DAR F23 Project Status Notebook - Assignment 06

Hockey Analytics

Caleb Smith

2023-11-17

Contents

Weekly Work Summary

- RCS ID: smithc22
- Project Name: Hockey Analytics
- Summary of work since last week

  I analyzed Jeff's discretezation of clusters, using UMAP + Kmeans as well as PCA to see if the categorization of everything would fix PCA's issue of not treating continuous and categorical variables evenly.

  I also added a slider to allow the user to subset the data based on features in Play_Plotter. I'll discuss at the next meeting whether to allow the user to slice data on multiple features and how to implement it in a manner that would make since in the integration

  Finally, I attended the CommD meeting with Ashley about colors to get some advice for the color scheme. The new color scheme for clusters has been applied. They also recommend we switch the rink plot to grayscale.

- Summary of github issues added and worked

  n/a

- Summary of github commits

  Branch Name: dar-smithc22 Files pushed: https://github.rpi.edu/DataINCITE/Hockey_Fall_2023/blob/dar-smithc22/ShinyApps/Play_Plotter/app.R

  https://github.rpi.edu/DataINCITE/Hockey_Fall_2023/blob/dar-smithc22/StudentNotebooks/Assignment06/smithc22_assignment06.Rmd

  https://github.rpi.edu/DataINCITE/Hockey_Fall_2023/raw/dar-smithc22/StudentNotebooks/Assignment06/smithc22_assignment06.pdf

  App Links:

  https://lp01.idea.rpi.edu/shiny/smithc22/Play_Plotter/

- List of presentations, papers, or other outputs
    - N/A
- List of references (if necessary)

N/A

- Indicate any use of group shared code base

I used some of Lieben's code from his app to help with the sliders in my app

- Indicate which parts of your described work were done by you or as part of joint efforts

Most of the work not already attributed was done by me, choosing colors was a joint effort between me, Ashley, and Jessica from CommD

Personal Contribution

- Clearly defined, unique contribution(s) done by you: code, ideas, writing…

The code in this notebook is my own. I fixed some bugs in my app relating to cluster colors and also added a method of filtering out features based on a range, although I got the code for the UI from Lieben.

- Include github issues you've addressed

N/A

Analysis: Question 1: Does categorizing continuous variables change the clustering?

Question being asked

Provide in natural language a statement of what question you're trying to answer

How does categorizing all the data change the clustering performed by UMAP?

Data Preparation

Provide in natural language a description of the data you are using for this analysis

The categorical data from Jeff, where each continuous feature is put into tertiles.

Include a step-by-step description of how you prepare your data for analysis

1. Read in Jeff's data
2. Select only the categorized parts
3. Set things up so UMAP works properly *If you're re-using dataframes prepared in another section, simply re-state what data you're using*

```r
# Include all data processing code (if necessary), clearly commented
library(data.table)
library(mltools)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:data.table':
##
##     between, first, last
## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
library(plotly)
```

```
## Loading required package: ggplot2

##
## Attaching package: 'plotly'

## The following object is masked from 'package:ggplot2':
##
##     last_plot

## The following object is masked from 'package:stats':
##
##     filter

## The following object is masked from 'package:graphics':
##
##     layout
```

```r
library(umap)
library(ggplot2)
library(scatterplot3d)
library(rgl)
```

```
## Warning in rgl.init(initValue, onlyNULL): RGL: unable to open X11 display

## Warning: 'rgl.init' failed, running with 'rgl.useNULL = TRUE'.
```

```r
library(car)
```

```
## Loading required package: carData

##
## Attaching package: 'car'

## The following object is masked from 'package:dplyr':
##
##     recode
```

```r
library(grid)
library(gridExtra)
```

```
##
## Attaching package: 'gridExtra'

## The following object is masked from 'package:dplyr':
##
##     combine
```

```r
library(kableExtra)
```

```
##
## Attaching package: 'kableExtra'
## The following object is masked from 'package:dplyr':
##
##     group_rows
```

```
library(heatmaply)
```

```
## Loading required package: viridis
```

```
## Loading required package: viridisLite
```

```
##
## ======================
## Welcome to heatmaply version 1.4.2
##
## Type citation('heatmaply') for how to cite the package.
## Type ?heatmaply for the main documentation.
##
## The github page is: https://github.com/talgalili/heatmaply/
## Please submit your suggestions and bug-reports at: https://github.com/talgalili/heatmaply/issues
## You may ask questions at stackoverflow, use the r and heatmaply tags:
##   https://stackoverflow.com/questions/tagged/heatmaply
## ======================
```

```
library(gplots)
```

```
##
## Attaching package: 'gplots'
```

```
## The following object is masked from 'package:stats':
##
##     lowess
```

```
library(tibble)
library(ggbiplot)
```

```
## Loading required package: plyr
```

```
## ------------------------------------------------------------------------------
```

```
## You have loaded plyr after dplyr - this is likely to cause problems.
## If you need functions from both plyr and dplyr, please load plyr first, then dplyr:
## library(plyr); library(dplyr)
```

```
## ------------------------------------------------------------------------------
```

```
##
## Attaching package: 'plyr'
```

```
## The following objects are masked from 'package:plotly':
##
##     arrange, mutate, rename, summarise
```

```
## The following objects are masked from 'package:dplyr':
##
##     arrange, count, desc, failwith, id, mutate, rename, summarise,
##     summarize
```

```
## Loading required package: scales
```

```
##
## Attaching package: 'scales'
## The following object is masked from 'package:viridis':
##
##     viridis_pal
```

```
library(stats)
setwd("~/Hockey_Fall_2023/StudentNotebooks/Assignment06")
shots_stats.df <- readRDS("~/Hockey_Fall_2023/StudentData/categorized_shots_stats_goal.df.Rds")
justCat <- cbind.data.frame(shots_stats.df[,16:23], shots_stats.df[,8:10])
str(justCat)
```

```
## 'data.frame':    210 obs. of  11 variables:
##  $ puckSpeedCategory     : num  2 1 1 1 2 2 2 2 1 2 ...
##  $ puckAngleCategory     : num  0 1 0 2 0 1 0 1 0 1 ...
##  $ puckDistCategory      : num  2 1 1 1 0 2 0 1 2 2 ...
##  $ posTimeCategory       : num  0 0 0 2 2 1 2 2 1 0 ...
##  $ goalieDist_qCategory  : num  2 0 1 0 2 1 1 1 1 0 ...
##  $ shooterSpeed_qCategory: num  1 0 1 1 2 0 2 2 0 0 ...
##  $ goalieAngleCategory   : num  0 1 0 2 1 1 0 1 1 1 ...
##  $ defDistCategory       : num  2 0 2 0 1 1 1 2 1 1 ...
##  $ NumOffense            : int  1 1 0 0 1 2 0 2 2 1 ...
##  $ NumDefense            : int  2 2 2 2 2 3 1 2 4 2 ...
##  $ rightHanded           : num  0 0 0 0 1 0 0 0 1 1 ...
```

```
custom.config <- umap.defaults
custom.config$random_state <- 11102023
set.seed(100)

select <- dplyr::select

wssplot <- function(data, nc=15, seed=100){
  wss <- data.frame(cluster=1:nc, quality=c(0))
  for (i in 1:nc){
    set.seed(seed)
    wss[i,2] <- kmeans(data, centers=i)$tot.withinss}
  ggplot(data=wss,aes(x=cluster,y=quality)) +
    geom_line() +
    ggtitle("Quality of k-means by Cluster")
}
```

Analysis: Methods and results

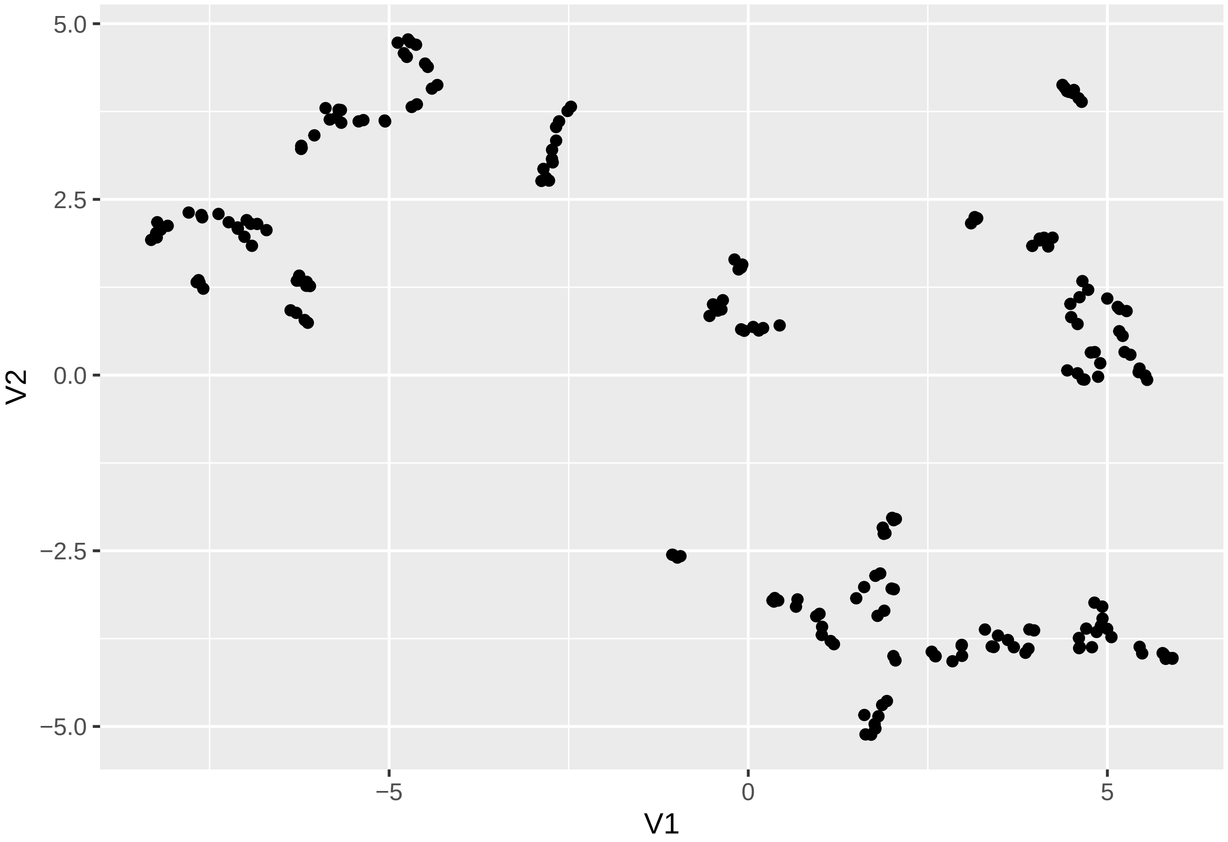
Describe in natural language a statement of the analysis you're trying to do

Trying to run UMAP and then Kmeans clustering on Jeff's data

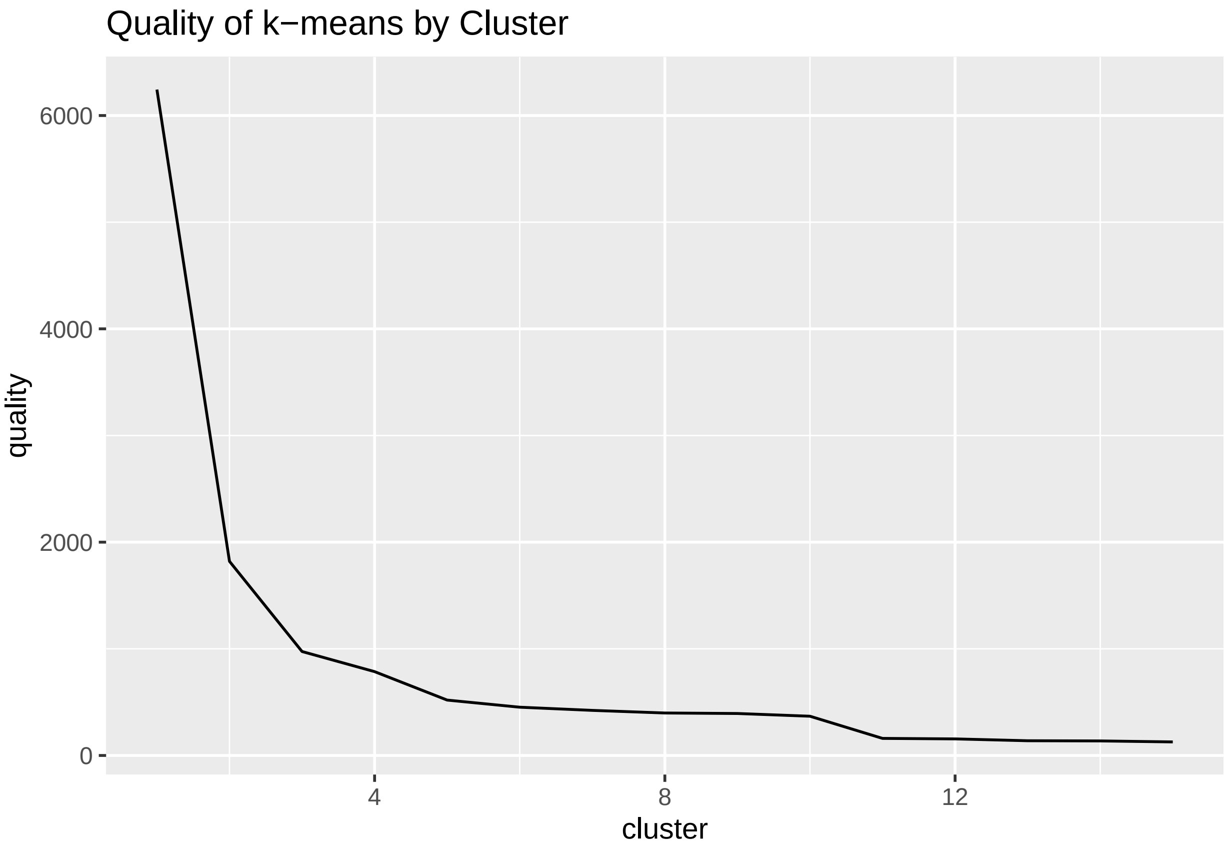
Provide clearly commented analysis code; include code for tables and figures!

```
# Include all analysis code, clearly commented
# If not possible, screen shots are acceptable.
# If your contributions included things that are not done in an R-notebook,
#   (e.g. researching, writing, and coding in Python), you still need to do
#   this status notebook in R.  Describe what you did here and put any products
#   that you created in github. If you are writing online documents (e.g. overleaf
#   or google docs), you can include links to the documents in this notebook
#   instead of actual text.
umapCat <- umap(justCat,n_components = 2, config = custom.config)
```

```
ggplot()+
 geom_point(data = as.data.frame(umapCat$layout),aes(x = V1,y = V2))
```

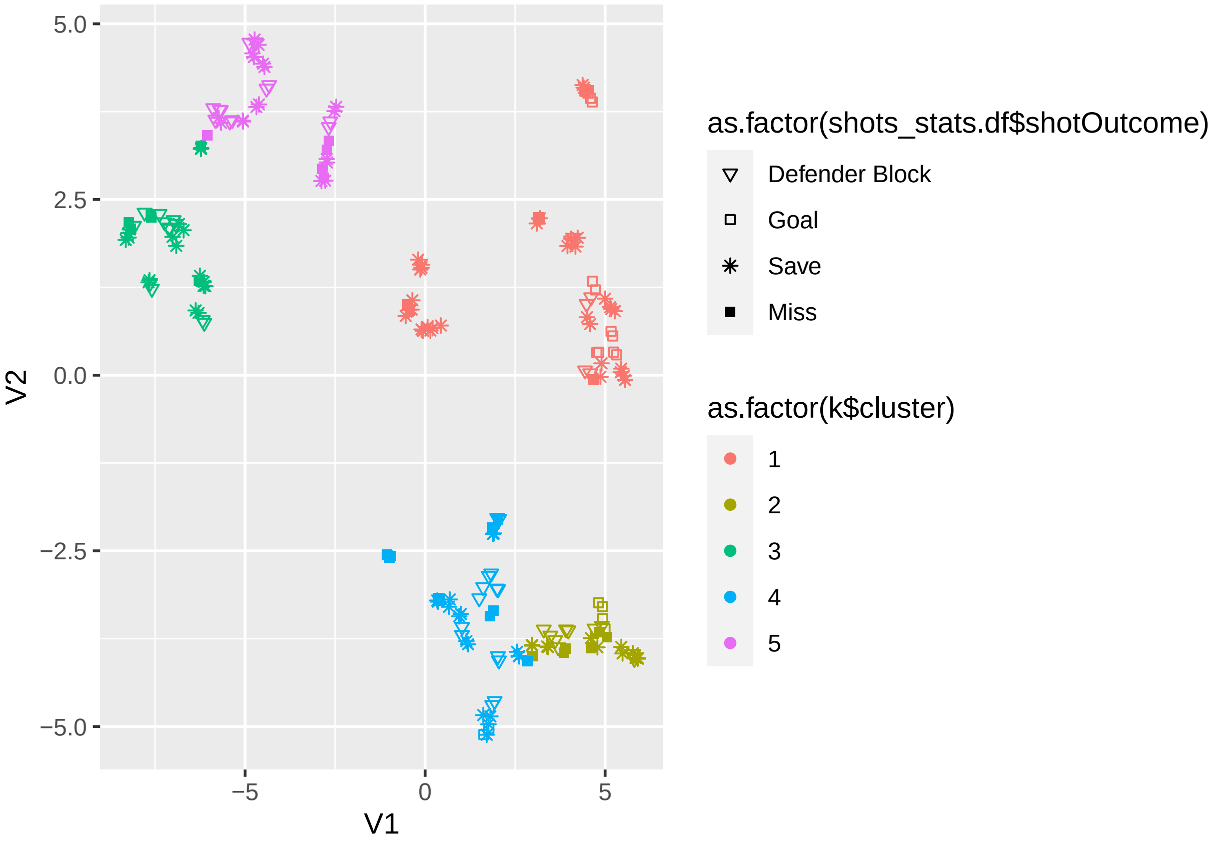


```
wssplot(umapCat$layout)
```



Kind of looks like 5 here, which is consistent with previous results. I'll try the clustering and see if it looks alright, and also generate a summary table to compare the clusters.

```
k <- kmeans(umapCat$layout,5) #4 also looks pretty reasonable

ggplot()+
  geom_point(data = as.data.frame(umapCat$layout),aes(x = V1,y = V2,shape = as.factor(shots_stats.df$sho
  scale_shape_manual(values = c(25,22,8,15,12))
```



```
#summary table
tabData <- cbind.data.frame(shots_stats.df[,1:10],k$cluster)
tabData <- cbind.data.frame(tabData,shots_stats.df[,12:13])
head(tabData)
```

```
##   puckDist puckAngle puckSpeed shooterSpeed goalieDist goalieAngle posTime
## 1 688.5900  56.32323  35.60191    12.741100   79.35496    40.29081       2
## 2 370.3595  75.22510  26.85647    10.126463   30.50606    62.68679      15
## 3 379.3229  47.26643  52.76842    18.196121   67.94116    32.18629       1
## 4 408.4321 152.88808  51.62253    13.477262   42.66974   146.41294      39
## 5 278.5743  59.24320  50.14920    20.367026   86.24243    54.11996      60
## 6 606.1261  64.86992  45.44330     5.240912   73.35530    62.38697      37
##   NumOffense NumDefense rightHanded k$cluster   defDist  defAngle
## 1          1          2           0         3 554.55208 140.29522
## 2          1          2           0         1  77.14952 129.70627
## 3          0          2           0         1 208.92192 125.24775
## 4          0          2           0         2  77.19322  59.64689
## 5          1          2           1         5 129.34843 139.81344
## 6          2          3           0         3 141.06063 169.62383
```

```
tab <- tabData %>% group_by(`k$cluster`) %>% summarise_all(.funs = mean)
kable(tab[,1:9])
```

k$cluster	puckDist	puckAngle	puckSpeed	shooterSpeed	goalieDist	goalieAngle	posTime	NumOffense
1	207.4834	73.65368	39.20384	15.39178	46.70010	56.00655	19.36667	0.1333333
2	405.4547	116.65869	45.44805	22.05085	82.84534	116.10651	34.84848	0.2424242
3	580.3942	68.41547	37.87735	11.50763	81.53488	61.28986	21.78378	1.1081081
4	479.0101	117.39438	44.46145	13.56194	65.22730	114.58703	24.08889	0.6666667
5	455.3005	54.40157	42.36026	18.35611	86.30226	46.35483	54.74286	0.6571429

```
kable(tab[,10:ncol(tab)])
```

NumDefense	rightHanded	defDist	defAngle
1.333333	0.2333333	141.9952	115.55221
2.212121	0.8787879	112.7941	80.56646
3.108108	0.4594595	194.8992	137.05835
2.022222	0.4666667	288.3022	29.70404
2.085714	0.4857143	198.2589	135.21538

3 is traffic jam, 2 is panic shots, 1 is perfect shots, not sure about 4 and 5. Categorical data seems to focus on angle a lot more. Defender block as a cluster was there because of the defender angle and the shooter angle lining up, so I think the categorizing of all the angles prevented it from forming, as everything became left, center, and right. While defenders being directly in front or 60 degrees to the side are very different for a shooter, both of these scenarios can be placed similarly in the categorized data.

For the UMAP graph: Very interesting, seems the goals all have the same (very high) V1 projection, resulting in them being clustered nicely. I'll take a look at a PCA to see if this is captured linearly as well, as having a more explanable dimensionality reduction would be nice to have

Discussion of results

Provide in natural language a clear discussion of your observations.

3 of the clusters lined up with the clusterings of the continuous, while wide open and defender block were no where to be seen. I explained defender block's absence above, but wide open is probably mixed in with perfect shot for a similar reason to defender block not being present, except this time it is the angle between the goalie and the shooter not being captured instead of the shooter-defender angle.

Analysis: Question 2: Can relationships between these categorical variables be linearly described?

Question being asked

Provide in natural language a statement of what question you're trying to answer

Can we also see such a clean division with a more explainable dimensionality reduction method?

Data Preparation

Provide in natural language a description of the data you are using for this analysis

Include a step-by-step description of how you prepare your data for analysis

If you're re-using dataframes prepared in another section, simply re-state what data you're using

Just reusing the dataframe of categorical variables

```
# Include all data processing code (if necessary), clearly commented
```

Analysis: Methods and Results

Describe in natural language a statement of the analysis you're trying to do

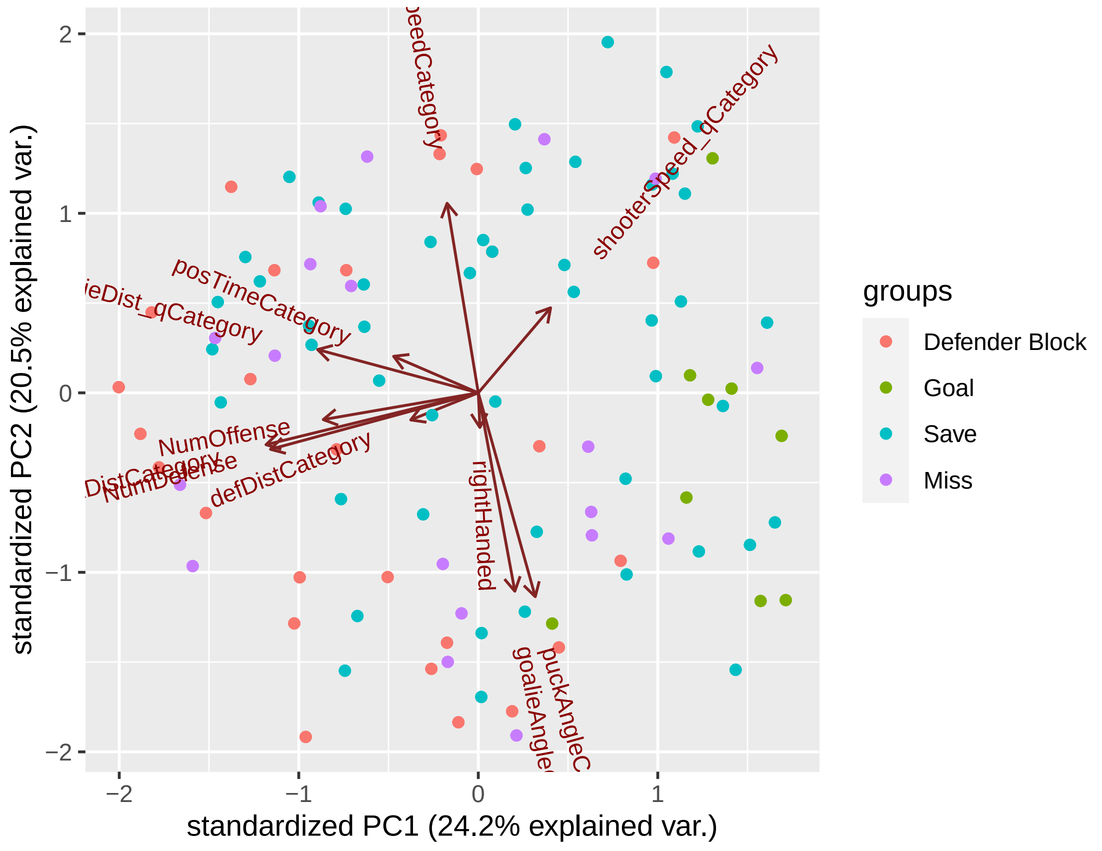
Run a PCA and see how it compares with UMAP

Provide clearly commented analysis code; include code for tables and figures!

```
# Include all analysis code, clearly commented
# If not possible, screen shots are acceptable.
# If your contributions included things that are not done in an R-notebook,
#   (e.g. researching, writing, and coding in Python), you still need to do
#   this status notebook in R.  Describe what you did here and put any products
#   that you created in github. If you are writing online documents (e.g. overleaf
#   or google docs), you can include links to the documents in this notebook
#   instead of actual text.

catPCA <- prcomp(justCat)
summary(catPCA)
```

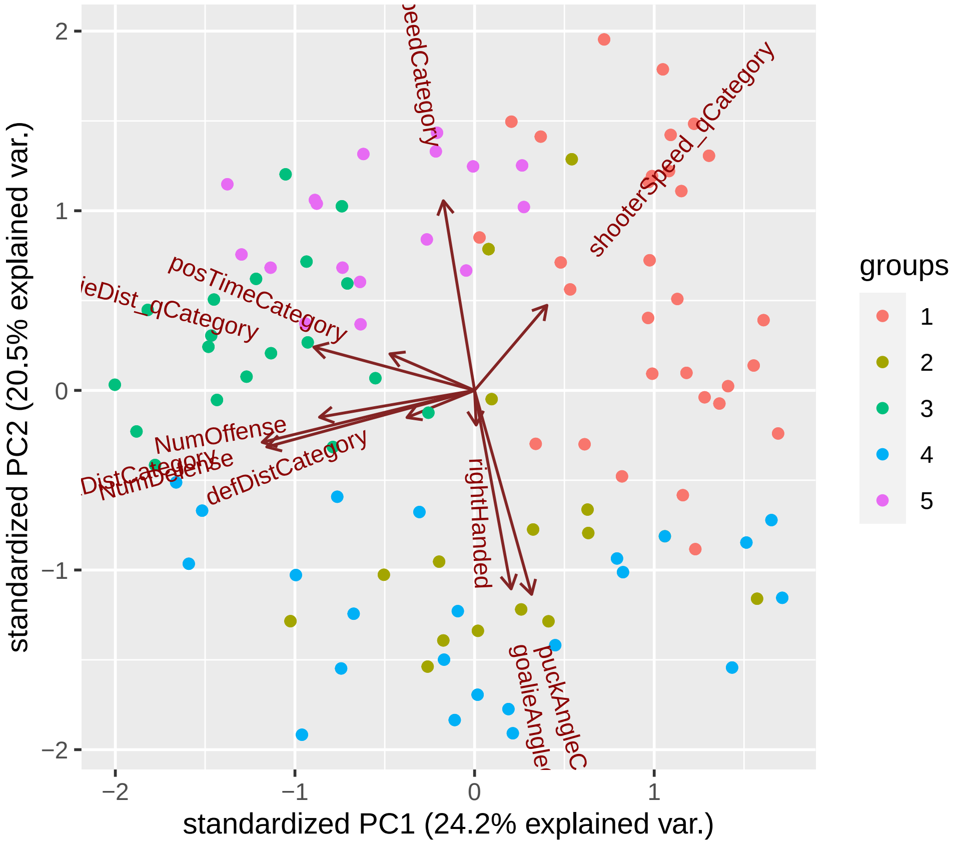
```
## Importance of components:
##                           PC1    PC2    PC3    PC4    PC5     PC6    PC7
## Standard deviation     1.2925 1.1886 0.9892 0.8718 0.7126 0.61771 0.5851
## Proportion of Variance 0.2425 0.2051 0.1420 0.1103 0.0737 0.05538 0.0497
## Cumulative Proportion  0.2425 0.4476 0.5896 0.6999 0.7736 0.82898 0.8787
##                            PC8     PC9    PC10    PC11
## Standard deviation     0.52525 0.48016 0.43388 0.37569
## Proportion of Variance 0.04005 0.03347 0.02733 0.02049
## Cumulative Proportion  0.91872 0.95219 0.97951 1.00000
```

```
ggbiplot(catPCA,groups = as.factor(shots_stats.df$shotOutcome))
```

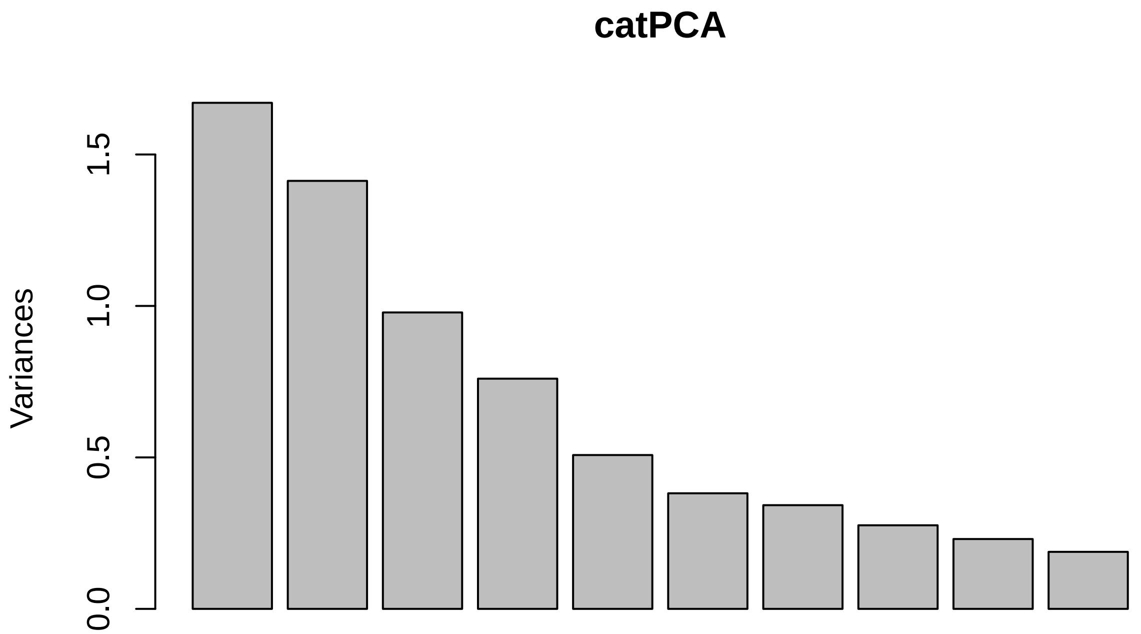


Goals lining up at a particular value isn't as pronounced as in UMAP, although it is still there. I'll do another graph to show the UMAP clusters instead of the outcomes, as well as a screeplot

```
ggbiplot(catPCA,groups = as.factor(k$cluster))
```

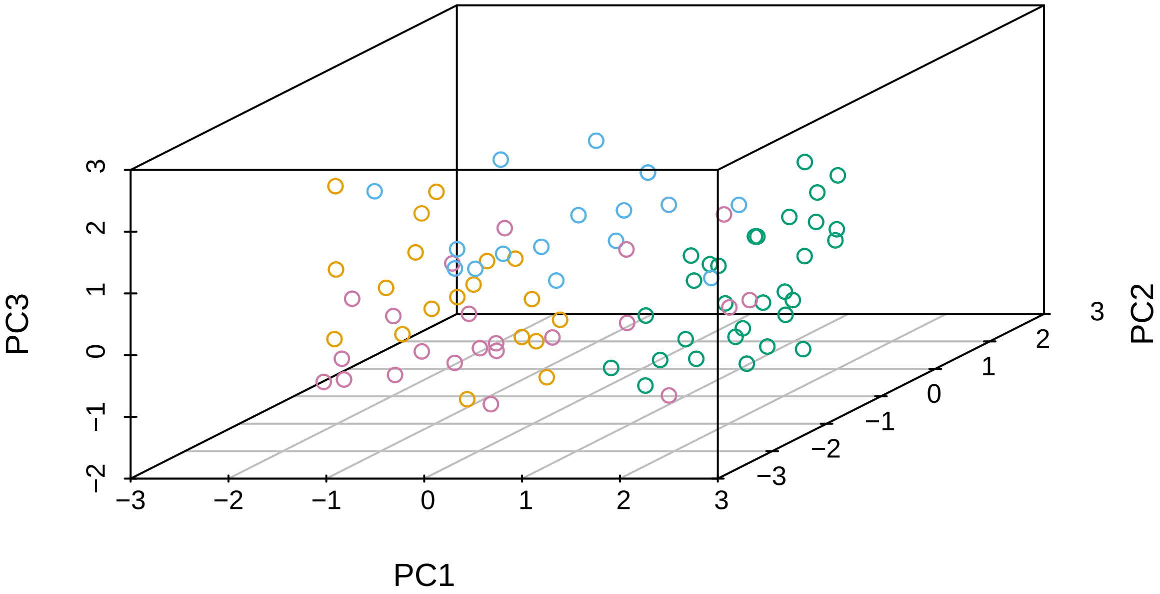


```
screeplot(catPCA)
```

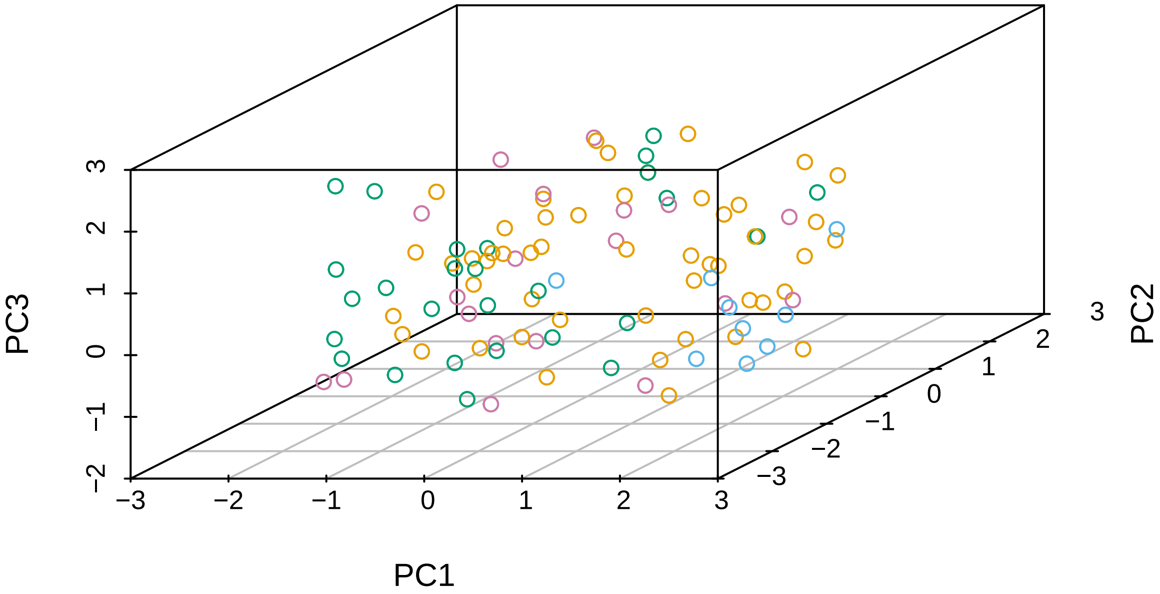


```
colorPalette <- c( rgb(0,0.62,0.45),rgb(86/255,180/255,233/255),
                   rgb(230/255,159/255,0/255),rgb(204/255,121/255,167/255))
```

```
colors <- colorPalette[k$cluster]

scatterplot3d(catPCA$x[,1:3],color = colors)
```



```
#Clusters don't nicely align

#s3d <- scatterplot3d(catPCA$x[,1:3],color = colors)
#legend(s3d$xyz.convert(7.5, 3, 4.5), legend = levels(shots_stats.df),
#      col =  c("#999999", "#E69F00", "#56B4E9"), pch = 16)

colors2 <- colorPalette[shots_stats.df$outcomes.goal]
scatterplot3d(catPCA$x[,1:3],color = colors2)
```



```
#light blue is goals, yeah doesn't really do much
```

Note the first 3D graph is by cluster, the second is by outcome. While there is still some seperation of clusters, and the UMAP clusters line up alright, it looks like clusters 3 and 5 are getting projected into the same plane in PCA, as are clusters 2 and 4. Unfortunately, going up to 3 dimensions doesn't stop these overlapping clusters. All the features related to players who aren't the shooter or the goalie are heavily correlated with eachother and puck distance, which makes sense. Puck angle and goalie angle are also correlated for rather obvious reasons, although the lack of representation right handed is getting in the 2D plot is bizzare. Puck speed is strongly negatively correlated with the angle category, which I wasn't really expecting. Shooter speed is off doing it's own thing.

Discussion of results

Provide in natural language a clear discussion of your observations.

PCA really doesn't do much with the data, there isn't a clear clustering to any great extent. I don't think Jeff's discretezation has too much impact on clustering, and I think the fact that all the categoricals got lumped together orginally in PCA hasn't changed, as things that were orginally categorical are still correlated in this new PCA.

Analysis: Question 3 (Provide short name)

Question being asked

Provide in natural language a statement of what question you're trying to answer

Data Preparation

Provide in natural language a description of the data you are using for this analysis

Include a step-by-step description of how you prepare your data for analysis

If you're re-using dataframes prepared in another section, simply re-state what data you're using

```
# Include all data processing code (if necessary), clearly commented
```

Analysis methods used

Describe in natural language a statement of the analysis you're trying to do

Provide clearly commented analysis code; include code for tables and figures!

```
# Include all analysis code, clearly commented
# If not possible, screen shots are acceptable.
# If your contributions included things that are not done in an R-notebook,
#   (e.g. researching, writing, and coding in Python), you still need to do
#   this status notebook in R.  Describe what you did here and put any products
#   that you created in github. If you are writing online documents (e.g. overleaf
#   or google docs), you can include links to the documents in this notebook
#   instead of actual text.
```

Discussion of results

Provide in natural language a clear discussion of your observations.

Summary and next steps

Provide in natural language a clear summary and your proposed next steps.

I'm done with clustering analyses, as the results here for clustering have been dead ends. I'll focus on integrating and finalizing the app, and the final paper for the rest of the semester