Review: Exam II

Rebecca C. Steorts

Overview of the hashing/PCA

- Let's review hashing and principal components analysis for the second exam.

Locality sensitive hashing

Why does locality sensitive hashing help us with similarity search of documents (like songs)?

It provides a type of dimension reduction and also brings similar entities close together.

The type of dimension reduction we focused on is the minwise hash. The type of similarity we focused on is the Jaccard similarity.

Could these change? Yes! (Then we would be dealing with a different LSH family).

Terminology to know

- Shingle
- Jaccard similarity
- Hash function
- Characteristic matric
- Permutation
- Signature matrix
- Banding
- Minwise hash
- Connection between minwise hash and the Jaccard similarity.

Hashing

- Why did we introduce locality sensitive hashing?
- Recall we have some number of documents that we want to compare, and we want to avoid doing all-to-all document (entity) comparisons.

Locality sensitive hashing

- We avoid doing all-to-all document comparisons by filtering pairs of documents (entities) that are not similar.
- How do we do this? Let's start with the all-to-all comparison algorithm and then look at how we do the speed up.

Minwise hashing (or LSH)

1. Construct shingles of all documents in your corpus.
2. Hash all of your shingled documents.
3. Compute pairwise Jaccard similarity coefficients for all documents.
   (a) To do this in a computationally more efficient way, use the characteristic matrix and a random permutation.
   (b) Then create the signature matrix by using the minhash. Repeat this process using many random permutations in order to avoid collisions. This will increase the size of your signature matrix.

Why is this computationally intensive?

Speed up variant of minwise hashing (or LSH)

1. Construct shingles of all documents in your corpus.
2. Hash all of your shingled documents.
3. Compute pairwise Jaccard similarity coefficients for all documents.
   (a) To do this in a computationally more efficient way, use the characteristic matrix and a random permutation.
   (b) Then create the signature matrix by using the minhash. Repeat this process using many random permutations in order to avoid collisions. This will increase the size of your signature matrix.

To avoid performing all-to-all comparisons, compute the Jaccard similarity only for candidate pairs using b bands and r rows of the signature matrix, which provide a threshold $t = (1/b)^{1/r}$ using the steps above but now using these extra conditions of filtering out documents that are unlikely to be the same.

Application

- See the Beatles' example or your homework for a review in terms of a full running example.

Principal Componenents Analysis

PCA is just one type of unsupervised learning, where one tries to visualiaze and learning something from the data when we have observations but no response variable.

What is PCA

- PCA seeks to to find a low-dimensional representation of the data that captures as much of the information as possible.
- Each of the dimensions found by PCA is a linear combination of the p features.
- How many principle components do we have?

Mathematics behind PCA

- I don't expect you to be able to sovle the optimzation problem behind PCA since it's beyond the scope of this class.
- You can read more advanced details about PCA with the mathematical details in ESL.

PCA

- features (data points)
- loadings
- principal components
- scree plot
- biplot

PCA

- features are just the data points
- the first PC is the direction along which the data have the most variance.
- the second PC is the direction orthogonal to the first component with the most variance. Why is this true? For two reasons.
- Because it is orthogonal to the first eigenvector, their projections will be uncorrelated.
- Projections on to all the principal components are uncorrelated with each other.

Biplot

A biplot plots the data, along with the projections of the original variables, on to the first two components

Scree plot

- We can figure out the number of principal components by fitting what's called a scree plot.
- Choose the smallest number of principal components that are required such that an adequate amount of variability is explained.
- We look for the point at which the proportion of variance explained by each subsequent principal drops off.
- This is called the elbow of the scree plot.
- These plots are application specific and ad-hoc.

Practice Question on PCA

- Let's investigate a data set on PCA about cars and see what we find.

Cars data set

Let's read in the data

```
cars04 = read.csv("cars-fixed04.dat")
```

Summary information

```
head(cars04)
```

```
##                          Sports SUV Wagon Minivan Pickup AWD RWD Retail
## Acura 3.5 RL                  0   0     0       0      0   0   0  43755
## Acura 3.5 RL Navigation       0   0     0       0      0   0   0  46100
## Acura MDX                     0   1     0       0      0   1   0  36945
## Acura NSX S                   1   0     0       0      0   0   1  89765
## Acura RSX                     0   0     0       0      0   0   0  23820
## Acura TL                      0   0     0       0      0   0   0  33195
##                          Dealer Engine Cylinders Horsepower CityMPG
## Acura 3.5 RL              39014    3.5         6        225      18
## Acura 3.5 RL Navigation   41100    3.5         6        225      18
## Acura MDX                 33337    3.5         6        265      17
## Acura NSX S               79978    3.2         6        290      17
## Acura RSX                 21761    2.0         4        200      24
## Acura TL                  30299    3.2         6        270      20
##                          HighwayMPG Weight Wheelbase Length Width
## Acura 3.5 RL                     24   3880       115    197    72
## Acura 3.5 RL Navigation          24   3893       115    197    72
## Acura MDX                        23   4451       106    189    77
## Acura NSX S                      24   3153       100    174    71
## Acura RSX                        31   2778       101    172    68
## Acura TL                         28   3575       108    186    72
```

```
summary(cars04)
```

```
##      Sports            SUV              Wagon             Minivan
##  Min.   :0.0000   Min.   :0.0000   Min.   :0.00000   Min.   :0.00000
##  1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.00000   1st Qu.:0.00000
##  Median :0.0000   Median :0.0000   Median :0.00000   Median :0.00000
##  Mean   :0.1163   Mean   :0.1525   Mean   :0.07235   Mean   :0.05426
##  3rd Qu.:0.0000   3rd Qu.:0.0000   3rd Qu.:0.00000   3rd Qu.:0.00000
##  Max.   :1.0000   Max.   :1.0000   Max.   :1.00000   Max.   :1.00000
##      Pickup        AWD              RWD              Retail
##  Min.   :0   Min.   :0.0000   Min.   :0.0000   Min.   : 10280
##  1st Qu.:0   1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.: 20997
##  Median :0   Median :0.0000   Median :0.0000   Median : 28495
##  Mean   :0   Mean   :0.2016   Mean   :0.2429   Mean   : 33231
##  3rd Qu.:0   3rd Qu.:0.0000   3rd Qu.:0.0000   3rd Qu.: 39552
##  Max.   :0   Max.   :1.0000   Max.   :1.0000   Max.   :192465
##      Dealer          Engine        Cylinders        Horsepower
##  Min.   :  9875   Min.   :1.400   Min.   : 3.000   Min.   : 73.0
##  1st Qu.: 19575   1st Qu.:2.300   1st Qu.: 4.000   1st Qu.:165.0
##  Median : 26155   Median :3.000   Median : 6.000   Median :210.0
##  Mean   : 30441   Mean   :3.127   Mean   : 5.757   Mean   :214.4
```

```
##  3rd Qu.: 36124   3rd Qu.:3.800   3rd Qu.: 6.000   3rd Qu.:250.0
##  Max.   :173560   Max.   :6.000   Max.   :12.000   Max.   :493.0
##     CityMPG        HighwayMPG        Weight       Wheelbase
##  Min.   :10.00   Min.   :12.00   Min.   :1850   Min.   : 89.0
##  1st Qu.:18.00   1st Qu.:24.00   1st Qu.:3107   1st Qu.:103.0
##  Median :19.00   Median :27.00   Median :3469   Median :107.0
##  Mean   :20.31   Mean   :27.26   Mean   :3532   Mean   :107.2
##  3rd Qu.:21.50   3rd Qu.:30.00   3rd Qu.:3922   3rd Qu.:112.0
##  Max.   :60.00   Max.   :66.00   Max.   :6400   Max.   :130.0
##      Length        Width
##  Min.   :143   Min.   :64.00
##  1st Qu.:177   1st Qu.:69.00
##  Median :186   Median :71.00
##  Mean   :185   Mean   :71.28
##  3rd Qu.:193   3rd Qu.:73.00
##  Max.   :221   Max.   :81.00
```

Scale versus not-scale

```
cars04.pca = prcomp(cars04[,8:18], scale.=TRUE)
cars04.pca2 = prcomp(cars04[,8:18], scale.=FALSE)
```

What's the difference in these two commands? Which command should we use? How would you verify this in practice?

Recall that TRUE normalizes the features to be on the same scale. This will be application specific, so it depends on what type of data you are working with. Note: many times an un-normalized version of a PCA can be very strange looking and this is because it treats the features as being un-normalized.

Principle components

```
round(cars04.pca$rotation[,1:2],2)
```

```
##              PC1   PC2
## Retail     -0.26 -0.47
## Dealer     -0.26 -0.47
## Engine     -0.35  0.02
## Cylinders  -0.33 -0.08
## Horsepower -0.32 -0.29
## CityMPG     0.31  0.00
## HighwayMPG  0.31  0.01
## Weight     -0.34  0.17
## Wheelbase  -0.27  0.42
## Length     -0.26  0.41
## Width      -0.30  0.31
```

What do we observe?

- All the variables except the gas-mileages have a negative projection on to the first PC.
- There is a negative correlation between mileage and everything else.

The first and second PC's

- The first PC tells us if we are getting a big, expensive gas-guzzling car with a powerful engine, OR whether we are getting a small, cheap, fuel-efficient car with a wimpy engine.

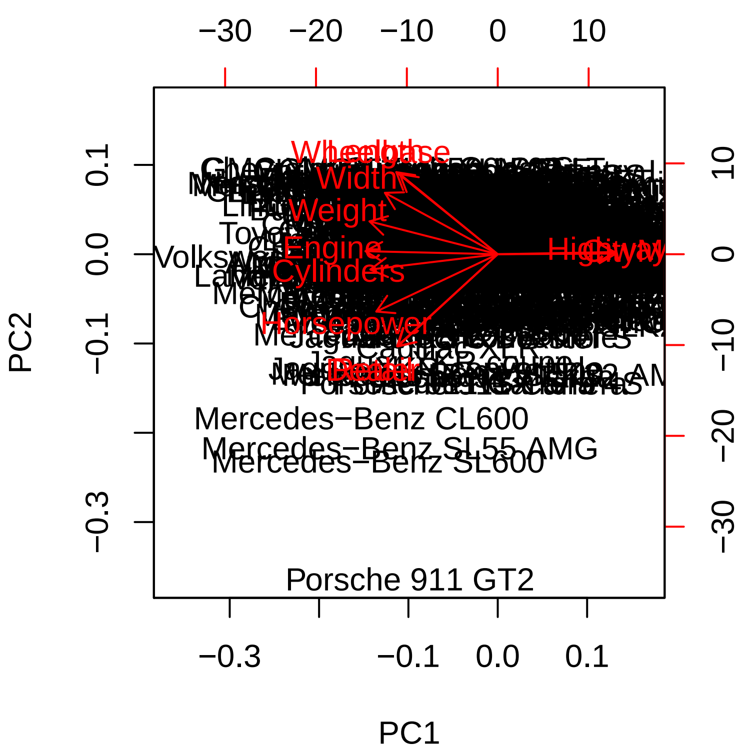
The second PC is a bit more interesting. It tell us:

- Engine size and gas mileage hardly project on to it at all.
- Contrast between the physical size of the car (positive projection) and the price and horsepower.
- This axis separates mini-vans, trucks and SUVs (big, not so expensive, not so much horse-power) from sports-cars (small, expensive, lots of horse-power).

How could we check this interpretation?

Biplot

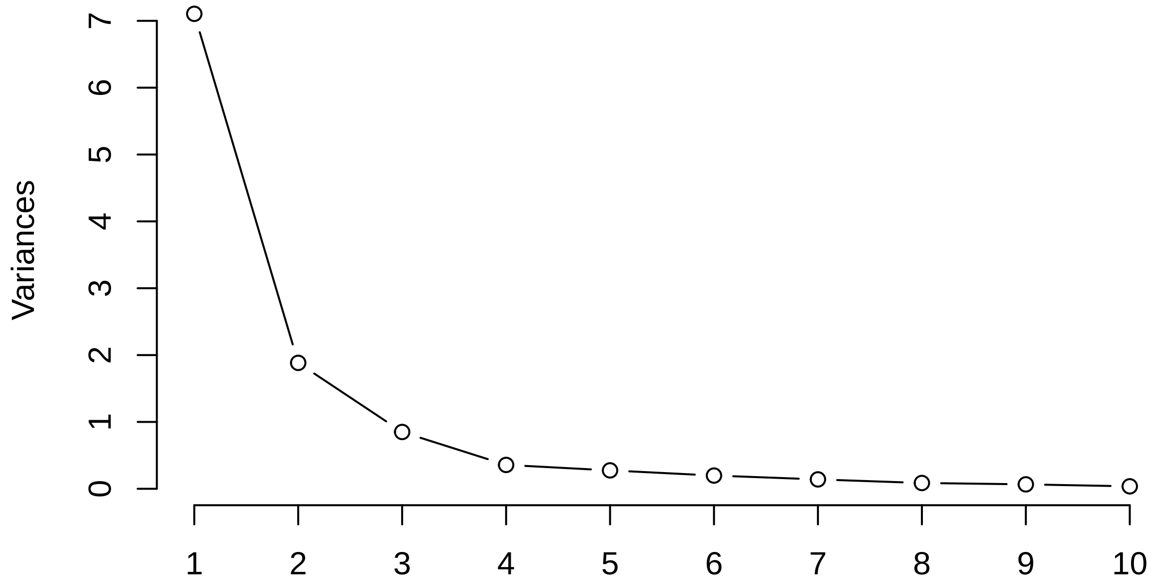
```
biplot(cars04.pca)
```



We see that the lowest value of the second component is a Porsche 911. The highest values of the first component happen to be hybrids.

Scree plot

```
plot(cars04.pca,type="l",main="")
```



What is the optimal number of principal components based on the spree plot?