title: “Portfolio Component: Searching for Similarity”

author: ” Huy Nguyen”

output: html_notebook

Portfolio Component: Searching for Similarity Dimensionality Reduction

Created by Huy Nguyen on March 25, 2023

Here I import libraries and read in our wine dataset

```
library(caret)
```

```
## Warning: package 'caret' was built under R version 4.2.3
```

```
## Loading required package: ggplot2
```

```
## Warning: package 'ggplot2' was built under R version 4.2.3
```

```
## Loading required package: lattice
```

```
library(ggplot2)

wineData <- read.csv("winemag-data-130k-v2.csv", na.strings = "NA", header = TRUE)
data(wineData)
```

```
## Warning in data(wineData): data set 'wineData' not found
```

```
attach(wineData)
```

Dividing Data into train/test sets

```
i <- sample(1:10000, 10000*0.80, replace=FALSE)
train <- wineData[i,]
test <- wineData[-i,]
set.seed(1234)
```

create regression model without PCA

```
lmPrice <- lm(price~points, data = train)
summary(lmPrice)
```

```
##
## Call:
## lm(formula = price ~ points, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
##  -52.41  -15.23   -5.18    7.81 1806.54
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -497.0444    13.6305  -36.47   <2e-16 ***
## points         6.0255     0.1542   39.09   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 40.42 on 7445 degrees of freedom
##   (553 observations deleted due to missingness)
## Multiple R-squared:  0.1703, Adjusted R-squared:  0.1702
## F-statistic:  1528 on 1 and 7445 DF,  p-value: < 2.2e-16
```

```
par(mfrow = c(2,2))
```

Apply PCA dimensionality reduction

```
pca_out <- preProcess(train[,1:4], method=c("center", "scale", "pca"))
```

```
## Warning in pre_process_options(method, column_types): PCA is a group
## transformation and only a single predictor is listed. This method is
## eliminated.
```

```
pca_out
```

```
## Created from 8000 samples and 4 variables
##
## Pre-processing:
##   - centered (1)
##   - ignored (3)
##   - scaled (1)
```

create regression model with the PCA data

```
lmPricePCA <- lm(price~points, data = train)
summary(lmPricePCA)
```

```
##
## Call:
## lm(formula = price ~ points, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
##  -52.41  -15.23   -5.18    7.81 1806.54
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -497.0444    13.6305  -36.47   <2e-16 ***
## points         6.0255     0.1542   39.09   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 40.42 on 7445 degrees of freedom
##   (553 observations deleted due to missingness)
## Multiple R-squared:  0.1703, Adjusted R-squared:  0.1702
## F-statistic:  1528 on 1 and 7445 DF,  p-value: < 2.2e-16
```

```
par(mfrow = c(2,2))
```

Test both models to compare results

```
pred1 <- predict(lmPrice, newdata = test)
pred1 <- exp(pred1)
cor1 <- cor(pred1, test$points)
mse1 <- mean((pred1-test$points)^2)
rmse1 <- sqrt(mse1)

pred2 <- predict(lmPricePCA, newdata = test)
pred2 <- exp(pred2)
cor2 <- cor(pred2, test$points)
mse2 <- mean((pred2-test$points)^2)
rmse2 <- sqrt(mse2)

print(paste("PCA correlation: ", cor1))
```

```
## [1] "PCA correlation:  0.0450229221490798"
```

```
print(paste("correlation: ", cor2))
```

```
## [1] "correlation:  0.0450229221490798"
```

```
print(paste("PCA mse: ", mse1))
```

```
## [1] "PCA mse:  6.13195326257468e+87"
```

```
print(paste("mse: ", mse2))
```

```
## [1] "mse:  6.13195326257468e+87"
```

```
print(paste("PCA rse: ", rmse1))
```

```
## [1] "PCA rse:  7.83067893772608e+43"
```

```
print(paste("rse: ", rmse2))
```

```
## [1] "rse:  7.83067893772608e+43"
```

PCA is a group transformation and only a single predictor is listed. This caused PCA to not be applied to this data and provides us with the same data with no modifications.

Attempting to perform LDA dimensionality reduction

```
library(MASS)

wineData <- subset(wineData, select = c(X, points, price))
#lda1 <- lda(price~., data=train)
```

I could not run the commented line to perform LDA because of the dataset I would assume that this kind of data set is not meant for LDA reduction

Calculations to test the model after LDA has been performed if was

```
#pred3 <- predict(lda1, newdata = test)
#pred3 <- exp(pred3)
#cor3 <- cor(pred3, test$points)
#mse3 <- mean((pred3-test$points)^2)
#rmse3 <- sqrt(mse3)

#print(paste("correlation: ", cor3))
#print(paste("mse: ", mse3))
#print(paste("rse: ", rmse3))
```

Because of this dataset I was not able to perform PCA or LDA therefore I cannot calculate the loss of accuracy from both reductions.