DAR F23 Project Status Notebook 04

Hockey Analytics

Ashley Woodson

2023-10-16

Contents

Weekly Work Summary

- RCS ID: woodsa4
- Project Name: Hockey Analytics
- File names: woodsa4_assignment04.Rmd/.pdf
- Summary of work since last week
  - Using a variety of colors and scaling methods for the pheatmap function to improve intuitive readability.
  - Apply the udpated data from the rest of the group
- NEW: Summary of github issues added and worked
  - Not applicable
- Summary of github commits
  - branch name - dar-woodsa4
  - include browsable links to all external files on github - none
  - Include links to shared Shiny apps - none
- List of presentations, papers, or other outputs
  - Include browsable links - none
- List of references (if necessary)
- Indicate any use of group shared code base

  I have used the dataset created by Amy and Dr. Morgan https://github.rpi.edu/DataINCITE/Hockey_Fall_2023/blob/dar-enyena/StudentData/shots_stats_goal.df.Rds is the location of the most updated file at the moment I've typed this.
- Indicate which parts of your described work were done by you or as part of joint efforts

All the code in this notebook was created as an individual effort by me, however the dataframes I read in were created by Amy and Dr. Morgan's efforts.

- **Required:** Provide illustrating figures and/or tables

  The figures produced in the code below will be attached at the bottom of the pdf submitted to gradescope.

Personal Contribution

- Clearly defined, unique contribution(s) done by you: code, ideas, writing…
  - I have created the pheatmaps and all associated analysis

Package Installation and Data Loading

Here I have included the package installation and loading required for this notebook.

```
#load the required packages
library(knitr)
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v ggplot2 3.4.3     v purrr   1.0.2
## v tibble  3.2.1     v dplyr   1.1.3
## v tidyr   1.3.0     v stringr 1.5.0
## v readr   2.1.3     v forcats 0.5.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(tidymodels)
```

```
## -- Attaching packages -------------------------------------- tidymodels 1.1.1 --
## v broom        1.0.5     v rsample      1.2.0
## v dials        1.2.0     v tune         1.1.2
## v infer        1.0.4     v workflows    1.1.3
## v modeldata    1.2.0     v workflowsets 1.0.1
## v parsnip      1.1.1     v yardstick    1.2.0
## v recipes      1.0.8
## -- Conflicts ----------------------------------------- tidymodels_conflicts() --
## x scales::discard() masks purrr::discard()
## x dplyr::filter()   masks stats::filter()
## x recipes::fixed()  masks stringr::fixed()
## x dplyr::lag()      masks stats::lag()
## x yardstick::spec() masks readr::spec()
## x recipes::step()   masks stats::step()
## * Use suppressPackageStartupMessages() to eliminate package startup messages
```

```
library(ggplot2)
library(gplots)
```

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

```
library(heatmaply)
```

```
## Loading required package: plotly
##
## 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
##
## Loading required package: viridis
## Loading required package: viridisLite
##
## Attaching package: 'viridis'
##
## The following object is masked from 'package:scales':
##
##     viridis_pal
##
##
## ======================
## 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(pheatmap)
```

Analysis: Alternate scaling for pheatmap

Question being asked: Which scaling type is most appropriate?

Comparing scaling methods: 1. Scaling the data using min/max instead of mean/standard deviation 2. Applying teh scale function externally to the pheatmap function 3. Allowing the pheaetmap to scale automatically (my original work)

Which will produce the most legible visual & accurately depict the data?

Data Preparation

This is the dataframe Amy provided from before Dr. Morgan had introduced the new features. It is essentially the original data with the added goal outcomes to include the miss/defender block/goalie block in addition to goals.

```
shots_stats_goal.df <- read_rds("shots_stats_goal.df.Rds")

head(shots_stats_goal.df)
```

```
##   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
##   sameDefenders oppDefenders goalieScreened rightHanded outcomes
## 1             1            2           TRUE           0        3
## 2             1            2           TRUE           0        1
## 3             0            2          FALSE           0        3
## 4             0            2           TRUE           0        4
## 5             1            2           TRUE           1        1
## 6             2            3           TRUE           0        3
```

```
# custom function to implement min max scaling
minMax <- function(x) {
  (x - min(x)) / (max(x) - min(x))
}

#separate the features from the goals outcome
features <- subset(shots_stats_goal.df, select = -outcomes)
goal <- subset(shots_stats_goal.df , select = outcomes)

#convert all the non-numeric features to numeric
#done manually since the only non-numeric feature is known to be goalieScreened
features$goalieScreened <- as.numeric(features$goalieScreened)
goal$outcomes <- as.numeric(goal$outcomes)

#normalise data using custom function
mmScale <- as.data.frame(lapply(features, minMax))

#formatting the traditionally scaled features
pfeatures <- as.matrix(features, rownames.force = TRUE)

#formatting the min max scaled features
pmmScale <-  as.matrix(mmScale, rownames.force = TRUE)

#manually scale the data for the last pheatmap option
manScale <- scale(pfeatures)

#Create the colors for the pheatmap side column identifying the goal outcomes
# Defender blocked, goal, goalie blocked, miss
resultColors <- list(outcomes=c("yellow", "black", "green", "red"))
```

Analysis: Methods and results

Here is where I create the various pheatmaps. They are located at the bottom of the pdf submission to gradescope since they are too large to load automatically.

```
#make a png pheatmap using the goals as a row annotation
#and scaling by column
#use cutree_rows to divide the clusters



#First pheatmap uses the builtin scaling
png("pheatmapBuiltin.png", height = 1000, width = 800)
pheatmap(pfeatures,
         scale = 'column',
         main = "PHeatmap builtin scaling",
         col = colorRampPalette(c("blue", "white", "red"))(75),
         cutree_rows = 5,
         annotation_row = goal,
         annotation_colors = resultColors
         )



#second heatmap uses the manual min/max scaling
png("pheatmapMinMax.png", height = 1000, width = 800)
pheatmap(pmmScale,
         scale = 'none',
         main = "PHeatmap min/max scaling",
         col = colorRampPalette(c("blue","white", "red"))(75),
         cutree_rows = 5,
         annotation_row = goal,
         annotation_colors = resultColors
         )

#third heatmap uses the manual mean scaling external from the function
png("pheatmapManMean.png", height = 1000, width = 800)
pheatmap(manScale,
         scale = 'none',
         main = "PHeatmap manual scaling",
         col = colorRampPalette(c("blue", "white", "red"))(75),
         cutree_rows = 5,
         annotation_row = goal,
         annotation_colors = resultColors
         )
```

Discussion of results

The min/max scaled pheatmap seems to be overvaluing the categorical variables, since they have essentially been pushed to the extremes. With only 0 or 1, they become more pronounced than the continuous variables which tend towards 0.5 when min/max scaled. This has affected the pheatmap's ability to cluster the data, such that goals have become a bit scattered. This does not seem to be an effective method.

Both the manual and builtin scaling methods have produced the same clustering, although slightly different color scales. They both apply subtraction of the mean and division by standard deviation, but the builtin also applies some additional calculations that appear to tend the data more towards the mean, probably with the intention of muting outliers.

From this, I have concluded that the manual scaling is most effective to move forward with.

Analysis: Question 2 Removing categorical variables

Question being asked

We discussed in class whether removing the categorical variables would produce a beneficial analysis for the continuous variables in the min/max scale.

Data Preparation

This section begins with the min/max scaled matrix from Question 1.

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

#remove the two most extremely categorical variables
#handedness and screened
mm2 <- subset(mmScale , select = -goalieScreened )
mm2 <- subset(mm2 , select = -rightHanded )

head(mm2)
```

```
##    puckDist puckAngle puckSpeed shooterSpeed goalieDist goalieAngle     posTime
## 1 0.6837829 0.2818805 0.3280662   0.24334292  0.5220045   0.2185021 0.018018018
## 2 0.3533344 0.4140248 0.2269748   0.18660635  0.1564106   0.3499548 0.135135135
## 3 0.3626420 0.2185638 0.5264995   0.36171463  0.4365815   0.1709328 0.009009009
## 4 0.3928688 0.9569721 0.5132538   0.25931732  0.2474458   0.8413837 0.351351351
## 5 0.2580252 0.3022942 0.4962230   0.40882235  0.5735515   0.2996720 0.540540541
## 6 0.5981528 0.3416310 0.4418260   0.08059193  0.4771019   0.3481950 0.333333333
##   sameDefenders oppDefenders
## 1           0.5    0.3333333
## 2           0.5    0.3333333
## 3           0.0    0.3333333
## 4           0.0    0.3333333
## 5           0.5    0.3333333
## 6           1.0    0.6666667
```

```
#remove all categorical variables
mm4 <- subset(mm2 , select = -sameDefenders )
mm4 <- subset(mm4 , select = -oppDefenders )

head(mm4)
```

```
##    puckDist puckAngle puckSpeed shooterSpeed goalieDist goalieAngle     posTime
## 1 0.6837829 0.2818805 0.3280662   0.24334292  0.5220045   0.2185021 0.018018018
## 2 0.3533344 0.4140248 0.2269748   0.18660635  0.1564106   0.3499548 0.135135135
## 3 0.3626420 0.2185638 0.5264995   0.36171463  0.4365815   0.1709328 0.009009009
## 4 0.3928688 0.9569721 0.5132538   0.25931732  0.2474458   0.8413837 0.351351351
## 5 0.2580252 0.3022942 0.4962230   0.40882235  0.5735515   0.2996720 0.540540541
## 6 0.5981528 0.3416310 0.4418260   0.08059193  0.4771019   0.3481950 0.333333333
```

Analysis: Methods and Results

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!

```
#this uses the data with half the categoricals (2 value) having been removed
png("pheatmapNoTF.png", height = 1000, width = 800)
pheatmap(mm2,
         scale = 'none',
         main = "PHeatmap min/max scaling with less categorical variables",
         col = colorRampPalette(c("blue","white", "red"))(75),
         cutree_rows = 5,
         annotation_row = goal,
         annotation_colors = resultColors
         )

#This has all categorical variables removed
png("pheatmapNoCategorical.png", height = 1000, width = 800)
pheatmap(mm4,
         scale = 'none',
         main = "PHeatmap min/max scaling with no categorical variables",
         col = colorRampPalette(c("blue", "white", "red"))(75),
         cutree_rows = 5,
         annotation_row = goal,
         annotation_colors = resultColors
         )
```

Discussion of results

To see if the removeal of the categoriacal variables would show any new analysis for the continuous variables, I have removed them in stages. The partial removal does not appear to contribute anything useful, but the complete removal reiterates some things we had seen in the mean/standard deviation scaling. Principally, that the most goals are scored when the goalie and puck angles differ greatly. While this shows some more extreme coloring than the mean scale, it isn't demonstrating any new insight.

Analysis: Question 3 New Data Frame

Question being asked

After concluding above that the best representation is the manual mean scaling, how does this apply to the new data which includes new variables?

Data Preparation

Using the new variables coded for by Dr. Morgan.

```
shots_stats_goal2.df <- read_rds("shots_stats_goal.df-2.Rds")

head(shots_stats_goal2.df)
```

```
##   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 closestDef   defDist  defAngle
## 1          1          2           0         H1 554.55208 140.29522
## 2          1          2           0         H1  77.14952 129.70627
## 3          0          2           0         H2 208.92192 125.24775
## 4          0          2           0         H1  77.19322  59.64689
## 5          1          2           1         H1 129.34843 139.81344
## 6          2          3           0         H1 141.06063 169.62383
##      shotOutcome outcomes.goal
## 1           Save             3
## 2 Defender Block             1
## 3           Save             3
## 4           Miss             4
## 5 Defender Block             1
## 6           Save             3
```

```
#separate the features from the goals outcome
features2 <- subset(shots_stats_goal2.df, select = -outcomes.goal)
features2 <- subset(features2, select = -shotOutcome)

goal2 <- subset(shots_stats_goal2.df , select = outcomes.goal)

#remove the closestDef since the type of defender is non-numeric nor
#can it logically be converted easily
features2 <- subset(features2, select = -closestDef)

features2 <- scale(features2)

#Create the colors for the pheatmap side column identifying the goal outcomes
# Defender blocked, goal, goalie blocked, miss
resultColors <- list(outcomes=c("white","yellow", "black", "green", "red"))
```

Analysis methods used

Give the manually scaled pheatmap with the newest data

```
#This is the most up-to-date data frame
png("pheatmapNewData.png", height = 1000, width = 800)
pheatmap(features2,
         scale = 'none',
         main = "PHeatmap manual scaling new data",
         col = colorRampPalette(c("blue", "white", "red"))(75),
         cutree_rows = 5,
         annotation_row = goal2,
         annotation_colors = resultColors
         )
```

Discussion of results

The new data appears to desire a different number of clusters than is currently being applied. There are two clusters which contain only a few shots, defined rather sharply by high defDist and shooterSpeed values, respectively. Neither of these contain goals, so perhaps this shows, although rather niche, new modes of faliure we had not seen in the previous dataset.

Summary and next steps

Most of this work was performed before the updated dataframe had been released, so much of the analysis is now out-of date, but provides some direction towards further steps. Alternative scaling and other methods can be applied to this new data, following the format already established.

Additionally, my next steps will include analysis using the max/min scaling on Jeff's categorized versions of the continuous variables. This will hopefully mitigate the preference distortions seen in the above min/max scaled graph, since when it is entirely categorical variables the scales should self-balance.