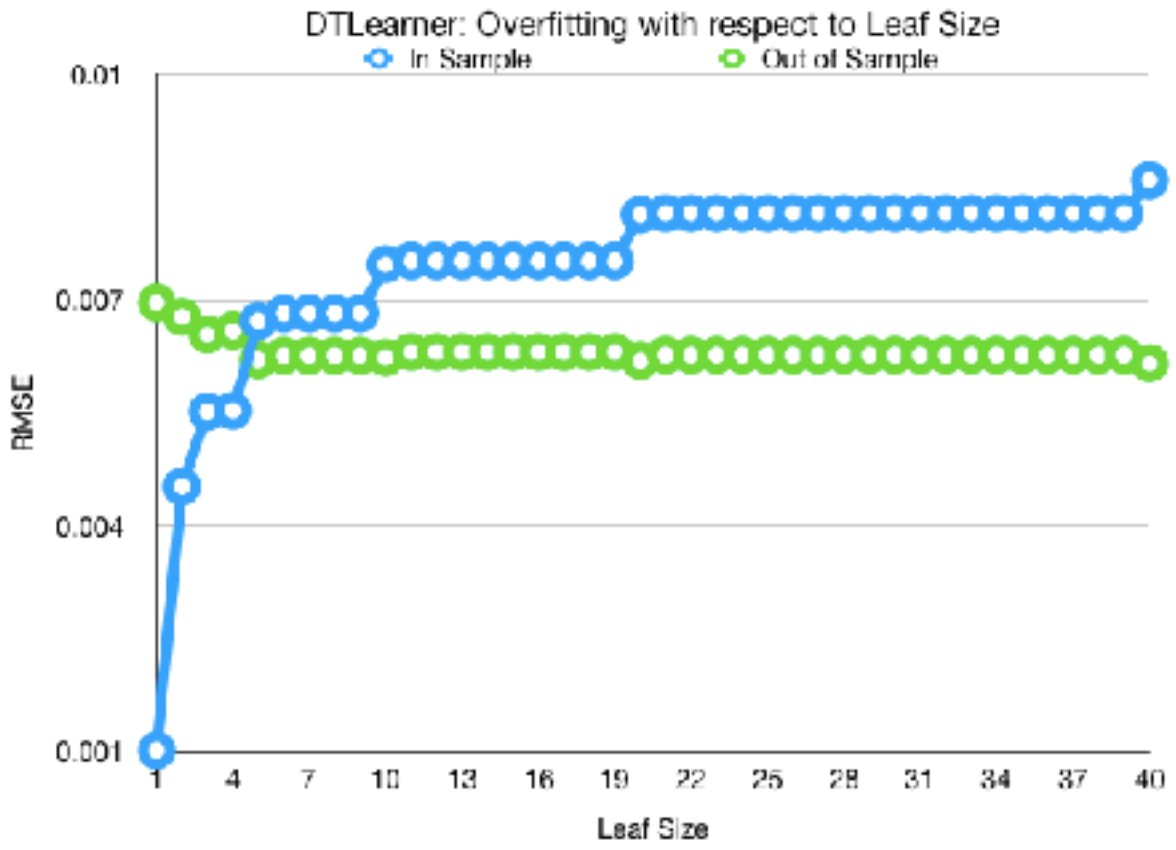
Joshua Reno
CS 4646: Machine Learning for Trading
Project 2: Assess Learners
Due: 2/11/2018

Question 1: Does overfitting occur with respect to leaf_size? Consider the dataset `istanbul.csv` with DTLearner. For which values of leaf_size does overfitting occur? Use RMSE as your metric for assessing overfitting. Support your assertion with graphs/charts. (Don't use bagging).

I generated a graph showing the RMSE (root-mean square error) value of both in sample (blue) and out of sample (green) data using the DTLearner, the istanbul.csv dataset, and leaf values from 1 to 40. The blue line represents in sample or training RMSE while the green line represents out of sample or validation RMSE. The lower the RMSE value, the lower the error between training and testing data. For our purposes, I am going to define overfitting as the point at which the model learns the exact pattern of the training data which forces it to poorly generalize to testing data.

I conclude that overfitting does occur with respect to leaf size. The value for which this occurs is at leaf size 4 to 1 (between leaf size 4 and 5). It is noticeable that from leaf size 40 to leaf size 5, the out of sample RMSE remains relatively stable and the in sample RMSE falls steadily. From leaf size 5 to 4, the RMSE value of the out of sample data begins to rise steadily and the value of the in sample RMSE begins to fall dramatically. These trends continue from leaf size 4 to 1. Simply, this is an obvious sign of overfitting since the model performs poorly with data it has "never seen before" and performs well with data it has "seen before." This is characteristic of overfitting. The graph and RMSE table with increments of 3 leaves (the full table could not be properly visualized) are shown below.

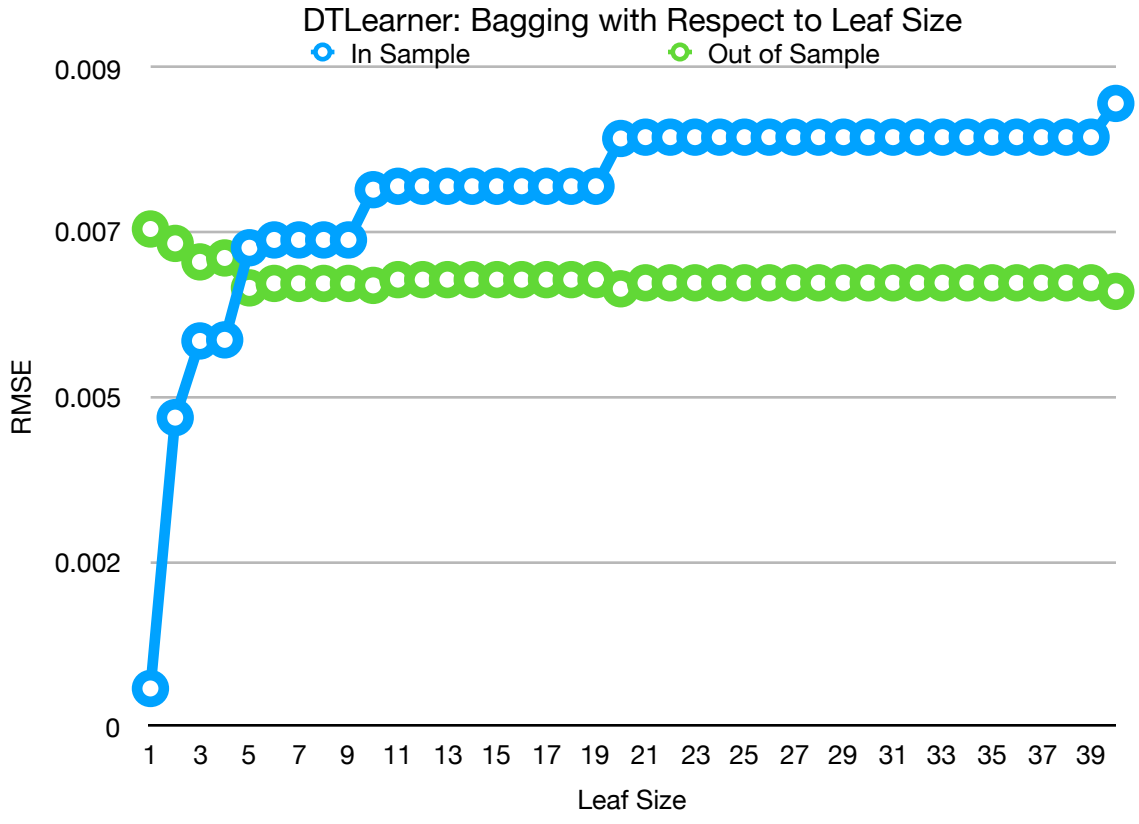


	In Sample	Out of Sample
1	0.000516658582313	0.00679885086812
4	0.0052830330399	0.00640303967484
7	0.0066492318142	0.00605469395262
10	0.00733615444199	0.00602771327511
13	0.0073828909620 6	0.00610634126511
16	0.0073828909620 6	0.00610634126511
19	0.0073828909620 6	0.00610634126511
22	0.0080544430888	0.00606281762135
25	0.0080544430888	0.00606281762135
28	0.0080544430888	0.00606281762135
31	0.0080544430888	0.00606281762135
34	0.0080544430888	0.00606281762135
37	0.0080544430888	0.00606281762135
40	0.00851582971436	0.00594604464748

Question 2: Can bagging reduce or eliminate overfitting with respect to leaf_size? Again consider the dataset `istanbul.csv` with DTLearner. To investigate this choose a fixed number of bags to use and vary leaf_size to evaluate. Provide charts and/or tables to validate your conclusions.

I generated a graph showing the RMSE (root-mean square error) value of both in sample (blue) and out of sample (green) data using the DTLearner, the istanbul.csv dataset, leaf values from 1 to 40, and a fixed value of bags of 20. Bag sizes of 100 and 10 were also used but the data is not shown. The results using bag sizes of 100 and 10 were extremely similar to those for bag sizes of 20.

I conclude that overfitting occurs at leaf size 4 regardless of the number of bags (between leaf size 4 and 5). In fact, the data received and the corresponding graph are identical to the graph in Question 1. I can therefore conclude that bagging has little to no effect on overfitting with respect to leaf size using a DTLearner. The graph and corresponding table are shown below. Out of curiosity, I also tested an RTLearner with bagging and found that using a bag size of 20 reduced the point of overfitting from leaf size 9 to leaf size 5 or 6.



	In Sample	Out of Sample
1	0.000516658582313	0.00679885086812
4	0.0052830330399	0.00640303967484
7	0.0066492318142	0.00605469395262
10	0.00733615444199	0.00602771327511
13	0.00738289096206	0.00610634126511
16	0.00738289096206	0.00610634126511
19	0.00738289096206	0.00610634126511
22	0.0080544430888	0.00606281762135
25	0.0080544430888	0.00606281762135
28	0.0080544430888	0.00606281762135
31	0.0080544430888	0.00606281762135
34	0.0080544430888	0.00606281762135
37	0.0080544430888	0.00606281762135
40	0.00851582971436	0.00594604464748

Question 3: Quantitatively compare "classic" decision trees (DTLearner) versus random trees (RTLearner). In which ways is one method better than the other?

Several metrics were used for the quantitative comparison. Namely, the RMSE value (explained in the explanation of Question 1), the correlation coefficient, and the leaf size values that lead to overfitting. The RMSE value is defined as the square root of the quotient received by dividing the sum of the square of the error between the predicted and actual value by N where N is the number of data points tested. The correlation coefficient (Pearson's r) is defined as the covariance of the prediction and actual value dividing by the product of the standard deviation of the prediction and the standard deviation of the actual values. The reason these two values were chosen is:

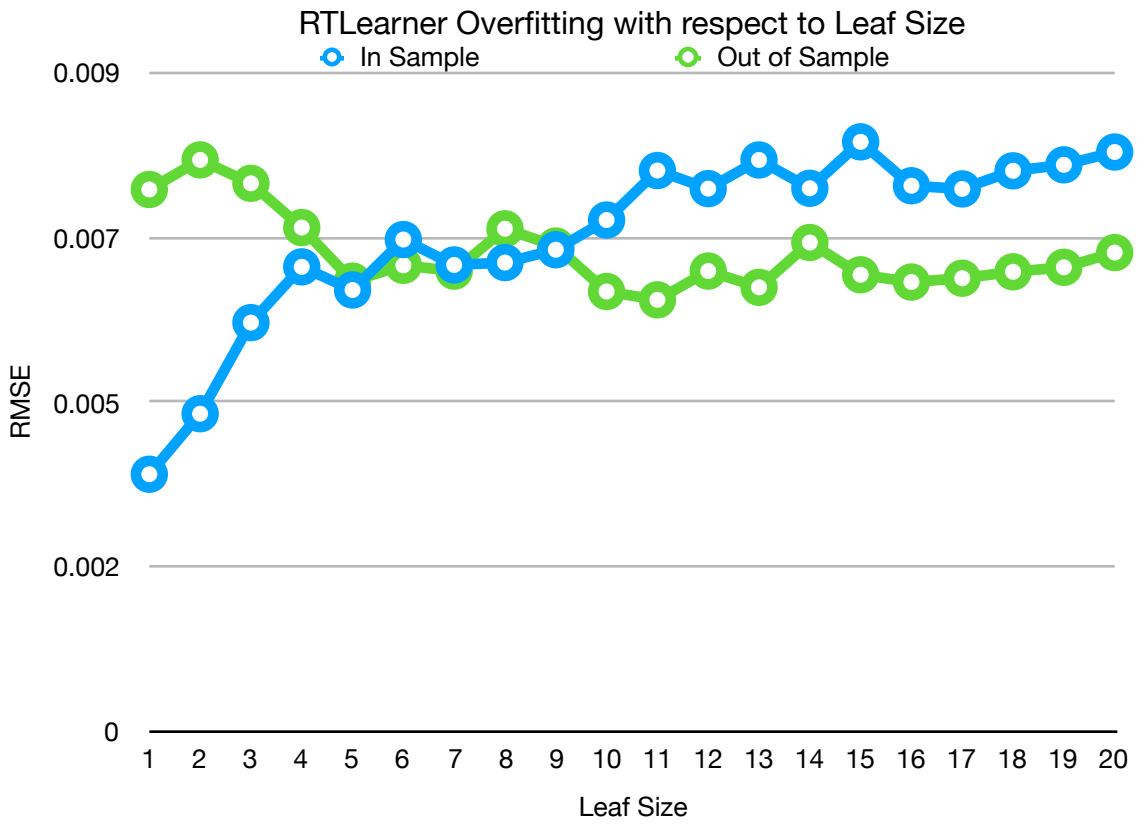
1. RMSE is a measurement of the difference between values (in this case, predicted and actual values). This allows us to measure how "off" the model is in value.
2. Pearson's correlation coefficient is a measure of the linear relationship between variables (in this case, predicted and actual values). This allows us to measure the relationship between our predictions and the model's results.
3. Overfitting is defined in Question 1. For the purpose of this analysis, we will discuss the value of leaf sizes that leads to overfitting.

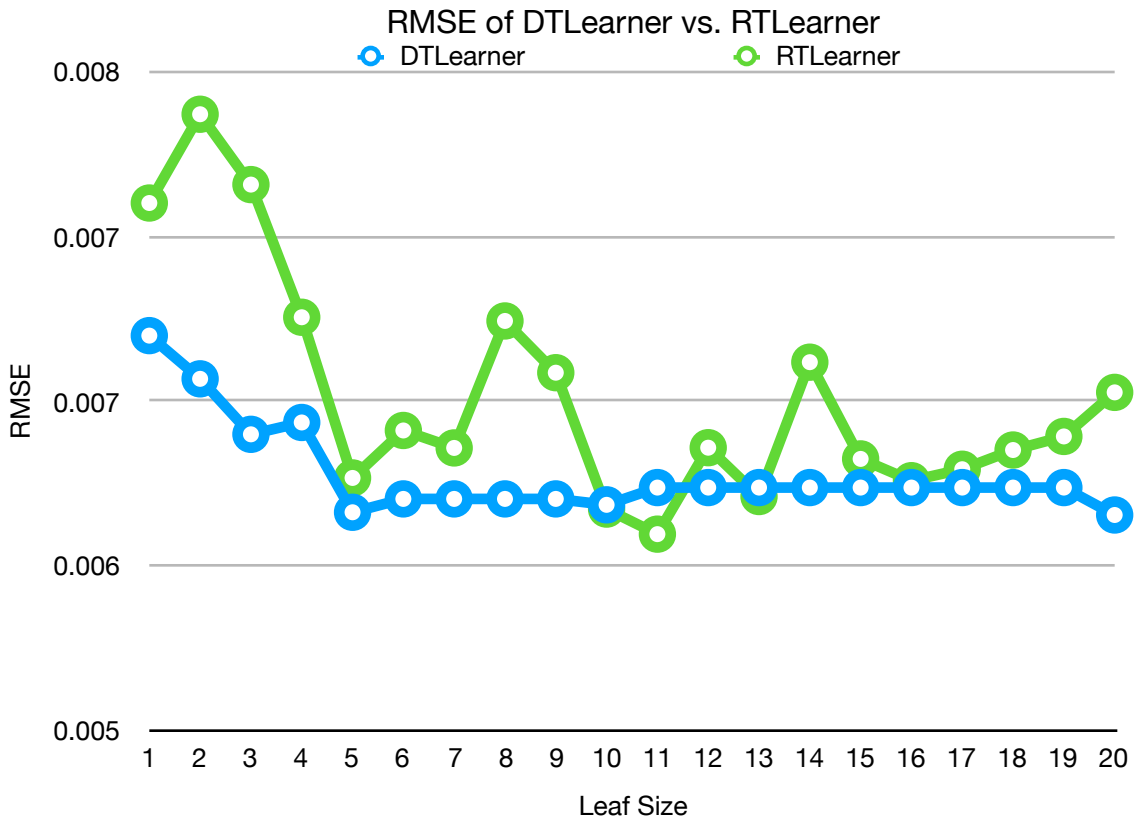
Ranging leaf size from 1 to 20, I constructed a graph of the RMSE values for the DTLearner and RTLearner (Figure: RMSE of DTLearner vs. RTLearner)). Generally, as leaf size increases, the RMSE, which is a method of calculating error, remains lower for the DTLearner than the RTLearner. It is noticeable that, as leaf size increases, the RMSE for both the DTLearner and the RTLearner fall and become somewhat stable (more so for the DTLearner).

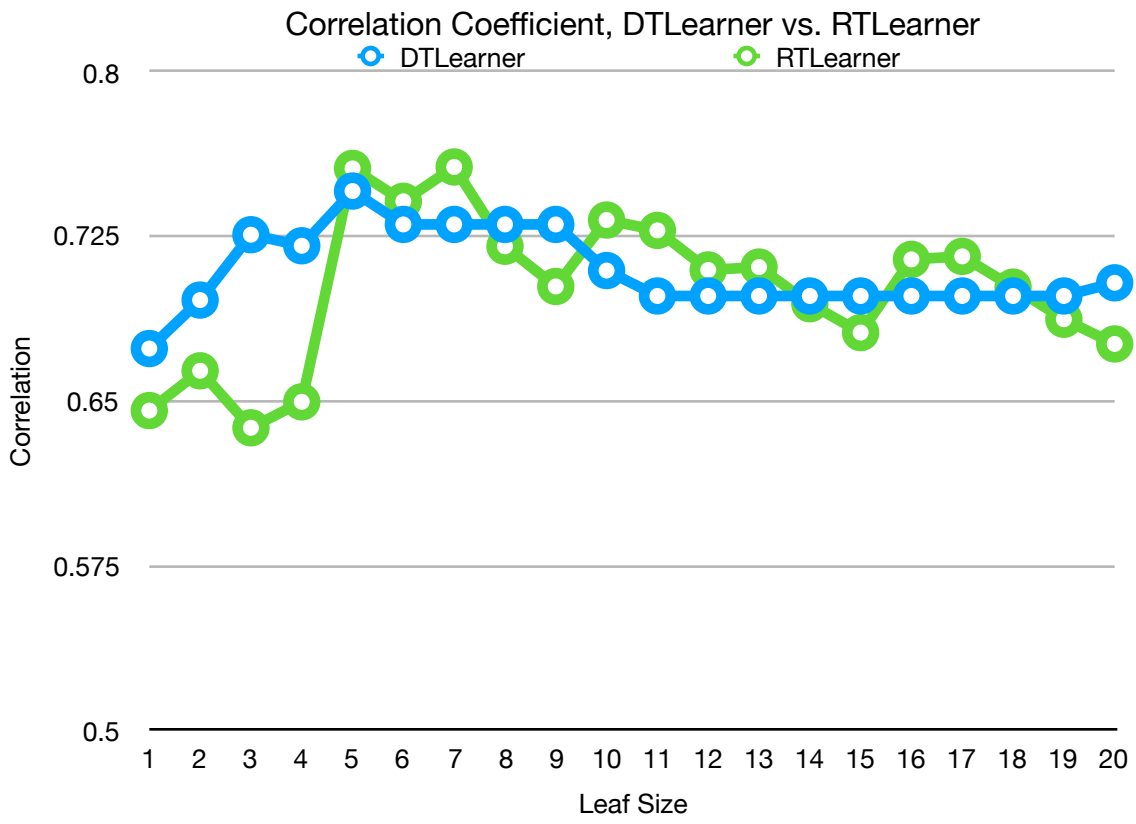
Similarly, ranging leaf size from 1 to 20, I constructed a graph of the correlation coefficient values for the DTLearner and the RTLearner (Figure: Correlation Coefficient, DTLearner vs. RTLearner). Generally, as the leaf size increases, the correlation coefficient for both the DTLearner and the RTLearner increases for the first 5 values of leaf size and then fall in sporadic increments. As far as comparing the two, the correlation coefficients for both the DTLearner and the RTLearner remain close to each other as they steadily fall.

Finally, ranging from leaf size 1 to 20, I constructed a graph of RMSE values for the RTLearner (in sample and out of sample) and compared it to the DTLearner from Question 1 (Figure: RTLearner Overfitting with respect to leaf size) . In this case, value of leaf size that demonstrates overfitting occurs from leaf size 5 to 9. There are cases where the out of sample error and the in sample error rise and fall above each other several times in that interval. This analysis does not exactly compare to the learner from Question 1 as we are only analyzing values from 1 to 20 instead of 1 to 40, but the results remain nonetheless. Finally, the effect of bagging on the learner is not shown in graphs but, as discussed in Question 2, bagging reduced the effect of overfitting in the RTLearner but not in the DTLearner.

In conclusion, the DTLearner produces less error between the model's results and the actual results. This may be because the RTLearner utilizes randomness. Also, there is no significant difference between the DTLearner and the RTLearner with a measure of correlation coefficients, other than that the RTLearner experiences the highest correlation coefficient at leaf size 7. Finally, the RTLearner experiences greater overfitting, which, unlike with the DTLearner, can be reduced with bagging (based on data in Question 2).


RTLearner Overfitting with respect to Leaf Size
In Sample
Out of Sample
RMSE
0.009
0.007
0.005
0.002
0
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
Leaf Size


RMSE of DTLearner vs. RTLearner
DTLearner
RTLearner
RMSE
0.008
0.007
0.007
0.006
0.005
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
Leaf Size


Correlation Coefficient, DTLearner vs. RTLearner
DTLearner
RTLearner
0.8
0.725
0.65
0.575
0.5
Correlation
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
Leaf Size

DTLearner

	In Sample	In Sample Corr	Out of Sample	Out of Sample Core
1	0.000516658582313	0.99903610833	0.00679885086812	0.67396187319
2	0.00421998331599	0.933516296962	0.0066019096803	0.696051438751
3	0.00526827156001	0.894234472336	0.0063490239784	0.725844211229
4	0.0052830330399	0.893605619595	0.00640303967484	0.720798422703
5	0.00654231269081	0.831287340385	0.00599376245781	0.745738421385
6	0.0066492318142	0.825140952467	0.00605469395262	0.730525962951
7	0.0066492318142	0.825140952467	0.00605469395262	0.730525962951
8	0.0066492318142	0.825140952467	0.00605469395262	0.730525962951
9	0.00664923182441	0.825140951873	0.00605469368532	0.730526202292
10	0.00733615444199	0.781991816693	0.00602771327511	0.709513708108
11	0.00738289096206	0.778810332771	0.00610634126511	0.697923831667
12	0.00738289096206	0.778810332771	0.00610634126511	0.697923831667
13	0.00738289096206	0.778810332771	0.00610634126511	0.697923831667
14	0.00738289096206	0.778810332771	0.00610634126511	0.697923831667
15	0.00738289096206	0.778810332771	0.00610634126511	0.697923831667
16	0.00738289096206	0.778810332771	0.00610634126511	0.697923831667
17	0.00738289096206	0.778810332771	0.00610634126511	0.697923831667
18	0.00738289096206	0.778810332771	0.00610634126511	0.697923831667
19	0.00738289096206	0.778810332771	0.00610634126511	0.697923831667
20	0.00803642520822	0.730619766688	0.00598139221655	0.703924644832

RTLearner

	In Sample	In Sample Corr	Out of Sample	Out of Sample Core
1	0.00350981025735	0.954504099864	0.00740319051499	0.645702221575
2	0.00433714342381	0.929632094572	0.00780774500564	0.663818346507
3	0.00558254443172	0.880363372001	0.00748714130249	0.637850003775
4	0.00634528578144	0.842239666989	0.00688256048704	0.649686572999
5	0.00602931323743	0.858831696764	0.00615023975257	0.756072094419
6	0.00671872438389	0.821067523742	0.00636658116577	0.741084575704
7	0.00637139515708	0.840815655675	0.00628702040616	0.756711027125
8	0.00640395208235	0.839028391702	0.00686569877193	0.720572796377
9	0.00657933222659	0.829175643296	0.00662987520333	0.702253668078
10	0.00698562898986	0.804828367756	0.00600833234393	0.732541649113
11	0.00766223762513	0.759083626835	0.00589410220955	0.727812823103
12	0.00741497089505	0.776607269441	0.00628808475411	0.709775884347
13	0.00780690157194	0.748367824529	0.00606556794288	0.711108029257
14	0.00741957743421	0.776289620036	0.00667780692025	0.694393661655
15	0.00805434092198	0.729194307258	0.00623699025494	0.681232054526
16	0.0074540494085	0.773902143106	0.0061377359554	0.714601448626
17	0.00741077281757	0.776896468054	0.0061899129887	0.715970694813
18	0.00765821712185	0.75937643978	0.0062774996096	0.701959914809
19	0.00774022330576	0.753350784437	0.00633940785452	0.687312638743
20	0.00791555039725	0.740083593197	0.00654018881284	0.676063998247