

NHL Shots per Goal Model

Sasank Vishnubhatla

January 4th, 2019

Contents

Data Formatting	1
Shots Per Goal	2
Goalie Aspect	2
Skater Aspect	12
Positional Aspect	14
Locational Aspect	14
Player Analysis	14

Last Update: January 7th, 2019

GitHub Repo: [sv4u/nhl-shots-per-goal-model](https://github.com/sv4u/nhl-shots-per-goal-model)

In the NHL, goalies are arguably one of the most important team members. Most deep postseason runs rely on consistent goalie play. However, being able to compare goalies tends to be difficult, since consistency is extremely hard for a goalie. Few goalies have been consistent their entire career. Of the modern day goalies, very few exemplify extreme consistency: Henrik Lundqvist, Marc-Andre Fleury, Roberto Luongo and Jonathan Quick to name a few.

In this document, I'll be detailing an analysis and seeing how it pertains to consistency. The data used has been taken from MoneyPuck.

Data Formatting

So, let's load in our data:

```
data_2017 = read.csv("data/2017.csv")
```

We'll be starting with the 2017-2018 season. Since we have an extremely large amount of data, we need to clean it up. We'll only be looking at regular season games, initially, so we can start by subsetting our data on that.

```
regular_season_2017 = subset(data_2017, isPlayoffGame == 0)
```

Some important columns we need to keep are:

- xCord
- yCord
- xCordAdjusted
- yCordAdjusted
- shotAngle
- shotAngleAdjusted
- shotDistance
- playerPositionThatDidEvent
- goalieIdForShot
- goalieNameForShot
- shooterPlayerId

- shooterName
- game_id

We'll rename some of the columns, so here is a handy table of the old column names and the new column names.

Old Column Name	New Column Name
xCord	x
yCord	y
xCordAdjusted	x_adj
yCordAdjusted	y_adj
shotAngle	angle
shotAngleAdjusted	angle_adj
goal	goal
goalieIdForShot	goalie_id
goalieNameForShot	goalie_name
shooterPlayerId	skater_id
shooterName	skater_name
playerPositionThatDidEvent	pos
game_id	game

Now, let's create a new dataframe with just those columns.

```
analysis_2017 = data.frame(x = regular_season_2017$xCord,
  y = regular_season_2017$yCord,
  x_adj = regular_season_2017$xCordAdjusted,
  y_adj = regular_season_2017$yCordAdjusted,
  angle = regular_season_2017$shotAngle,
  angle_adj = regular_season_2017$shotAngleAdjusted,
  goal = regular_season_2017$goal,
  goalie_id = regular_season_2017$goalieIdForShot,
  goalie_name = regular_season_2017$goalieNameForShot,
  skater_id = regular_season_2017$shooterPlayerId,
  skater_name = regular_season_2017$shooterName,
  pos = regular_season_2017$playerPositionThatDidEvent,
  game = regular_season_2017$game_id)
```

With this `analysis_2017` dataframe, we can start looking at a proposed statistic: shots per goal.

Shots Per Goal

Shots per goal is a new statistic. Initially, when I had this idea, it was only for goalies. This article details an initial attempt at analyzing what a shot per goal means. In that initial analysis, I looked at several goalies and tried to determine if there was any significance in what a shot per goal meant. However, now, with further thought, we'll be looking at what a shot per goal means to more than just goalies.

Goalie Aspect

Let's start by looking at a few select goalies from the 2017-2018 season:

1. Braden Holtby
2. Marc-Andre Fleury

3. Connor Hellebuyck
4. Carey Price
5. Andrei Vasilevskiy
6. Robin Lehner
7. Antti Niemi
8. Jaroslav Halak
9. Peter Budaj
10. Jacob Markstrom

To start, we need a function that will take a goalie's name and return all their data. So, let's write our specified subset function:

```
get_goalie_data = function(data, name) {
  subset(data, goalie_name == name)
}
```

So, let's now start with Brayden Holtby:

```
holtby = get_goalie_data(analysis_2017, "Braden Holtby")
```

Shots per Goals for a Season

With this data, we can now calculate a few shot per goal stats. Let's start with the most basic: shots per goal for the entire season.

```
calculate_spg = function(data) {
  total_shots = length(data$goal)
  temp = subset(data, goal == 1)
  total_goals = length(temp$goal)
  if (total_goals == 0) {
    200
  } else {
    total_shots / total_goals
  }
}
```

Note that if a goalie has a shutout, the SPG will be 200. This is due to R using Inf and not throwing an error.

Discussion: The Shutout

When calculating SPG, we must decide what should be awarded for a shutout. R handles a division by 0 as infinity, as it should. However, this does not bode well with visualizations. Therefore, I have set the value to 200. In my research and sifting through data, I have noticed that no goalie has had any SPG above 175. Therefore, it would make sense to set a shutout as something above 200. However, in the future, I will be looking to keep a shutout as infinity so it tell the true story about the goalie.

Now with this function, let's calculate the total shots per goal for the 2017-2018 season for Braden Holtby.

```
holtby_spg_season = calculate_spg(holtby)
```

We see that Holtby gives up one goal per 15.3398693 shots. We can calculate the season shots per goal for each goaltender now.

```
fleury = get_goalie_data(analysis_2017, "Marc-Andre Fleury")
fleury_spg_season = calculate_spg(fleury)

hellebuyck = get_goalie_data(analysis_2017, "Connor Hellebuyck")
```

```

hellebuyck_spg_season = calculate_spg(hellebuyck)

price = get_goalie_data(analysis_2017, "Carey Price")
price_spg_season = calculate_spg(price)

vasilevskiy = get_goalie_data(analysis_2017, "Andrei Vasilevskiy")
vasilevskiy_spg_season = calculate_spg(vasilevskiy)

lehner = get_goalie_data(analysis_2017, "Robin Lehner")
lehner_spg_season = calculate_spg(lehner)

niemi = get_goalie_data(analysis_2017, "Antti Niemi")
niemi_spg_season = calculate_spg(niemi)

halak = get_goalie_data(analysis_2017, "Jaroslav Halak")
halak_spg_season = calculate_spg(halak)

budaj = get_goalie_data(analysis_2017, "Peter Budaj")
budaj_spg_season = calculate_spg(budaj)

markstrom = get_goalie_data(analysis_2017, "Jacob Markstrom")
markstrom_spg_season = calculate_spg(markstrom)

```

Let's take a look at what the results are:

Goalie	Shots per Goal
Fleury	19.08
Hellebuyck	18.3076923
Price	13.4864865
Vasilevskiy	16.6566265
Lehner	14.8661972
Niemi	15.952381
Halak	14.65625
Budaj	10.7777778
Markstrom	15.9868421

With each of these shots per goal calculations, we can begin to delve deeper into what a shot per goal (SPG) means for a goalie.

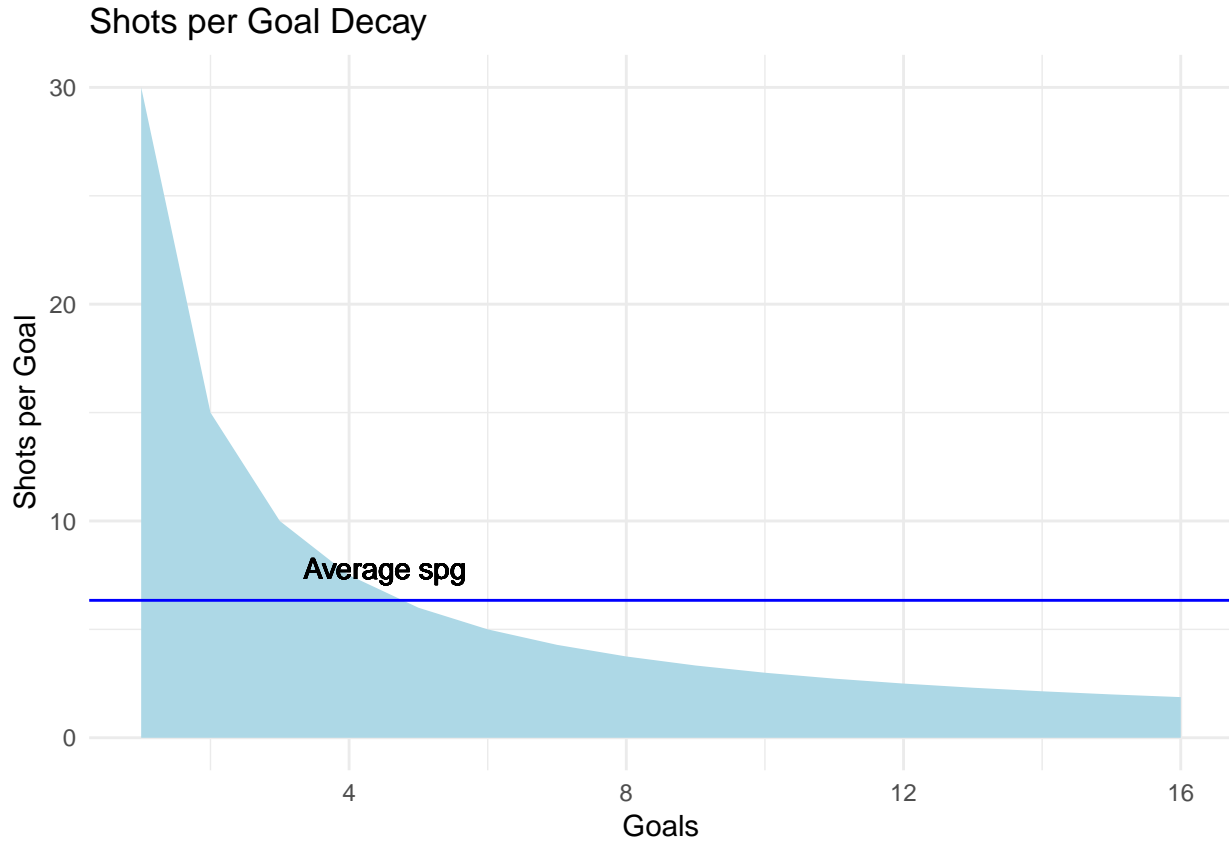
A Mathematical Aside

Ideally, during a hockey game, a goalie would like to have a shutout. A shutout, the best goalie performance possible, would equate to an infinite SPG. Now, let's continue this hypothetical exercise. Let us say that our goalie faces a total of 30 shots per game, a figure that now seems quite average in the NHL. With a shutout, the SPG is infinite. With one goal allowed, the SPG will drop down to 30. With another goal allowed, the spg drops to 15. Here is a table which depicts the decay of SPG as goals are allowed for a theoretical 30 shot game:

Goals Allowed	Shots per Goal (30 shots)
0	infinite
1	30
2	15

Goals Allowed	Shots per Goal (30 shots)
3	10
4	7.5
5	6
6	5

If one were to plot this trend, it would look like this:



Mathematically speaking, we are able to formulaically determine the equation of this curve. We see that this curve is:

$$spg(shots = 30, goals) = \frac{shots}{goals} = \frac{30}{goals}$$

Therefore, whenever we take the limit to determine the lower bound of this function, we see that the lowest possible SPG is:

$$\lim_{goals \rightarrow \infty} spg(shots = 30, goals) = \lim_{goals \rightarrow \inf} \frac{30}{goals} = 0$$

Therefore, we now know that the lowest possible SPG is 0.

Understanding Shots Per Goal

In the NHL, a goalie (and a team ideally) strives to lower his goals against average (GAA). However, given the nature of GAA, it can be misleading. Similar to a pitchers ERA, GAA is a general stat. It does not give

insight into his performance.

For example, a goalie could have a GAA of 3.0 on a given night, but saved 45 or 48 shots. A GAA of 3.0 is considered not very good in the current NHL, but the goalie played exceptionally well given that he had a 93.75% save percentage. Whenever GAA is paired with save percentage, we are given a better picture. However, this picture is still slightly misleading.

Now, let's look at the other extreme; a goalie has a GAA of 1.0 for a game, but only faced 20 shots. With this, we are looking at a GAA of 1.0 and a save percentage of 90.00%. The GAA says the goalie is elite, but the save percentage says the goalie is a benchwarmer. Pairing GAA and save percentage will only take an analysis so far, but there is more to look at.

If we add SPG into the picture, we might be able to gain more insight into the goalie's performance. Let's revisit our goalie with a 3.00 GAA and 93.75% save percentage. Since he faced a 48 shots and gave up 3 goals, he has a SPG of 16.0. Our goalie on average will save 16 shots before letting up a goal. On a good day, that means a period or two of shutout hockey. On a bad day, they let up a goal per period.

Now, let's revisit our GAA of 1.0 and save percentage of 90.00% goalie. His SPG, given he faced 20 shots and gave up 2 goals, is 10.0. So, every 10 shots, we expect a goal to be given up. Let me make a table to help understand the differences between the two goalies.

Goalie	GAA	Save Percentage	Shots Per Goal	Shots Faced
A	3.00	93.75%	16.0	48
B	1.00	90.00%	10.0	20

From here, we can determine which goalie is better. Categorically, goalie A wins as he has a better save percentage and SPG. Though goalie B has a better GAA, we know through SPG and save percentage, that goalie B will give up more goals given a higher shot volume.

Drawbacks of Shots per Goal

An immediate drawback of SPG is that it does not take the defense into account. However, both GAA and save percentage, also have this drawback. Taking defense out of the picture in a statistic may be not as useful as assessing the goalie by himself takes a large chunk of a team out of the statistic. Both GAA and save percentage reflect on the defensive play of team since a goalie can only do so much. A perfect example of this is Cam Talbot in 2016-2017. Cam Talbot started **73** games for the Edmonton Oilers and put up a 2.39 GAA and 91.9% save percentage (numbers taken from Hockey Reference). These numbers are not spectacular, but it does reflect on the team's defense: it was terrible. His SPG that season was 12.38. This SPG additionally reflects on his team's defensive play.

Additionally, SPG does not account for shot attempts, only shots on goal. There is a significant difference between the two, as a defense can be excellent at shot blocking. However, their goalie may not be as good, so the defense is forced to block shots more. This may lead to a slightly inflated SPG. In the future, more research can be done on this, leading to a new statistic: shot attempts per goal (SaPG). Currently, I do not have the data to calculate this. However, I do suspect that SaPG can be used not only as a metric to determine the effectiveness of a goalie, but also determine how good a defense is.

Determining Shots Per Goal for a Single Game

In our goalie data, we have a column called **game**. This column tells us what game we are looking at. So, we can subset on a specific game and determine the shots per goal that game. Let's write a few functions to help us do that.

```

get_games = function(data) {
  unique(data$game)
}

get_game_data = function(data, game_id) {
  subset(data, game == game_id)
}

```

With the use of some functional programming, we can create a list of dataframes for each game.

```

get_all_games = function(data) {
  games = get_games(data)
  Map(function(x) get_game_data(data, x), games)
}

```

Note that `get_all_games` doesn't return a vector, but a large list.

Now, with our `calculate_spg` function, we can get the SPG for each game. Here's a function that does that:

```

get_spg_games = function(data) {
  gameframes = get_all_games(data)
  gamespg = map(gameframes, function(x) calculate_spg(x))
  unlist(gamespg, use.names = FALSE)
}

```

Please note that the `map` function used here comes from the `purrr` library.

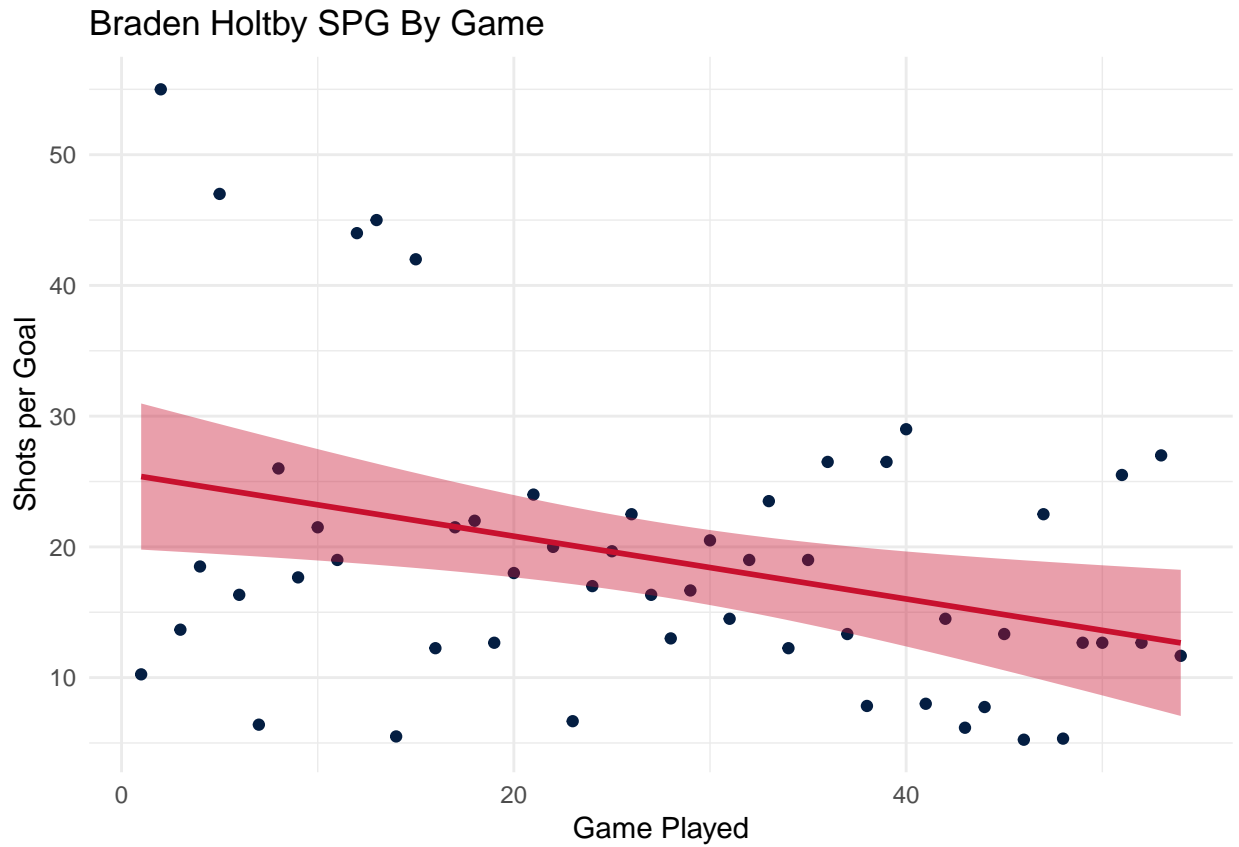
With `get_spg_games`, we can start looking at trends in performances. Let's start by first calculating SPG for each game for each of our select goalies.

```

holtby_spg_games = get_spg_games(holtby)
fleury_spg_games = get_spg_games(fleury)
hellebuyck_spg_games = get_spg_games(hellebuyck)
price_spg_games = get_spg_games(price)
vasilevskiy_spg_games = get_spg_games(vasilevskiy)
lehner_spg_games = get_spg_games(lehner)
niemi_spg_games = get_spg_games(niemi)
halak_spg_games = get_spg_games(halak)
budaj_spg_games = get_spg_games(budaj)
markstrom_spg_games = get_spg_games(markstrom)

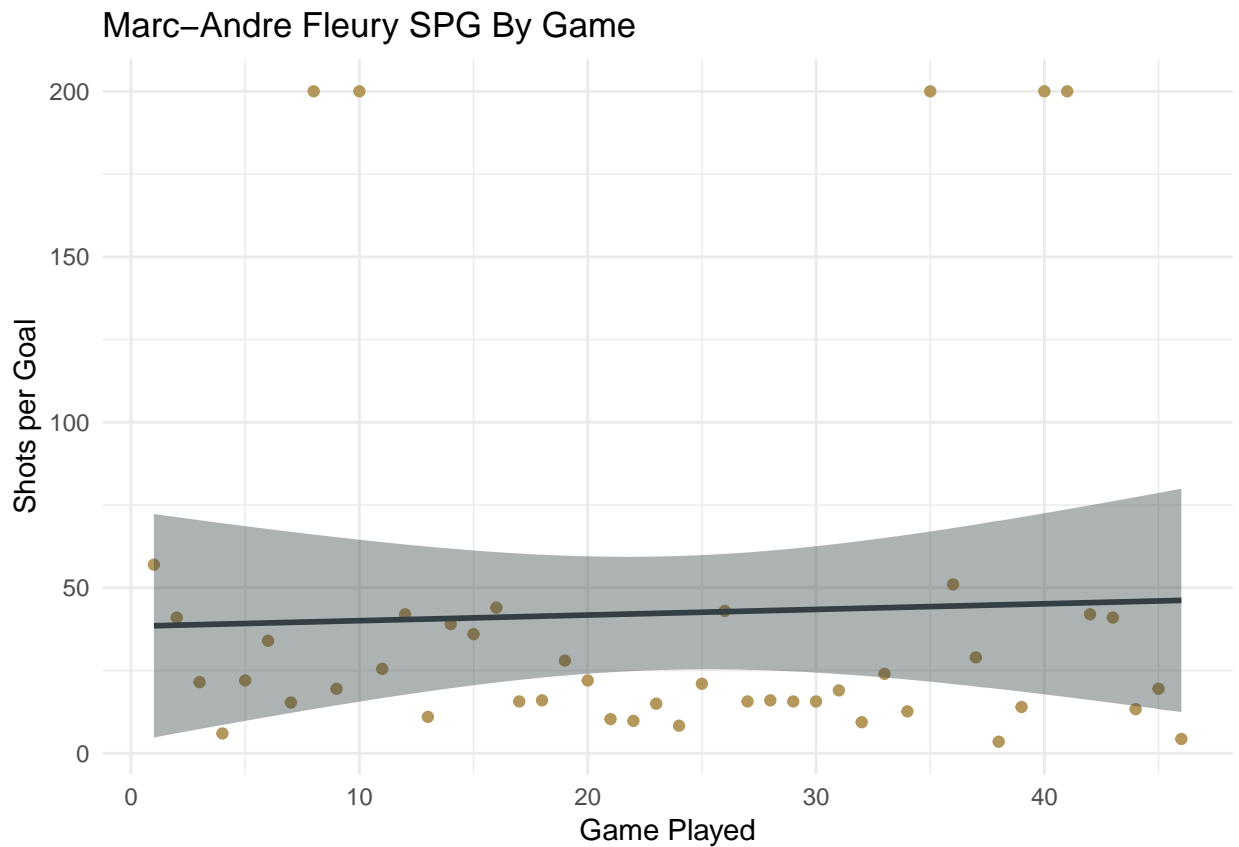
```

Let's take a look at these goalies: Holtby, Fleury, Price, Lehner, and Markstrom. Here's the trend of Braden Holtby's SPG by game.



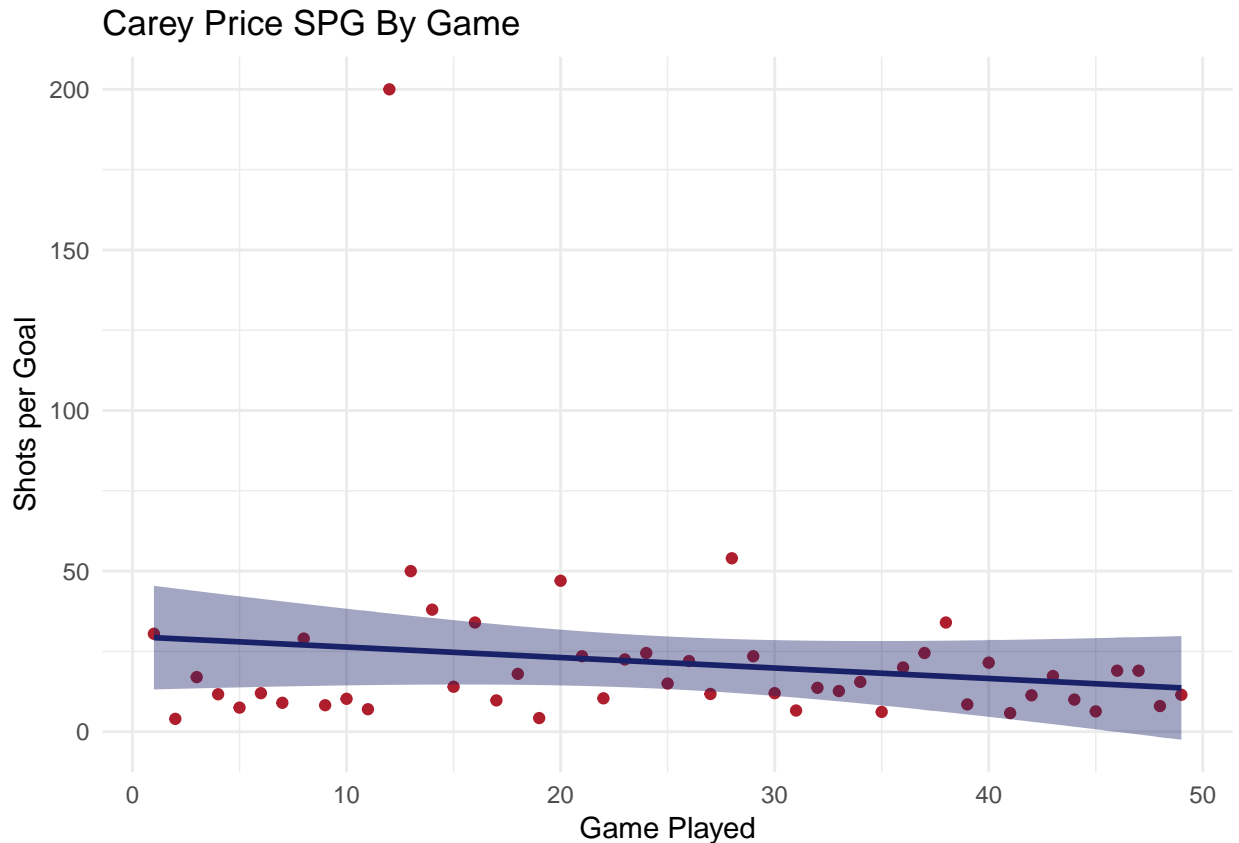
In this graph, we see that each point is Holtby's SPG for a specific game. The line is a linear model which shows that over the season, Holtby seemed to follow a decreasing linear trend.

Let's now look at someone who he faced in the Stanley Cup finals: Marc-Andre Fleury



Fleury's trend seems to be more stable, unlike Holtby. However, Fleury had 5 shutouts, which are seen as 200 instead as Inf. This leads to there being some minor skewing in the linear model.

A final player I'd like to look at is Carey Price. Price is supposed to be a Vezina caliber goaltender every year. However, injuries have set him back. Let's see how his trend is for the 2017-2018 season.



Similar to Holtby, Price has a decreasing trend in his SPG. However, Price's decreasing trend is less than Holtby's.

Shots per Goal from Specific Locations

In my initial attempt at analyzing shots per goal, I determined three danger zones in the attacking end: low, medium, and high. In the goalie dataframe, we've included x and y coordinates since these zones are in specific locations. With landmarks on the ice, like the slot and circles, I determined which were low, medium, and high danger. Here is how I determined:

Position	X Coordinate (Adj)	Y Coordinate (Adj)	Danger Level
Slot	From 40 to 90	From -22 to 22	High
Left Circle	From 32 to 82	From -43 to 7	Medium
Right Circle	From 32 to 82	From 7 to 43	Medium
Point	From 20 to 35	From -43 to 43	Low

I do know that this is not as accurate as it could be. There is significantly more areas that can be studied, such as high and low slot, net front, and the half-walls. That being said, this is not meant to be an in-depth analysis; instead I am looking for generic trends.

So, without further ado, let's start by writing a few subsetting functions to help us look at the trends in shots per goal.

```
get_low_danger = function(data) {
  subset(data, x_adj >= 20 & x_adj <= 35 & y_adj <= 43 & y_adj >= -43)
}
```

```

get_medium_danger = function(data) {
  subset(data, x_adj >= 32 & x_adj <= 82 & ((y_adj >= 7 & y_adj <= 43) | (y_adj <= -7 & y_adj >= -43))
}

get_high_danger = function(data) {
  subset(data, x_adj >= 40 & x <= 90 & y_adj >= -22 & y_adj <= 22)
}

```

With these functions, we can start looking at how a goalie performs when a shot is taken from a certain location. Let's start by writing a few more functions which will calculate the SPG for a certain danger level.

```

low_danger_spg = function(data) {
  calculate_spg(get_low_danger(data))
}

medium_danger_spg = function(data) {
  calculate_spg(get_medium_danger(data))
}

high_danger_spg = function(data) {
  calculate_spg(get_high_danger(data))
}

```

With these functions, let's compute the SPG for each danger level for each goalie now.

Low Danger Shots

```

holtby_low_spg = low_danger_spg(holtby)
fleury_low_spg = low_danger_spg(fleury)
hellebuyck_low_spg = low_danger_spg(hellebuyck)
price_low_spg = low_danger_spg(price)
vasilevskiy_low_spg = low_danger_spg(vasilevskiy)
lehner_low_spg = low_danger_spg(lehner)
niemi_low_spg = low_danger_spg(niemi)
halak_low_spg = low_danger_spg(halak)
budaj_low_spg = low_danger_spg(budaj)
markstrom_low_spg = low_danger_spg(markstrom)

```

Medium Danger Shots

```

holtby_medium_spg = medium_danger_spg(holtby)
fleury_medium_spg = medium_danger_spg(fleury)
hellebuyck_medium_spg = medium_danger_spg(hellebuyck)
price_medium_spg = medium_danger_spg(price)
vasilevskiy_medium_spg = medium_danger_spg(vasilevskiy)
lehner_medium_spg = medium_danger_spg(lehner)
niemi_medium_spg = medium_danger_spg(niemi)
halak_medium_spg = medium_danger_spg(halak)
budaj_medium_spg = medium_danger_spg(budaj)
markstrom_medium_spg = medium_danger_spg(markstrom)

```

High Danger Shots

```

holtby_high_spg = high_danger_spg(holtby)
fleury_high_spg = high_danger_spg(fleury)
hellebuyck_high_spg = high_danger_spg(hellebuyck)
price_high_spg = high_danger_spg(price)
vasilevskiy_high_spg = high_danger_spg(vasilevskiy)
lehner_high_spg = high_danger_spg(lehner)
niemi_high_spg = high_danger_spg(niemi)
halak_high_spg = high_danger_spg(halak)
budaj_high_spg = high_danger_spg(budaj)
markstrom_high_spg = high_danger_spg(markstrom)

```

Putting It All Together

Here's a table showing each of the goalies with their low, medium, and high SPG values.

Goalie	Low Danger SPG	Medium Danger SPG	High Danger SPG
Holtby	113.5	25.3396226	10.3475177
Fleury	200	28.5454545	13.2365591
Hellebuyck	51.8	25.739726	12.5271318
Price	98.5	20.483871	8.5725191
Vasilevskiy	99.5	23.6202532	11.6486486
Lehner	30.2857143	23.9642857	9.8991597
Niemi	45	22.3928571	12
Halak	47.25	20.4225352	11.4285714
Budaj	200	16.25	7.2
Markstrom	40.4	21.7857143	11.5692308

Note: a blank means that they have an infinite SPG i.e. gave up no goals from that zone

As we can see, there seems to be a distinction between elite goalies (like Holtby, Fleury, and Vasilevskiy) and struggling goalies (Niemi, Halak, and Markstrom).

Skater Aspect

Similar to goalies, SPG can be used on skaters. When SPG is used on a skater, we see that it is very similar to a strike rate. Let's start by first selecting 15 skaters: 4 centers, 3 left wingers, 3 right wingers, and 5 defenseman. I've chosen to look at the following:

- Centers
 - Sidney Crosby
 - Connor McDavid
 - Anze Kopitar
 - Christian Dvorak
- Left Wingers
 - Alex Ovechkin
 - Taylor Hall
 - Jujhar Khaira
- Right Wingers
 - Phil Kessel
 - Nikita Kucherov
 - Alex Tuch
- Defensemen

- Brent Burns
- John Klingberg
- Drew Doughty
- Olli Maatta
- Deryk Engelland

Let's start with the centers. We need a function to subset the skater data. Similar to `get_goalie_data`, we'll write a function to subset based on a skater's name.

```
get_skater_data = function(data, name) {
  subset(data, skater_name == name)
}
```

With this, we can use our `calculate_spg` function to calculate the SPG for a season. We've nicely written our `get_spg_games` function to work on any player, skater or goalie. Furthermore, our location functions will work on any player, skater or goalie. So, we can do a complete skater analysis without having to write any supplemental functions.

Centers

```
crosby = get_skater_data(analysis_2017, "Sidney Crosby")
mcdavid = get_skater_data(analysis_2017, "Connor McDavid")
kopitar = get_skater_data(analysis_2017, "Anze Kopitar")
dvorak = get_skater_data(analysis_2017, "Christian Dvorak")
```

Now, we can calculate SPG.

```
crosby_spg = calculate_spg(crosby)
mcdavid_spg = calculate_spg(mcdavid)
kopitar_spg = calculate_spg(kopitar)
dvorak_spg = calculate_spg(dvorak)
```

Let's take a look at what each player has done.

Player	SPG
Crosby	11.7586207
McDavid	8.6829268
Kopitar	7.5428571
Dvorak	14.1333333

As we can see, it takes Christian Dvorak 14.1333333 shots to score a goal, while Anze Kopitar takes about half of that with 7.5428571 shots in order to score a goal.

Let's now look at locational SPG.

```
crosby_low_spg = low_danger_spg(crosby)
crosby_medium_spg = medium_danger_spg(crosby)
crosby_high_spg = high_danger_spg(crosby)

mcdavid_low_spg = low_danger_spg(mcdavid)
mcdavid_medium_spg = medium_danger_spg(mcdavid)
mcdavid_high_spg = high_danger_spg(mcdavid)

kopitar_low_spg = low_danger_spg(kopitar)
kopitar_medium_spg = medium_danger_spg(kopitar)
```

```

kopitar_high_spg = high_danger_spg(kopitar)

dvorak_low_spg = low_danger_spg(dvorak)
dvorak_medium_spg = medium_danger_spg(dvorak)
dvorak_high_spg = high_danger_spg(dvorak)

```

Let's put it all in a table so we can compare.

Player	Low Danger SPG	Medium Danger SPG	High Danger SPG
Crosby	200	16.9	10.6538462
McDavid	200	9.8888889	8.9142857
Kopitar	8	8.4736842	7.3846154
Dvorak	4	17.2	14.2307692

As we can see here, Anze Kopitar is not a high volume shooter, while Crosby and Dvorak are. Using SPG, we can analyze player shooting trends and how it affects their scoring.

Left Wingers

Similar to the center analysis, we will just look at SPG over an entire season and SPG at danger locations.
To be done.

Right Wingers

Similar to the center analysis, we will just look at SPG over an entire season and SPG at danger locations.
To be done.

Defensemen

Similar to the center analysis, we will just look at SPG over an entire season and SPG at danger locations.
To be done.

Positional Aspect

To be done.

Locational Aspect

To be done.

Player Analysis

Matt Murray

Matt Murray is not even 25 yet and he has already won 2 Stanley Cups. In his young career, Murray has displayed phenomenal poise in net. However, he has also battled concussions and various injuries. This

season, Murray has somewhat struggled. Due to injury or something else, Murray was placed on injured reserve in November 2018 and was re-instated a month later. Since his comeback, Murray has been on fire. I want to see how Murray has grown as a goalie during his 2016-2017 campaign, 2017-2018 campaign, his current season.

Let's start with simply loading in the data. We already have the 2017-2018 data, so let's load in his 2016-2017 data.

```
data_2016 = read.csv("data/2016.csv")
regular_season_2016 = subset(data_2016, isPlayoffGame == 0)
analysis_2016 = data.frame(x = regular_season_2016$xCord,
                           y = regular_season_2016$yCord,
                           x_adj = regular_season_2016$xCordAdjusted,
                           y_adj = regular_season_2016$yCordAdjusted,
                           angle = regular_season_2016$shotAngle,
                           angle_adj = regular_season_2016$shotAngleAdjusted,
                           goal = regular_season_2016$goal,
                           goalie_id = regular_season_2016$goalieIdForShot,
                           goalie_name = regular_season_2016$goalieNameForShot,
                           skater_id = regular_season_2016$shooterPlayerId,
                           skater_name = regular_season_2016$shooterName,
                           pos = regular_season_2016$playerPositionThatDidEvent,
                           game = regular_season_2016$game_id)

murray_2016 = get_goalie_data(analysis_2016, "Matt Murray")
murray_2017 = get_goalie_data(analysis_2017, "Matt Murray")
```

With this data, we can select Murray's data and start looking at his trends. Let's get all his import SPG data now.

```
murray_spg_2016 = calculate_spg(murray_2016)
murray_spg_2017 = calculate_spg(murray_2017)

murray_spg_games_2016 = get_spg_games(murray_2016)
murray_spg_games_2017 = get_spg_games(murray_2017)

murray_low_spg_2016 = low_danger_spg(murray_2016)
murray_medium_spg_2016 = medium_danger_spg(murray_2016)
murray_high_spg_2016 = high_danger_spg(murray_2016)

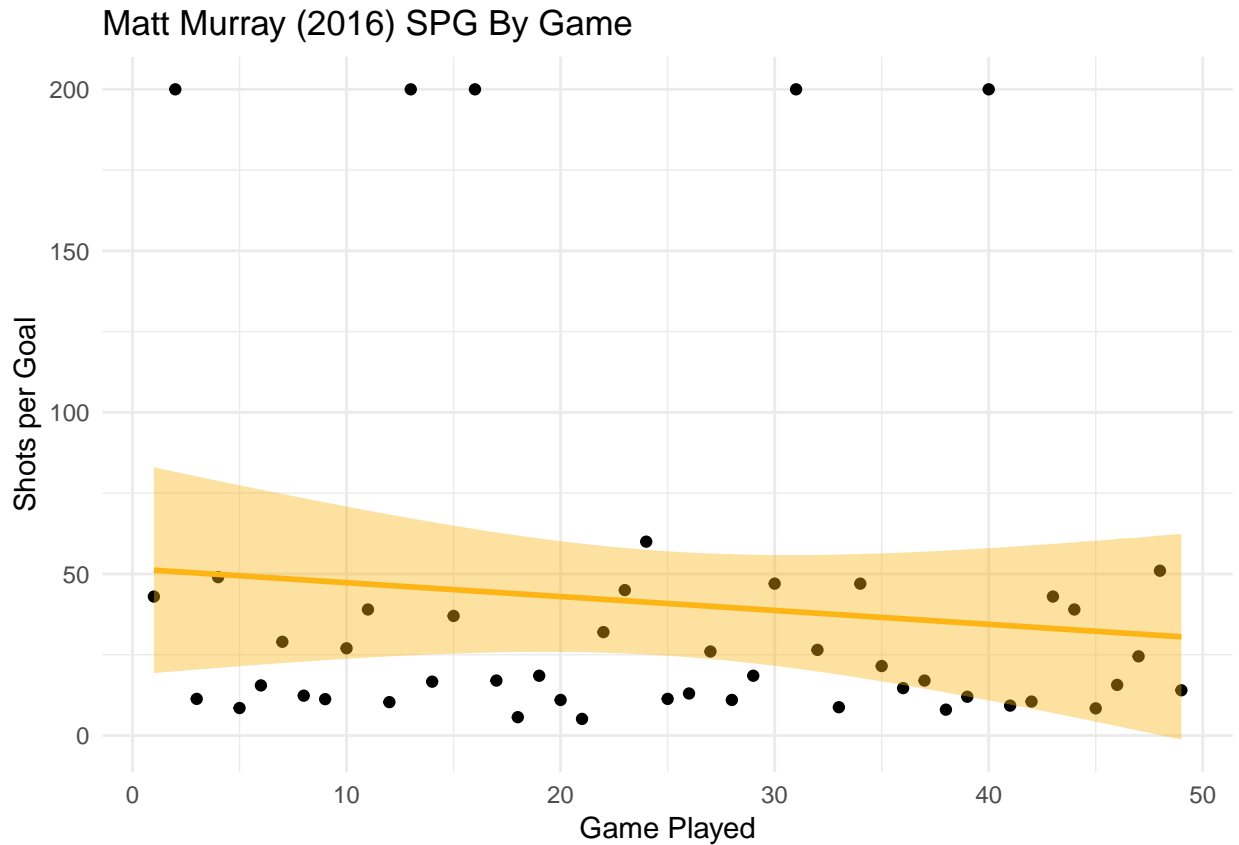
murray_low_spg_2017 = low_danger_spg(murray_2017)
murray_medium_spg_2017 = medium_danger_spg(murray_2017)
murray_high_spg_2017 = high_danger_spg(murray_2017)
```

With all these calculations, we can start to look at some trends. Here is a table so we can inspect the data:

Season	SPG	Low Danger SPG	Medium Danger SPG	High Danger SPG
2016	17.5315315	200	30.675	10.2231405
2017	14.3233083	116	17.5211268	10.2231405

From this, we can see that Murray has always been strong against low danger shots. However, there is a downward trend in his total SPG as well as medium danger SPG shots. With a difference of 13.1538732, there should be some concern for Murray.

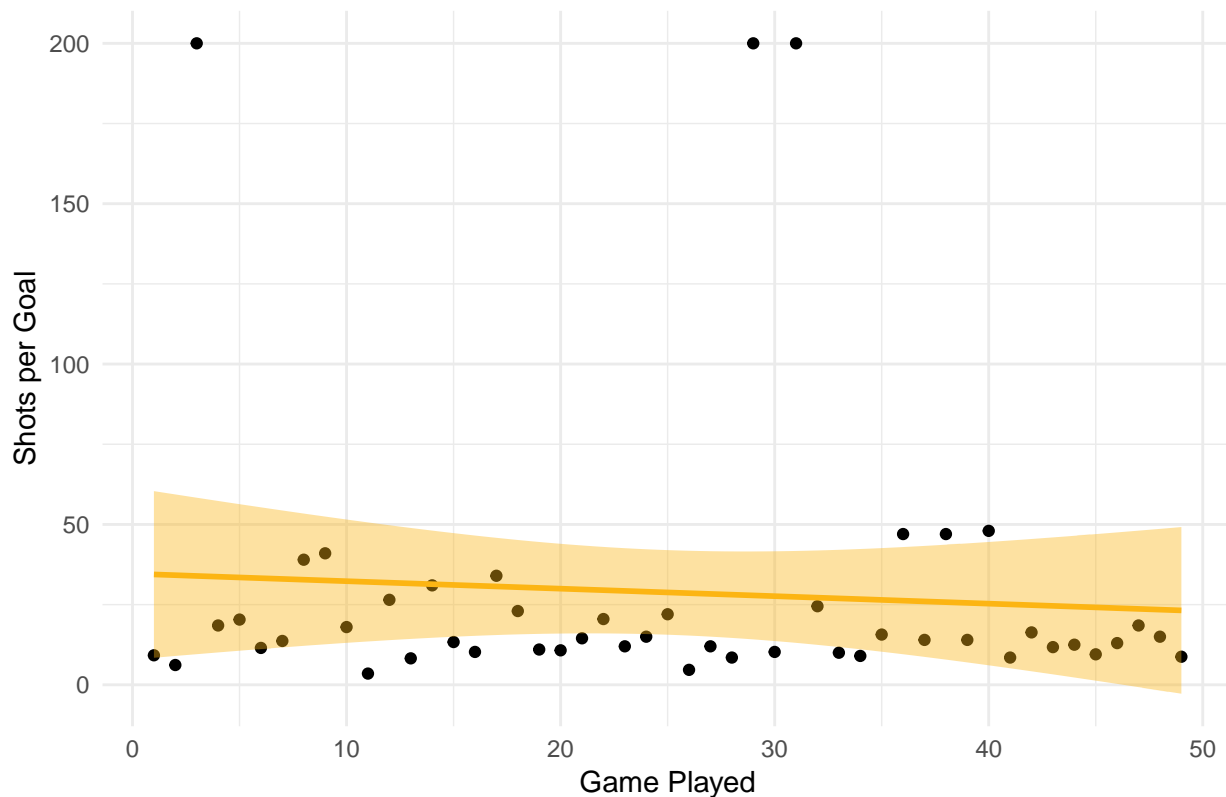
Let's now take a look at his trends for each game he played for both season. First, we'll look at 2016-2017.



We can see that Murray followed most trends that other goalies did: a decreasing trend. However, I believe this trend is slightly skewed as Murray did have 5 shutouts and therefore 5 outliers in the data. That being said, The shutouts are pretty evenly spaced in the data, skewing the data up completely. This does not discount the fact that Murray performed exceptionally well that season.

Let's now take a look at 2017-2018.

Matt Murray (2017) SPG By Game



Unlike his 2016-2017 season, Murray struggled for stretches, but then had elite play for other stretches. With 3 shutouts, the data is slightly skewed, but not by that much. Murray in 2017 almost had a stable neutral trend, but still tailed off near the end of the season.

Murray's change in performance can be attributed to multiple factors. First, personal issues during the 2017-2018 season. Second, injuries plagued him the entire season. Finally, a *theoretical* sophomore slump.

Let's now start looking at 2018-2019 data. This data, is incomplete as the season isn't over.

Note: this data contains all shots up to 1/7/19

```
data_2018 = read.csv("data/2018.csv")
regular_season_2018 = subset(data_2018, isPlayoffGame == 0)

analysis_2018 = data.frame(x = regular_season_2018$xCoord,
                           y = regular_season_2018$yCoord,
                           x_adj = regular_season_2018$xCoordAdjusted,
                           y_adj = regular_season_2018$yCoordAdjusted,
                           angle = regular_season_2018$shotAngle,
                           angle_adj = regular_season_2018$shotAngleAdjusted,
                           goal = regular_season_2018$goal,
                           goalie_id = regular_season_2018$goalieIdForShot,
                           goalie_name = regular_season_2018$goalieNameForShot,
                           skater_id = regular_season_2018$shooterPlayerId,
                           skater_name = regular_season_2018$shooterName,
                           pos = regular_season_2018$playerPositionThatDidEvent,
                           game = regular_season_2018$game_id)
```

With `analysis_2018`, we can start looking at Murray's data.

```

murray_2018 = get_goalie_data(analysis_2018, "Matt Murray")

murray_spg_2018 = calculate_spg(murray_2018)
murray_spg_games_2018 = get_spg_games(murray_2018)

murray_low_spg_2018 = low_danger_spg(murray_2018)
murray_medium_spg_2018 = medium_danger_spg(murray_2018)
murray_high_spg_2018 = high_danger_spg(murray_2018)

```

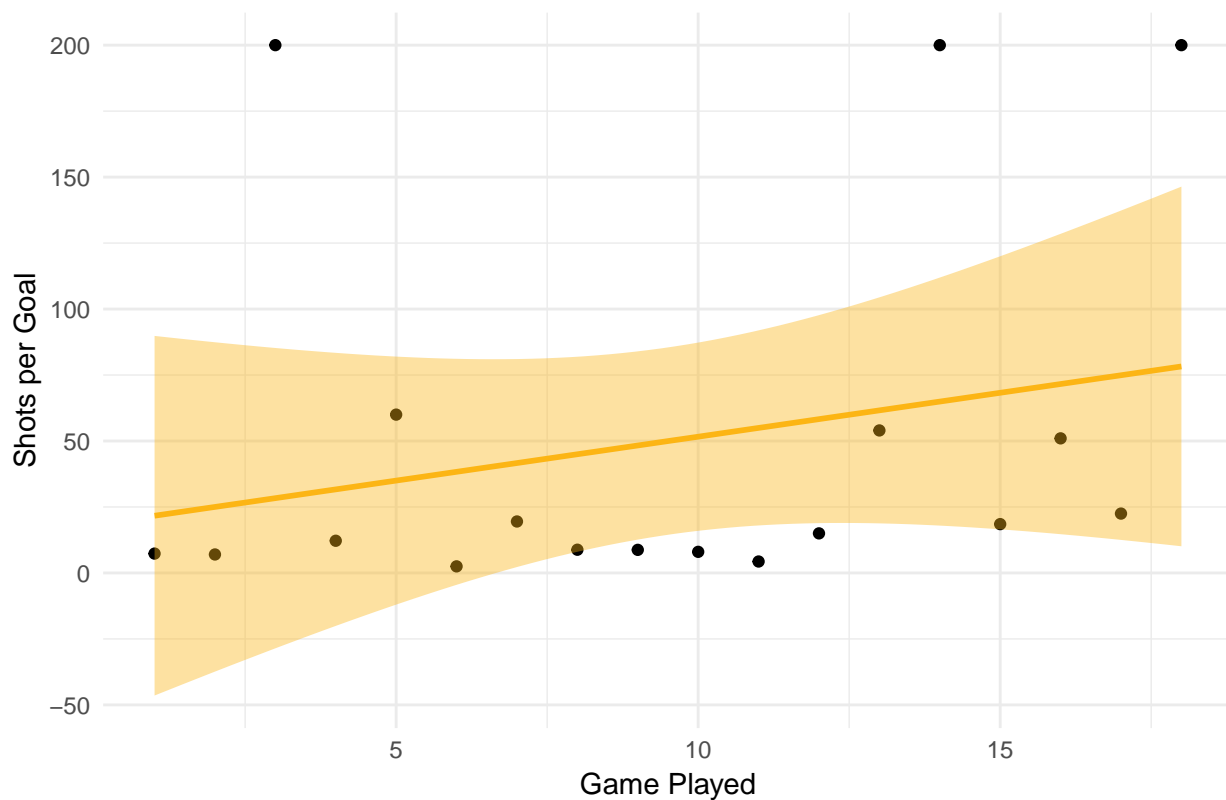
Let's compare all of his data together now.

Season	SPG	Low Danger SPG	Medium Danger SPG	High Danger SPG
2016	17.5315315	200	30.675	10.2231405
2017	14.3233083	116	17.5211268	10.2231405
2018	15.9791667	43	26.8333333	10.4565217

This 2018-2019 season has been an interesting one for Murray. With a steep decline in low danger SPG, it seems like Murray is unable to save shots from the point on a consistent basis. However, there is improvement: his medium danger SPG has risen close to his excellent 2016 season. Additionally, his total SPG has increased as he improves his game.

Let's take a look at his trend now.

Matt Murray (2018) SPG By Game



Though Murray hasn't played many games, we can see improvement. With 3 shutouts this season, we see Murray's trendline is somewhat skewed. That being said, this season seems more promising than last year, as Murray seems to finally be healthy at the right time.

Casey DeSmith

At 27-years-old, Casey DeSmith is a rookie. However, even though he has broken into the NHL late, he has been nothing but money for the Penguins. Having just played 40 games in his entire NHL career, DeSmith looks like a veteran when playing in net. His freakish athleticism helps him make superb saves. However, he is currently serving as backup to Matt Murray. The Penguins are in an interesting position as both DeSmith and Murray have been playing exceptionally as of late (1/7/19). So, let's take a look at DeSmith's numbers from last season, when he played 14 games.

```
desmith_2017 = get_goalie_data(analysis_2017, "Casey DeSmith")

desmith_spg_2017 = calculate_spg(desmith_2017)
desmith_spg_games_2017 = get_spg_games(desmith_2017)

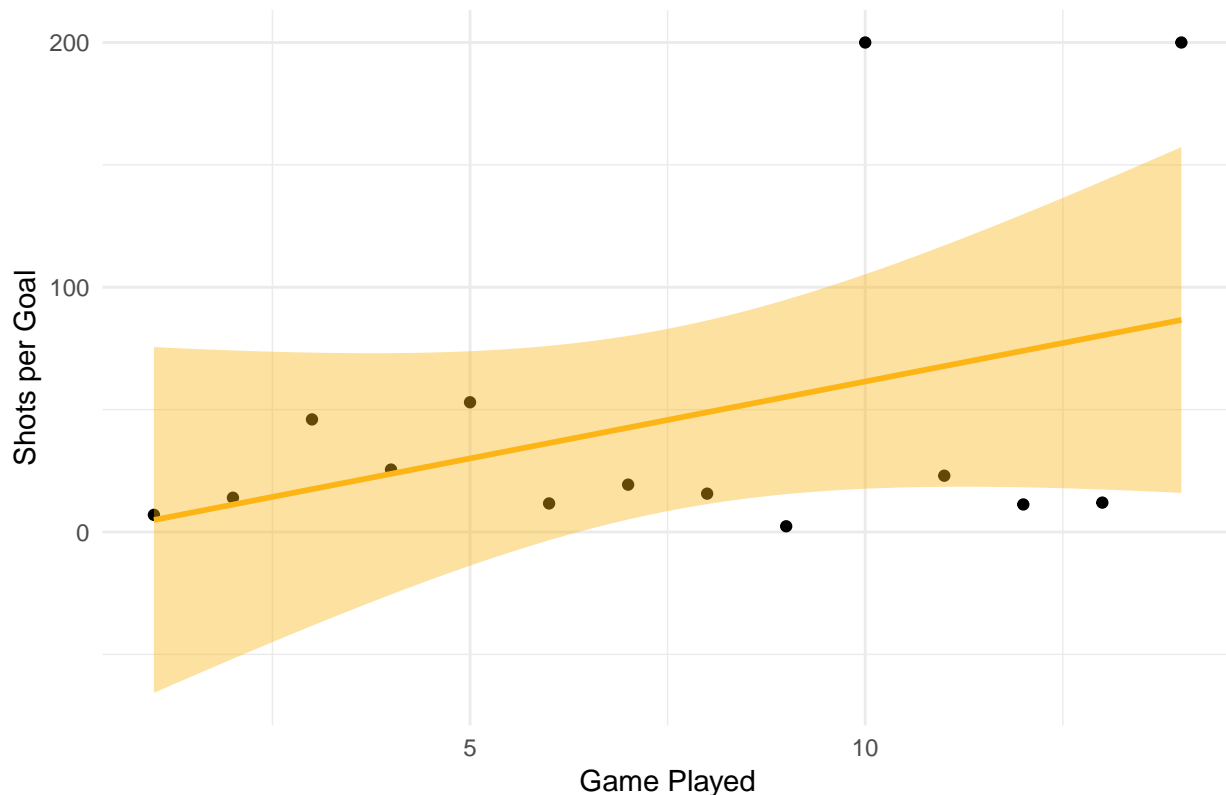
desmith_low_spg_2017 = low_danger_spg(desmith_2017)
desmith_medium_spg_2017 = medium_danger_spg(desmith_2017)
desmith_high_spg_2017 = high_danger_spg(desmith_2017)
```

With this data, we can now take a look at his performance.

Season	SPG	Low Danger SPG	Medium Danger SPG	High Danger SPG
2017	17.4642857	200	39.625	11.6923077

With a SPG of 17.4642857, DeSmith should surely be in conversation as a potential starting goalie for a rebuilding team. Let's see how his trends look like:

Casey DeSmith (2017) SPG By Game



Though there are only 14 games played by DeSmith, he shows something exceptionally rare: an increasing

trend in SPG. This trend however is slightly skewed by his two shutout performances near the end of the season.

It is easy to overlook DeSmith due to his age, but he has pulled his weight. Let's now start looking at 2018-2019 data. This data, is incomplete as the season isn't over.

Note: this data contains all shots up to 1/7/19

```
desmith_2018 = get_goalie_data(analysis_2018, "Casey DeSmith")

desmith_spg_2018 = calculate_spg(desmith_2018)
desmith_spg_games_2018 = get_spg_games(desmith_2018)

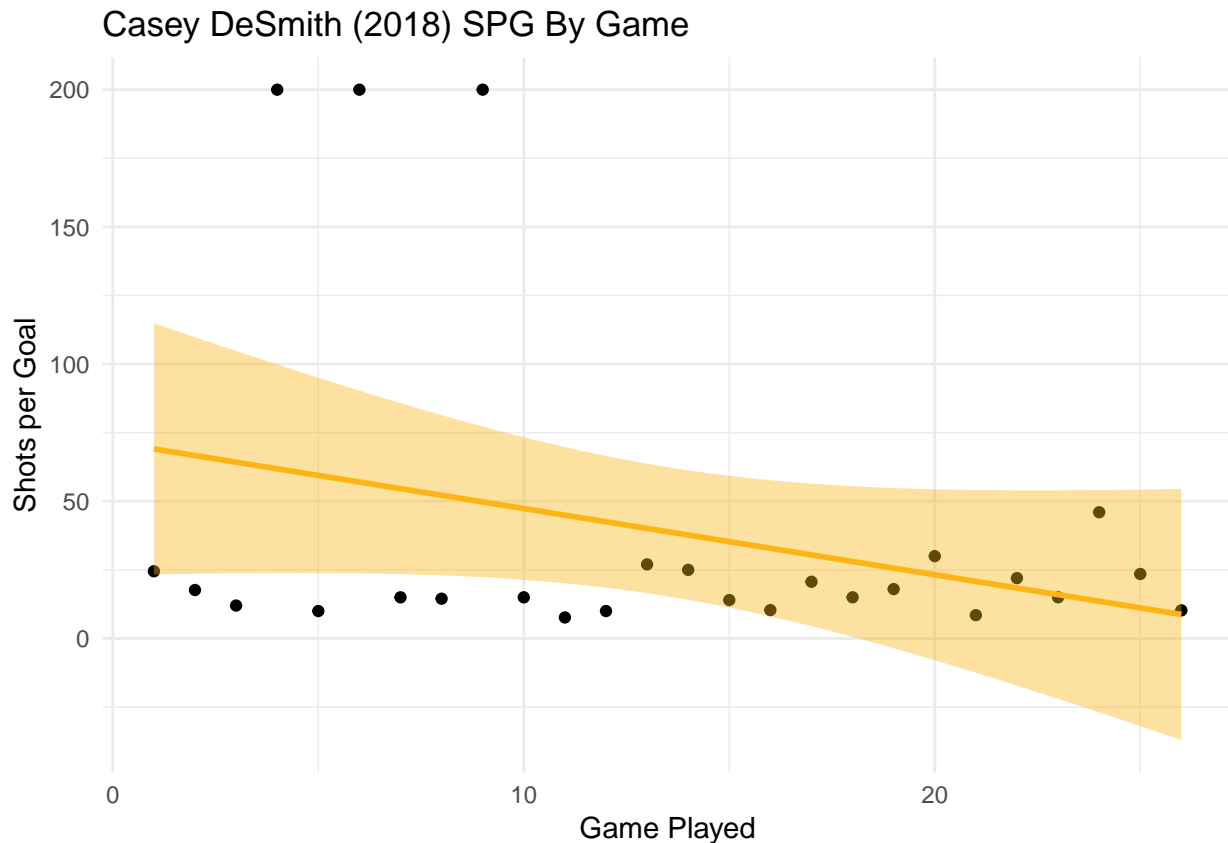
desmith_low_spg_2018 = low_danger_spg(desmith_2018)
desmith_medium_spg_2018 = medium_danger_spg(desmith_2018)
desmith_high_spg_2018 = high_danger_spg(desmith_2018)
```

Let's look at both seasons together now.

Season	SPG	Low Danger SPG	Medium Danger SPG	High Danger SPG
2017	17.4642857	200	39.625	11.6923077
2018	17.3275862	73	19.6969697	12.875

We can see there are some improvement and falters for DeSmith. A major falter for DeSmith is his SPG against low danger shots. As he sees more shots, his data should normalize towards that of a backup goaltender. However, that hasn't happened yet, which is quite peculiar.

Let's look at his trend for the current season:



With three early season shutouts, DeSmith's trend is skewed to his detriment. If we remove those three data points, we see that DeSmith is actually on a slight up-swing. With excellent play like this, it would be hard to deny the fact that he is performing well above his expectation.