Rounding to integer better than submitting floats?

15

posted in Web Traffic Time Series Forecasting 6 months ago

We know that visits per page must be a non negative integer. If your algorithm spits out a floating point number, we know it must not be right. A better answer **must** be an integer above or below it but not the float number itself. What should we do?

While we may leave the floating point as is for submission, it bloats the submission file size significantly. Therefore I thought it might be a good idea to convert it to integers first instead, knowing that it might also improve the score if done properly.

My first thought was since the SMAPE function is asymmetric, which penalizes the score less when we overestimate than underestimate, I tried my best floating point submission at 44.7 score and applied the ceiling function. That brought the score to 44.6. That was encouraging.

These small deltas, however, are naturally more sensitive at the small visits range. This is particularly true when the target is *0* and we estimate it to be a small factional number. We know SMAPE gives a score of 200 even with a very small fractional number as the estimate for a target of 0. So it was unclear if ceiling was the best way and I tried rounding the visits instead. Rounding the answer gave me a bigger jump of a score to 44.2 (with the added benefit of a smaller submission file).

Your mileage may vary but rounding to the nearest integer might be the way to go when submitting your final answer.

EDIT

I realize the conclusions in this post are rather general. It is not only applicable for this contest provided that some reasonable conditions are satisfied. The rough conditions follows from this **sketch of a proof:**

Let y be the target restricted to integer values and y_hat be your floating point estimate. Let's assume y_hat deviates from y by a zero mean symmetric random noise (this is where we assume your algorithm is reasonably good and there is little bias). Then if the standard deviation of the noise is small (around 0.5), you are most likely unilaterally improving your score when you round. When the standard deviation of your noise is big, you might guess towards to right or wrong side of rounding with equal probability so it becomes a wash (disregarding asymmetries on SMAPE).

So roughly you are likely to benefit the most when your estimates are already pretty good but only off by a bit because of the floating point processing. Rounding should not hurt on the average though. If it does, either SMAPE is playing with us or you might be applying an artificial bias to compensate the known SMAPE bias.

Rudolph • (39th in this Competition) • 6 months ago • Options • Reply 1

I published a simple kernel to demonstrate the impact of rounding - in general it can make a difference of 0.5 as Oscar pointed out but in other cases the impact can be quite small (eg 0.05) - in particular for the median prediction.

CPMP • (2nd in this Competition) • 6 months ago • Options • Reply 1

The large improvement is for cases where you don't already round small values to 0. This was proposed and discussed quite a while ago in this forum. Rounding small values to 0 has no effect on median models because they at most produce a 0.5 fractional value; nothing smaller. What Oscar has shown is that rounding other values helps as well.

Oscar Take... • (67th in this Competition) • 6 months ago • Options • Reply 0

@Rudolph nice kernel. I ran some different experiments myself and concluded ceiling and floor are not good in general but rounding is. You may want to add bars in your kernel to check that rounding is better than all four (float, round, ceiling, floor).

Oscar Takesh... • (67th in this Competition) • 6 months ago • Options • Reply 0

For those interested in what's called a **sketch of a proof** it follows as this.

Let y be the target and y_hat be your estimate. Let's assume y_hat deviates from y by a zero mean symmetric random noise (this is where we assume your algorithm is reasonably good and there is little bias). Then if the standard deviation of the noise is small (around 0.5), you are most likely unilaterally improving your score when you round. When the standard deviation of your noise is big, you might guess towards to **right** or **wrong** side of rounding with equal probability so it becomes a wash (disregarding asymmetries on SMAPE).

So roughly you are likely to benefit the most when your estimates are already pretty good but only off by a bit because of the floating point processing. Rounding should not hurt on the average though. If it does, either SMAPE is playing with us or you might be applying an artificial bias to compensate the known SMAPE bias.