tvannoy.github.io/randomized-subkmeans-presentation

Improved Subspace K-Means Performance via a Randomized Matrix Decomposition

Trevor Vannoy

Jacob Senecal, Veronika Strnadova-Neeley

IEEE GlobalSIP 2019 | Nov. 13 2019

Outline

- Clustering in high dimensions
- Subspace k-means
- Randomized Subspace k-means
- Experiments
- Results
- Conclusions

Clustering in High Dimensions

Problems

- curse of dimensionality
- computational complexity
- visualization

Solutions

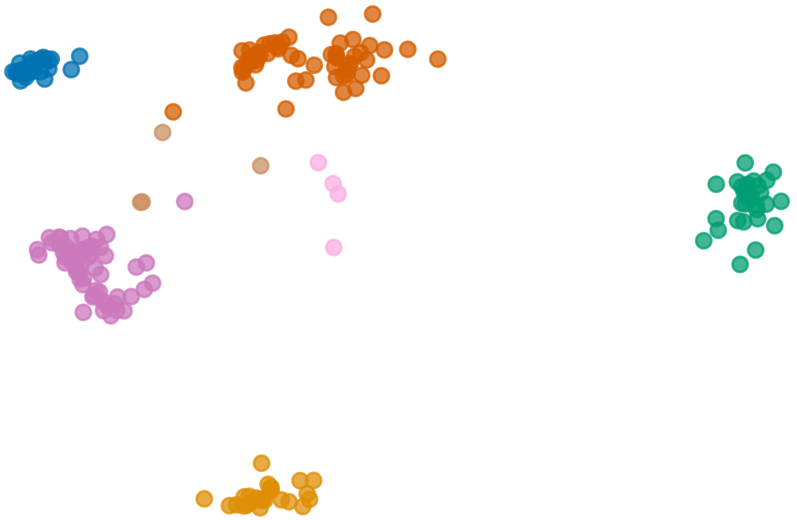
- subspace clustering
  - separate subspaces
  - common subspace

Subspace K-means

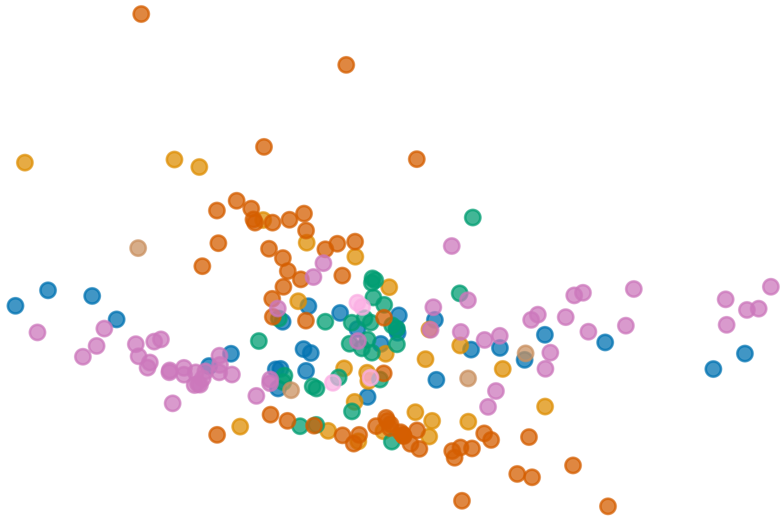
Subspace k-means

- transforms data into a cluster subspace and a noise subspace
- alternates between subspace estimation and clustering

cluster subspace



noise subspace



Objective Function

$$\mathcal{J} = \left[\sum_{i=1}^{k} \sum_{\mathbf{x} \in C_i} || \quad \mathbf{x} - \quad \boldsymbol{\mu}_i ||^2 \right]$$

Objective Function

$$\mathcal{J} = \left[\sum_{i=1}^{k} \sum_{\mathbf{x} \in C_i} || \, P_C^T \, V^T \, \mathbf{x} - P_C^T \, V^T \, \boldsymbol{\mu}_i ||^2 \right]$$

$$+ \sum_{\mathbf{x} \in \mathcal{D}} || \, P_N^T \, V^T \, \mathbf{x} - P_N^T \, \boldsymbol{\mu}_{\mathcal{D}} ||^2$$

Objective Function

$$\mathcal{J} = \left[\sum_{i=1}^{k} \sum_{\mathbf{x} \in C_i} || P_C^T V^T \mathbf{x} - P_C^T V^T \boldsymbol{\mu}_i ||^2 \right] + \sum_{\mathbf{x} \in \mathcal{D}} || P_N^T V^T \mathbf{x} - P_N^T \boldsymbol{\mu}_{\mathcal{D}} ||^2$$

$P_C \equiv$ cluster space projection matrix

Objective Function

$$\mathcal{J} = \left[\sum_{i=1}^{k} \sum_{\mathbf{x} \in C_i} || P_C^T V^T \mathbf{x} - P_C^T V^T \boldsymbol{\mu}_i ||^2 \right] + \sum_{\mathbf{x} \in \mathcal{D}} || P_N^T V^T \mathbf{x} - P_N^T \boldsymbol{\mu}_{\mathcal{D}} ||^2$$

$P_C \equiv$ cluster space projection matrix

$P_N \equiv$ noise space projection matrix

Objective Function

$$\mathcal{J} = \left[\sum_{i=1}^{k} \sum_{\mathbf{x} \in C_i} || P_C^T V^T \mathbf{x} - P_C^T V^T \boldsymbol{\mu}_i ||^2 \right] + \sum_{\mathbf{x} \in \mathcal{D}} || P_N^T V^T \mathbf{x} - P_N^T \boldsymbol{\mu}_{\mathcal{D}} ||^2$$

$P_C \equiv$ cluster space projection matrix

$P_N \equiv$ noise space projection matrix

$V \equiv$ transformation matrix

Objective Function

$$\mathcal{J} = \mathrm{tr}\left(P_C P_C^T V^T \underbrace{\left(\left[\sum_{i=1}^{k} S_i\right] - S_{\mathcal{D}}\right)}_{\Sigma} V\right)$$

$$+ \underbrace{\mathrm{tr}(V^T S_{\mathcal{D}} V)}_{\text{const. w.r.t } V}$$

Objective Function

$$\mathcal{J} = \mathrm{tr}\left(P_C P_C^T V^T \underbrace{\left(\left[\sum_{i=1}^{k} S_i \right] - S_{\mathcal{D}} \right)}_{\Sigma} V \right)$$

$$+ \underbrace{\mathrm{tr}(V^T S_{\mathcal{D}} V)}_{\text{const. w.r.t } V}$$

$S_i \equiv$ cluster scatter matrix

Objective Function

$$\mathcal{J} = \mathrm{tr}\left(P_C P_C^T V^T \underbrace{\left(\left[\sum_{i=1}^{k} S_i\right] - S_{\mathcal{D}}\right)}_{\Sigma} V\right)$$

$$+ \underbrace{\mathrm{tr}(V^T S_{\mathcal{D}} V)}_{\text{const. w.r.t } V}$$

$S_i \equiv$ cluster scatter matrix

$S_{\mathcal{D}} \equiv$ dataset scatter matrix

Minimization

$$\mathcal{J} = \mathrm{tr}\left(P_C P_C^T V^T \underbrace{\left(\left[\sum_{i=1}^{k} S_i \right] - S_{\mathcal{D}} \right)}_{\Sigma} V \right) \dots$$

- put eigenvectors of Σ into V in ascending order
- keep the negative eigenvalues via $P_C P_C^T$

Computational Complexity

$$\mathcal{O}\Big(I \left(mk|\mathcal{D}| + d^2|\mathcal{D}| + d^3 \right)\Big)$$

Computational Complexity

$$\mathcal{O}\Big(\, I \,(\, mk|\mathcal{D}| + d^2|\mathcal{D}| + d^3 \,)\Big)$$

k-means

Computational Complexity

$$\mathcal{O}\Big(I\,(\,mk|\mathcal{D}| + d^2|\mathcal{D}| + d^3\,)\Big)$$

k-means

scatter matrix

Computational Complexity

$$\mathcal{O}\left(I \left(mk|\mathcal{D}| + d^2|\mathcal{D}| + d^3 \right)\right)$$

k-means

scatter matrix

eigenvalue decomposition

Randomized Subspace k-means

Transformation Matrix Approximation

- $P_C P_C^T$ only keeps the first m eigenvalues
- compute rank-m approximation, $\widetilde{V}$, using a randomized eigenvalue decomposition

Transformation Matrix Approximation

- $P_C P_C^T$ only keeps the first m eigenvalues
- compute rank-m approximation, $\widetilde{V}$, using a randomized eigenvalue decomposition

$$\mathcal{J} = \mathrm{tr}\left(\widetilde{V}^T \underbrace{\left(\left[\sum_{i=1}^{k} S_i\right] - S_{\mathcal{D}}\right)}_{\Sigma} \widetilde{V}\right) + \underbrace{\mathrm{tr}(\widetilde{V}^T S_{\mathcal{D}} \widetilde{V})}_{\text{const. w.r.t } \widetilde{V}}.$$

Computational Complexity

before:

$$\mathcal{O}\Big(I\big(mk|\mathcal{D}| + d^2|\mathcal{D}| + d^3\big)\Big)$$

Computational Complexity

before:

$$\mathcal{O}\Big(I\big(mk|\mathcal{D}| + d^2|\mathcal{D}| + d^3\big)\Big)$$

after:

$$\mathcal{O}\Big(I\big(mk|\mathcal{D}| + d^2|\mathcal{D}| + d^2\log m\big)\Big)$$

Computational Complexity

before:

$$\mathcal{O}\Big(I\big(mk|\mathcal{D}| + d^2|\mathcal{D}| + d^3\big)\Big)$$

after:

$$\mathcal{O}\Big(I\big(mk|\mathcal{D}| + d^2|\mathcal{D}| + d^2 \log m\big)\Big)$$

Experiments

Synthetic Data

- runtime vs dimensions
- runtime vs instances

Real Datasets

- clustering quality
- runtime

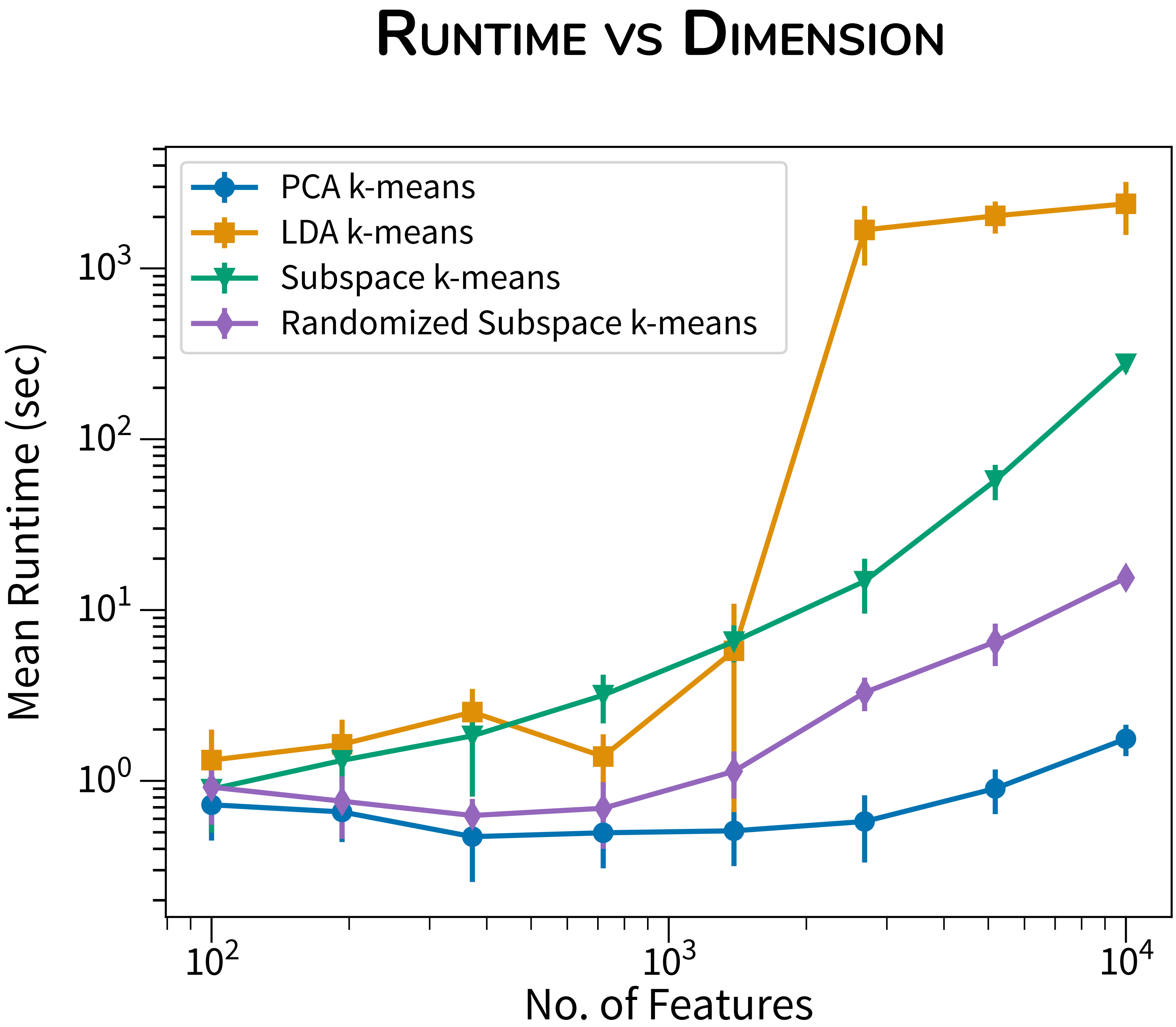
Algorithms

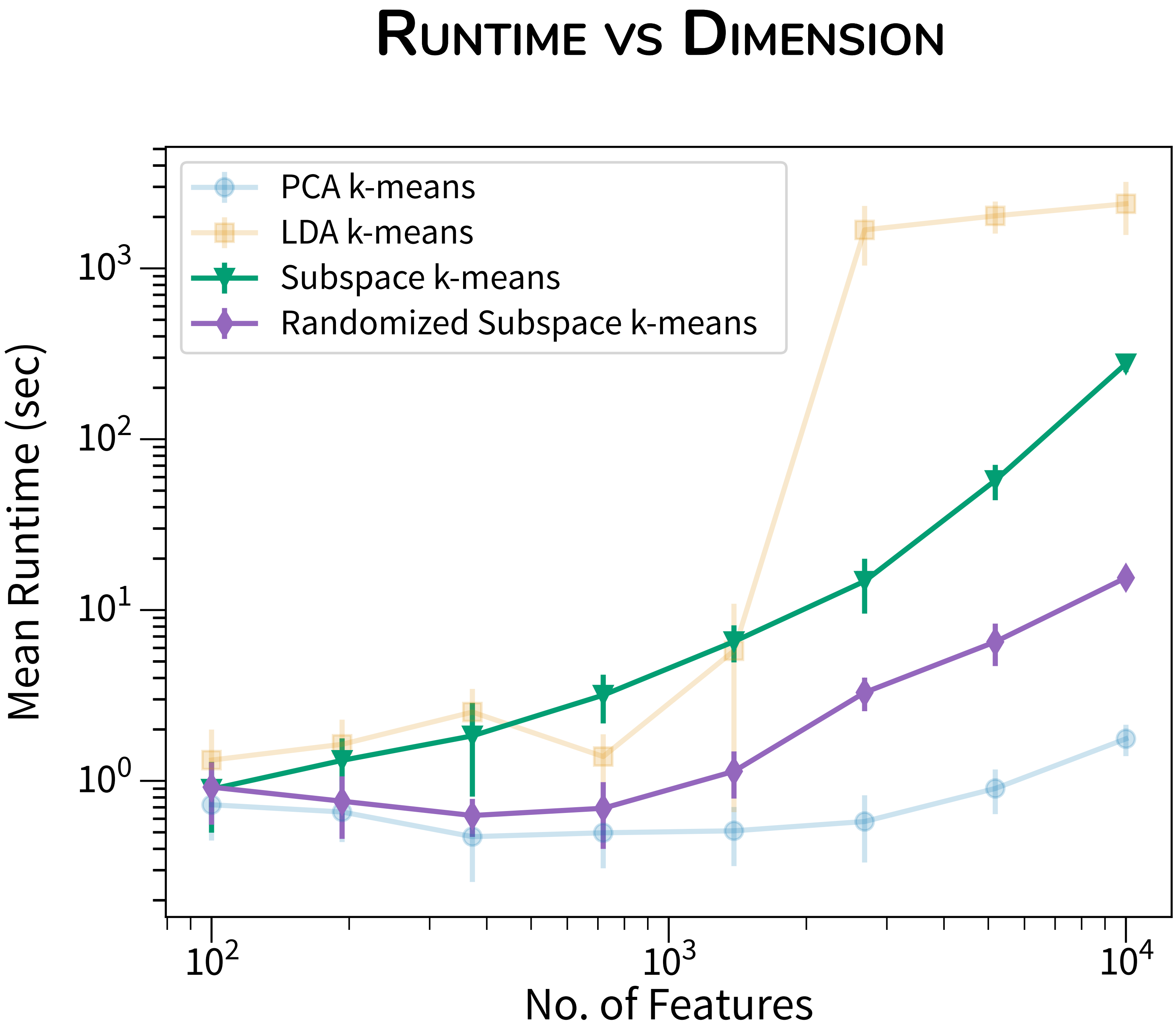
- Subspace k-means
- Randomized Subspace k-means
- PCA k-means
- LDA k-means

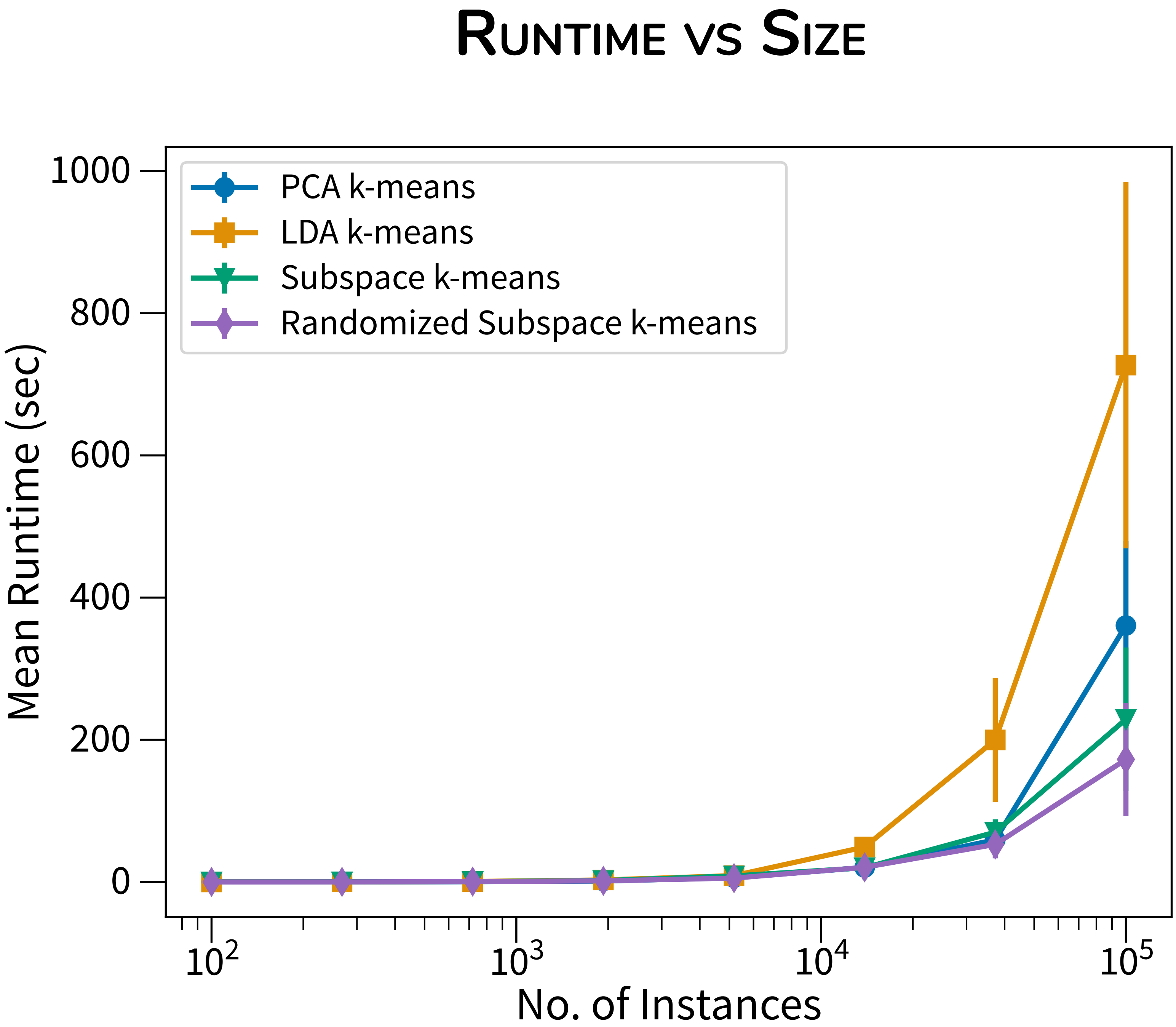
Datasets

	Features	Instances	Classes
Plane	114	210	7
Symbols	398	1020	6
OliveOil	570	60	4
StarLightCurves	1024	9236	3
DrivFace	6400	606	3
RNA-Seq	20531	801	5

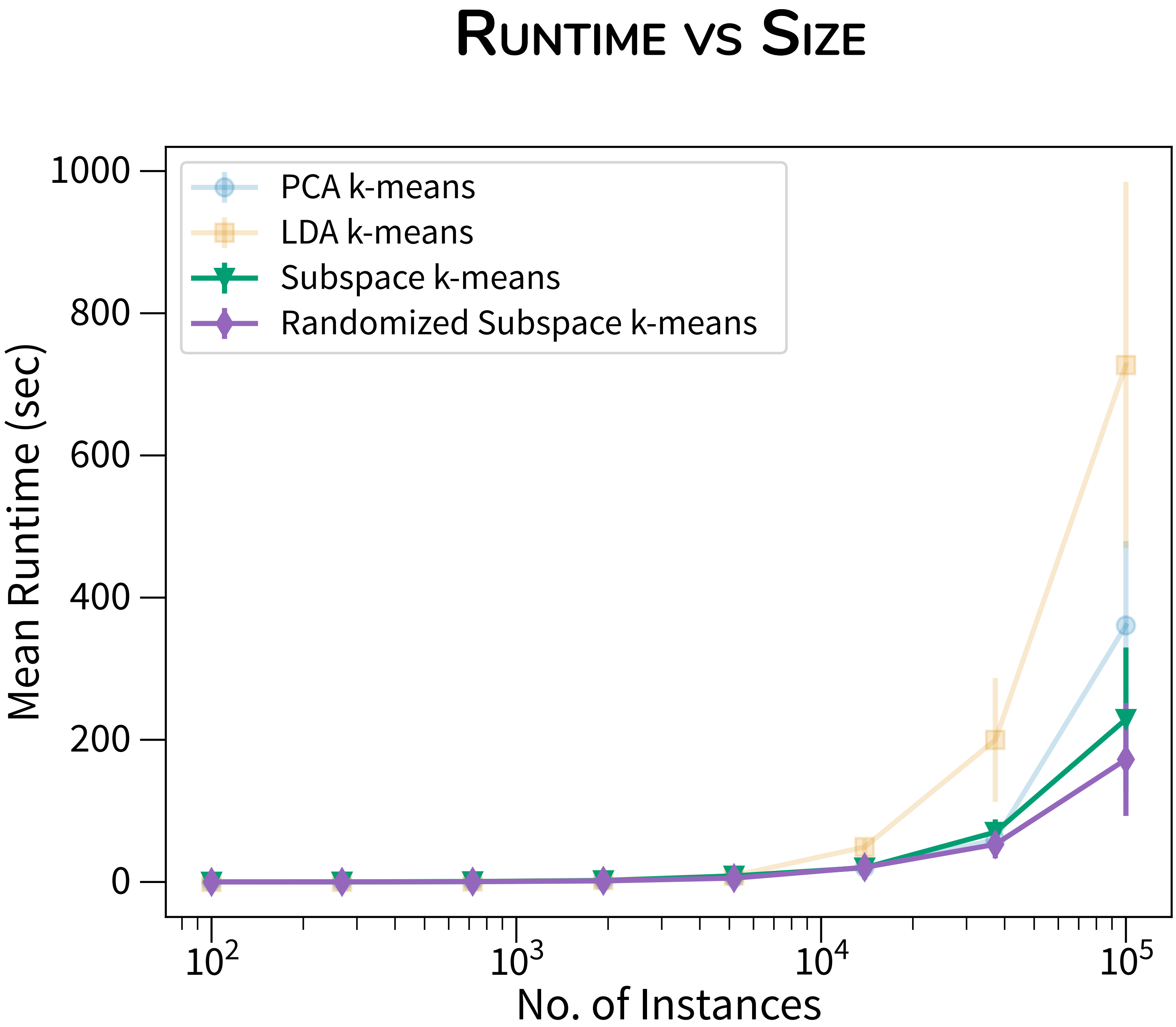
Results


Runtime vs Dimension
PCA k-means
LDA k-means
Subspace k-means
Randomized Subspace k-means
Mean Runtime (sec)
10^0
10^1
10^2
10^3
10^2
10^3
10^4
No. of Features

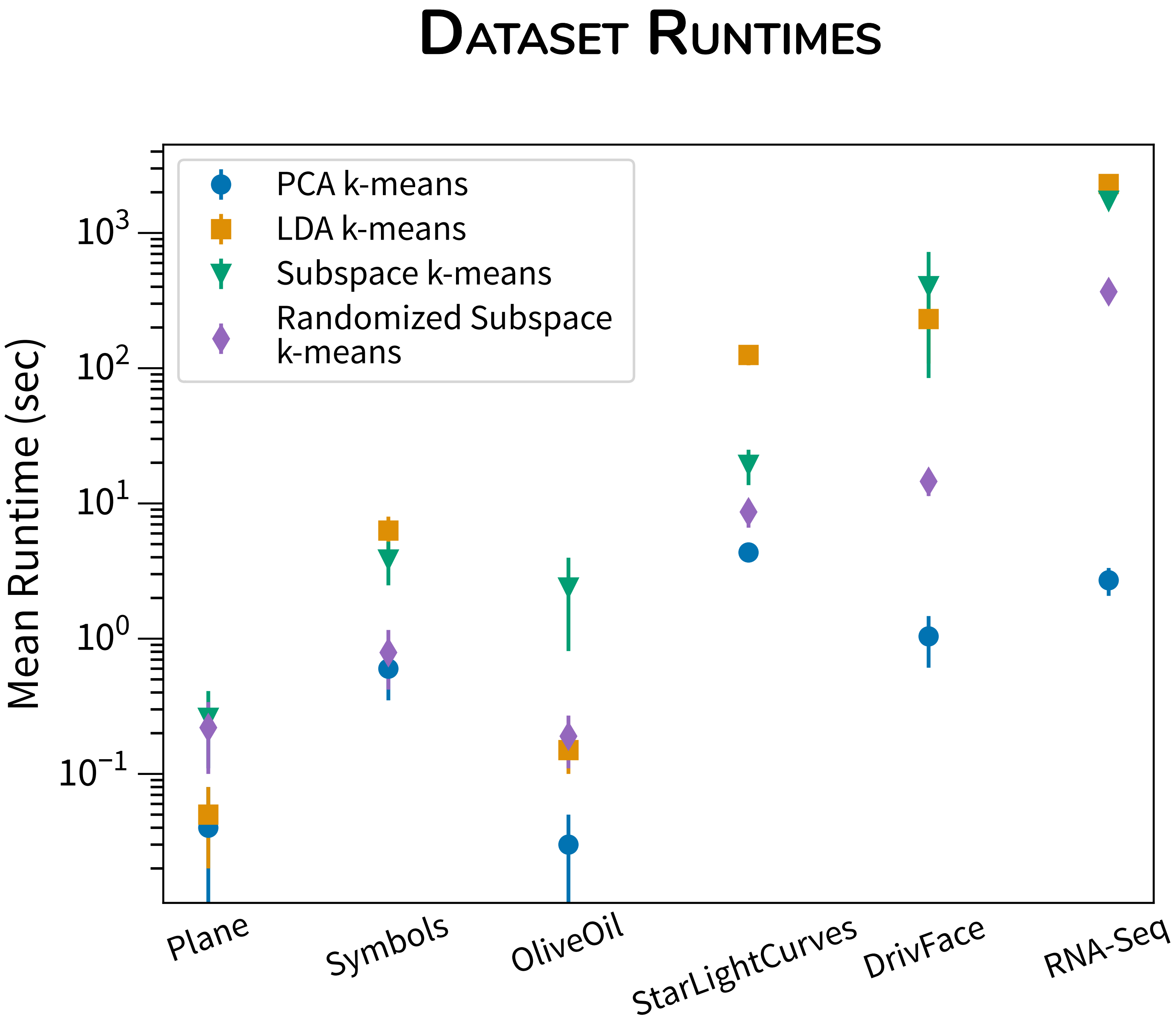

Runtime vs Dimension
PCA k-means
LDA k-means
Subspace k-means
Randomized Subspace k-means
Mean Runtime (sec)
No. of Features
10^0
10^1
10^2
10^3
10^2
10^3
10^4



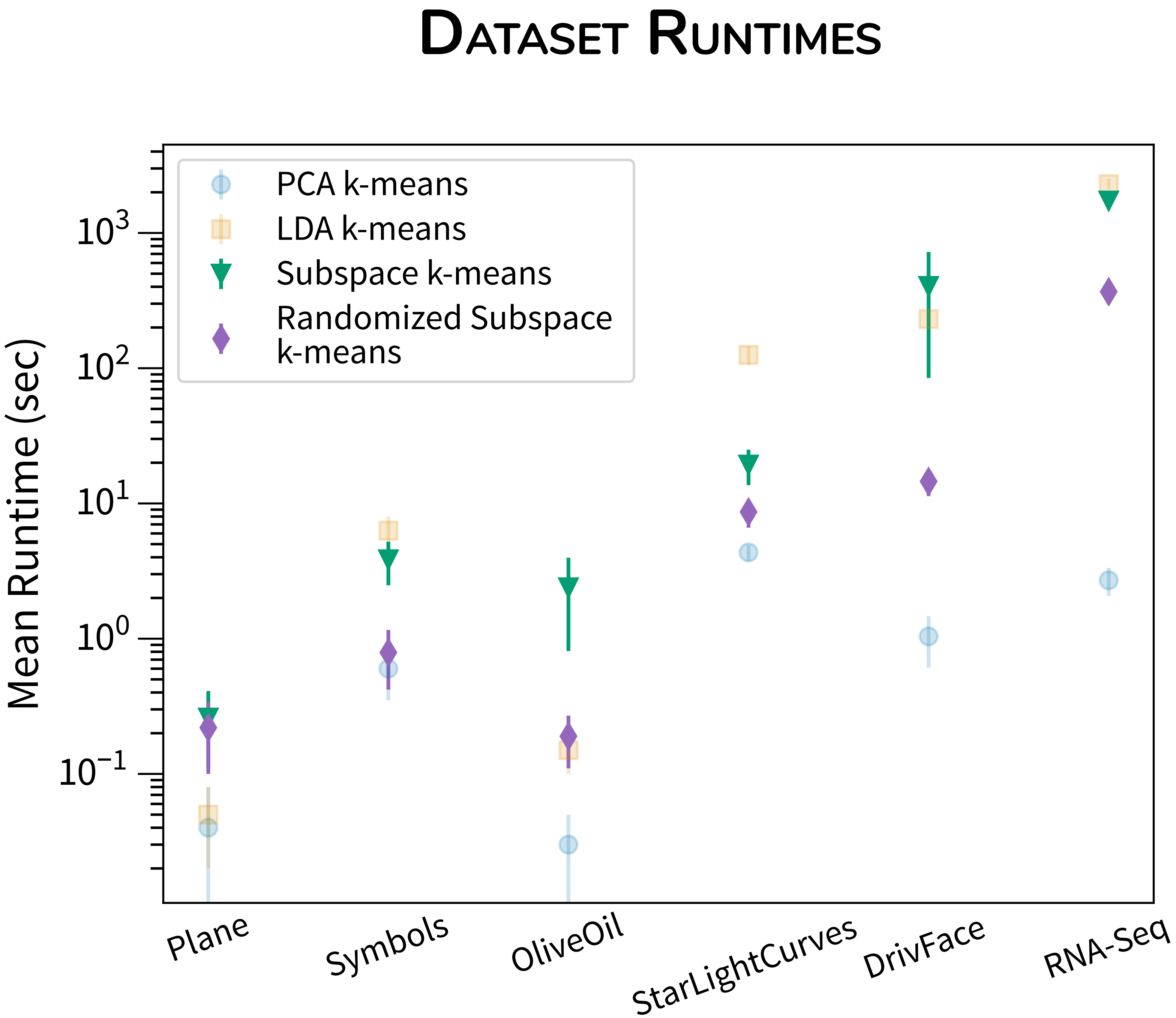
Runtime vs Size
Mean Runtime (sec)
No. of Instances
PCA k-means
LDA k-means
Subspace k-means
Randomized Subspace k-means
0
200
400
600
800
1000
10^2
10^3
10^4
10^5



Runtime vs Size
PCA k-means
LDA k-means
Subspace k-means
Randomized Subspace k-means
Mean Runtime (sec)
No. of Instances
1000
800
600
400
200
0
10^2
10^3
10^4
10^5



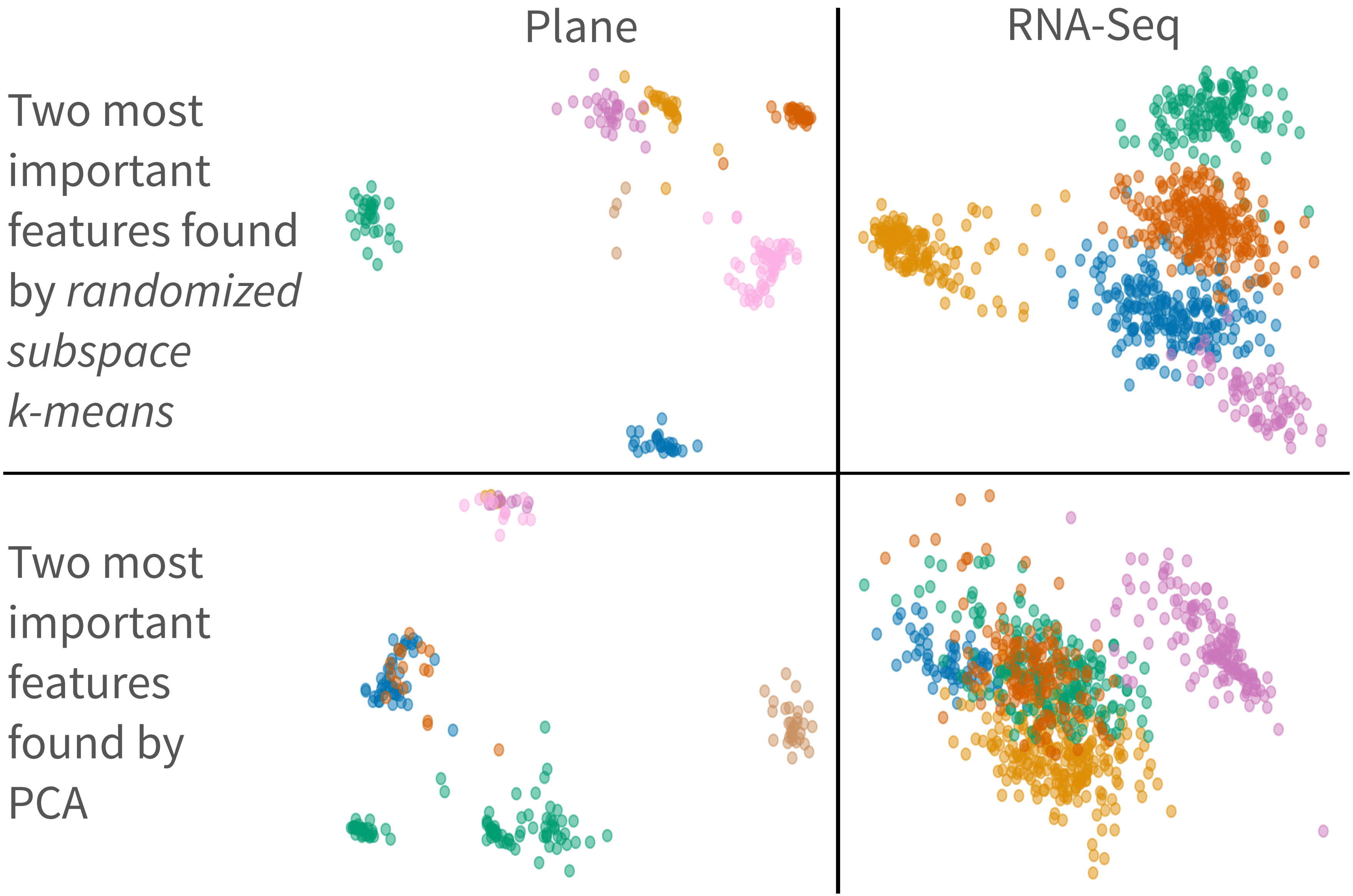
Dataset Runtimes
PCA k-means
LDA k-means
Subspace k-means
Randomized Subspace k-means
Mean Runtime (sec)
10^3
10^2
10^1
10^0
10^{-1}
Plane
Symbols
OliveOil
StarLightCurves
DrivFace
RNA-Seq


Dataset Runtimes
Mean Runtime (sec)
10^3
10^2
10^1
10^0
10^{-1}
PCA k-means
LDA k-means
Subspace k-means
Randomized Subspace k-means
Plane
Symbols
OliveOil
StarLightCurves
DrivFace
RNA-Seq

Clustering Quality (NMI)

	Randomized Subspace k-means	Subspace k-means	PCA k-means	LDA k-means
Plane	**0.835**	0.825	0.804	0.728
Symbols	**0.788**	0.742	0.762	0.745
OliveOil	0.609	0.657	0.673	**0.690**
StarLight Curves	**0.546**	0.542	0.507	0.542
DrivFace	0.191	0.205	0.203	**0.209**
RNA-Seq	0.659	0.679	**0.680**	0.668

Subspace Projections



Conclusions

Highlights

- significant performance increase
- no reduction in clustering quality

Future Work

- improve scatter matrix complexity
- k-means extensions
- test on more/larger datasets

QUESTIONS?

NMI

$$NMI(C,T) = \frac{I(C,T)}{\sqrt{H(C)H(T)}}$$

$C \equiv$ cluster assignments

$T \equiv$ ground truth

$I(C,T) \equiv$ mutual information

$H(C) \equiv$ entropy of cluster assignments

$H(T) \equiv$ entropy of ground truth

Randomized EVD

Approximate range: $Y = A\Omega$

Obtain orthonormal basis: $Y = QR$

Factorize: $A \approx QQ^*A$

EVD on $B = Q^*A$