Principal components analysis II

Patrick Breheny

November 29

svd

- Singular value decompositions can be computed in R via the svd function:

  ```
  S <- svd(X)
  ```

  where Su, Sd, and S$v contain the three elements of the decomposition
- Thus, principal components can be computed manually provided that we first center and scale the matrix:

  ```
  P <- svd(scale(X))
  ```

prcomp

- However, it is generally more convenient to use the `prcomp` function instead:

  ```
  P <- prcomp(X,scale=TRUE)
  ```

  where it is worth pointing out that for certain historical reasons, the default is `scale=FALSE`, but generally, scaling the matrix is advisable
- It is also possible to specify a subset of variables using a formula interface:
  `prcomp(~ldl+sbp+obesity,data=heart)`
- The components of `P` that are most useful are `P$x`, which contains the principal components, `P$rotation`, which contains the loadings ($\mathbf{V}$), and `P$dev`, which contains the standard deviations of each of the principal components

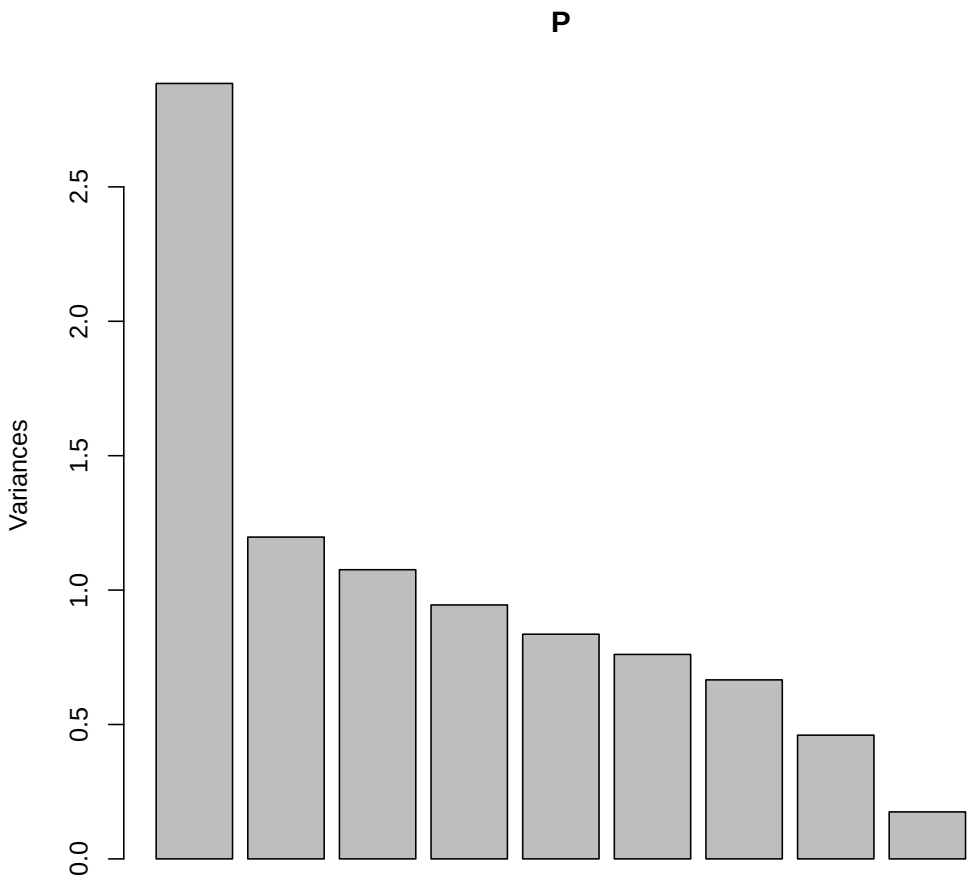
summary

There are several advantages of using `prcomp` instead of manually computing the principal components via `svd`; namely, that it provides `summary`, `plot`, and `predict` methods:

```
> summary(P)
Importance of components:
                          PC1   PC2   PC3   PC4
Standard deviation     1.698 1.094 1.037 0.972
Proportion of Variance 0.321 0.133 0.120 0.105
Cumulative Proportion  0.321 0.454 0.573 0.678
```

plot

```
plot(P)
```



predict

- Perhaps the most useful reason to use `prcomp` over `svd`, however, is the `predict` method
- Suppose you want to use your principal components to make predictions for new data; applying `prcomp` to the new data does not work, as the new principal components will not be the same as the old ones
- Fortunately, `predict(P,newX)` calculates the original principal components of `P` for the new data

PROC PRINCOMP

- In SAS, principal components analysis is carried out via `PROC PRINCOMP`:

```
PROC PRINCOMP DATA=X OUT=P;
RUN;
```

  where it is worth pointing out that SAS has the more sensible default of scaling the matrix, although you do have the option to leave `X` unscaled
- The output data set `P` contains the original variables as well as principal components, named `Prin1`, `Prin2`, ...

Supervised vs. unsupervised learning

- We will now take a more in-depth look at the heart disease data from the previous lecture
- Before we do so, however, it is worth noting once again that principal components have many uses, and is not solely a method for transforming inputs to regression problems
- Thus, although this problem has an outcome (CHD) that we are trying to predict, principal components are also widely used in problems were there is no "outcome", simply a number of variables for which we are interesting their interrelationships

Principal components regression

Results of a logistic regression fit to the principal components:

	β	SE	z value	p
PC1	-0.63	0.08	-7.71	< 0.0001
PC2	0.16	0.10	1.68	0.09
PC3	0.36	0.11	3.25	< 0.01
PC4	0.49	0.11	4.32	< 0.0001
PC5	-0.46	0.12	-3.83	< 0.001
PC6	-0.22	0.12	-1.79	0.07
PC7	-0.13	0.13	-0.95	0.34
PC8	-0.32	0.17	-1.89	0.06
PC9	-0.00	0.29	-0.00	1.00

Pros and cons of principal components regression

- The obvious advantages of regressing on the principal components are:
  - Variables are ordered in terms of standard error
  - Thus, the also *tend* to be ordered in terms of statistical significance
  - Variables are orthogonal, so including, say, PC9 in the model has no bearing on, say, PC3
- The primary disadvantage is that this model is far more difficult to interpret than a regular logistic regression model

Interpretability

- One of the main reasons why principal components are difficult to interpret is that every variable shows up in every factor (and vice versa)
- There are literally dozens of approaches that have been proposed to try to alleviate this problem and produce more interpretable components
- Typically, these approaches are grouped under the umbrella of *factor analysis*, with *sparse principal components analysis* a more modern take on the problem

Exploratory approaches

- Factor analysis and sparse PCA are automatic approaches, but the problem can be approached in a more exploratory fashion as well – or indeed, the two approaches can be combined
- A combination approach was taken by Harrell *et al.* in the original paper on the WHO-ARI data set that you have looked at in your homework
- The initial principal components were "rotated" using factor analysis, and then scientific judgment was used to pare the components down into compact, interpretable quantities that represented meaningful biological concepts

Multiple comparisons concerns

- One key thing to keep in mind is that exploration of factors/components can be done without worrying about multiple comparisons
- All we are contemplating is various transformations of the design matrix, and we are evaluating them without looking at the outcome, thereby retaining statistical validity
- If we were to tinker with the components based on whether they improve the significance of the resulting regression, on the other hand, statistical validity would be lost

Reconsidering factors

- Regardless of the method used, automatic construction of factors is rarely a good idea if the interpretability of the model is important
- For example, in the heart data set, age and body fat are highly correlated (0.63), but that doesn't mean it makes sense to think of them as arising from one common underlying factor
- Indeed, this is one of the goals of regression: the ability to separate out the effects of correlated predictors

Constructing factors for the heart data

- We could, for instance, fit a model in which age, tobacco, alcohol, family history, and stress are all included as conventional predictors, but that we use principal components to combine cholesterol, blood pressure, obesity, and adiposity
- On the positive side, this approach is more parsimonious and produces, on the whole, lower p-values
- On the negative side, it appears to miss the fact that LDL cholesterol is significantly associated with coronary heart disease

Condition numbers

- Besides their use in constructing principal components, the singular values themselves are often useful in detecting multicollinearity
- A common diagnostic for multicollinearity is the *condition number* κ, which is simply the largest singular value divided by the smallest
- A general rule of thumb is that if $\kappa > 15$, you have a multicollinearity problem, and if $\kappa > 30$, you have a big multicollinearity problem
- In the heart disease data set, $\kappa = 4$, suggesting that although some of the variables are correlated, multicollinearity is not an overwhelming concern

Final remarks

- On the whole, the coronary heart disease data set is not particularly well-suited for a principal components approach
- However, the method can dramatically improve estimation and insight in problems where multicollinearity is a large problem – as well as aid in detecting it
- It is very difficult to make sweeping generalizations about principal components; their use is very much decided on a case by case basis, as the homework assignment hopefully illustrates