Chapter 10: Unsupervised Learning

Credit: https://xkcd.com/1425/

This chapter will focus on methods intended for the setting in which we only have a set of features $X_1, \dots, X_p$ measured on n observations. no longer have Y!

We are not interested in prediction because we have no associated Y.

Goal: discovering interesting things about measurements $X_1, \dots, X_p$

- Is there an informative way to plot the data?
- Can we discover subgroups among variables or observations?

1 The Challenge of Unsupervised Learning

Supervised learning is a well-understood area.

You now have a good grasp of supervised learning.

If you were asked to predict a binary response -

Logistic regression, SVM, LDA, classification trees, RF, bagged trees, boosted trees,

and a clear understanding of how to assess your results -

validation on an independent test set

cross-validation error.

In contrast, unsupervised learning is often much more challenging.

More subjective, no simple goal for the analysis, e.g. prediction.

Unsupervised learning is often performed as part of an *exploratory data analysis*.

1st part of an analysis, before fitting any models.

It can be hard to assess the results obtained from unsupervised learning methods.

No universally accepted mechanism for performing cross-validation or validation on a test set.

Because there is no way to "check our work" with no response variable.

→ We don't know the true answer!

Techniques for unsupervised learning are of growing importance in a number of fields.

cancer research — assay gene expression levels in 100 patients and look for subgroups among cancer sample to better understand the disease.

online shopping — identify similar groups of shoppers show preferential items that they might be particularly interested.

My research — Entity resolution — Many noisy databases without unique identifying attributes → can we find the matches/links?

2 Principal Components Analysis

We have already seen principal components as a method for dimension reduction.

When faced w/ a large set of correlated variables, we can use principal components to summarize this set w/ a smaller number of representative variables that collectively explain most of the variability in the original data set.

Principal component directions = directions in feature space along which original data are highly variable.

define lines and subspaces that are close as possible to the data cloud.

Principal component regression

used principal components as predictors in a regression model instead of original variables.

Principal Components Analysis (PCA) refers to the process by which principal components are computed and the subsequent use of these components to understand the data.

Unsupervised approach

involves only features $X_1, \ldots, X_p$, no response Y.

Apart from producing derived variables for use in supervised learning, PCA also serves as a tool for data visualization.

Visualize observations or of variables.

as a tool for visualization / EDA.

2.1 What are Principal Components?

$X_1, \ldots, X_p$

Suppose we wish to visualize n observations with measurements on a set of p features as part of an exploratory data analysis.

We could do this by examining 2D plots of the data which contain n observations measured on 2 features.

ggpairs, pairs

$\Rightarrow \binom{p}{2} = \frac{p(p-1)}{2}$ plots. e.g. w/ $p = 10 \Rightarrow 45$ plots!

- Too many to look at
- likely no plot will be informative because they contain a small fraction of information present in our data.

For visualization in high dimensions

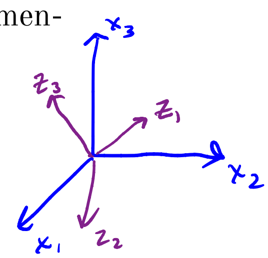
Goal: We would like to find a low-dimensional representation of the data that captures as much of the information as possible.

Then plot observations in low-dimensional space.

PCA provides us a tool to do just this.

It finds low dimensional representation of a data set that contains as much as possible of the variation (information).

Idea: Each of the n observations lives in p dimensional space, but not all of these dimensions are equally interesting.



PCA seeks a small number of dimensions that are as interesting as possible.

"interesting" = amount the observations vary along each dimension.

Each dimension found by PCA is a linear combination of the p features.

The *first principal component* of a set of features $X_1, \ldots, X_p$ is the normalized linear combination of the features

$$Z_1 = \phi_{11} X_1 + \phi_{21} X_2 + \cdots + \phi_{p1} X_p$$

normalized: $\sum_{j=1}^{p} \phi_{j1}^2 = 1$ $\leftarrow$ otherwise could result in arbitrarily large variances

$\phi_{11}, \ldots, \phi_{p1}$ are called "loadings" of the first principal component.

$\phi_1 = (\phi_{11}, \ldots, \phi_{p1})^T$ = "loading vector".

that has the largest variance. of Z_1

$$\text{Var}(\phi_{11} X_1 + \cdots + \phi_{p1} X_p).$$

Given a $n \times p$ data set $\boldsymbol{X}$, how do we compute the first principal component?

1. Assume each variable has been centered (i.e. columns have mean 0) — only care about maximizing variance.
(do it)

2. Look for linear combinations of the form

$$z_{i1} = \phi_{11} x_{i1} + \phi_{21} x_{i2} + \cdots + \phi_{p1} x_{ip}$$

w/ largest sample variance subject to $\sum_{j=1}^{p} \phi_{j1}^2 = 1$.

i.e. solve the following optimization problem:

$$\underset{\phi_{11}, \ldots, \phi_{p1}}{\text{maximize}} \left\{ \frac{1}{n} \sum_{i=1}^{n} \left(\sum_{j=1}^{p} \phi_{j1} x_{ij} \right)^2 \right\}$$

$$\text{subject to } \sum_{j=1}^{p} \phi_{j1}^2 = 1$$

can write this way because columns centered

$\Rightarrow \frac{1}{n} \sum_{i=1}^{n} x_{ij} = 0$

$\Rightarrow \frac{1}{n} \sum_{i=1}^{n} z_{i1} = 0$

$\Rightarrow$ this is sample variance of $z_{i1}, i = 1, \ldots, n$.

$z_{11}, \ldots, z_{n1}$ are called "scores" of the first principal component.

There is a nice geometric interpretation for the first principal component.

The loading vector ϕ_1 defines a direction in the feature space along which the data vary the most.

If we project n data points onto this direction we get the scores $z_{11}, \ldots, z_{n1}$

After the first principal component Z_1 of the features has been determined, we can find the second principal component, Z_2. The second principal component is the linear combination of $X_1, \ldots, X_p$ that has maximal variance out of all linear combinations that are uncorrelated with Z_1.

First principal component.

The second principal component score are

$$z_{i2} = \phi_{12} x_{i1} + \cdots + \phi_{p2} x_{ip}$$

$\phi_2 =$ second principal component loading vector.

Z_2 uncorrelated w/ Z_1

$\Longleftrightarrow$

ϕ_2 orthogonal to ϕ_1

$$\sum_{j=1}^{p} \phi_{j2} \cdot \phi_{j1} = 0$$

$p = 2$ in 2D space, there only one possibility for ϕ_2.

But w/ $p > 2$, there are multiple ϕ_2 orthogonal to ϕ_1

To find ϕ_2 (and Z_2), solve a similar optimization problem w/ additional constraint.

$$\underset{\phi_{21}, \ldots, \phi_{2p}}{\text{maximize}} \left\{ \frac{1}{n} \sum_{i=1}^{n} \left(\sum_{j=1}^{p} \phi_{j2} x_{ij} \right)^2 \right\} \text{ subject to } \sum_{j=1}^{p} \phi_{j2}^2 = 1 \text{ and } \sum_{j=1}^{p} \phi_{j2} \phi_{j1} = 0.$$

Once we have computed the principal components, we can plot them against each other to produce low-dimensional views of the data.

each of the 50 states, # arrests per 100,000 residents for each of 3 crimes.

```
str(USArrests)
```

```
## 'data.frame':    50 obs. of  4 variables:
## $ Murder  : num  13.2 10 8.1 8.8 9 7.9 3.3 5.9 15.4 17.4 ...
## $ Assault : int  236 263 294 190 276 204 110 238 335 211 ...
## $ UrbanPop: int  58 48 80 50 91 78 77 72 80 60 ...
## $ Rape    : num  21.2 44.5 31 19.5 40.6 38.7 11.1 15.8 31.9 25.8 ...
```

1 (Murder), 2 (Assault), 3 (Rape)

% pop in state living in urban area.

```
pca <- prcomp(USArrests, center = TRUE, scale = TRUE) # get loadings

summary(pca) # summary
```

PVE

Cumulative PVE

```
## Importance of components:
##                           PC1    PC2     PC3     PC4
## Standard deviation     1.5749 0.9949 0.59713 0.41645
## Proportion of Variance 0.6201 0.2474 0.08914 0.04336
## Cumulative Proportion  0.6201 0.8675 0.95664 1.00000
```

visualize 87% of variability retained.

```
pca$rotation # principal components loading matrix
```

→ missing 13% variability

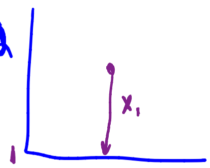
ϕ_1 ϕ_2 ϕ_3 ϕ_4

```
##                 PC1        PC2        PC3         PC4
## Murder   -0.5358995  0.4181809 -0.3412327  0.64922780
## Assault  -0.5831836  0.1879856 -0.2681484 -0.74340748
## UrbanPop -0.2781909 -0.8728062 -0.3780158  0.13387773
## Rape     -0.5434321 -0.1673186  0.8177779  0.08902432
```

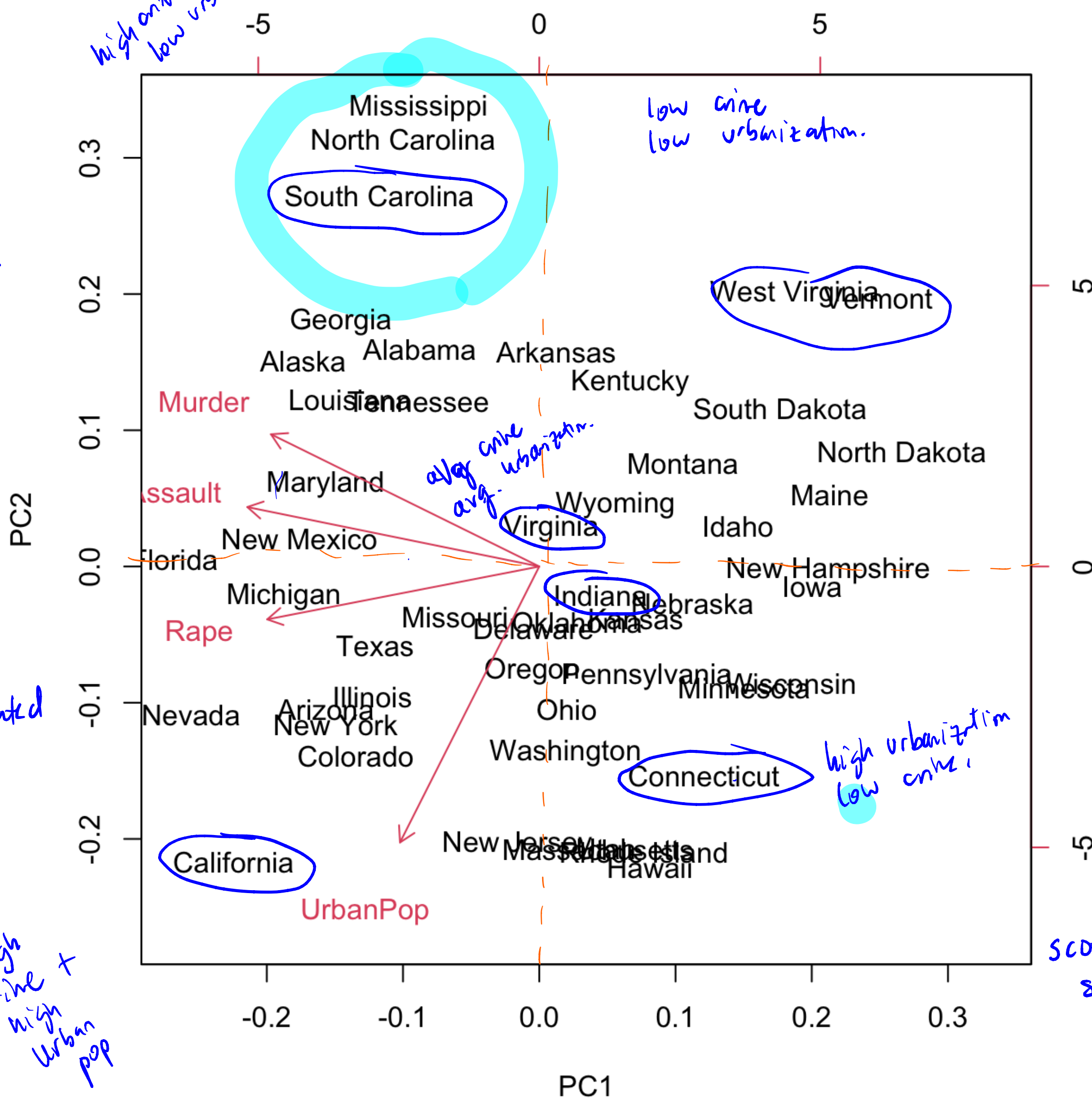
(-.58, 0.18)

```
## plot scores + directions
biplot(pca)
```

high crime
low urbanization.



PC2



PC1

Crime related variables close to each other, far away from urban pop.

$\Rightarrow$ crime related variables are correlated.

high crime + high urban pop

scores for 50 states.

First loading places approximately equal weight on murder, assault, rape.
less weight on Urban pop.

$\Rightarrow$ this first component $\approx$ measure of rate of serious crimes

Second loading places most weight on Urban population.

$\Rightarrow$ 2nd component $\approx$ level of urbanization of a state.

2.2 Scaling Variables

We've already talked about how when PCA is performed, the variables should be centered to have mean zero.

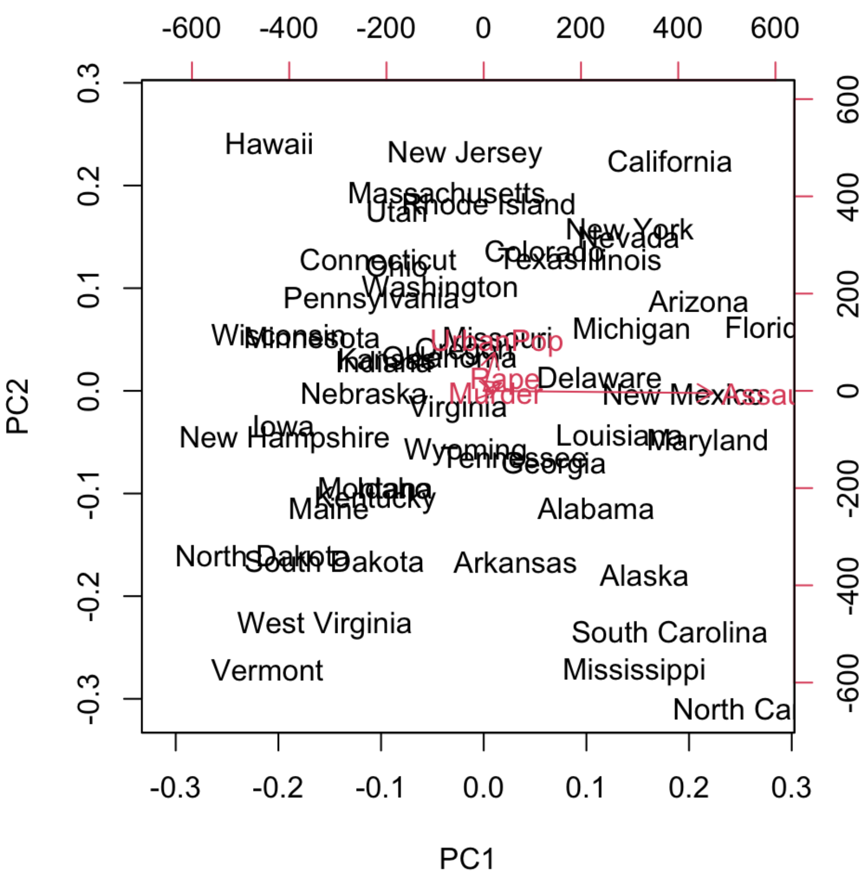
Also results depend on whether variables have been individually scaled.
(to have same sd).

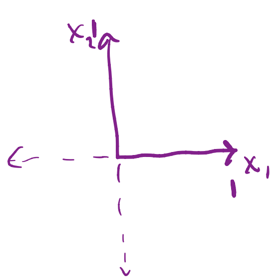
This is in contrast to other methods we've seen before.

e.g. linear regression when multiply by c, the corresponding coefficient is changed by a factor $1/c$.

e.g. LASSO is scale dependent
ridge regression.

same data as before, didn't scale.





Variables measured in different units
crime: #/100,000
UrbanPop: percentage.

← large loading on assault for 1^{st} PC.
(large variance b/c of scale).

Undesireable for PCA to depend on something as arbitrary as scale ⇒ scale each variable to have sd = 1.

UNLESS: all variables are measured on same units ⇒ might not want to scale them.

2.3 Uniqueness

Each principal component loading vector is unique, up to a sign flip.

$\Rightarrow$ software should result in same princ. comp. loading vectors, but sign might flip.

signs might differ because each prin. comp. loading specifies a direction in p-space.

line that extends in either direction.

Flipping sign has no effect since the directions.

Similarly, the score vectors are unique up to a sign flip.

$$Var(Z) = Var(-Z)$$

2.4 Proportion of Variance Explained

We have seen using the `USArrests` data that we can summarize 50 observations in 4 dimensions using just the first two principal component score vectors and the first two principal component vectors.

Question:

How much of the information in a given data set is lost by projecting observations onto first 2 princ. comp?

i.e, how much of variability is not contained in the first 2 prin. comp?

More generally, we are interested in knowing the *proportion of variance explained (PVE)* by each principal component.

Total variance: $\sum_{j=1}^{p} Var(X_j) = \sum_{j=1}^{p} \frac{1}{n} \sum_{i=1}^{n} x_{ij}^2$

variance explained by m^{th} Prin. Comp.: $\frac{1}{n} \sum_{i=1}^{n} z_{im}^2 = \frac{1}{n} \sum_{i=1}^{n} \left(\sum_{j=1}^{p} \phi_{jm} x_{ij} \right)^2$

$\Rightarrow$ PVE by m^{th} Prin. Comp: $\dfrac{\sum_{i=1}^{n} \left(\sum_{j=1}^{p} \phi_{jm} x_{ij} \right)^2}{\sum_{j=1}^{p} \sum_{i=1}^{n} x_{ij}^2}$ (positive quantity).

Cumulative PVE for 1st M components: sum PVE first M.

2.5 How Many Principal Components to Use

In general, a *ntimesp* matrix $\boldsymbol{X}$ has $\min(n-1, p)$ distinctt principal components.

$n \times p$

We are probably not interested in all of them

Rather, we would like to just use the first few principal components in order to visualize or interpret the data.

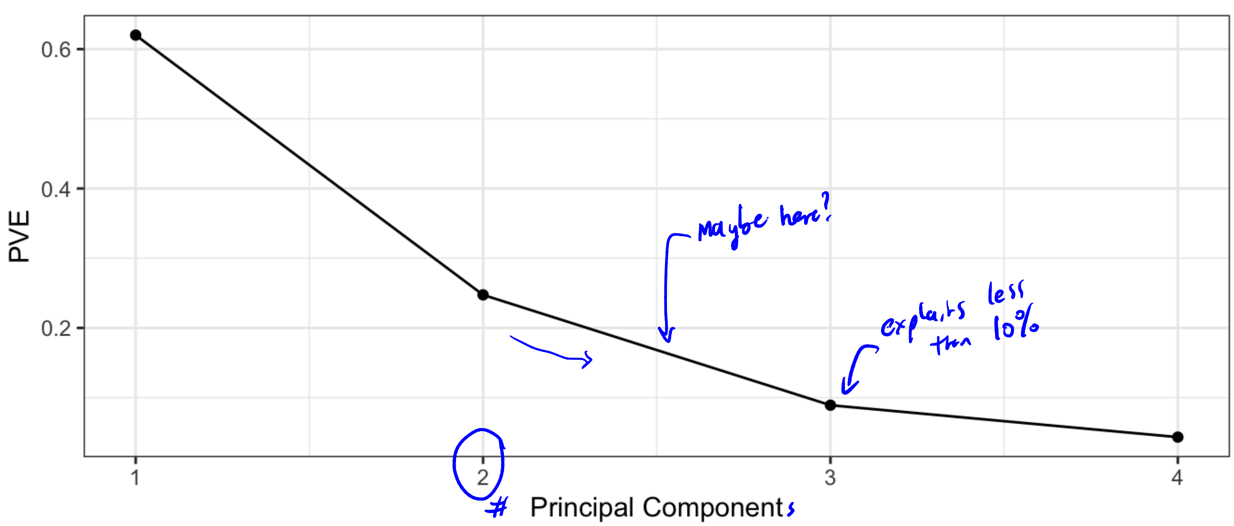
Want to use the smallest # required to get a good understanding of the data.

How many?

No one simple answer!

We typically decide on the number of principal components required by examining a *scree plot*.

look for a point that has an "elbow", where plot stops dropping so sharply.



this is ad hoc, because the question of how many is "enough" is not well defined.

Depends on problem, the data, and your goals.

Unsupervised EDA

Usually plot first two and look for "interesting patterns". If there are none, probably won't be so interesting in later components.

If first 2 are interesting, keep looking!

Supervised PCR

There is a good way to choose # of components: Cross validation.

2.6 Other Uses for Principal Components

We've seen previously that we can perform regression using the principal component score vectors as features for dimension reduction.

Many statistical techniques can be easily adapted to use the $n \times M$ matrix whose columns are the first $M << p$ principal components.

e.g. (other types of) regression, classification, clustering (next).

This can lead to *less noisy* results.

Since usually the signal is concentrated in its first few principal components.