msheridan econ 1042 goali project

Matt Sheridan

2023-04-11

```r
#https://www.geeksforgeeks.org/how-to-calculate-auc-area-under-curve-in-r/
# Should be using auc roc because we're concerned about ranking here, not necessarily all out probabili
library(ggplot2)
library(tidyverse)
library("xtable")
library(broom)
library(stargazer)
library(MASS)
options(scipen = 99)
```

```r
team_stand = data.frame(readxl::read_excel("TeamStandingsFinal.xlsx"))

team_stand$WinsPerGame = team_stand$W / team_stand$GP

years = unique(team_stand$Year)

coefs_1 = rep(NULL,length(years))

for (i in 3:length(years)){
  current = years[i]
  dta = data.frame(subset(team_stand, Year %in% c(years[i-2],years[i-1], years[i])))
  dta$weight = 0
  dta[dta$Year == years[i-2], ]$weight = 1
  dta[dta$Year == years[i-1], ]$weight = 2
  dta[dta$Year == years[i], ]$weight = 3
  coefs_1 = c(coefs_1, summary(lm(GD.GP ~ WinsPerGame, data = dta, weights = weight))$coefficients[2,1]]
}

GPW = data.frame(readxl::read_excel("GoalsPerWinStat.xlsx"))[,c(1,3)]

GPW = rbind(GPW, data.frame(Year = years[6:7], Goals.Per.Win = coefs_1[4:5]))[2:16,]
GPW$Year = as.numeric(GPW$Year)
GPW
```

```
##    Year Goals.Per.Win
## 2  2008      5.525000
## 3  2009      5.525000
## 4  2010      5.600000
## 5  2011      5.733000
## 6  2012      5.389000
## 7  2013      5.279000
## 8  2014      5.252000
## 9  2015      5.182000
## 10 2016      5.312000
## 11 2017      5.132000
## 12 2018      5.364000
## 13 2019      5.620000
## 14 2020      5.571000
## 15 2021      5.250578
## 16 2022      5.543946
```

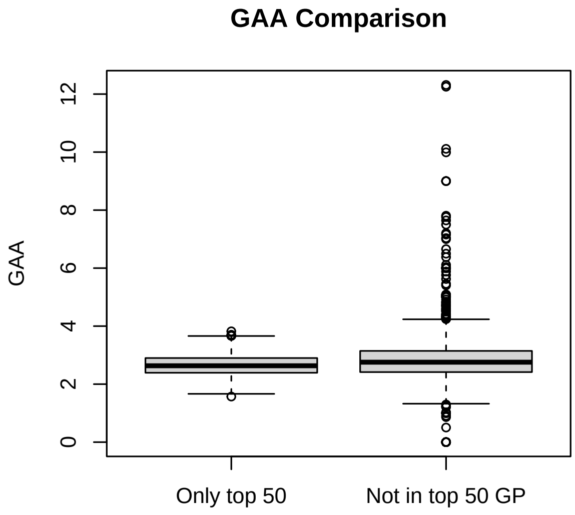
```
#Data loading and cleaning
goalie_lagged = data.frame(readxl::read_excel("goaliedata2.xlsx"))
goalie_lagged = goalie_lagged[goalie_lagged$ongoal > 0, ]

#GSAX variables
goalie_lagged$GSAX = goalie_lagged$XGA - goalie_lagged$GA
goalie_lagged$lagged_GSAX = goalie_lagged$lagged_xga - goalie_lagged$lagged_ga

#flurry adjusted
goalie_lagged$flurry_GSAX = goalie_lagged$flurryAdjustedxGoals - goalie_lagged$GA
goalie_lagged$lagged_flurry_GSAX = goalie_lagged$lagged_flurryadjxg - goalie_lagged$lagged_ga

goalie_lagged$flurryGSAXper60 = (60 * goalie_lagged$flurry_GSAX) / (goalie_lagged$TOI/60)
goalie_lagged$lagged_flurryGSAXper60 = (60 * goalie_lagged$lagged_flurry_GSAX) / (goalie_lagged$lagged_

#GSAX Per Game
goalie_lagged$GSAXper = (goalie_lagged$XGA - goalie_lagged$GA) / goalie_lagged$GP
goalie_lagged$GSAXper_lagged = (goalie_lagged$lagged_xga - goalie_lagged$lagged_ga) / goalie_lagged$lag

#GSAX per 60
goalie_lagged$GSAXper60 = (60 * goalie_lagged$GSAX) / (goalie_lagged$TOI/60)
goalie_lagged$lagged_GSAXper60 = (60 * goalie_lagged$lagged_GSAX) / (goalie_lagged$lagged_toi/60)

#GP Percentage
goalie_lagged$GPPCT = goalie_lagged$GP / 82
goalie_lagged$lagged_GPPCT = goalie_lagged$lagged_gp / 82

#lockout adjusting - this year is weird because the gppcts could be higher since there were less games.
goalie_lagged[goalie_lagged$Year==2012,]$GPPCT = goalie_lagged[goalie_lagged$Year==2012,]$GPPCT * 82/48
goalie_lagged[goalie_lagged$Year==2012,]$lagged_GPPCT = goalie_lagged[goalie_lagged$Year==2012,]$lagged

#covid adjusting - this year is weird because the gppcts could be higher since there were less games.
goalie_lagged[goalie_lagged$Year==2012,]$GPPCT = goalie_lagged[goalie_lagged$Year==2012,]$GPPCT * 82/70
goalie_lagged[goalie_lagged$Year==2012,]$lagged_GPPCT = goalie_lagged[goalie_lagged$Year==2012,]$lagged

#SVPCT
goalie_lagged$SVPCT = (goalie_lagged$ongoal - goalie_lagged$GA) / goalie_lagged$ongoal
goalie_lagged$lagged_SVPCT = (goalie_lagged$lagged_ongoal - goalie_lagged$lagged_ga) / goalie_lagged$la

#GAA
goalie_lagged$GAA = (60*goalie_lagged$GA) / (goalie_lagged$TOI/60)
goalie_lagged$lagged_GAA = (60*goalie_lagged$lagged_ga) / (goalie_lagged$lagged_toi/60)

#low danger goals saved above expected
```

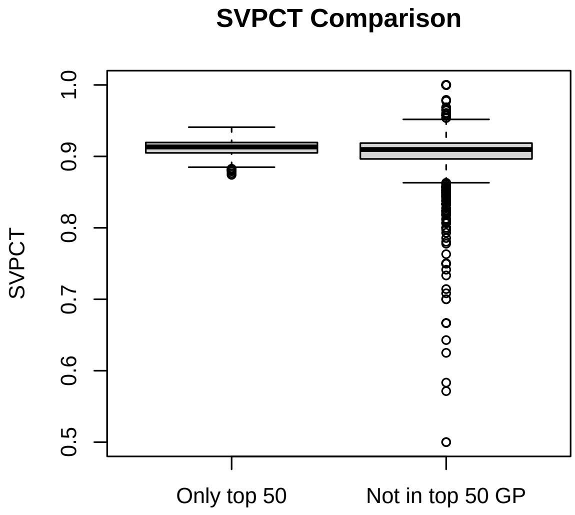
```
goalie_lagged$LDGSAX = goalie_lagged$lowDangerxGoals - goalie_lagged$lowDangerGoals
goalie_lagged$lagged_LDGSAX = goalie_lagged$lagged_ldxg - goalie_lagged$lagged_ldg

#medium danger goals saved above expected
goalie_lagged$MDGSAX = goalie_lagged$mediumDangerxGoals - goalie_lagged$mediumDangerGoals
goalie_lagged$lagged_MDGSAX = goalie_lagged$lagged_mdxg - goalie_lagged$lagged_mdg

#high danger goals saved above expected
goalie_lagged$HDGSAX = goalie_lagged$highDangerxGoals - goalie_lagged$highDangerGoals
goalie_lagged$lagged_HDGSAX = goalie_lagged$lagged_hdxg - goalie_lagged$lagged_hdg

#low danger goals saved above expected per 60
goalie_lagged$LDGSAXper = (60 * goalie_lagged$LDGSAX) / (goalie_lagged$TOI/60)
goalie_lagged$lagged_LDGSAXper = (60 * goalie_lagged$lagged_LDGSAX) / (goalie_lagged$lagged_toi/60)

#medium danger goals saved above expected per 60
goalie_lagged$MDGSAXper = (60 * goalie_lagged$MDGSAX) / (goalie_lagged$TOI/60)
goalie_lagged$lagged_MDGSAXper = (60 * goalie_lagged$lagged_MDGSAX) / (goalie_lagged$lagged_toi/60)

#high danger goals saved above expected per 60
goalie_lagged$HDGSAXper = (60 * goalie_lagged$HDGSAX) / (goalie_lagged$TOI/60)
goalie_lagged$lagged_HDGSAXper = (60 * goalie_lagged$lagged_HDGSAX) / (goalie_lagged$lagged_toi/60)

#Win PCT
goalie_lagged$WGP = goalie_lagged$W / goalie_lagged$GP
```

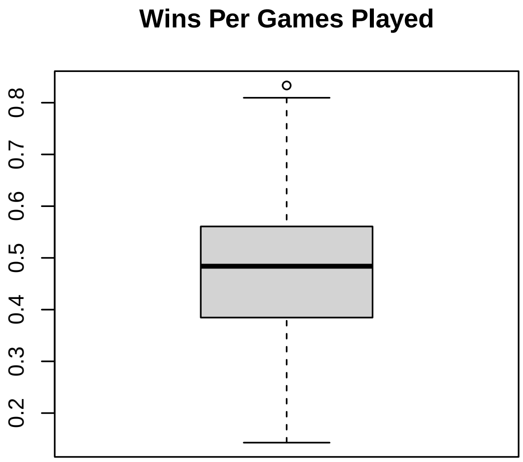
```
par(mfrow = c(2,2))

goalies_2022 = subset(goalie_lagged, (Year == 2022) & !is.na(W))

goalies_not_2022 = subset(goalie_lagged, (Year != 2022) & !is.na(W))

goalies_not_2022_nowin = subset(goalie_lagged, (Year != 2022))

boxplot(goalies_not_2022$GAA, goalies_not_2022_nowin$GAA,
        names=c("Only top 50", "Not in top 50 GP"), main = "GAA Comparison", ylab = "GAA")
boxplot(goalies_not_2022$SVPCT, goalies_not_2022_nowin$SVPCT,
        names=c("Only top 50", "Not in top 50 GP"), main = "SVPCT Comparison", ylab = "SVPCT")

print(xtable(t(summary(1:8))), type="html", file="xt.html", include.rownames=FALSE)

colnames = c("WGP", "GPPCT")

goalies_not_2022[,colnames(goalies_not_2022) %in% colnames]
```

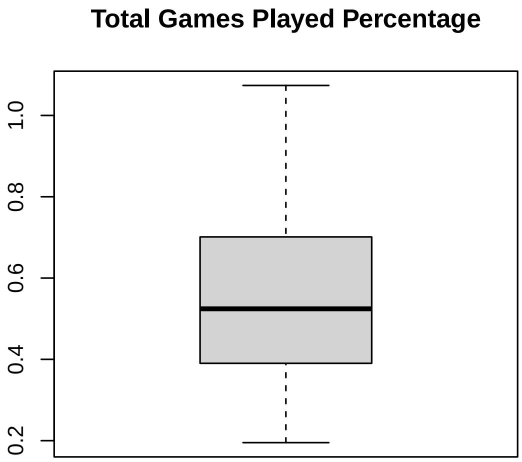
```
##          GPPCT       WGP
## 1    0.5609756 0.4130435
## 4    0.6829268 0.3571429
## 6    0.6341463 0.5576923
## 7    0.6341463 0.4230769
## 9    0.3292683 0.2962963
## 16   0.9268293 0.5921053
## 19   0.7439024 0.5409836
## 20   0.6707317 0.5272727
## 22   0.3414634 0.5714286
## 24   0.3780488 0.3225806
## 25   0.7560976 0.5645161
## 27   0.5975610 0.2857143
## 28   0.3414634 0.2857143
## 29   0.4878049 0.6250000
## 30   0.3780488 0.3870968
## 31   0.3780488 0.6129032
## 32   0.8536585 0.5428571
## 33   0.5609756 0.5652174
## 40   0.5365854 0.4772727
## 41   0.4756098 0.6410256
## 42   0.3780488 0.5161290
## 43   0.6463415 0.6226415
## 44   0.9024390 0.4459459
## 45   0.2682927 0.5454545
## 48   0.6463415 0.6792453
## 52   0.3780488 0.4838710
## 53   0.2926829 0.2083333
## 55   0.5121951 0.5952381
## 56   0.7195122 0.5762712
## 57   0.7682927 0.4444444
## 58   0.5609756 0.5000000
## 59   0.6951220 0.4736842
## 60   0.5243902 0.3720930
## 62   0.5000000 0.4878049
## 64   0.8292683 0.5735294
## 65   0.5609756 0.4130435
## 66   0.3170732 0.3846154
## 70   0.7560976 0.6612903
## 73   0.7804878 0.4062500
## 75   0.4268293 0.3142857
## 78   0.5000000 0.3414634
## 81   0.6341463 0.4423077
## 82   0.7195122 0.4406780
## 83   0.4024390 0.3939394
## 85   0.3536585 0.4482759
## 87   0.4146341 0.5294118
## 93   0.8414634 0.5942029
## 94   0.3902439 0.2812500
## 96   0.6707317 0.5272727
## 98   0.5609756 0.3913043
## 99   0.7682927 0.3650794
## 100  0.5121951 0.3809524
## 102  0.8292683 0.5882353
## 105  0.5121951 0.3095238
## 109  0.6951220 0.3508772
## 111  0.2926829 0.3750000
## 112  0.3048780 0.3600000
## 113  0.5121951 0.3333333
## 115  0.3536585 0.5517241
## 117  0.6341463 0.4230769
## 118  0.5853659 0.5416667
## 121  0.4146341 0.3529412
## 122  0.3170732 0.3846154
## 123  0.5487805 0.5777778
## 125  0.4024390 0.2727273
## 127  0.8658537 0.6197183
## 128  0.5487805 0.4888889
## 129  0.4268293 0.3428571
## 132  0.9390244 0.5844156
## 133  0.5000000 0.3170732
## 134  0.2804878 0.3043478
## 135  0.8902439 0.4794521
## 136  0.8414634 0.6086957
## 137  0.8780488 0.5416667
## 140  0.3414634 0.3571429
## 142  0.2804878 0.3913043
## 144  0.3536585 0.3103448
## 147  0.3780488 0.4838710
## 148  0.5731707 0.4468085
## 149  0.8902439 0.4794521
## 151  0.7439024 0.4918033
## 152  0.3170732 0.5769231
## 153  0.8658537 0.5352113
## 155  0.4268293 0.2857143
## 156  0.8048780 0.5606061
## 159  0.7073171 0.5517241
## 161  0.5243902 0.3953488
## 162  0.7195122 0.5084746
## 170  0.4756098 0.6666667
## 172  0.6097561 0.4600000
## 173  0.8414634 0.4927536
## 174  0.3414634 0.3214286
## 176  0.4024390 0.3939394
## 177  0.3536585 0.3793103
## 179  0.6829268 0.4107143
## 181  0.2804878 0.2608696
## 182  0.6219512 0.4705882
## 184  0.9024390 0.5000000
## 188  0.7317073 0.5833333
## 189  0.8048780 0.5151515
## 190  0.8292683 0.5294118
## 193  0.5853659 0.5625000
## 194  0.4146341 0.5294118
## 196  0.4512195 0.5405405
## 198  0.3048780 0.4400000
## 204  0.5731707 0.2127660
## 205  0.3536585 0.3793103
## 207  0.2804878 0.5652174
## 210  0.7439024 0.5737705
## 211  0.6951220 0.5789474
## 212  0.4024390 0.3333333
## 213  0.8292683 0.5294118
## 215  0.3048780 0.3200000
## 217  0.6585366 0.5185185
## 219  0.3292683 0.4074074
## 220  0.7317073 0.6333333
## 221  0.8658537 0.5211268
## 222  0.4146341 0.4411765
## 225  0.3048780 0.6400000
## 226  0.7804878 0.5156250
## 231  0.6951220 0.3859649
## 232  0.6951220 0.6140351
## 234  0.5487805 0.3333333
## 235  0.6951220 0.4736842
## 236  0.3170732 0.3076923
## 240  0.4390244 0.2777778
## 243  0.5975610 0.5306122
## 245  0.7073171 0.3620690
## 247  0.3780488 0.2580645
## 248  0.5365854 0.4772727
## 249  0.4268293 0.3428571
## 250  0.8780488 0.5277778
## 251  0.6585366 0.4444444
## 255  0.6585366 0.4444444
## 257  0.6707317 0.2727273
## 259  0.7926829 0.5538462
## 260  0.7195122 0.5423729
## 262  0.7195122 0.5932203
## 263  0.5853659 0.4791667
## 264  0.8292683 0.4411765
## 265  0.2804878 0.4782609
## 266  0.7560976 0.6290323
## 267  0.8170732 0.6268657
## 274  0.7926829 0.4000000
## 277  0.4634146 0.6052632
## 279  0.5609756 0.3478261
## 281  0.7682927 0.5238095
## 282  0.5731707 0.4255319
## 285  0.2317073 0.1578947
## 288  0.6463415 0.4905660
## 291  0.8902439 0.5890411
## 294  0.3658537 0.4666667
## 295  0.8536585 0.5000000
## 296  0.5121951 0.4047619
## 298  0.3902439 0.4687500
## 299  0.2560976 0.5714286
## 300  0.3536585 0.4827586
## 301  0.4634146 0.3421053
## 302  0.5853659 0.5208333
## 304  0.7439024 0.5081967
## 305  0.8292683 0.5000000
## 306  0.5609756 0.5652174
## 307  0.4878049 0.3000000
## 309  0.4024390 0.6060606
## 312  0.7195122 0.5593220
## 316  0.8414634 0.5072464
## 321  0.4878049 0.3250000
## 325  0.3170732 0.3076923
## 327  0.2317073 0.4736842
## 328  0.4146341 0.4117647
## 329  0.3780488 0.2903226
## 331  0.8170732 0.5671642
## 333  0.6951220 0.5263158
## 335  0.8902439 0.3972603
## 336  0.5121951 0.4523810
## 337  0.4146341 0.3823529
## 339  0.7195122 0.5254237
## 341  0.2439024 0.4000000
## 342  0.3292683 0.6296296
## 344  0.4390244 0.2777778
## 345  0.6707317 0.5636364
## 346  0.4146341 0.4411765
## 348  0.8292683 0.4264706
## 349  0.8785714 0.4166667
## 350  0.4636905 0.1578947
## 351  0.6101190 0.6000000
## 353  0.5857143 0.5833333
## 354  0.5369048 0.5000000
## 357  0.9273810 0.5526316
## 359  0.4636905 0.2105263
## 363  1.0494048 0.5581395
## 364  0.8053571 0.5757576
## 366  0.4880952 0.3500000
## 367  1.0494048 0.5581395
## 368  0.4392857 0.2777778
## 371  0.5125000 0.3809524
## 372  1.0005952 0.5609756
## 373  0.5125000 0.8095238
## 374  0.8785714 0.5277778
## 375  0.3660714 0.6000000
## 376  0.3416667 0.6428571
## 377  0.8785714 0.6388889
## 378  0.4148810 0.5294118
## 381  0.6345238 0.5769231
## 383  0.3904762 0.3750000
## 384  0.5857143 0.3333333
## 387  0.9273810 0.3684211
## 388  1.0494048 0.3488372
## 392  0.9761905 0.4750000
## 393  0.9029762 0.4864865
## 397  0.4636905 0.3157895
## 400  0.3660714 0.2666667
## 402  0.4880952 0.3500000
## 405  0.8297619 0.4411765
## 406  0.8541667 0.3142857
## 408  0.9761905 0.4250000
## 410  0.8053571 0.6969697
## 411  0.4880952 0.6500000
## 413  0.7321429 0.6333333
## 416  0.4392857 0.2777778
## 418  1.0738095 0.4772727
## 421  0.7321429 0.5666667
## 423  0.7077381 0.4482759
## 424  0.5857143 0.5000000
## 425  0.4880952 0.4500000
## 426  0.4636905 0.3157895
## 427  0.3416667 0.6428571
## 430  0.9517857 0.5384615
## 431  0.7926829 0.5076923
## 432  0.3414634 0.3571429
## 433  0.4878049 0.3750000
## 437  0.4146341 0.3235294
## 438  0.5853659 0.3750000
## 441  0.4390244 0.3333333
## 443  0.3780488 0.5806452
## 444  0.6829268 0.4464286
## 452  0.3170732 0.4615385
## 453  0.6341463 0.5576923
## 454  0.7195122 0.5423729
## 457  0.7804878 0.6093750
## 460  0.7682927 0.5873016
## 462  0.5853659 0.4791667
## 463  0.7682927 0.6507937
## 465  0.6463415 0.4716981
## 466  0.7073171 0.5517241
## 467  0.6097561 0.5800000
## 468  0.3902439 0.3750000
## 469  0.7195122 0.5762712
## 474  0.7682927 0.5238095
## 476  0.7439024 0.5409836
## 478  0.3780488 0.2903226
## 482  0.3414634 0.3214286
## 485  0.5487805 0.3555556
## 486  0.7195122 0.4237288
## 490  0.7073171 0.6206897
## 491  0.3414634 0.4642857
## 493  0.6707317 0.4727273
## 495  0.6951220 0.3859649
## 498  0.4878049 0.5000000
## 500  0.3658537 0.3333333
## 503  0.3048780 0.3200000
## 504  0.3292683 0.6296296
## 506  0.5975610 0.5510204
## 507  0.3536585 0.6206897
## 509  0.3414634 0.7142857
## 512  0.4390244 0.3333333
## 514  0.3414634 0.1428571
## 515  0.3414634 0.3928571
## 517  0.7560976 0.4354839
## 519  0.4878049 0.4000000
## 520  0.3048780 0.5200000
## 521  0.4390244 0.5277778
## 522  0.3292683 0.5925926
## 526  0.4756098 0.4871795
## 527  0.7804878 0.6093750
## 528  0.7926829 0.5230769
## 529  0.3902439 0.3750000
## 535  0.4512195 0.5945946
## 536  0.8902439 0.5616438
## 539  0.7439024 0.4590164
## 540  0.3536585 0.5517241
## 541  0.5487805 0.6444444
## 542  0.8048780 0.6666667
## 548  0.2439024 0.4500000
## 549  0.5609756 0.6521739
## 550  0.7804878 0.6406250
## 551  0.6219512 0.3529412
## 553  0.8414634 0.3768116
## 555  0.7073171 0.3620690
## 556  0.6951220 0.5614035
## 557  0.1951220 0.4375000
## 559  0.3902439 0.2187500
## 561  0.8780488 0.5000000
## 562  0.4268293 0.2571429
## 564  0.4146341 0.2352941
## 565  0.6341463 0.5000000
## 567  0.6219512 0.4313725
## 568  0.7195122 0.6440678
## 570  0.7560976 0.6451613
## 573  0.7439024 0.5081967
## 575  0.6097561 0.4400000
## 576  0.6951220 0.4912281
## 579  0.7804878 0.5312500
## 582  0.6951220 0.2631579
## 584  0.3048780 0.3600000
## 587  0.6585366 0.6481481
## 588  0.4268293 0.4000000
## 592  0.8536585 0.4857143
## 593  0.7560976 0.2258065
## 595  0.6219512 0.5882353
## 596  0.2682927 0.3636364
## 598  0.3170732 0.2307692
## 600  0.3780488 0.4516129
## 603  0.6097561 0.3600000
## 608  0.4634146 0.5526316
## 609  0.3780488 0.3225806
## 610  0.2804878 0.5652174
## 611  0.2439024 0.3000000
## 613  0.5609756 0.5652174
## 614  0.2926829 0.8333333
## 619  0.7073171 0.6206897
## 620  0.5243902 0.5813953
## 622  0.7926829 0.5692308
## 623  0.7560976 0.5645161
## 624  0.4390244 0.5000000
## 629  0.3536585 0.4137931
## 630  0.7439024 0.5737705
## 631  0.7317073 0.5166667
## 636  0.6585366 0.5000000
## 639  0.8048780 0.7272727
## 642  0.5731707 0.5531915
## 643  0.8292683 0.5882353
## 644  0.4512195 0.4054054
## 646  0.6707317 0.3818182
## 649  0.4878049 0.5250000
## 650  0.5487805 0.4888889
## 653  0.4634146 0.3157895
## 655  0.7073171 0.4655172
## 657  0.7926829 0.5384615
## 662  0.8048780 0.5151515
## 664  0.3170732 0.3461538
## 665  0.5000000 0.5609756
## 666  0.5121951 0.5476190
## 671  0.3780488 0.5161290
## 673  0.4024390 0.3939394
## 674  0.2926829 0.2916667
## 675  0.7073171 0.6034483
## 677  0.5853659 0.5208333
## 681  0.6341463 0.4423077
## 682  0.7804878 0.4843750
## 683  0.5243902 0.5116279
## 685  0.3048780 0.4400000
## 686  0.3658537 0.3000000
## 687  0.6951220 0.4736842
## 689  0.4878049 0.4250000
## 692  0.4756098 0.3846154
## 693  0.3536585 0.3448276
## 695  0.3902439 0.5625000
## 697  0.2926829 0.4583333
## 703  0.3902439 0.4687500
## 704  0.7073171 0.6034483
## 705  0.8170732 0.4776119
## 707  0.6219512 0.3333333
## 708  0.3048780 0.4800000
## 710  0.3170732 0.5000000
## 711  0.6585366 0.4259259
## 712  0.7195122 0.3728814
## 718  0.7195122 0.3898305
## 720  0.3170732 0.3846154
## 721  0.6707317 0.4727273
## 723  0.4634146 0.4736842
## 726  0.6463415 0.3584906
## 727  0.7073171 0.4482759
## 729  0.3414634 0.3214286
## 736  0.5975610 0.5306122
## 738  0.7926829 0.6153846
## 741  0.2926829 0.5416667
## 742  0.6097561 0.3000000
## 743  0.3780488 0.3548387
## 744  0.4878049 0.4250000
## 747  0.6585366 0.3333333
## 751  0.4390244 0.5000000
## 752  0.7439024 0.5081967
## 753  0.7317073 0.5000000
## 757  0.3658537 0.4333333
## 758  0.5121951 0.4285714
## 763  0.4756098 0.5384615
## 765  0.7560976 0.5967742
## 766  0.6097561 0.4600000
## 767  0.7804878 0.5468750
## 768  0.6219512 0.5098039
## 769  0.5000000 0.4634146
## 770  0.4756098 0.4615385
## 772  0.4878049 0.6250000
## 774  0.3414634 0.4285714
## 775  0.7926829 0.5692308
## 776  0.2926829 0.2500000
## 777  0.7439024 0.5409836
## 778  0.3902439 0.5625000
## 779  0.6707317 0.5818182
## 780  0.3658537 0.5333333
## 784  0.7317073 0.3333333
## 785  0.7560976 0.6774194
## 786  0.6097561 0.3600000
## 788  0.6829268 0.5535714
## 790  0.6341463 0.4807692
## 791  0.8048780 0.5000000
## 792  0.4512195 0.3243243
## 793  0.3170732 0.3076923
## 798  0.5975610 0.6530612
## 801  0.7439024 0.4262295
## 804  0.7682927 0.6507937
## 806  0.3414634 0.3928571
## 807  0.4512195 0.4054054
## 809  0.7317073 0.5000000
## 811  0.8048780 0.5757576
## 814  0.5365854 0.5000000
## 815  0.5243902 0.5348837
## 816  0.7804878 0.5156250
## 817  0.3414634 0.5714286
## 819  0.5000000 0.6341463
## 821  0.5609756 0.6304348
## 822  0.8170732 0.6567164
## 824  0.3414634 0.4285714
## 825  0.3780488 0.1612903
## 826  0.5975610 0.5510204
## 829  0.3902439 0.5312500
## 831  0.6463415 0.2641509
## 832  0.4268293 0.4285714
## 833  0.4268293 0.2857143
## 834  0.3292683 0.4814815
## 838  0.7195122 0.7118644
## 839  0.7195122 0.4576271
## 840  0.7317073 0.5166667
## 843  0.6585366 0.6296296
## 845  0.3414634 0.3571429
## 847  0.5243902 0.5348837
## 848  0.4878049 0.4250000
## 849  0.4390244 0.5277778
## 850  0.7073171 0.3965517
## 851  0.5243902 0.3023256
## 853  0.3170732 0.4230769
## 855  0.5975610 0.3265306
## 856  0.6219512 0.4705882
## 862  0.7926829 0.5692308
## 864  0.7926829 0.6769231
## 867  0.3536585 0.5172414
## 868  0.6341463 0.5000000
## 869  0.4390244 0.2777778
## 871  0.7682927 0.4126984
## 872  0.7317073 0.5833333
## 875  0.4634146 0.3684211
## 881  0.3780488 0.5161290
## 887  0.3292683 0.2592593
## 888  0.4268293 0.5142857
## 889  0.6585366 0.3703704
## 890  0.3536585 0.4137931
## 893  0.5731707 0.4468085
## 894  0.6585366 0.6296296
## 899  0.6707317 0.4545455
## 902  0.4024390 0.6060606
## 903  0.3780488 0.5161290
## 905  0.8170732 0.4626866
## 906  0.7195122 0.5423729
## 910  0.5609756 0.3478261
## 911  0.3780488 0.5483871
## 912  0.3780488 0.3225806
## 914  0.6707317 0.4909091
## 915  0.5243902 0.4186047
## 920  0.4878049 0.5750000
## 922  0.5000000 0.3902439
## 925  0.6097561 0.5800000
## 928  0.4268293 0.2571429
## 930  0.6097561 0.3400000
## 935  0.4756098 0.3589744
## 937  0.4390244 0.3611111
## 939  0.4024390 0.4848485
## 946  0.4024390 0.4242424
## 948  0.6463415 0.7358491
## 949  0.7317073 0.6000000
## 950  0.5609756 0.5869565
## 953  0.6707317 0.4545455
## 954  0.4878049 0.5500000
## 955  0.5487805 0.6000000
## 956  0.5609756 0.4130435
## 957  0.4390244 0.3888889
## 958  0.5243902 0.5348837
## 959  0.7682927 0.5396825
## 960  0.6341463 0.3461538
## 961  0.3902439 0.7500000
## 965  0.3292683 0.5925926
## 967  0.4512195 0.4864865
## 970  0.4390244 0.4166667
## 972  0.5609756 0.5869565
## 973  0.6829268 0.5357143
## 975  0.7439024 0.5737705
## 976  0.7073171 0.4482759
## 980  0.5000000 0.3658537
## 981  0.6097561 0.3600000
## 982  0.3292683 0.3703704
## 985  0.5121951 0.5476190
## 986  0.4512195 0.4054054
## 987  0.5975610 0.4081633
## 989  0.5609756 0.5434783
## 990  0.7560976 0.5806452
## 992  0.7560976 0.5967742
## 994  0.8048780 0.5303030
## 1001 0.3658537 0.3333333
## 1002 0.5365854 0.4772727
## 1003 0.5975610 0.5510204
## 1004 0.4634146 0.5263158
## 1005 0.7073171 0.5344828
## 1006 0.3658537 0.4000000
## 1007 0.6341463 0.5576923
## 1008 0.4024390 0.6060606
## 1009 0.4146341 0.5000000
## 1013 0.4390244 0.5277778
## 1014 0.4634146 0.5263158
## 1015 0.5121951 0.5714286
## 1019 0.3780488 0.5161290
## 1024 0.5000000 0.4146341
## 1025 0.4390244 0.5000000
## 1026 0.5731707 0.4680851
## 1027 0.5853659 0.5208333
## 1028 0.4878049 0.4000000
## 1029 0.4390244 0.5000000
## 1033 0.5000000 0.6341463
## 1035 0.3780488 0.5161290
## 1038 0.4878049 0.4250000
## 1041 0.5975610 0.4897959
## 1042 0.6219512 0.5882353
## 1043 0.6341463 0.3846154
## 1044 0.4146341 0.3235294
## 1046 0.3780488 0.3870968
## 1048 0.7073171 0.4655172
## 1049 0.3658537 0.5333333
## 1050 0.6341463 0.6730769
## 1053 0.4878049 0.5250000
## 1055 0.5121951 0.3809524
## 1056 0.4634146 0.4736842
## 1059 0.5609756 0.3260870
## 1061 0.4024390 0.4545455
## 1062 0.4512195 0.5135135
## 1063 0.3536585 0.5517241
## 1064 0.3170732 0.4230769
## 1065 0.3902439 0.4062500
## 1067 0.4756098 0.4871795
## 1068 0.6097561 0.4600000
## 1069 0.4024390 0.3636364
## 1070 0.4146341 0.5000000
## 1075 0.3292683 0.4814815
## 1076 0.5487805 0.4222222
## 1077 0.3780488 0.5806452
## 1079 0.4146341 0.6176471
## 1081 0.2682927 0.6818182
## 1082 0.4390244 0.5833333
## 1083 0.5487805 0.5333333
## 1085 0.2682927 0.2727273
## 1087 0.4268293 0.2571429
## 1090 0.3292683 0.3333333
## 1094 0.4268293 0.4571429
## 1097 0.3292683 0.3703704
## 1098 0.3902439 0.3750000
## 1099 0.3658537 0.5000000
## 1100 0.2804878 0.6086957
## 1102 0.4756098 0.6410256
## 1103 0.2682927 0.7727273
## 1105 0.2317073 0.6842105
## 1106 0.3292683 0.3703704
## 1109 0.2682927 0.5909091
## 1110 0.3048780 0.4800000
## 1111 0.4146341 0.4411765
## 1112 0.4268293 0.4000000
## 1116 0.4146341 0.2352941
## 1118 0.3170732 0.5000000
## 1120 0.2682927 0.5000000
## 1123 0.2926829 0.4166667
## 1126 0.4390244 0.5277778
## 1128 0.2926829 0.5416667
## 1129 0.4024390 0.2727273
## 1137 0.2439024 0.4500000
## 1139 0.5121951 0.7380952
## 1140 0.2317073 0.6842105
## 1142 0.2560976 0.3333333
## 1144 0.4390244 0.7222222
## 1145 0.2926829 0.3750000
## 1148 0.3780488 0.6129032
## 1150 0.4512195 0.5675676
## 1155 0.4878049 0.7500000
## 1156 0.3414634 0.2857143
## 1157 0.2439024 0.5500000
## 1158 0.3902439 0.6562500
## 1160 0.5121951 0.4285714
## 1161 0.4512195 0.4594595
## 1164 0.3536585 0.3793103
## 1168 0.3536585 0.3793103
## 1169 0.2317073 0.4736842
## 1172 0.2926829 0.6250000
## 1173 0.4268293 0.4571429
## 1178 0.5487805 0.6000000
## 1180 0.6341463 0.5000000
## 1181 0.6829268 0.3214286
## 1182 0.5609756 0.4782609
## 1183 0.5487805 0.2888889
## 1186 0.4512195 0.3513514
## 1189 0.5000000 0.5609756
## 1192 0.3170732 0.4230769
## 1194 0.4268293 0.2571429
## 1197 0.7804878 0.5156250
## 1198 0.3536585 0.3448276
## 1200 0.3292683 0.3333333
## 1202 0.7195122 0.4576271
## 1210 0.3780488 0.5483871
## 1211 0.6951220 0.6491228
## 1212 0.5975610 0.6326531
## 1217 0.3414634 0.5714286
## 1220 0.7682927 0.6190476
## 1222 0.6829268 0.5000000
## 1227 0.8048780 0.4393939
## 1229 0.4512195 0.4864865
## 1230 0.3780488 0.3225806
## 1232 0.5853659 0.3958333
## 1242 0.6463415 0.6792453
## 1244 0.3902439 0.2500000
## 1246 0.5121951 0.4761905
## 1248 0.3414634 0.4285714
## 1252 0.4878049 0.6250000
## 1253 0.4268293 0.3428571
## 1254 0.6341463 0.6730769
## 1255 0.5365854 0.5227273
## 1258 0.4512195 0.5405405
## 1263 0.3902439 0.5937500
## 1264 0.5609756 0.5000000
## 1265 0.3414634 0.5357143
## 1266 0.5365854 0.5227273
## 1270 0.6585366 0.7222222
## 1271 0.5853659 0.6250000
## 1275 0.6341463 0.2500000
## 1276 0.3780488 0.3225806
## 1280 0.7073171 0.5862069
## 1281 0.5000000 0.6341463
## 1282 0.8170732 0.5671642
## 1293 0.4024390 0.4545455
## 1294 0.6707317 0.3272727
```

```
boxplot(goalies_not_2022$WGP, main = "Wins Per Games Played")
boxplot(goalies_not_2022$GPPCT, main = "Total Games Played Percentage")
```

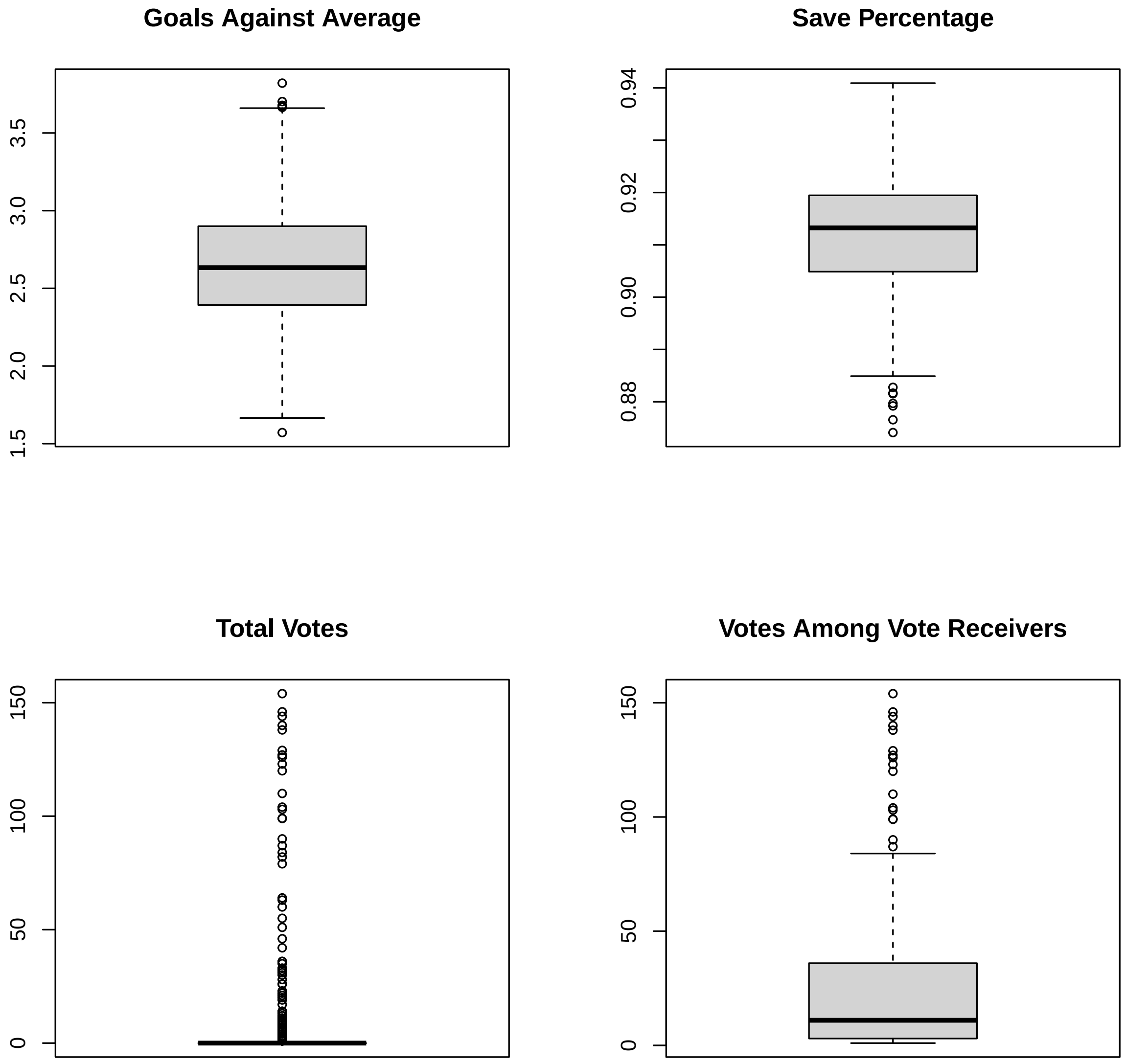








```
boxplot(goalies_not_2022$GAA, main = "Goals Against Average")
boxplot(goalies_not_2022$SVPCT, main = "Save Percentage")
boxplot(goalies_not_2022$Votes, main = "Total Votes")
boxplot(goalies_not_2022[goalies_not_2022$Votes>0,]$Votes, main = "Votes Among Vote Receivers")
```



```
nb_model_1 = glm.nb(Votes ~ WGP + GPPCT + SVPCT + GAA, data = goalies_not_2022)
summary(nb_model_1)
```

```
##
## Call:
## glm.nb(formula = Votes ~ WGP + GPPCT + SVPCT + GAA, data = goalies_not_2022,
##     init.theta = 0.239499014, link = log)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.6593  -0.5388  -0.1630  -0.0232   4.0063
##
## Coefficients:
```

```
##              Estimate Std. Error z value             Pr(>|z|)    
## (Intercept) -215.0915    27.7919  -7.739  0.00000000000000999 ***
## WGP           16.5274     1.7974   9.195 < 0.0000000000000002 ***
## GPPCT          8.8823     0.8139  10.913 < 0.0000000000000002 ***
## SVPCT        219.1445    28.3956   7.718  0.00000000000001186 ***
## GAA           -0.3073     0.8219  -0.374                0.708    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## (Dispersion parameter for Negative Binomial(0.2395) family taken to be 1)
## 
##     Null deviance: 1025.14  on 643  degrees of freedom
## Residual deviance:  228.66  on 639  degrees of freedom
## AIC: 1203.1
## 
## Number of Fisher Scoring iterations: 1
## 
## 
##               Theta:  0.2395 
##           Std. Err.:  0.0310 
## 
##  2 x log-likelihood:  -1191.1110
```

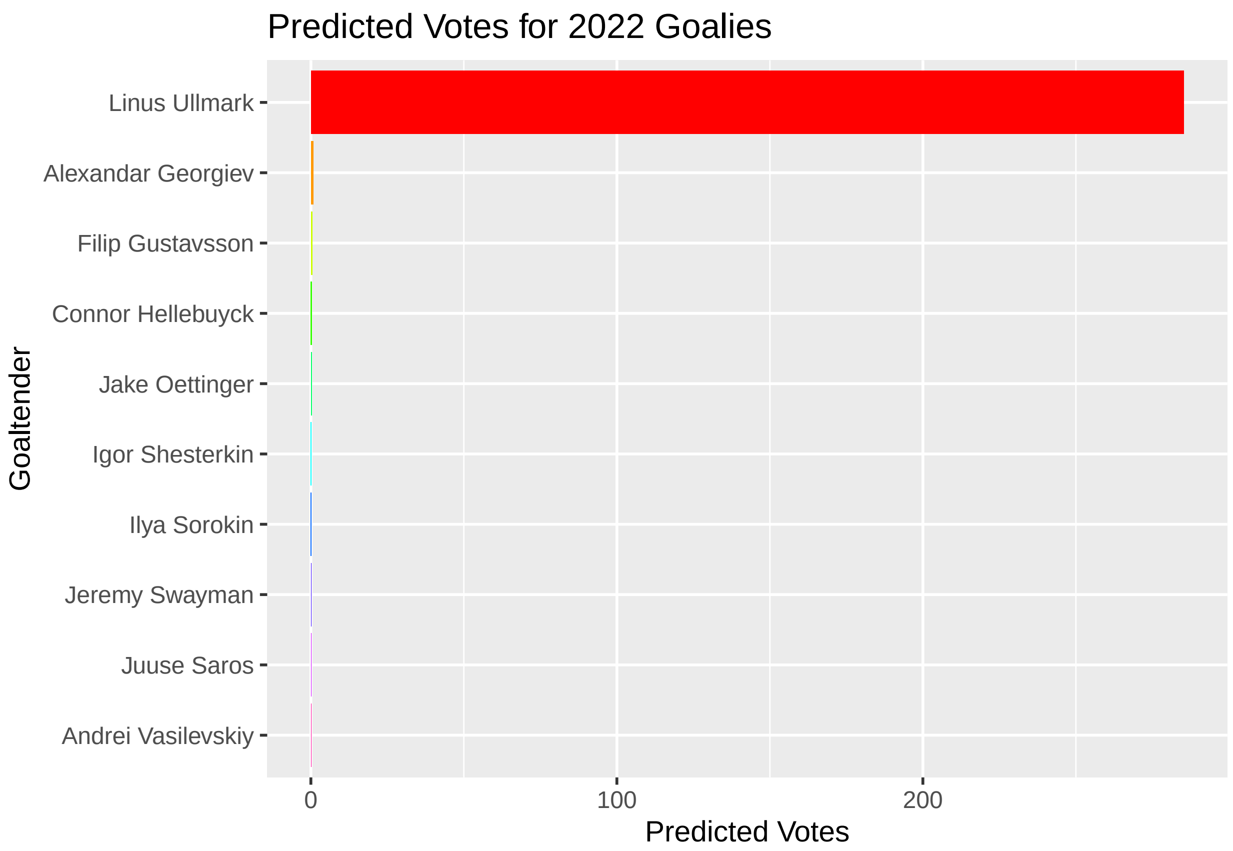
```
stargazer(nb_model_1, type='latex')
```

```
## 
## % Table created by stargazer v.5.2.3 by Marek Hlavac, Social Policy Institute. E-mail: marek.hlavac a
## % Date and time: Fri, May 05, 2023 - 11:54:07 PM
## \begin{table}[!htbp] \centering 
##   \caption{} 
##   \label{} 
## \begin{tabular}{@{\extracolsep{5pt}}lc} 
## \\[-1.8ex]\hline 
## \hline \\[-1.8ex] 
##  & \multicolumn{1}{c}{\textit{Dependent variable:}} \\ 
## \cline{2-2} 
## \\[-1.8ex] & Votes \\ 
## \hline \\[-1.8ex] 
##  WGP & 16.527$^{***}$ \\ 
##   & (1.797) \\ 
##   & \\ 
##  GPPCT & 8.882$^{***}$ \\ 
##   & (0.814) \\ 
##   & \\ 
##  SVPCT & 219.145$^{***}$ \\ 
##   & (28.396) \\ 
##   & \\ 
##  GAA & $-$0.307 \\ 
##   & (0.822) \\ 
##   & \\ 
##  Constant & $-$215.091$^{***}$ \\ 
##   & (27.792) \\ 
##   & \\ 
## \hline \\[-1.8ex] 
## Observations & 644 \\
## Log Likelihood & $-$596.555 \\
## $\theta$ & 0.239$^{***}$  (0.031) \\
## Akaike Inf. Crit. & 1,203.111 \\
## \hline
## \hline \\[-1.8ex]
## \textit{Note:}  & \multicolumn{1}{r}{$^{*}$p$<$0.1; $^{**}$p$<$0.05; $^{***}$p$<$0.01} \\
## \end{tabular}
## \end{table}
```

```
mean((predict(nb_model_1) - goalies_not_2022$Votes)^2)
```

```
## [1] 470.4179
```

```
preds = predict(nb_model_1, newdata = goalies_2022, type = 'response')

#288 * (preds / sum(preds))

preds_df_22 = data.frame(Name = goalies_2022$Name, pred_votes = 288 * (preds / sum(preds)))
top_10 = head(preds_df_22[order(preds_df_22$pred_votes, decreasing = T),],10)
top_10$Name = factor(top_10$Name, levels = top_10$Name)

predict_year = function(year, mod){
  data = subset(subset(goalie_lagged, (Year == year) & !is.na(W)))
  predictions = predict(mod, data, type='response')
  predictiondf = data.frame(Name = data$Name, pred_votes = 288 * (predictions / sum(predictions)))
  top_10 = head(predictiondf[order(predictiondf$pred_votes, decreasing = T),],10)
  top_10$Name = factor(top_10$Name, levels = top_10$Name)
  ggplot(top_10, mapping = aes(x =forcats::fct_rev(Name),y=pred_votes)) +
  geom_bar(stat='identity', fill=rainbow(10)) + coord_flip() +
  labs(title=paste("Predicted Votes for", year, "Goalies")) + ylab("Predicted Votes") +
    xlab("Goaltender")
}


ggplot(top_10, mapping = aes(x =forcats::fct_rev(Name),y=pred_votes)) +
  geom_bar(stat='identity', fill=rainbow(10)) + coord_flip() +
  labs(title="Predicted Votes for 2022 Goalies") + ylab("Predicted Votes") +
  xlab("Goaltender")
```



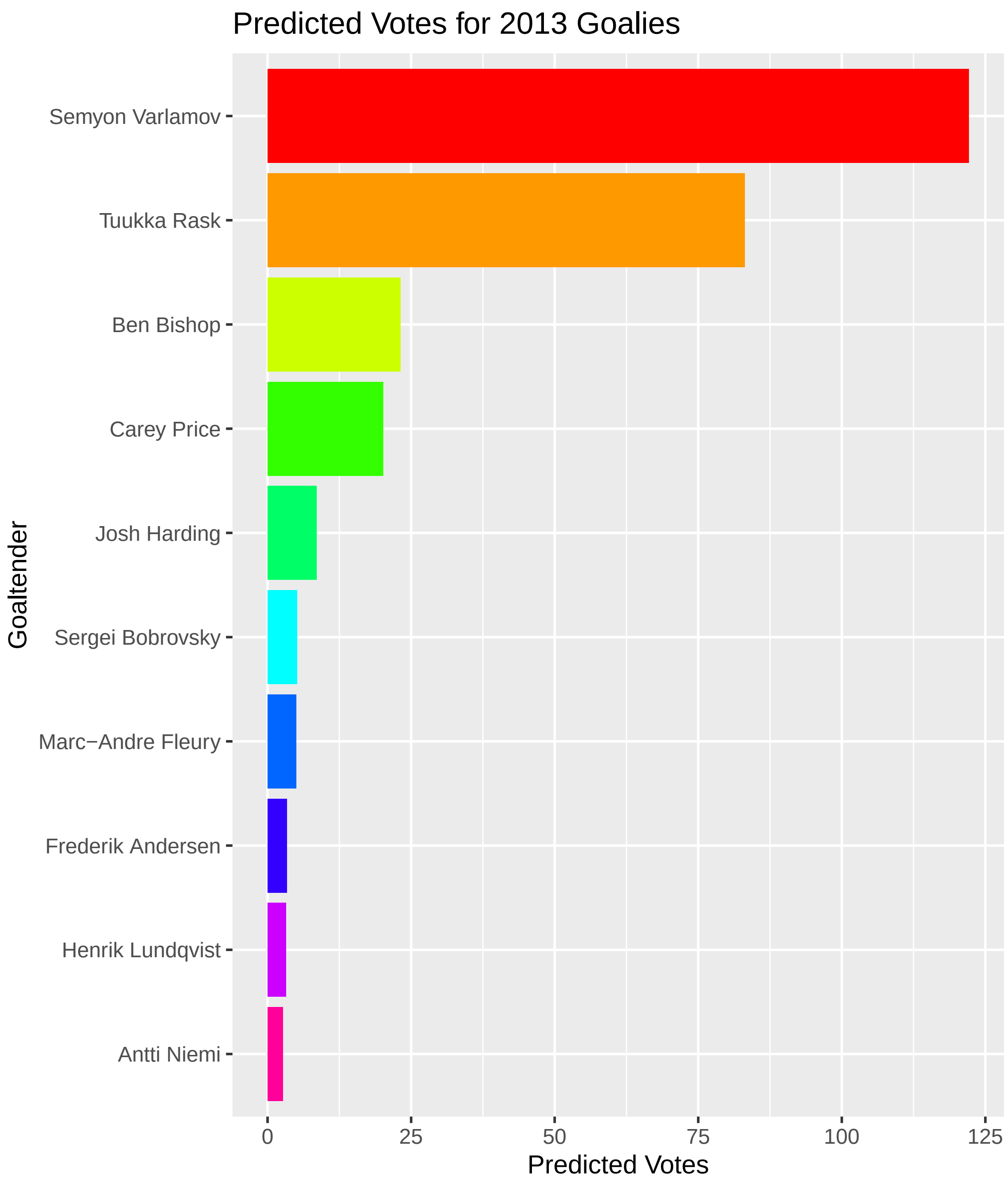
```
library("ggpubr")
```

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

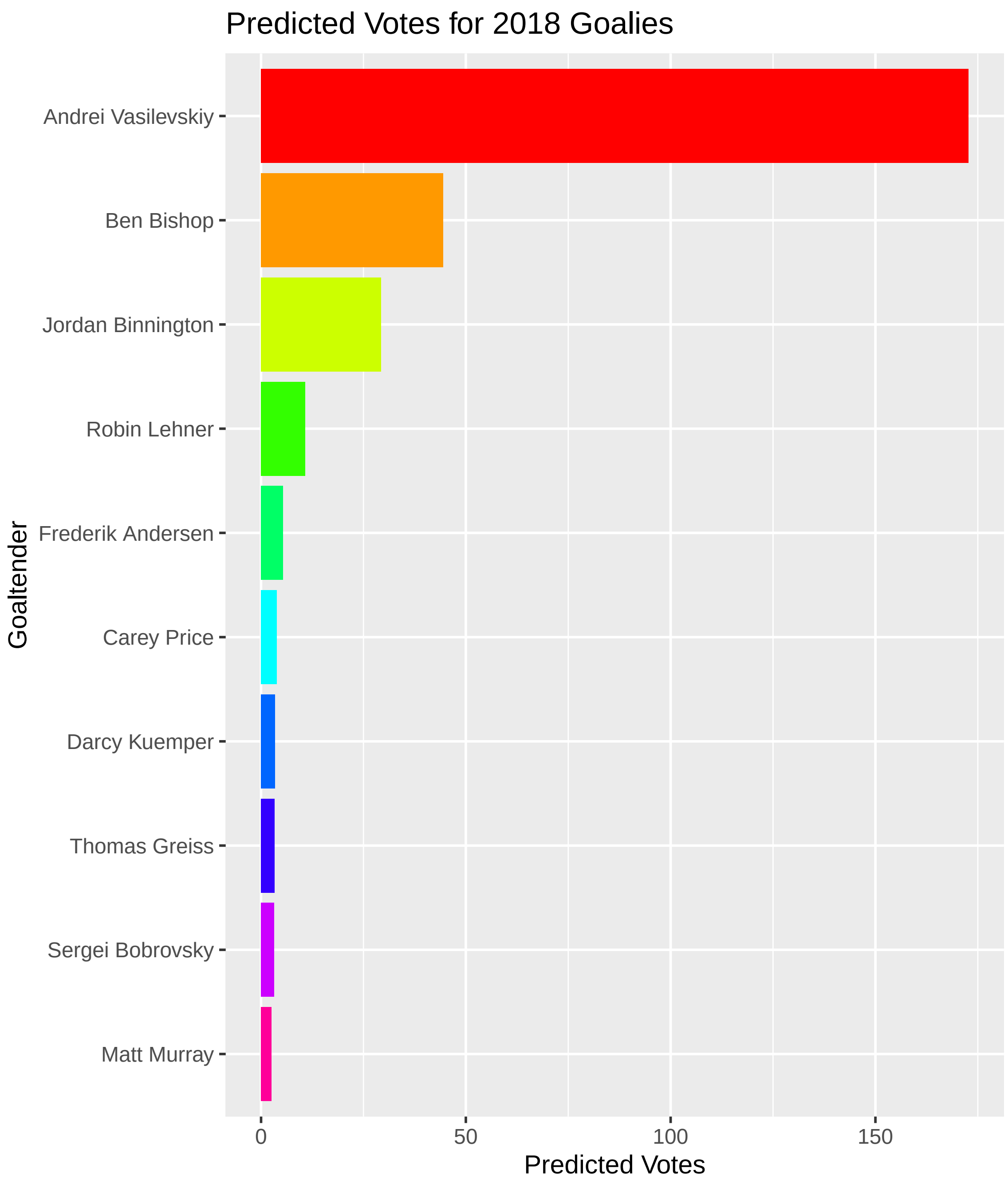
```
par(mfrow = c(5,3))
ggexport(predict_year(2008,nb_model_1),predict_year(2009,nb_model_1),
         predict_year(2010,nb_model_1),predict_year(2011,nb_model_1),
         predict_year(2013,nb_model_1),predict_year(2014,nb_model_1),
         predict_year(2015,nb_model_1),predict_year(2016,nb_model_1),
         predict_year(2017,nb_model_1),predict_year(2018,nb_model_1),
         predict_year(2019,nb_model_1),predict_year(2020,nb_model_1),
         predict_year(2021,nb_model_1),predict_year(2022,nb_model_1), "images.pdf")
```

```
## file saved to axolsit.pdf
```

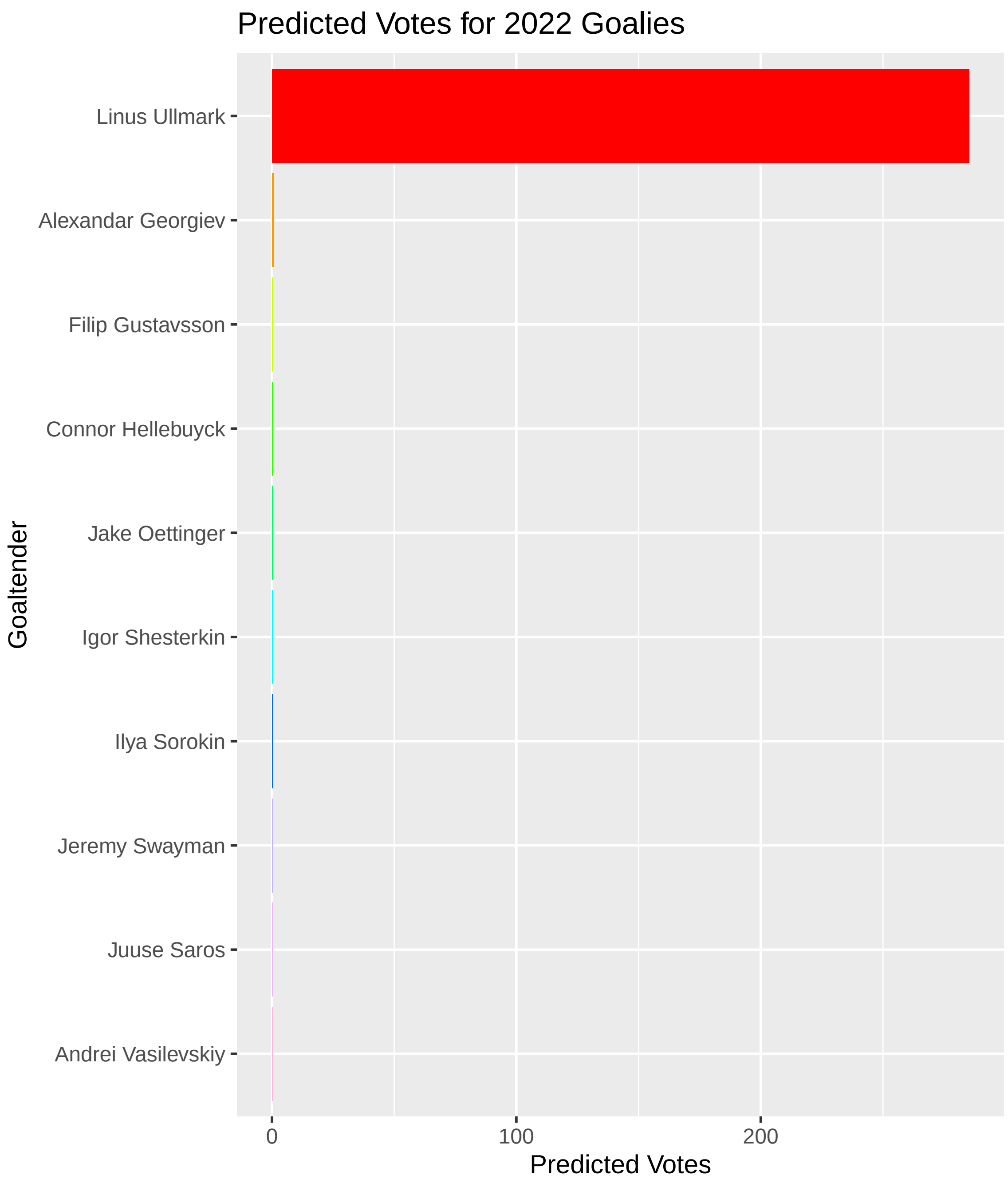
```
predict_year(2013,nb_model_1)
```



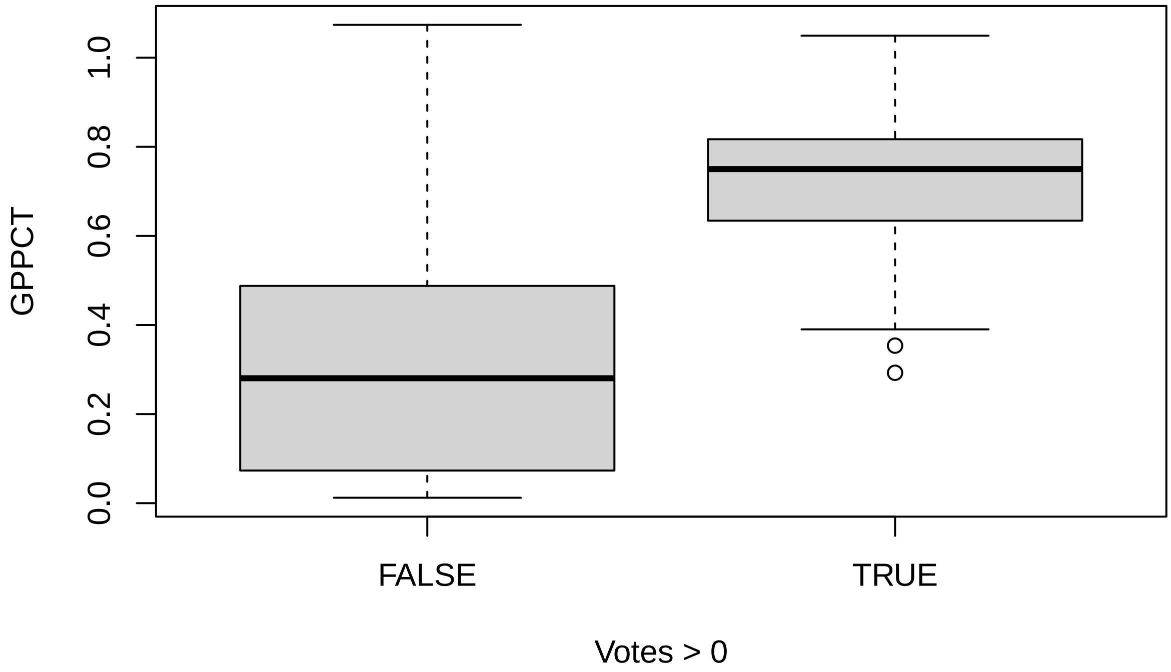
```
predict_year(2018,nb_model_1)
```



```
predict_year(2022,nb_model_1)
```



```r
#Data Expoloration

boxplot(GPPCT~Votes>0 , data = goalie_lagged)
```



```
goalies_lagged_contenders = subset(goalie_lagged, (GPPCT>0.28) & !is.na(W))
```

```
goalies_2022 = subset(goalies_lagged_contenders, Year == 2022)
goalies_not_2022 = subset(goalies_lagged_contenders, Year != 2022)
train_goalies = subset(goalies_lagged_contenders, ((Year!=2022) & (Year %% 2 == 0)))
test_goalies = subset(goalies_lagged_contenders, ((Year!=2022) & (Year %% 2 != 0)))

pois_model = glm(Votes ~ WGP + GPPCT + SVPCT + GAA, data = train_goalies, family = poisson)
summary(pois_model)
```

```
##
## Call:
## glm(formula = Votes ~ WGP + GPPCT + SVPCT + GAA, family = poisson,
##     data = train_goalies)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -15.7005   -1.4123   -0.4693   -0.1078   12.8131
##
## Coefficients:
##               Estimate Std. Error z value             Pr(>|z|)
## (Intercept) -151.5256     5.9330 -25.540 < 0.0000000000000002 ***
## WGP           11.3277     0.3903  29.024 < 0.0000000000000002 ***
## GPPCT          6.3699     0.1750  36.394 < 0.0000000000000002 ***
## SVPCT        153.2342     5.9678  25.677 < 0.0000000000000002 ***
## GAA            0.5325     0.1938   2.747              0.00601 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 8355.7  on 305  degrees of freedom
## Residual deviance: 2001.6  on 301  degrees of freedom
## AIC: 2249
##
## Number of Fisher Scoring iterations: 6
```

```
preds_df_22 = data.frame(Name = goalies_2022$Name, pred_votes = predict(pois_model, newdata = goalies_20
preds_df_22[order(preds_df_22$pred_votes, decreasing = T),]
```

```
##                      Name      pred_votes
## 1301        Linus Ullmark 592.32709674804
## 1394   Alexandar Georgiev  16.55797588172
## 1305    Connor Hellebuyck   9.11623188342
## 1302       Jake Oettinger   7.43967602821
## 1374     Filip Gustavsson   7.24983891928
## 1297      Igor Shesterkin   6.83973845906
## 1303         Ilya Sorokin   6.26544941875
## 1327          Juuse Saros   4.76845534851
## 1371   Andrei Vasilevskiy   4.00341889882
## 1304       Jeremy Swayman   3.64929777258
## 1306        Ilya Samsonov   2.90913230505
## 1345        Vitek Vanecek   1.59153391390
## 1310       Stuart Skinner   1.34438401363
## 1316       Logan Thompson   0.56521140829
## 1401         Antti Raanta   0.44903631650
## 1337        Tristan Jarry   0.36373927755
## 1356    Marc-Andre Fleury   0.29137871494
## 1334            Adin Hill   0.26947344255
## 1360     Joonas Korpisalo   0.21683765562
## 1384       Pheonix Copley   0.21239519862
## 1319        Darcy Kuemper   0.15505567958
## 1323    Frederik Andersen   0.15111119030
## 1378          Carter Hart   0.11429888229
## 1361     Sergei Bobrovsky   0.08276312392
## 1357          Ville Husso   0.05473664420
## 1314    Jordan Binnington   0.03864132341
## 1298       Karel Vejmelka   0.02520502313
## 1366        Casey DeSmith   0.02219047250
## 1341       Craig Anderson   0.02039662735
## 1351         Martin Jones   0.01660788139
## 1340        Jack Campbell   0.01451535017
## 1300       Thatcher Demko   0.01243700280
## 1368          John Gibson   0.01117645058
## 1347         Alex Stalock   0.01048926528
## 1362 Ukko-Pekka Luukkonen   0.00935094036
## 1369     Philipp Grubauer   0.00764626335
## 1358       Anton Forsberg   0.00719739898
## 1373     Charlie Lindgren   0.00667625914
## 1322           Jake Allen   0.00322747471
## 1363        Connor Ingram   0.00226375826
## 1359         James Reimer   0.00121542051
## 1390          Petr Mrazek   0.00103223135
## 1400       Jonathan Quick   0.00101846672
## 1329       Spencer Martin   0.00008497573
## 1311     Elvis Merzlikins   0.00004424917
```

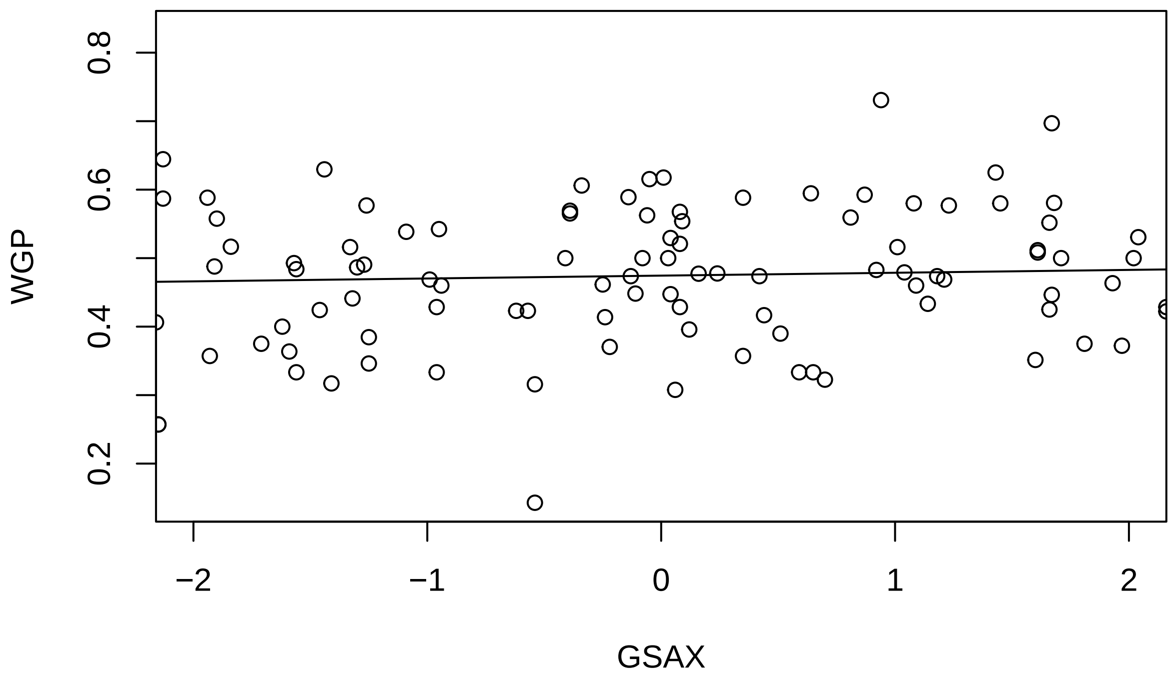
```
pred_year = function(year, model){
  pred_df = cbind((subset(goalies_not_2022, Year == year)[,c(2,3,4,5,42,44,46,60)]),
                  data.frame(pred_votes = predict(model, newdata = subset(goalies_not_2022, Year == yea
  pred_df$actual_votes = subset(goalies_not_2022, Year == year)$Votes
  pred_df[order(pred_df$pred_votes, decreasing = T),]
}

pred_year(2019, pois_model)
```

```
##                      Name Team GP Team_Wins lagged_flurry_GSAX
## 1033          Tuukka Rask  BOS 41        44              -6.55
## 1050   Andrei Vasilevskiy  TBL 52        43              10.99
## 1005    Connor Hellebuyck  WPG 58        37             -13.41
## 1079       Pavel Francouz  COL 34        42               0.48
## 1049       Anton Khudobin  DAL 30        37               1.60
## 1042    Jordan Binnington  STL 51        42               4.96
## 1008        Tristan Jarry  PIT 33        40              -4.13
## 1063        Darcy Kuemper  ARI 29        33               3.74
## 1013         Robin Lehner  VGK 36        39              11.79
## 1015          Carter Hart  PHI 42        41              -4.42
## 1002           Ben Bishop  DAL 44        37              12.31
## 1007    Frederik Andersen  TOR 52        36               4.29
## 1077       Jaroslav Halak  BOS 31        44               7.98
## 1026 Mackenzie Blackwood  NJD 47        28              -4.47
## 1048          Carey Price  MTL 58        31              -1.74
## 1056       Mikko Koskinen  EDM 38        37             -11.98
## 1003    Marc-Andre Fleury  VGK 49        39              -1.06
## 1025     Philipp Grubauer  COL 36        42              -5.70
## 1061         Antti Raanta  ARI 33        33              -1.96
## 1041        David Rittich  CGY 49        36               0.75
## 1004         Alex Stalock  MIN 38        35              -9.51
## 1009        Linus Ullmark  BUF 34        30             -18.07
## 1065     Elvis Merzlikins  CBJ 32        33                 NA
## 1062     Joonas Korpisalo  CBJ 37        33             -12.91
## 1035        Thomas Greiss  NYI 31        35               9.76
## 1076      Semyon Varlamov  NYI 45        35             -10.25
## 1028       Corey Crawford  CHI 40        32              -7.55
## 1038          Juuse Saros  NSH 40        35               1.32
## 1070   Alexandar Georgiev  NYR 34        37              -5.07
## 1053          Petr Mrazek  CAR 40        38               0.31
## 1027        Braden Holtby  WSH 48        41               4.42
## 1068     Sergei Bobrovsky  FLA 50        35               0.33
## 1043          John Gibson  ANA 52        29               4.83
## 1067           Mike Smith  EDM 39        37              -7.21
## 1014          Matt Murray  PIT 38        40              -2.90
## 1075       Thatcher Demko  VAN 27        36              -5.21
## 1059    Jonathan Bernier  DET 46        17              -9.84
## 1019       Brian Elliott  PHI 31        41              -2.41
## 1055      Jonathan Quick  LAK 42        29             -29.98
## 1029         Pekka Rinne  NSH 36        35               8.19
## 1069          Aaron Dell  SJS 33        29             -12.40
## 1024        Martin Jones  SJS 41        29             -28.79
## 1064       Jack Campbell  TOR 26        36               8.26
## 1001    Henrik Lundqvist  NYR 30        37             -21.23
## 1044      Craig Anderson  OTT 34        25             -22.57
## 1046       Carter Hutton  BUF 31        30             -13.92
## 1006        Devan Dubnyk  MIN 30        35             -20.77
##      lagged_flurryGSAXper60 GSAXper_lagged lagged_HDGSAX  pred_votes
## 1033           -0.149176304    -0.04630435         -3.27 9.605630099
## 1050            0.206879242     0.30735849          2.27 7.570920591
## 1005           -0.217227554    -0.10761905         -1.34 5.702251472
## 1079            0.470716426     0.28500000         -0.79 2.253969095
## 1049            0.043240971     0.13682927          2.97 1.627822827
## 1042            0.160024376     0.20531250         -0.25 1.264417703
## 1008           -2.064139942    -2.06000000          0.21 1.185609241
## 1063            0.069020669     0.14927273          0.67 1.163461793
## 1013            0.272625670     0.34391304          6.66 0.754981094
## 1015           -0.154509438    -0.04290323          4.99 0.681371192
## 1002            0.280463262     0.34391304          4.81 0.644993928
## 1007            0.073748651     0.20000000         -0.63 0.622225746
## 1077            0.209256588     0.29350000          4.30 0.587340240
## 1026           -0.212222720    -0.10565217         -0.47 0.384093010
## 1048           -0.026893469     0.05318182          2.53 0.349705593
## 1056           -0.241854623    -0.12545455          5.76 0.300663265
## 1003           -0.017602694     0.10065574         -0.94 0.262858579
## 1025           -0.170943019    -0.03513514         -3.61 0.257221467
## 1061           -0.171154126    -0.02000000          1.43 0.237830078
## 1041            0.017976032     0.12622222          6.51 0.191827695
## 1004           -0.545854592    -0.38904762         -5.44 0.184288117
## 1009           -0.516999666    -0.40621622         -0.52 0.167770007
## 1065                     NA             NA            NA 0.166437767
## 1062           -0.569447167    -0.40074074          2.04 0.150791646
## 1035            0.259872046     0.32069767          0.95 0.148770363
## 1076           -0.218049023    -0.09714286         -0.28 0.146420424
## 1028           -0.204713379    -0.08692308         -0.89 0.128394705
## 1038            0.046692148     0.13903226          2.55 0.124891457
## 1070           -0.162744871    -0.04424242          1.42 0.108994550
## 1053            0.007792697     0.07325000         -1.04 0.087214492
## 1027            0.077848120     0.19406780          3.13 0.062883920
## 1068            0.005598281     0.10758065         16.75 0.049328595
## 1043            0.089789005     0.22982759          5.16 0.047982260
## 1067           -0.180212456    -0.05238095          8.33 0.042629737
## 1014           -0.060386264     0.02900000         -1.56 0.033585343
## 1075           -0.585228868    -0.53888889          0.59 0.031188947
## 1059           -0.320887005    -0.16800000          7.63 0.022291423
## 1019           -0.107091279     0.03153846         -1.85 0.019682228
## 1055           -0.695273496    -0.52804348          2.00 0.018544698
## 1029            0.154541261     0.22392857          4.11 0.015737216
## 1069           -0.570865890    -0.42400000          2.13 0.013314501
## 1024           -0.485663544    -0.36435484          2.25 0.009475760
## 1064            0.311868104     0.34580645         -0.12 0.007658801
## 1001           -0.412382037    -0.30076923          0.27 0.006410753
## 1044           -0.486268126    -0.33860000         -4.74 0.005053695
## 1046           -0.294124207    -0.17340000         -5.05 0.004015860
## 1006           -0.325006629    -0.24388060         -1.69 0.001519601
##      actual_votes
## 1033           99
## 1050           31
## 1005          123
## 1079            0
## 1049            0
## 1042            1
## 1008            1
## 1063            1
## 1013            3
## 1015            0
## 1002            0
## 1007            0
## 1077            0
## 1026            0
## 1048            0
## 1056            0
## 1003            0
## 1025            0
## 1061            0
## 1041            0
## 1004            0
## 1009            0
## 1065            0
## 1062            0
## 1035            0
## 1076            0
## 1028            0
## 1038            0
## 1070            0
## 1053            0
## 1027            0
## 1068            0
## 1043            0
## 1067            0
## 1014            0
## 1075            0
## 1059            0
## 1019            0
## 1055            0
## 1029            0
## 1069            0
## 1024            0
## 1064            0
## 1001            0
## 1044            0
## 1046            0
## 1006            0
```

```
goalie_testing = subset(goalie_lagged, GP > 25)

lm1 = lm(WGP ~ GSAX, data = goalie_testing)

summary(lm1)
```

```
##
## Call:
## lm(formula = WGP ~ GSAX, data = goalie_testing)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.32936 -0.07049  0.00278  0.07140  0.25363
##
## Coefficients:
##              Estimate Std. Error t value            Pr(>|t|)
## (Intercept) 0.4744782  0.0040416  117.40 <0.0000000000000002 ***
## GSAX        0.0041786  0.0003122   13.38 <0.0000000000000002 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1002 on 615 degrees of freedom
##   (60 observations deleted due to missingness)
## Multiple R-squared:  0.2255, Adjusted R-squared:  0.2243
## F-statistic: 179.1 on 1 and 615 DF,  p-value: < 0.00000000000000022
```

```
plot(WGP ~ GSAX, data = goalie_testing, xlim = c(-2,2))
abline(lm1)
```



```
(24.2 * 0.0041786) * 0.4744782 * 58
```

```
## [1] 2.782854
```

```
17.7 / 5.6
```

```
## [1] 3.160714
```

```
#CREATING MY OWN WINS ABOVE REPLACEMENT VALUE FOR NHL GOALTENDERS
Complete_Data = subset(goalie_lagged, !is.na(W) & !is.na(Team_Wins))
Complete_Data = left_join(Complete_Data, GPW, by="Year")
Complete_Data$Goalie_WARs = (Complete_Data$GSAX / Complete_Data$Goals.Per.Win) * Complete_Data$W.TW

display_year = function(y){
  dta = subset(Complete_Data, Year == y)
  dta[order(dta$Goalie_WARs, decreasing=T), c(1,2,26,69)]
}


for (i in 1:length(unique(Complete_Data$Year))){
  print(display_year(unique(Complete_Data$Year)[i]))
}
```

```
##    Year                 Name Votes  Goalie_WARs
## 46 2016   Sergei Bobrovsky   138  5.100301205
## 37 2016      Braden Holtby    87  5.076054217
## 41 2016  Frederik Andersen     0  3.609375000
## 22 2016        Carey Price    19  2.674995194
## 6  2016         Mike Smith     0  2.287964357
## 44 2016        Matt Murray     0  2.015662651
## 40 2016        John Gibson     0  1.448729701
## 18 2016        Peter Budaj     0  1.398450947
## 1  2016      Kari Lehtonen     0  1.316774451
## 24 2016       Martin Jones     1  1.300582766
## 28 2016     Craig Anderson     0  1.132726930
## 34 2016     Corey Crawford     0  1.066265060
## 25 2016      Thomas Greiss     0  0.722248751
## 3  2016     Anders Nilsson     0  0.670294816
## 16 2016       Chad Johnson     0  0.579819277
## 33 2016      Scott Darling     0  0.534713855
## 35 2016       Antti Raanta     0  0.532128514
## 11 2016   Philipp Grubauer     0  0.512150876
## 20 2016       James Reimer     0  0.469556799
## 43 2016      Keith Kinkaid     0  0.396407057
## 14 2016     Roberto Luongo     0  0.322773236
## 17 2016        Pekka Rinne     0  0.229163606
## 9  2016      Brian Elliott     0  0.221887550
## 26 2016        Mike Condon     0  0.156891772
## 29 2016     Jaroslav Halak     0  0.119012636
## 23 2016 Andrei Vasilevskiy     0  0.112369119
## 2  2016       Robin Lehner     0  0.066915389
## 19 2016      Carter Hutton     0  0.060650210
## 5  2016  Marc-Andre Fleury     0  0.028463855
## 10 2016       Devan Dubnyk     8 -0.007683796
## 7  2016        Steve Mason     0 -0.013805221
## 27 2016         Ben Bishop     0 -0.021721501
## 30 2016        Tuukka Rask     0 -0.061738431
## 21 2016   Jonathan Bernier     0 -0.093676336
## 13 2016     Louis Domingue     0 -0.186370482
## 8  2016 Michael Hutchinson     0 -0.319371235
## 15 2016        Ryan Miller     0 -0.345632530
## 36 2016     Cory Schneider     0 -0.478700516
## 42 2016        Antti Niemi     0 -0.490343728
## 31 2016    Semyon Varlamov     0 -0.541141840
## 47 2016    Michal Neuvirth     0 -0.575571517
## 39 2016   Henrik Lundqvist     0 -0.723401418
## 4  2016  Connor Hellebuyck     0 -0.936088102
## 32 2016         Jake Allen     0 -0.987223350
## 45 2016           Cam Ward     0 -1.751171352
## 38 2016        Petr Mrazek     0 -1.823658269
## 12 2016     Calvin Pickard     0 -2.325780394
##    Year               Name Votes  Goalie_WARs
## 92 2017       Antti Raanta     0  4.247184670
## 79 2017   Sergei Bobrovsky     4  3.402961808
## 53 2017     Jonathan Quick     1  3.389451806
## 66 2017        Pekka Rinne   129  3.080560008
## 68 2017        John Gibson     0  2.998299440
## 94 2017         Mike Smith     0  1.877462029
## 56 2017  Marc-Andre Fleury     0  1.847042012
## 57 2017  Connor Hellebuyck    82  1.749355477
## 50 2017  Frederik Andersen    12  1.431036951
## 63 2017   Philipp Grubauer     0  1.104713125
## 89 2017     Roberto Luongo     1  1.095266775
## 54 2017     Corey Crawford     0  1.055291811
## 93 2017        Tuukka Rask     7  0.849337490
## 61 2017      Carter Hutton     0  0.824372919
## 78 2017    Semyon Varlamov     0  0.635139299
## 80 2017 Andrei Vasilevskiy    21  0.620796166
## 91 2017      Darcy Kuemper     0  0.589405219
## 58 2017        Ryan Miller     0  0.493693758
## 76 2017        Juuse Saros     0  0.383387991
## 52 2017      Brian Elliott     0  0.373473630
## 82 2017         Ben Bishop     0  0.243662547
## 87 2017     Anton Khudobin     0  0.062977397
## 49 2017       Martin Jones     0  0.003897116
## 51 2017       James Reimer     0 -0.039945440
## 81 2017         Aaron Dell     0 -0.154585607
## 70 2017       Alex Stalock     0 -0.314800381
## 59 2017        Mike Condon     0 -0.315944772
## 64 2017     Anton Forsberg     0 -0.371406977
## 48 2017      Kari Lehtonen     0 -0.374401514
## 55 2017      Keith Kinkaid     0 -0.401845816
## 88 2017     Anders Nilsson     0 -0.476956729
## 86 2017        Petr Mrazek     0 -0.492335672
## 69 2017      Braden Holtby     0 -0.597610829
## 60 2017        Matt Murray     0 -0.621258354
## 72 2017     Cory Schneider     0 -0.657992631
## 73 2017   Jonathan Bernier     0 -0.996166325
## 85 2017       Devan Dubnyk     0 -1.168485321
## 65 2017      Thomas Greiss     0 -1.258601492
## 83 2017       Chad Johnson     0 -1.449727202
## 84 2017   Henrik Lundqvist     0 -1.485603595
## 62 2017       Robin Lehner     0 -1.534216680
## 71 2017           Cam Ward     0 -1.564854508
## 90 2017     Jaroslav Halak     0 -1.625654159
## 75 2017      Scott Darling     0 -1.982874340
## 67 2017         Jake Allen     0 -2.385433643
## 77 2017        Carey Price     0 -3.156395302
## 74 2017     Craig Anderson     0 -3.711794344
##     Year               Name Votes Goalie_WARs
## 113 2018 Andrei Vasilevskiy   146  1.91031609
## 128 2018         Ben Bishop    64  1.85188076
## 131 2018        John Gibson     1  1.84606371
## 114 2018  Frederik Andersen     1  1.75080245
## 138 2018       Robin Lehner    17  1.53608936
## 129 2018        Pekka Rinne     8  1.49221762
## 98  2018      Braden Holtby     0  1.42306736
## 121 2018      Thomas Greiss     0  1.23186211
## 102 2018      Darcy Kuemper     9  1.05962829
## 117 2018     Jaroslav Halak     0  0.98266600
## 140 2018   Sergei Bobrovsky     4  0.97890587
## 130 2018  Marc-Andre Fleury     0  0.93170664
## 95  2018  Curtis McElhinney     0  0.77489220
## 124 2018  Jordan Binnington     9  0.65324385
## 101 2018      Jack Campbell     0  0.64468019
## 118 2018      David Rittich     0  0.57181208
## 141 2018        Carey Price     8  0.52051556
## 105 2018     Anton Khudobin     0  0.38915769
## 100 2018        Juuse Saros     0  0.29062941
## 104 2018        Petr Mrazek     0  0.27311708
## 106 2018        Matt Murray     0  0.17816589
## 127 2018      Casey DeSmith     0  0.02796421
## 96  2018        Carter Hart     0 -0.10722132
## 126 2018   Philipp Grubauer     0 -0.11480042
## 112 2018 Alexandar Georgiev     0 -0.11908091
## 135 2018         Mike Smith     0 -0.18866518
## 115 2018        Tuukka Rask     0 -0.21880564
## 125 2018     Pheonix Copley     0 -0.23924932
## 109 2018     Corey Crawford     0 -0.24577430
## 107 2018   Jonathan Bernier     0 -0.30830537
## 134 2018   Joonas Korpisalo     0 -0.42918114
## 137 2018    Semyon Varlamov     0 -0.46705130
## 120 2018     Anders Nilsson     0 -0.71639795
## 119 2018         Jake Allen     0 -0.77139780
## 110 2018       James Reimer     0 -0.78631204
## 133 2018      Carter Hutton     0 -0.88163514
## 122 2018  Connor Hellebuyck     0 -0.91437003
## 116 2018     Mikko Koskinen     0 -0.91882391
## 111 2018           Cam Ward     0 -1.16993951
## 136 2018      Linus Ullmark     0 -1.27364246
## 103 2018     Roberto Luongo     0 -1.54082774
## 123 2018   Henrik Lundqvist     0 -1.64010067
## 108 2018     Craig Anderson     0 -1.85020186
## 132 2018      Keith Kinkaid     0 -1.92501985
## 99  2018     Jonathan Quick     0 -2.33720623
## 97  2018       Devan Dubnyk     0 -2.55225024
## 139 2018       Martin Jones     0 -3.29588561
##     Year                Name Votes  Goalie_WARs
## 146 2019   Connor Hellebuyck   123 3.6137347312
## 161 2019         Tuukka Rask    99 1.8589453251
## 157 2019 Mackenzie Blackwood     0 1.1030757499
## 153 2019         Carter Hart     0 0.7166044614
## 175 2019    Jonathan Bernier     0 0.7159305003
## 178 2019       Darcy Kuemper     1 0.6599805888
## 170 2019      Anton Khudobin     0 0.6178705396
## 187 2019      Jaroslav Halak     0 0.6056292462
## 171 2019  Andrei Vasilevskiy    31 0.5503600099
## 176 2019        Antti Raanta     0 0.3081527014
## 159 2019      Corey Crawford     0 0.2686832740
## 186 2019     Semyon Varlamov     0 0.2086426029
## 151 2019        Robin Lehner     3 0.2002463728
## 163 2019         Juuse Saros     0 0.1434672089
## 156 2019    Philipp Grubauer     0 0.1304016268
## 165 2019   Jordan Binnington     1 0.0444839858
## 143 2019          Ben Bishop     0 0.0161585073
## 188 2019      Pavel Francouz     0 0.0008896797
## 184 2019  Alexandar Georgiev     0 -0.0065403482
## 174 2019      Mikko Koskinen     0 -0.0112532461
## 149 2019       Tristan Jarry     1 -0.0302491103
## 183 2019          Aaron Dell     0 -0.1170695791
## 180 2019    Elvis Merzlikins     0 -0.1514073115
## 142 2019    Henrik Lundqvist     0 -0.1562950851
## 185 2019      Thatcher Demko     0 -0.2595887703
## 172 2019         Petr Mrazek     0 -0.2645158269
## 179 2019       Jack Campbell     0 -0.3789541321
## 154 2019       Brian Elliott     0 -0.4569047826
## 167 2019      Craig Anderson     0 -0.5214234875
## 150 2019       Linus Ullmark     0 -0.6412811388
## 177 2019    Joonas Korpisalo     0 -0.7427477623
## 162 2019       Thomas Greiss     0 -0.7703101169
## 144 2019   Marc-Andre Fleury     0 -0.7785381878
## 168 2019       Carter Hutton     0 -0.7907473310
## 164 2019       David Rittich     0 -1.0047449585
## 181 2019          Mike Smith     0 -1.1101760123
## 145 2019        Alex Stalock     0 -1.1621759024
## 147 2019        Devan Dubnyk     0 -1.2549059481
## 173 2019      Jonathan Quick     0 -1.2673947724
## 169 2019         Carey Price     0 -1.4521294914
## 182 2019    Sergei Bobrovsky     0 -1.5107269954
## 160 2019         Pekka Rinne     0 -1.5812913066
## 155 2019        Martin Jones     0 -1.7846975089
## 152 2019         Matt Murray     0 -1.8282918149
## 158 2019       Braden Holtby     0 -1.8672424269
## 148 2019   Frederik Andersen     0 -2.0425563464
## 166 2019         John Gibson     0 -2.2113142717
##      Year               Name Votes  Goalie_WARs
## 216 2020  Andrei Vasilevskiy    99  2.817816470
## 191 2020   Connor Hellebuyck    13  2.764315204
## 219 2020   Marc-Andre Fleury     0  2.090827500
## 190 2020         Juuse Saros    10  1.599006375
## 226 2020          Mike Smith     2  1.553042542
## 195 2020      Thatcher Demko     0  0.846620309
## 201 2020       Jack Campbell     0  0.732364028
## 223 2020    Philipp Grubauer    36  0.723527056
## 233 2020     Igor Shesterkin     0  0.318049157
## 232 2020         Tuukka Rask     0  0.296992543
## 212 2020     Semyon Varlamov    11  0.286696733
## 217 2020        Robin Lehner     0  0.264270328
## 199 2020      Chris Driedger     0  0.201720299
## 229 2020          Jake Allen     0  0.186755819
## 204 2020        Ilya Sorokin     0  0.182305690
## 225 2020       Casey DeSmith     0  0.117403348
## 220 2020    Jonathan Bernier     0  0.078224641
## 227 2020   Jordan Binnington     0  0.009573386
## 203 2020       Darcy Kuemper     0 -0.016454257
## 205 2020         Carey Price     0 -0.131035721
## 224 2020    Elvis Merzlikins     0 -0.226570135
## 213 2020   Frederik Andersen     0 -0.316691028
## 202 2020       Ilya Samsonov     0 -0.360398093
## 215 2020       Linus Ullmark     0 -0.384491115
## 192 2020        Devan Dubnyk     0 -0.389655220
## 231 2020      Jaroslav Halak     0 -0.402898125
## 230 2020      Jake Oettinger     0 -0.408637900
## 208 2020       Thomas Greiss     0 -0.429290782
## 189 2020        James Reimer     0 -0.477173458
## 210 2020      Jonathan Quick     0 -0.542520365
## 211 2020         Pekka Rinne     0 -0.563980521
## 218 2020       Braden Holtby     0 -0.623882997
## 209 2020      Mikko Koskinen     0 -0.678718876
## 193 2020         John Gibson     0 -0.699420317
## 221 2020    Sergei Bobrovsky     0 -0.748470603
## 222 2020       Vitek Vanecek     0 -0.762280859
## 197 2020      Anton Khudobin     0 -0.937463417
## 207 2020 Mackenzie Blackwood     0 -1.130856220
## 194 2020         Carter Hart     0 -1.179321486
## 214 2020    Joonas Korpisalo     0 -1.240351822
## 196 2020         Matt Murray     0 -1.241678568
## 200 2020       Tristan Jarry     0 -1.426305142
## 228 2020      Kevin Lankinen     0 -1.816923951
## 198 2020       Brian Elliott     0 -2.080775444
## 206 2020        Martin Jones     0 -2.421981178
##     Year               Name Votes Goalie_WARs
## 257 2021    Igor Shesterkin   154  4.49620863
## 251 2021 Andrei Vasilevskiy    14  4.13769604
## 263 2021  Frederik Andersen    21  3.43419008
## 276 2021        Juuse Saros    32  3.32754925
## 270 2021   Sergei Bobrovsky     0  2.99543399
## 248 2021      Darcy Kuemper     0  2.63879132
## 253 2021  Connor Hellebuyck     0  2.35798234
## 274 2021      Tristan Jarry     1  1.79624130
## 267 2021     Jonathan Quick     0  1.77209941
## 235 2021       Ilya Sorokin    11  1.70905264
## 243 2021     Thatcher Demko     1  1.64981849
## 261 2021        Ville Husso     1  1.31569587
## 237 2021     Anton Forsberg     0  1.07416754
## 264 2021       Robin Lehner     0  0.61835943
## 246 2021   Elvis Merzlikins     0  0.56009282
## 275 2021      Linus Ullmark     0  0.55246956
## 250 2021         Mike Smith     0  0.45958015
## 240 2021     Jeremy Swayman     0  0.35645005
## 266 2021     Spencer Knight     0  0.24893813
## 268 2021       Antti Raanta     0  0.19521662
## 271 2021     Jake Oettinger     0  0.17762021
## 239 2021    Scott Wedgewood     0  0.08611889
## 273 2021    Semyon Varlamov     0  0.03603207
## 256 2021       James Reimer     0  0.01356994
## 241 2021      Casey DeSmith     0 -0.02823706
## 260 2021    Anthony Stolarz     0 -0.07077562
## 245 2021     Chris Driedger     0 -0.09903672
## 242 2021         Jake Allen     0 -0.16751404
## 249 2021      Jack Campbell     0 -0.25584487
## 277 2021 Alexandar Georgiev     0 -0.37578283
## 259 2021      Vitek Vanecek     0 -0.46401821
## 254 2021  Jordan Binnington     0 -0.57369782
## 244 2021    Dustin Tokarski     0 -0.63921539
## 238 2021        Carter Hart     0 -0.64968090
## 234 2021     Mikko Koskinen     0 -0.71782190
## 265 2021       Cal Petersen     0 -0.72546131
## 255 2021      Thomas Greiss     0 -0.80348303
## 262 2021       Martin Jones     0 -0.86116240
## 247 2021     Craig Anderson     0 -0.87924850
## 269 2021      Ilya Samsonov     0 -1.20064713
## 258 2021     Kevin Lankinen     0 -1.55737968
## 236 2021        John Gibson     0 -1.57696936
## 252 2021  Marc-Andre Fleury     0 -1.77188053
## 272 2021     Karel Vejmelka     0 -2.29765198
## 278 2021   Philipp Grubauer     0 -4.28397313
##     Year                 Name Votes  Goalie_WARs
## 295 2022          Juuse Saros     0  6.619962528
## 284 2022         Ilya Sorokin     0  5.220237476
## 282 2022        Linus Ullmark     0  4.730870874
## 279 2022      Igor Shesterkin     0  3.983068232
## 286 2022    Connor Hellebuyck     0  3.966645754
## 315 2022   Andrei Vasilevskiy     0  3.421044724
## 321 2022   Alexandar Georgiev     0  3.284982385
## 317 2022     Filip Gustavsson     0  2.122173908
## 287 2022        Ilya Samsonov     0  1.758134031
## 285 2022       Jeremy Swayman     0  1.595087153
## 280 2022       Karel Vejmelka     0  1.536425004
## 283 2022       Jake Oettinger     0  1.512287938
## 318 2022          Carter Hart     0  1.350499604
## 292 2022        Darcy Kuemper     0  1.015882924
## 308 2022     Joonas Korpisalo     0  0.983706749
## 304 2022    Marc-Andre Fleury     0  0.643710401
## 309 2022     Sergei Bobrovsky     0  0.639049729
## 319 2022       Pheonix Copley     0  0.432904655
## 306 2022       Anton Forsberg     0  0.411074421
## 288 2022       Stuart Skinner     0  0.406966451
## 301 2022        Vitek Vanecek     0  0.372027438
## 312 2022        Casey DeSmith     0  0.194807095
## 311 2022        Connor Ingram     0  0.142240101
## 323 2022         Antti Raanta     0  0.061952541
## 300 2022       Craig Anderson     0  0.049603658
## 297 2022            Adin Hill     0  0.049232294
## 291 2022       Logan Thompson     0  0.005941829
## 314 2022     Philipp Grubauer     0  0.002666442
## 302 2022         Alex Stalock     0 -0.078047714
## 293 2022           Jake Allen     0 -0.168448787
## 298 2022        Tristan Jarry     0 -0.240262084
## 316 2022     Charlie Lindgren     0 -0.261288881
## 294 2022    Frederik Andersen     0 -0.286279016
## 310 2022 Ukko-Pekka Luukkonen     0 -0.365778666
## 281 2022       Thatcher Demko     0 -0.532966389
## 320 2022          Petr Mrazek     0 -0.588306326
## 303 2022         Martin Jones     0 -0.733702808
## 322 2022       Jonathan Quick     0 -0.971630449
## 307 2022         James Reimer     0 -1.058648657
## 296 2022       Spencer Martin     0 -1.228604298
## 289 2022     Elvis Merzlikins     0 -1.306578401
## 299 2022        Jack Campbell     0 -1.384861992
## 313 2022          John Gibson     0 -1.393294222
## 290 2022    Jordan Binnington     0 -1.543977853
## 305 2022          Ville Husso     0 -1.922818177
```

```
by(Complete_Data$Goalie_WARs, Complete_Data$Year, summary)
```

```
## Complete_Data$Year: 2016
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -2.3258 -0.2529  0.1569  0.4783  0.8943  5.1003
## ------------------------------------------------------------
## Complete_Data$Year: 2017
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -3.71179 -0.82708 -0.03995  0.11106  0.95232  4.24718
## ------------------------------------------------------------
## Complete_Data$Year: 2018
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -3.29589 -0.83397 -0.11480 -0.07857  0.85330  1.91032
## ------------------------------------------------------------
## Complete_Data$Year: 2019
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -2.2113 -1.0575 -0.1563 -0.3062  0.2044  3.6137
## ------------------------------------------------------------
## Complete_Data$Year: 2020
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -2.4220 -0.6994 -0.3604 -0.1460  0.2643  2.8178
## ------------------------------------------------------------
## Complete_Data$Year: 2021
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -4.28397 -0.64968  0.03603  0.35578  1.31570  4.49621
## ------------------------------------------------------------
## Complete_Data$Year: 2022
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -1.9228 -0.3658  0.1422  0.7211  1.5123  6.6200
```

```
summaries = cbind(aggregate(Complete_Data$Goalie_WARs, by=list(Complete_Data$Year), min),
      aggregate(Complete_Data$Goalie_WARs, by=list(Complete_Data$Year), max)$x,
      aggregate(Complete_Data$Goalie_WARs, by=list(Complete_Data$Year), median)$x,
      aggregate(Complete_Data$Goalie_WARs, by=list(Complete_Data$Year), mean)$x)

colnames(summaries) = c("Year", "Min", "Max", "Median", "Mean")
summaries$Year = as.character(summaries$Year)

stargazer(t(summaries), type = 'latex')
```

```
##
## % Table created by stargazer v.5.2.3 by Marek Hlavac, Social Policy Institute. E-mail: marek.hlavac a
## % Date and time: Fri, May 05, 2023 - 11:54:13 PM
## \begin{table}[!htbp] \centering
##   \caption{}
##   \label{}
## \begin{tabular}{@{\extracolsep{5pt}} cccccccc}
## \\[-1.8ex]\hline
## \hline \\[-1.8ex]
## Year & 2016 & 2017 & 2018 & 2019 & 2020 & 2021 & 2022 \\
## Min & -2.325780 & -3.711794 & -3.295886 & -2.211314 & -2.421981 & -4.283973 & -1.922818 \\
## Max & 5.100301 & 4.247185 & 1.910316 & 3.613735 & 2.817816 & 4.496209 & 6.619963 \\
## Median &  0.15689177 & -0.03994544 & -0.11480042 & -0.15629509 & -0.36039809 &  0.03603207 &  0.14224
## Mean &  0.47830319 &  0.11106123 & -0.07856939 & -0.30621738 & -0.14597147 &  0.35577539 &  0.7210598
## \hline \\[-1.8ex]
## \end{tabular}
## \end{table}
```

```
dta = subset(Complete_Data, Year == 2022)
dta = head(dta[order(dta$Goalie_WARs, decreasing=T), c(2,69)], 5)
stargazer(t(t(dta)))
```

```
##
## % Table created by stargazer v.5.2.3 by Marek Hlavac, Social Policy Institute. E-mail: marek.hlavac
## % Date and time: Fri, May 05, 2023 - 11:54:13 PM
## \begin{table}[!htbp] \centering
##   \caption{}
##   \label{}
## \begin{tabular}{@{\extracolsep{5pt}} ccc}
## \\[-1.8ex]\hline
## \hline \\[-1.8ex]
##  & Name & Goalie\_WARs \\
## \hline \\[-1.8ex]
## 295 & Juuse Saros & 6.619963 \\
## 284 & Ilya Sorokin & 5.220237 \\
## 282 & Linus Ullmark & 4.730871 \\
## 279 & Igor Shesterkin & 3.983068 \\
## 286 & Connor Hellebuyck & 3.966646 \\
## \hline \\[-1.8ex]
## \end{tabular}
## \end{table}
```