulative Variance* of the PCA model in Figure 2 above. ***(11 marks)***



Component	Standard Deviation	Proportion of Variance	Cumulative Variance
PC 1	0.346	0.124	0.124
PC 2	0.299	0.092	0.216
PC 3	0.293	0.089	0.305
PC 4	0.285	0.084	0.389
PC 5	0.272	0.077	0.466
PC 6	0.271	0.076	0.542
PC 7	0.268	0.074	0.616
PC 8	0.260	0.070	0.686
PC 9	0.244	0.062	0.747
PC 10	0.239	0.059	0.807
PC 11	0.237	0.058	0.865
PC 12	0.228	0.054	0.919
PC 13	0.204	0.043	0.962
PC 14	0.193	0.038	1.000

May 2014, Q3a – Answer scheme:

a)Overview of PCA – points such as**:** **(6 marks)**

- Use the least number of attributes to capture all the relevant information in the dataset. These attributes are generated combinations of variables in the dataset.
- PCA attempts to capture the pattern of the dataset using less attributes than the original dataset
- The first principal component captures the main trend in the dataset
- The second principal components attempts to capture the majority of the remaining pattern not covered in the first principal component
- A third principal component would attempt to capture the remaining pattern not covered by the first two
- etc. . . .

Interpret the model: (**5 marks**)

- Proportion of Variance: What percentage of the variance of the original dataset has been captured by this component. E.g. PCA1 captures 12.4% of the information from the original dataset.
- Cumulative Variance illustrates how much of the variance of the original dataset has been captured by including all components up to this one, for example the model shows that only 14 principal components are needed to capture all the information from the original 200 attributes.

Summer 2015, Q3a

- Weight by Relief was used to select the top six terms for the training dataset in Table 2. The class label is *Topic*. Give an overview of how **Weight by Relief** assigns weights to terms in a document vector. How are term weights used to alleviate the curse of dimensionality? ***(9 marks)***

Summer 2015, Q3a: Answer scheme

5 Marks for understanding of:

- Weight by Relief: based on Nearest Neighbour, an attribute is evaluated based on two factors:
  - Does it have the same values as it’s nearest neighbours in the same class?
  - Does it have a different value from it’s nearest neighbours in different classes?

4 marks Curse of dimensionality: can be used to remove irelevant attributes by filtering by weight. Experimenting with different threshold values will help to select a weight threshold, below which attributes are filtered from the dataset, leaving the most predictive attributes as relevant to k-NN.