A better k-means++ Algorithm via Local Search

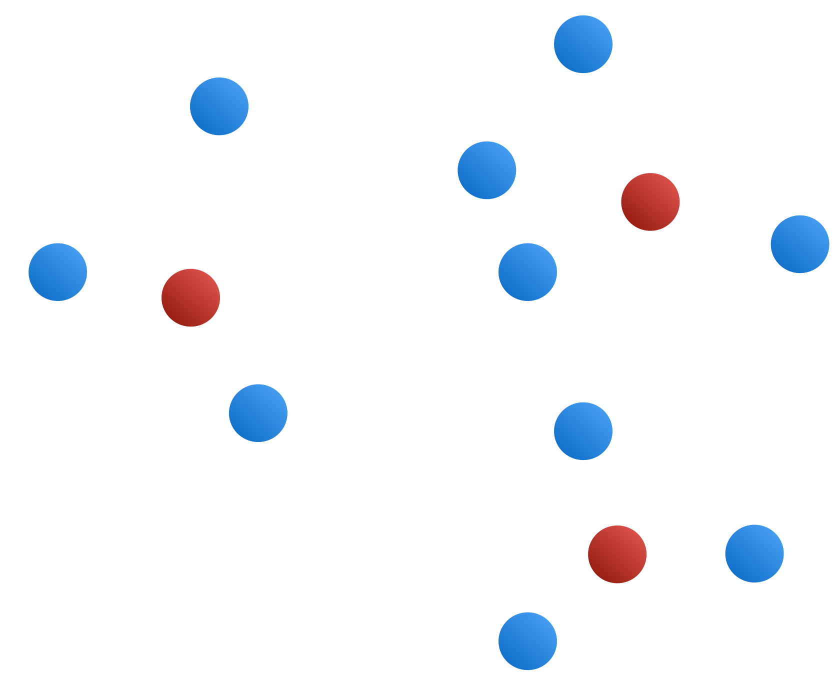
Silvio Lattanzi
Google Research

Christian Sohler
Google Research

ICML 2019

k-means

Find a set of k centers



$$\phi(X, C) = \sum_{x \in X} \min_{c \in C} d^2(x, c)$$

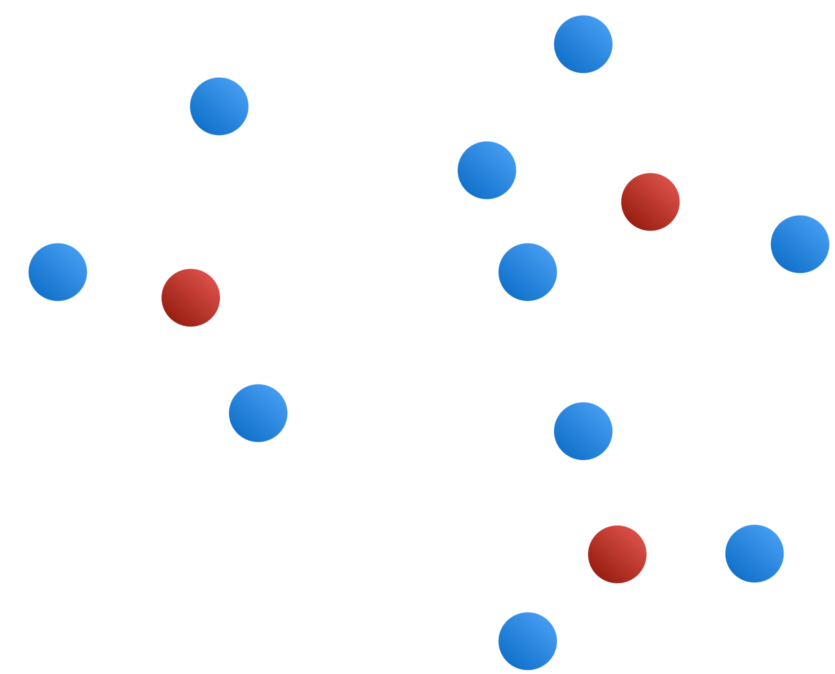
Constant approximation algorithms are known.

Goal is to design a constant approximation algorithm that is efficient, easy to implement and has good experimental results.

k-means++ seeding

Elegant and simple algorithm

Uniformly sample $p \in P$ and set $C = \{p\}$.
for $i \leftarrow 2, 3, \ldots, k$ **do**
  Sample $p \in P$ with probability $\frac{\text{cost}(\{p\},C)}{\sum_{q \in P} \text{cost}(\{q\},C)}$ and add it to C.
end for



Experimentally gives good results when combined with Lloyd's algorithm.

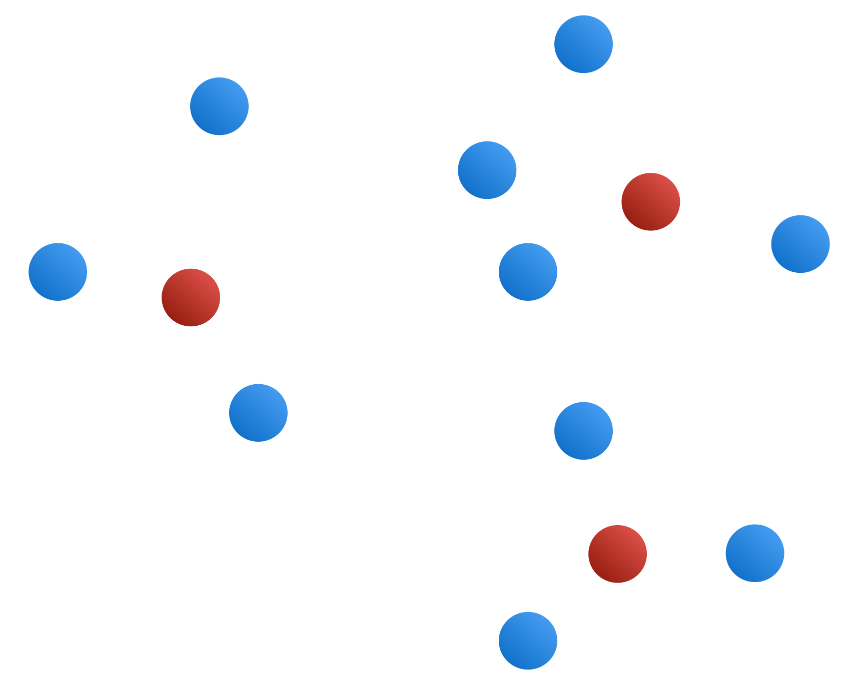
The solution is a $O(\log k)$ approximation in expectation.

David Arthur, Sergei Vassilvitskii: k-means++: the advantages of careful seeding. SODA 2007: 1027-1035

Local search

Elegant and simple algorithm

if $\exists q \in C$ s.t. $\text{cost}(P, C \setminus \{q\} \cup \{p\}) < \text{cost}(P, C)$ **then**
  Let $q \in C$ be the q s.t. $\text{cost}(P, C \setminus \{q\} \cup \{p\})$ is minimized
  $C = C \setminus \{q\} \cup \{p\}$
end if



It returns a constant approximation and nice experimental results.

The algorithm is a bit slow.

Tapas Kanungo, David M. Mount, Nathan S. Netanyahu, Christine D. Piatko, Ruth Silverman, Angela Y. Wu:
A local search approximation algorithm for k-means clustering. Comput. Geom. 28(2-3): 89-112 (2004)

Combining the two algorithms

Elegant and simple algorithm

Algorithm 1 k-means++ seeding with local search

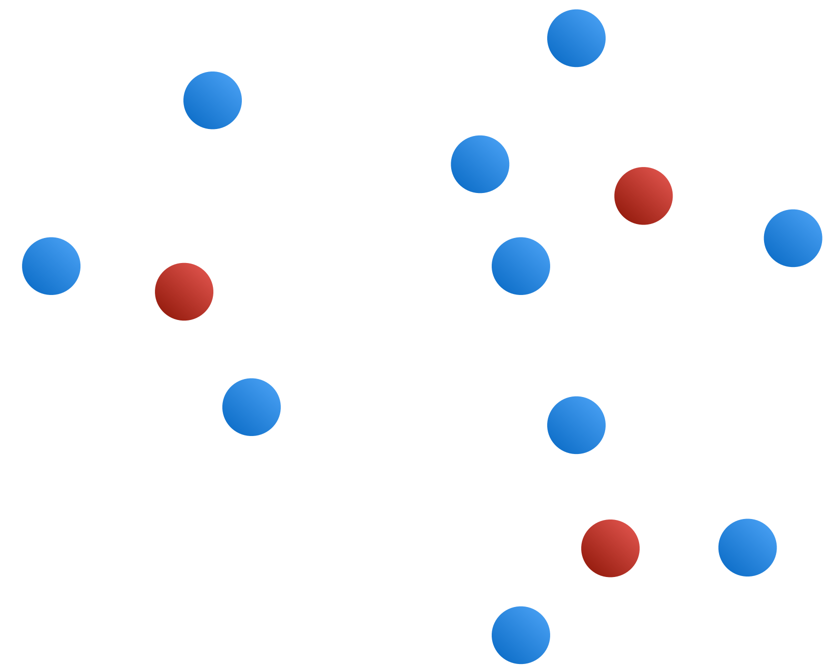
Require: P, k, Z

Uniformly sample $p \in P$ and set $C = \{p\}$.
for $i \leftarrow 2, 3, \dots, k$ **do**
  Sample $p \in P$ with probability $\frac{\text{cost}(\{p\},C)}{\sum_{q \in P} \text{cost}(\{q\},C)}$ and add it to C.
end for
for $i \leftarrow 2, 3, \dots, Z$ **do**
  $C = \text{LocalSearch++}(P, C)$
end for
return C

Algorithm 2 LocalSearch++

Require: P, C

Sample $p \in P$ with probability $\frac{\text{cost}(\{p\},C)}{\sum_{q \in P} \text{cost}(\{q\},C)}$
if $\exists q \in C$ s.t. $\text{cost}(P, C \setminus \{q\} \cup \{p\}) < \text{cost}(P, C)$ **then**
  Let $q \in C$ be the q s.t. $\text{cost}(P, C \setminus \{q\} \cup \{p\})$ is minimized
  $C = C \setminus \{q\} \cup \{p\}$
end if
return C



It returns a constant approximation, it is slightly slower than k-means++ and has better experimental results.

Main theoretical result

Theorem 1. *Let $P \subseteq \mathbb{R}^d$ be a set of points and C be the output of Algorithm 1 with $Z \geq 100000k \log\log k$ then we have $E[cost(P, C)] \in O(cost(P, C^*))$, where C^* is the set of optimum centers. The running time of the algorithm is $O(dnk^2 \log\log k)$.*

Main idea is to adapt local search analysis to show that in every step with constant probability we reduce the cost of the solution by a multiplicative $\left(1 - \frac{1}{100k}\right)$ factor

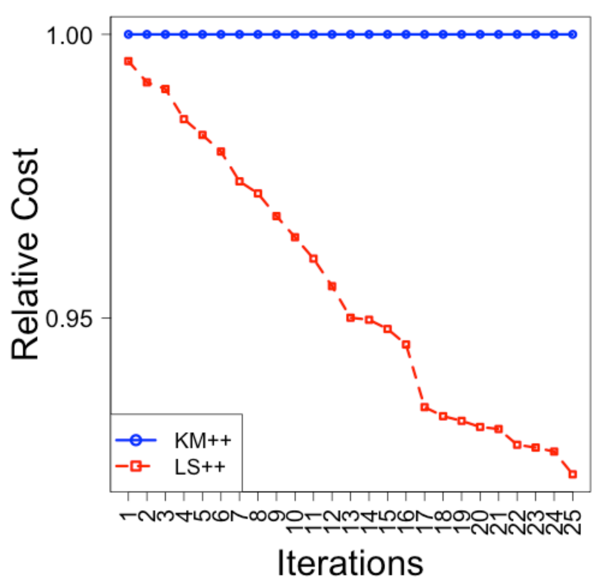
Experimental results

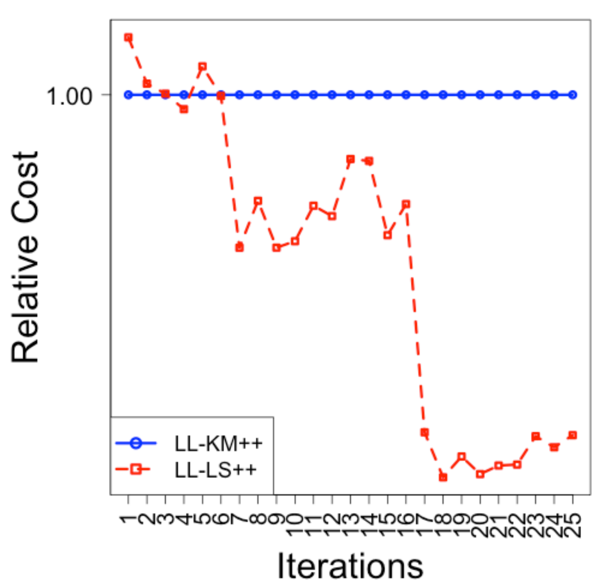
Datasets:

- **RNA**: 8 features from 488565 RNA input sequence pairs (Uzilov et al., 2006)
- **KDD-BIO**: 145751 samples with 74 features measuring the match between a protein and a native sequence (KDD)
- **KDD-PHY**: 100000 samples with 78 features representing a quantum physic task (KDD)

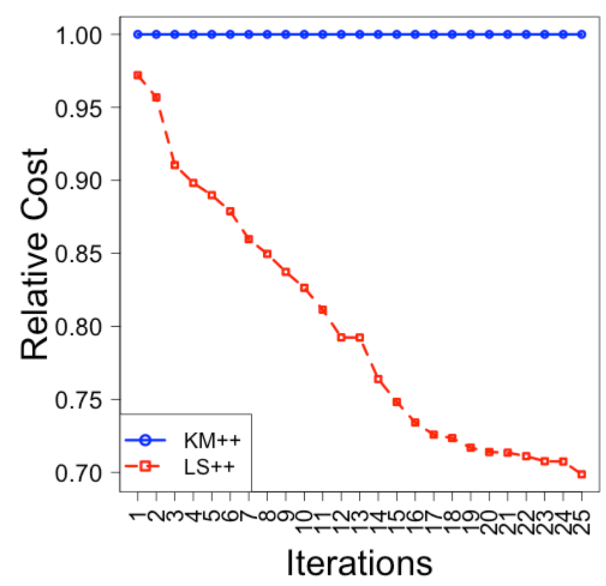
Experimental results

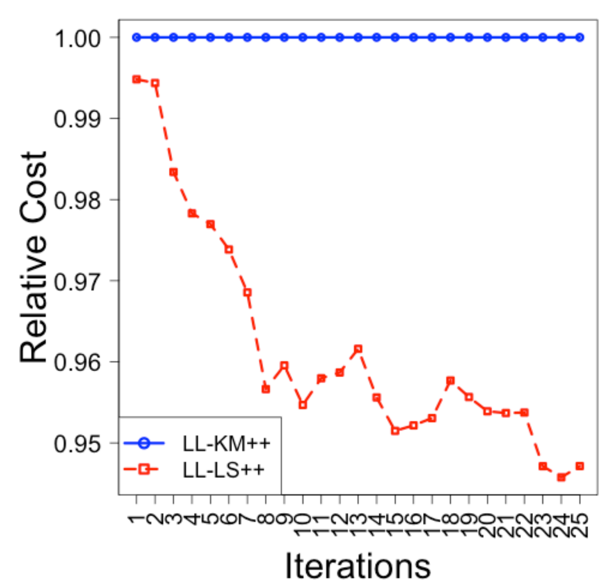
KDD-BIO



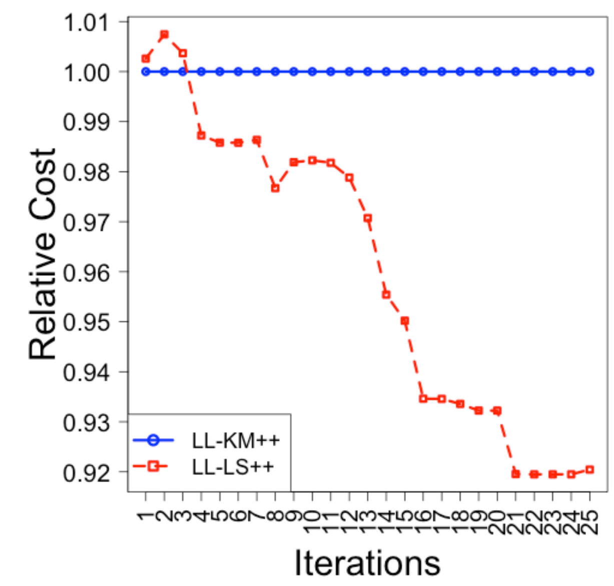


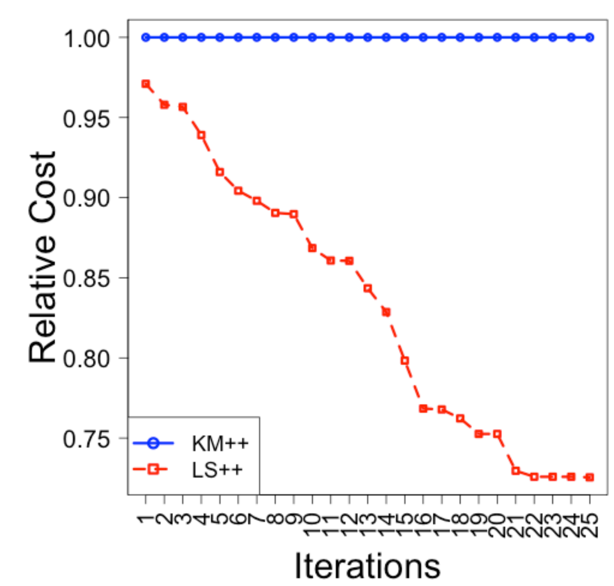
RNA





KDD-PHY





Thanks