Poker Adjusted Scoring Explanation

The main problem this scoring system hopes to solve is the flawed nature of not just the 9-6-3-1 tournament-style scoring, but any additive system. The fact remains that whatever intervals are used to number the ranks, there is no penalty for inconsistency, which in this game manifests itself as luck.

For example, the following situation is set up.

A	B	C	D	E	F	G	H
GAME:	**1**	**2**	**3**	**4**	**5**	**6**	**Total**
Player 1	9	9	-	-	-	-	18
Player 2	6	6	6	-	-	-	18
Player 3	3	3	3	3	-	-	12
Player 4	1	1	1	1	1	1	6
Count	1	1	1	1	1	1	

In this case, Player 1 and Player 2 are tied for the highest number of points, and player 3 is below them, although if we go by consistency, players 2 and 3 should be at the top.

In order to account for consistent play while still regarding the highest scorers, a multiplicative system is better. The new system, works as such:

E	F	G	H	I	J	K
4	**5**	**6**	**Total**	**G Mean**	**G*Cashes**	**Adjusted Score**
-	-	-	18	2.080084	4.160168	17.65
-	-	-	18	2.44949	7.348469	31.18
3	-	-	12	2.080084	8.320335	28.82
1	1	1	6	1	6	14.7

First, in column "I" the geometric mean of the numbers is taken (which is why a dash is used in place of a zero).

Code: I3 =PRODUCT(B3:G3)^(1/COUNT(B$7:G$7))
The "COUNT" uses a constant row in order to show how many games have been played.

Notice in column "I" the person who has all fourth place cashes only has 1 point. This is fixed in column "J," where each value is multiplied by the number of cashes.

Code: J3 =I3*COUNT(B3:G3)
This count is simply for the number of times a number appears in that player's row.

E	F	G	H	I	J	K
4	**5**	**6**	**Total**	**G Mean**	**G*Cashes**	**Adjusted Score**
-	-	-	18	2.080084	4.160168	17.65
-	-	-	18	2.44949	7.348469	31.18
3	-	-	12	2.080084	8.320335	28.82
1	1	1	6	1	6	14.7

Column “K” is where the magnitude of a person’s total score gets taken into account. In Column “J”, the leader is currently Player 3, with four third-place cashes. However, most people would agree that Player 2, having higher-placed finishes, and only one less of them, should be in the lead. Thus, the following formula is used to get the final score:

Code: K3 =J3*H3^(1/n)
The “n” value can change depending on how much the group of players values consistency over just a high score. The higher the “n” value, the less important the total score. For both to be mathematically weighted evenly, “n” would have to be the total number of games divided by the average number of cashes per player. Although this would vary depending on the number of players, changing this value does not affect the outcome to a great extent. In this example, I have used “2” because our group seems to favor consistency a little more. Remember that in any case, the effect of this value is relatively small.

The final score is then rounded to two decimal places for neatness.

The JavaScript version of this code is available in the GitHub repository as “pokerscoring.js”.