Report

1. Thread is implemented while creating trees, each thread would be assigned a certain amount of trees to speed up the process **currently using 100 trees, 20 threads, 500 data per tree**

2. Real Time

Thread / Tree	50	100	500	1000
1	42.945s	1m24.470s	7m23.623s	13m53.856s
5	20.324s	35.513s	2m29.160s	4m50.595s
10	11.982s	19.647s	1m32.627s	2m57.109s
20	9.829s	17.235s	1m21.455s	2m41.275s

a) As tree amount grows, time grows correspondingly, indicating that the process is mostly related to dealing with those trees.
b) As thread grows from 1 to 10, the improvement is significant. But from 10 to 20, little improvement is achieved, leading to the conclusion that thread limit is somewhere close.

3. Instructions

Thread / Tree	50	100	500	1000
1	129839299576	256823159141	1272888000721	2542907591194
5	129843315672	256831040206	1272927342455	2542986066969
10	129843729099	256832027250	1272932256606	2542997142187
20	129843950205	256832638896	1272935426649	2543002549500

a) As tree amount grows, instructions grow correspondingly, indicating that most of the process is related to dealing with those trees.(Agrees with the result of time)
b) As thread grows, instructions grow by a slow rate, which means that total work has been increased due to need to handle more threads, which agrees with the fact that user time grows as thread num grows. Also pointing out the fact that threads process things paralleled to achieve better real time.

4. Accuracy

Tree	50	100	500	1000
ans.Accuracy	0.8738	0.8744	0.8727	0.8739
train.Accuracy	0.8844	0.8849	0.8837	0.8838

a) Across different amount of trees, accuracy doesn't vary much. This either implies that similarity between data is high or model has already reached a stable point.
b) By modifying amount of data inside each tree, it can be shown that more data doesn't guarantee higher accuracy, due to it decreasing diversity between trees, leading to more similar votes.
c) Seems that it is hard for random forest to overfit, though proven possible (?