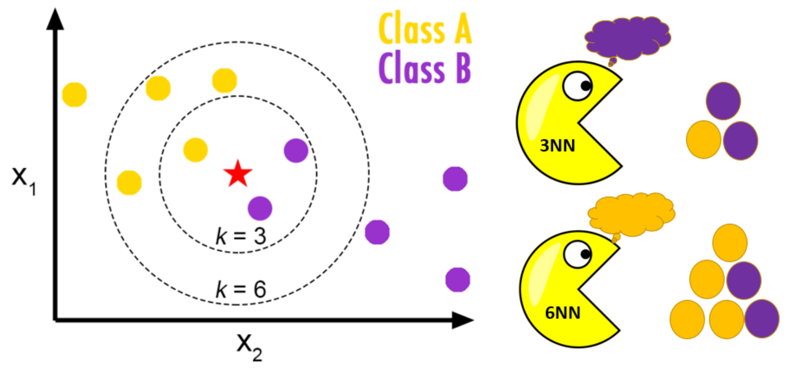
Statistical Guarantees of Distributed Nearest Neighbor Classification

Jiexin Duan[1], Xingye Qiao[2], Guang Cheng[1]

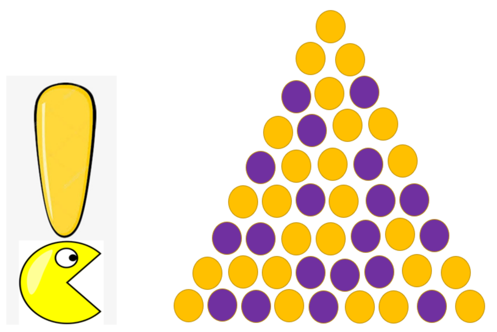
Purdue University[1], Binghamton University[2]

Motivation

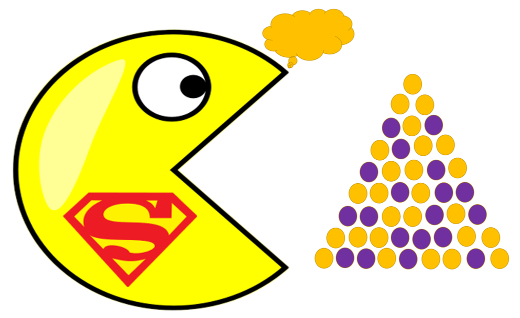
- k Nearest Neighbor Classifier (kNN)



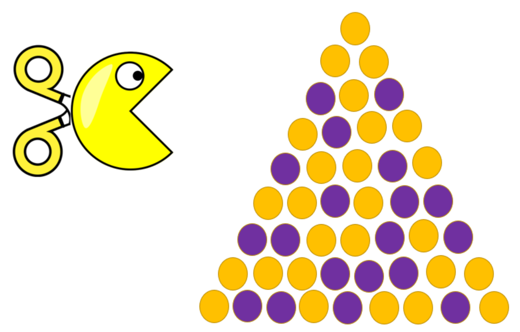
- Challenge of kNN for big data



- If we have a super computer (oracle kNN)
  - Large computational/space complexity
  - Expensive cost
  - Communication, privacy or ownership limitations

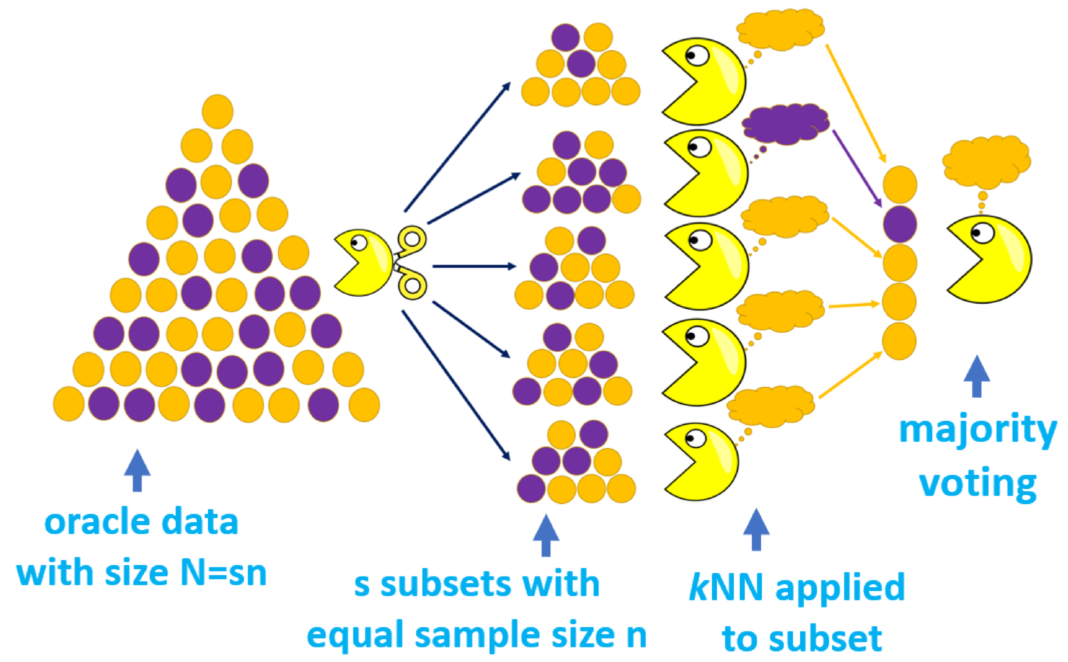


- Without a super computer, split the data!



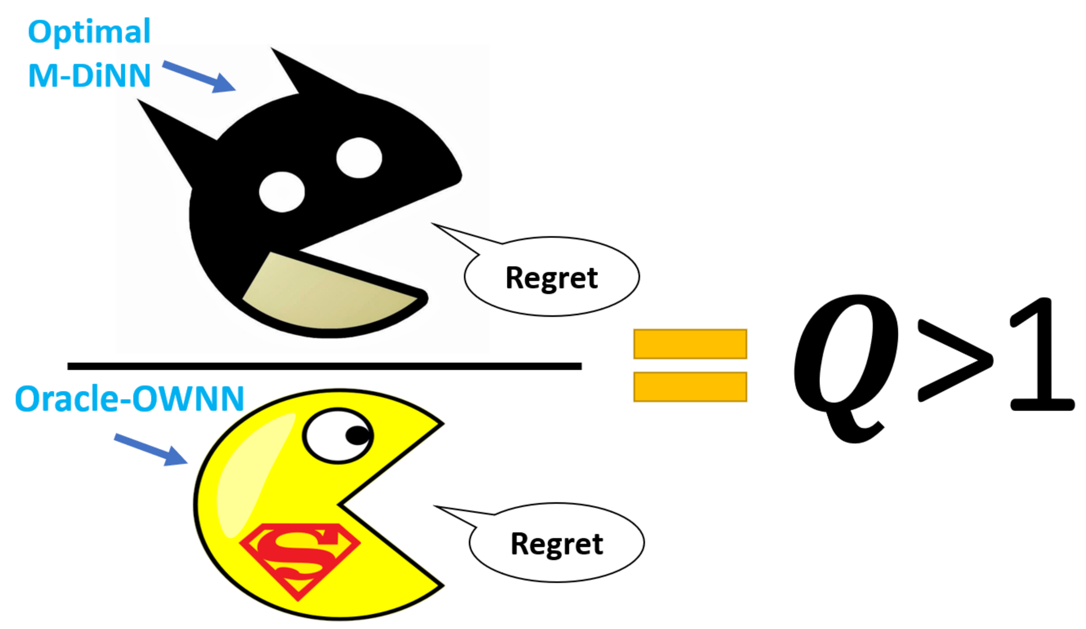
M-DiNN Classifier

- DiNN Classifier via Majority Voting



Performance Comparison

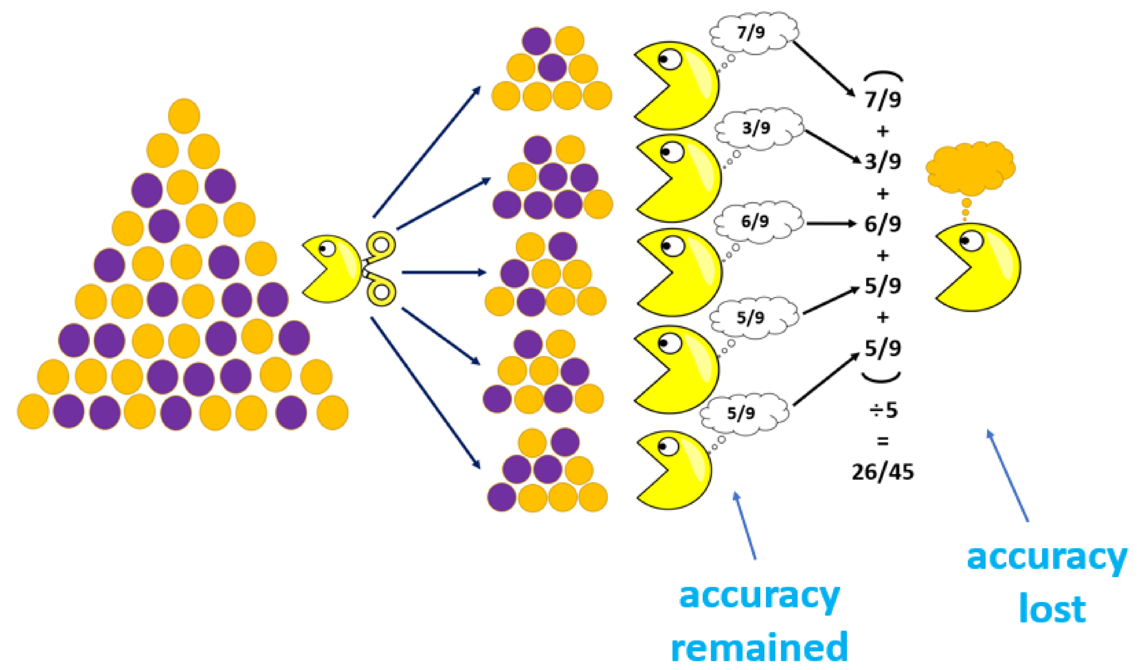
- Criterion: classification accuracy
- Regret=Expected Risk−Bayes Risk
- A small value of regret is preferred
- Benchmark: Oracle-OWNN is an NN classifier trained by entire dataset, and it minimizes the asymptotic regret of weighted nearest neighbor (Samworth 2012)
- Asymptotic Regret Comparison



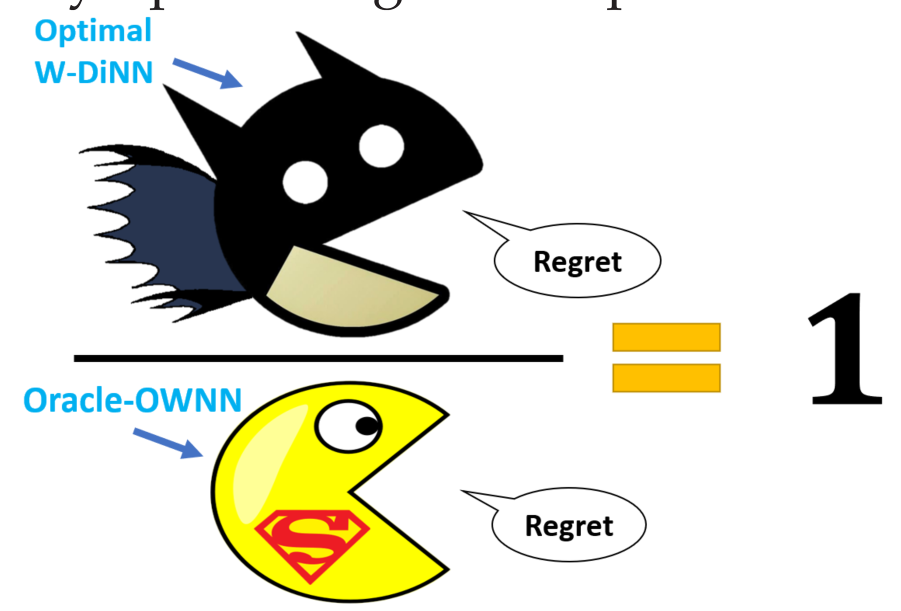
- Same convergence rate with a constant multiplicative accuracy loss Q

W-DiNN Classifier

- Motivation: to remove the constant multiplicative accuracy loss Q in M-DiNN
- DiNN Classifier via Weighted Voting



- Asymptotic Regret Comparison



Theorems: Asymptotic Regret for M-DiNN and W-DiNN

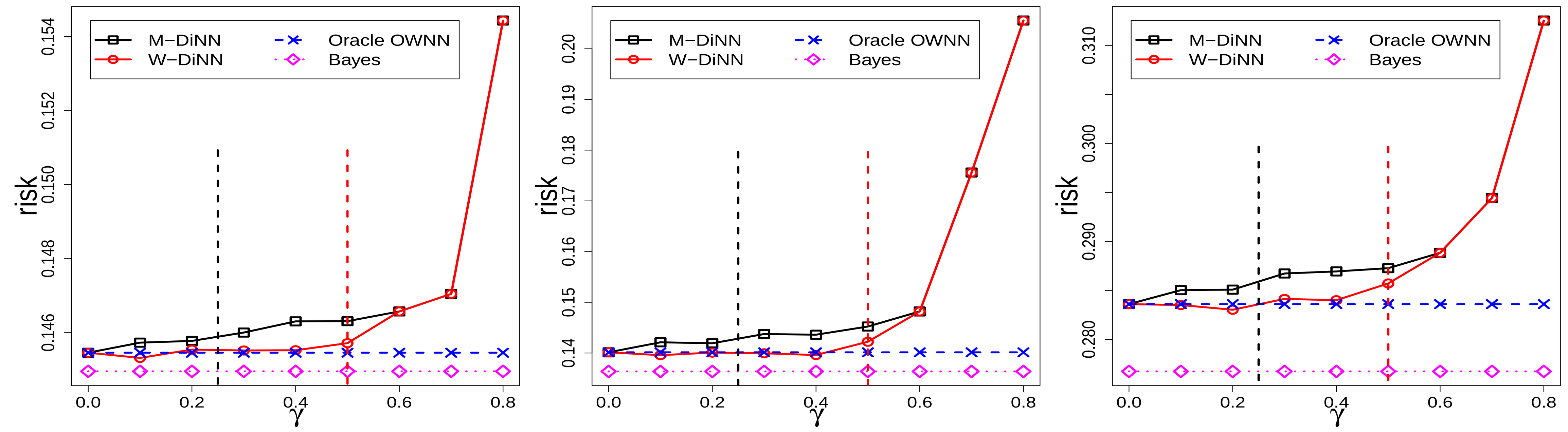
Under regularity assumptions, as $n, s \to \infty$, we have

$$\text{Regret(M-DiNN)} \approx B_1 s^{-1} \sum_{i=1}^{n} w_{ni}^2 + B_2 \Big(\sum_{i=1}^{n} \frac{\alpha_i w_{ni}}{n^{2/d}} \Big)^2, \text{ when } s < N^{2/(d+4)},$$

$$\text{Regret(W-DiNN)} \approx B_3 s^{-1} \sum_{i=1}^{n} w_{ni}^2 + B_2 \Big(\sum_{i=1}^{n} \frac{\alpha_i w_{ni}}{n^{2/d}} \Big)^2, \text{ when } s < N^{4/(d+4)},$$

where $\alpha_i = i^{1+\frac{2}{d}} - (i-1)^{1+\frac{2}{d}}$, w_{ni} are the local weights, constants B_1, B_2 and B_3 are based on the underlying distribution, s is the number of subsets, n is the size of subsets, $N = sn$ is the size of entire dataset.

Simulations



Notes: Risk of optimal M-DiNN, W-DiNN, Oracle OWNN and the Bayes rule for different γ. Left/middle/right: Simulation 1/2/3, $d = 4$. Upper bounds for number of subsamples in optimal M-DiNN ($\gamma = 1/4$) and W-DiNN ($\gamma = 1/2$) are shown as two vertical lines.

Real Data Examples

Data	N	d	γ	M(k)	W(k)	KT	CT	kNN	OWNN	SU-Di	SU-KT	SU-CT
			0.1	15.36	15.22					1.19		
Musk1	476	166	0.2	15.45	15.28	23.20	23.43	15.10	14.98	2.23	2.31	2.03
			0.3	15.82	15.53					3.30		
			0.1	4.01	3.70					2.54		
Gisette	6000	5000	0.2	4.18	3.94	7.11	7.16	3.62	3.48	4.55	1.56	1.13
			0.3	4.10	3.88					10.68		
			0.1	3.91	3.78					3.30		
Musk2	6598	166	0.2	3.91	3.75	6.17	6.14	3.54	3.28	5.69	3.67	1.9
			0.3	4.23	3.98					15.62		
			0.1	2.26	2.20					3.27		
HTRU2	17898	8	0.2	2.23	2.18	2.35	2.37	2.19	3.12	7.96	7.35	2.12
			0.3	2.30	2.22					21.90		
			0.1	19.37	19.28					3.00		
Credit	30000	24	0.2	19.31	19.23	22.78	22.77	19.08	18.96	7.67	7.53	3.36
			0.3	19.33	19.27					23.57		
			0.1	23.58	22.32					4.02		
SUSY	5000K	18	0.2	23.63	22.30	28.02	28.35	21.57	21.11	16.56	8.02	3.25
			0.3	23.76	22.51					72.78		

Notes: Risk (in %) of M-DiNN(k) (M(k)) and W-DiNN(k) (W(k)) compared to Fast Approximate Nearest Neighbor Search (FANN) (k-d tree (KT), cover tree (CT)), Oracle kNN and OWNN. Number of subsets $s = N^{\gamma}$. The speedup factors (SU-Di, SU-KT, SU-CT) are defined as the computing time of the Oracle kNN divided by the time of the slower of the two DiNN(k) methods, and the two FANN methods, respectively.