Intro to ML

March 12, 2019
Data Science CSCI 1951A
Brown University
Instructor: Ellie Pavlick
HTAs: Wennie Zhang, Maulik Dang, Gurnaaz Kaur

Announcements

- 😟 MR grades—talk to your UTA if there are concerns
- 😐 Stats HW—things have happened…
- 😀 ML HW out today, due next Friday (Spring Break = 1 late day)
- 😐 Data Viz HW after the break

Today

- Peak-Ahead: Overfitting/Train-Test Splits
- Supervised vs. Unsupervised Learning
- K-Means and EM

Regression Analysis in Stats	Regression in ML

Regression Analysis in Stats	Regression in ML
☑ Make claims about whether there is a meaningful relationship between X and Y	

Regression Analysis in Stats

- ☑ Make claims about whether there is a meaningful relationship between X and Y

Regression in ML

- ☑ Given X, predict Y; deploy a model to make predictions for new inputs

Regression Analysis in Stats

- ☑ Make claims about whether there is a meaningful relationship between X and Y
- ☑ (Often) interested in causation; focus on controls and removing colinearity

Regression in ML

- ☑ Given X, predict Y; deploy a model to make predictions for new inputs

Regression Analysis in Stats

- ☑ Make claims about whether there is a meaningful relationship between X and Y
- ☑ (Often) interested in causation; focus on controls and removing colinearity

Regression in ML

- ☑ Given X, predict Y; deploy a model to make predictions for new inputs
- ☑ Focused on prediction accuracy; exploiting correlation is totally fine

Regression Analysis in Stats

- ☑ Make claims about whether there is a meaningful relationship between X and Y
- ☑ (Often) interested in causation; focus on controls and removing colinearity
- ☑ A “result” is typically in the form of a significant relationship and/or practically relevant effect size

Regression in ML

- ☑ Given X, predict Y; deploy a model to make predictions for new inputs
- ☑ Focused on prediction accuracy; exploiting correlation is totally fine

Regression Analysis in Stats	Regression in ML
☑ Make claims about whether there is a meaningful relationship between X and Y	☑ Given X, predict Y; deploy a model to make predictions for new inputs
☑ (Often) interested in causation; focus on controls and removing colinearity	☑ Focused on prediction accuracy; exploiting correlation is totally fine
☑ A “result” is typically in the form of a significant relationship and/or practically relevant effect size	☑ A “result” is typically in the form of an improvement in prediction performance on a (held out) test set

Regression Analysis in Stats	Regression in ML
☑ Make claims about whether there is a meaningful relationship be	☑ Given X, predict Y; deploy a model to make predictions for
☑ (O ca co co	ction ng ly fine
☑ A "result" is typically in the form of a significant relationship and/or practically relevant effect size	☑ A "result" is typically in the form of an improvement in prediction performance on a (held out) test set

But! These are the same model.
These difference are "in general"/"by convention", not anything fundamental.

Regression Analysis in Stats	Regression in ML
☑ Make claims about whether there is a meaningful relationship between X and Y	☑ Given X, predict Y; deploy a model to make predictions for new inputs
☑ (n c ne collinearity	
☑ A "result" is typically in the form of a significant relationship and/or practically relevant effect size	☑ A "result" is typically in the form of an improvement in prediction performance on a (held out) test set

Different scientific communities with different goals.

Regression Analysis in Stats	Regression in ML
☑ Make claims about whether there is a meaningful relationship b	☑ Given X, predict Y; deploy a model to make predictions for
☑ (	n
☑ A	n
the form of a significant relationship and/or practically relevant effect size	the form of an improvement in prediction performance on a (held out) test set

Different scientific communities with different goals.

(and different software packages :))

<- R, stats_models, STATA

sklearn, matlab, pytorch ->

Regression Analysis in Stats

- Make claims about whether there is a meaningful relationship between X and Y
- (Often) interested in causation; focus on controls and removing colinearity
- pically in significant nd/or evant

Regression in ML

- Given X, predict Y; deploy a model to make predictions for new inputs
- Focused on prediction accuracy; exploiting correlation is totally fine
- A “result” is typically in the form of an improvement in prediction performance on a (held out) test set

In the limit, I think these goals *are* the same.
Even if we care about prediction (and we want to do it using as few models as possible), shouldn't we get the best performance by modeling the "true" underlying process?

A correct explanatory/causal models necessarily make right predictions, but not vice-versa.

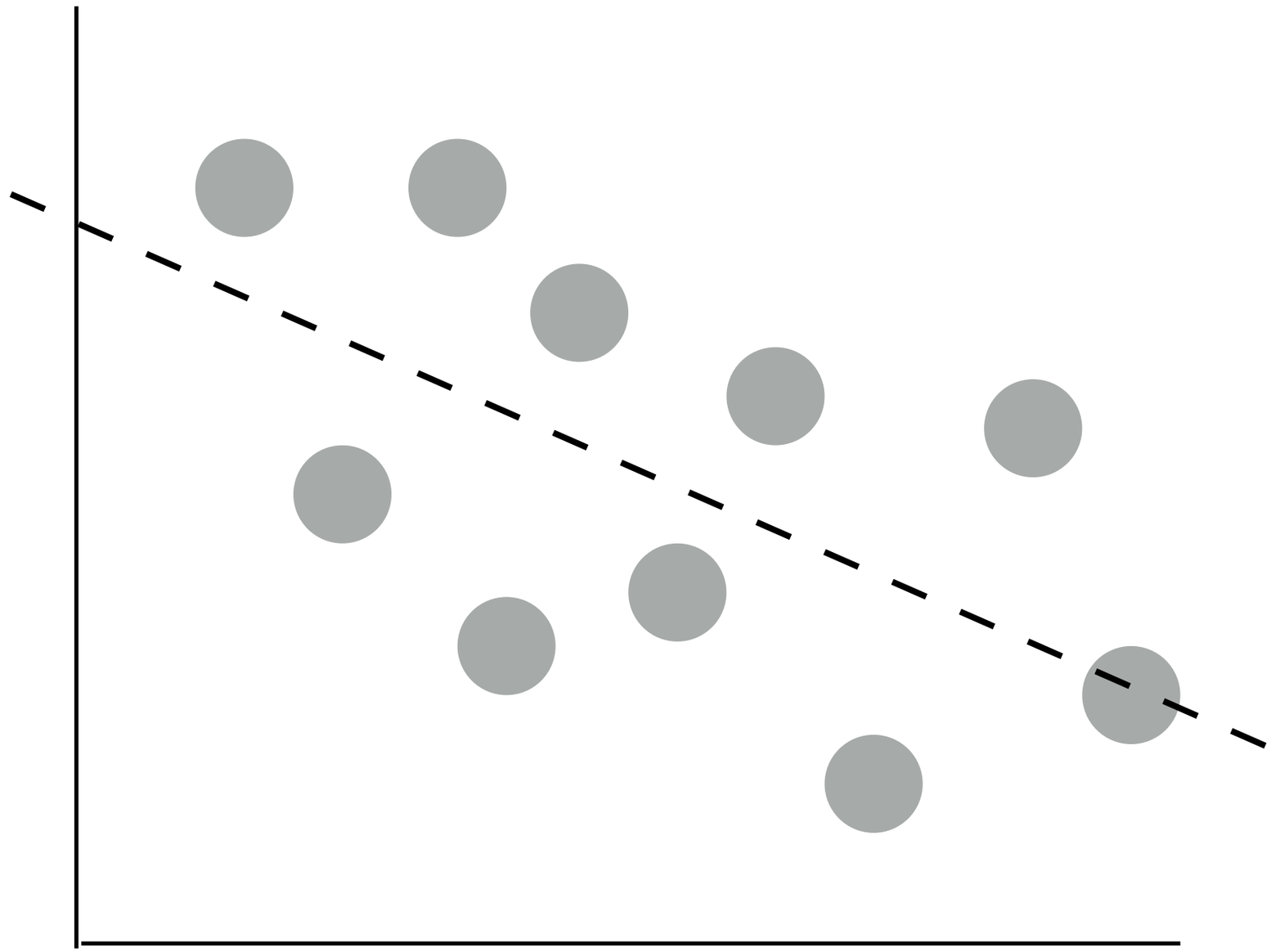
- ☑ A "result" is typically in the form of an improvement in prediction performance on a (held out) test set

Counter argument: You can get perfect* predictive performance with the wrong model. We were extremely good at predicting whether objects would fall or float long before we knew about gravity.

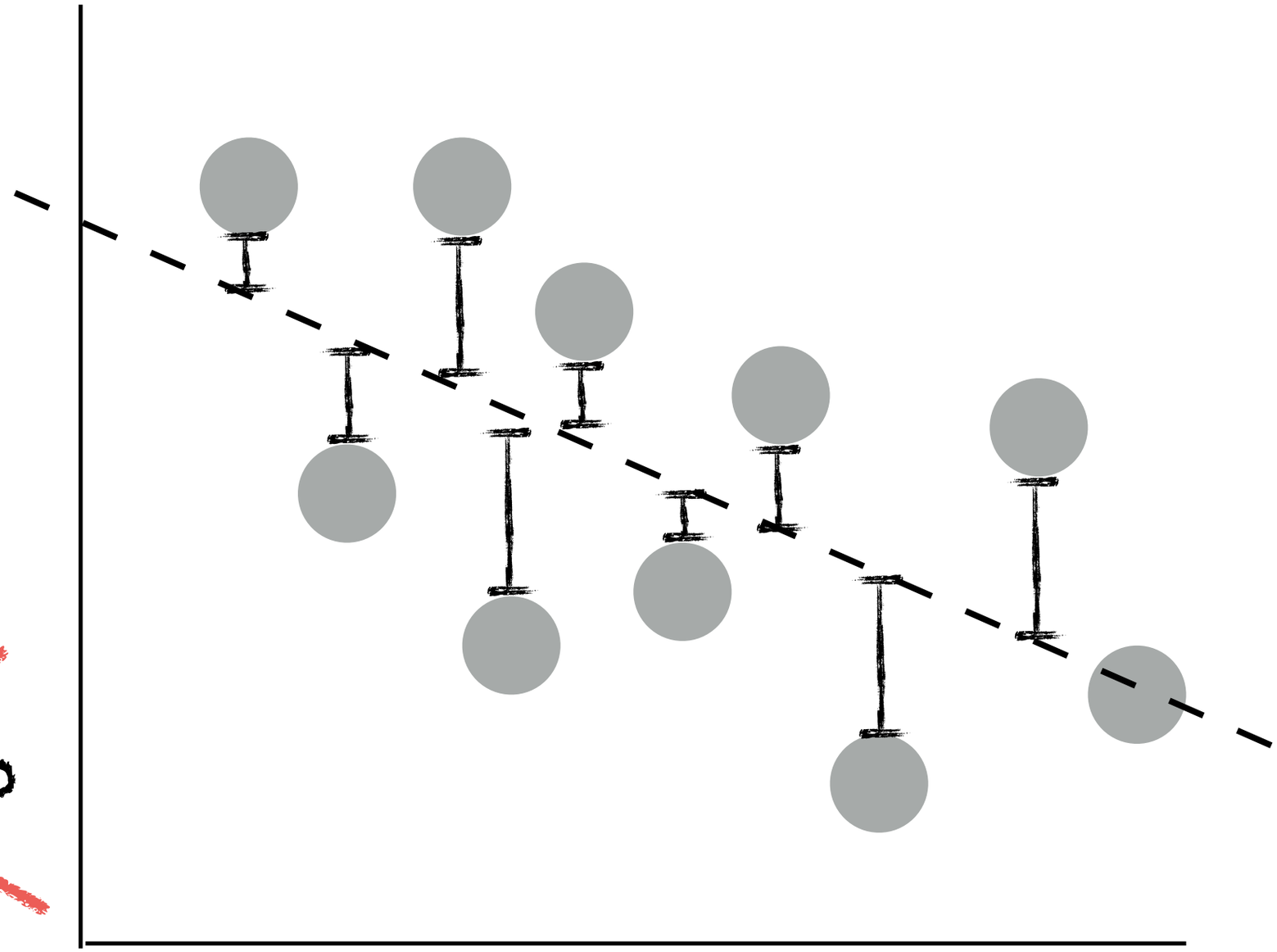
Explanatory/causal models are hard! We might never get there. Maybe empirically accurate predictions can lead, and theory/explanation will follow?

- A “result” is typically in the form of a significant relationship and/or practically relevant effect size

Train/Test Splits

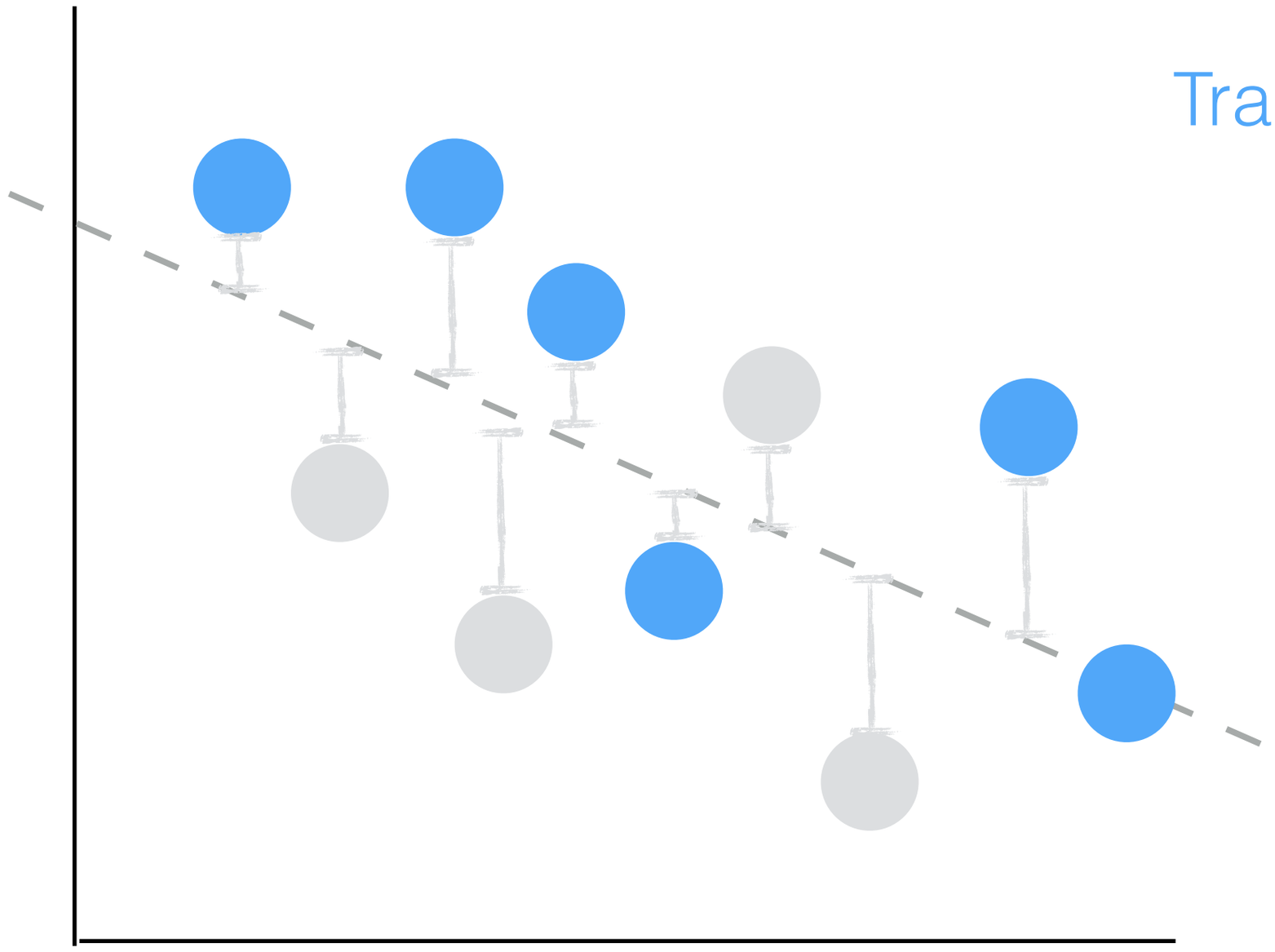


Train/Test Splits

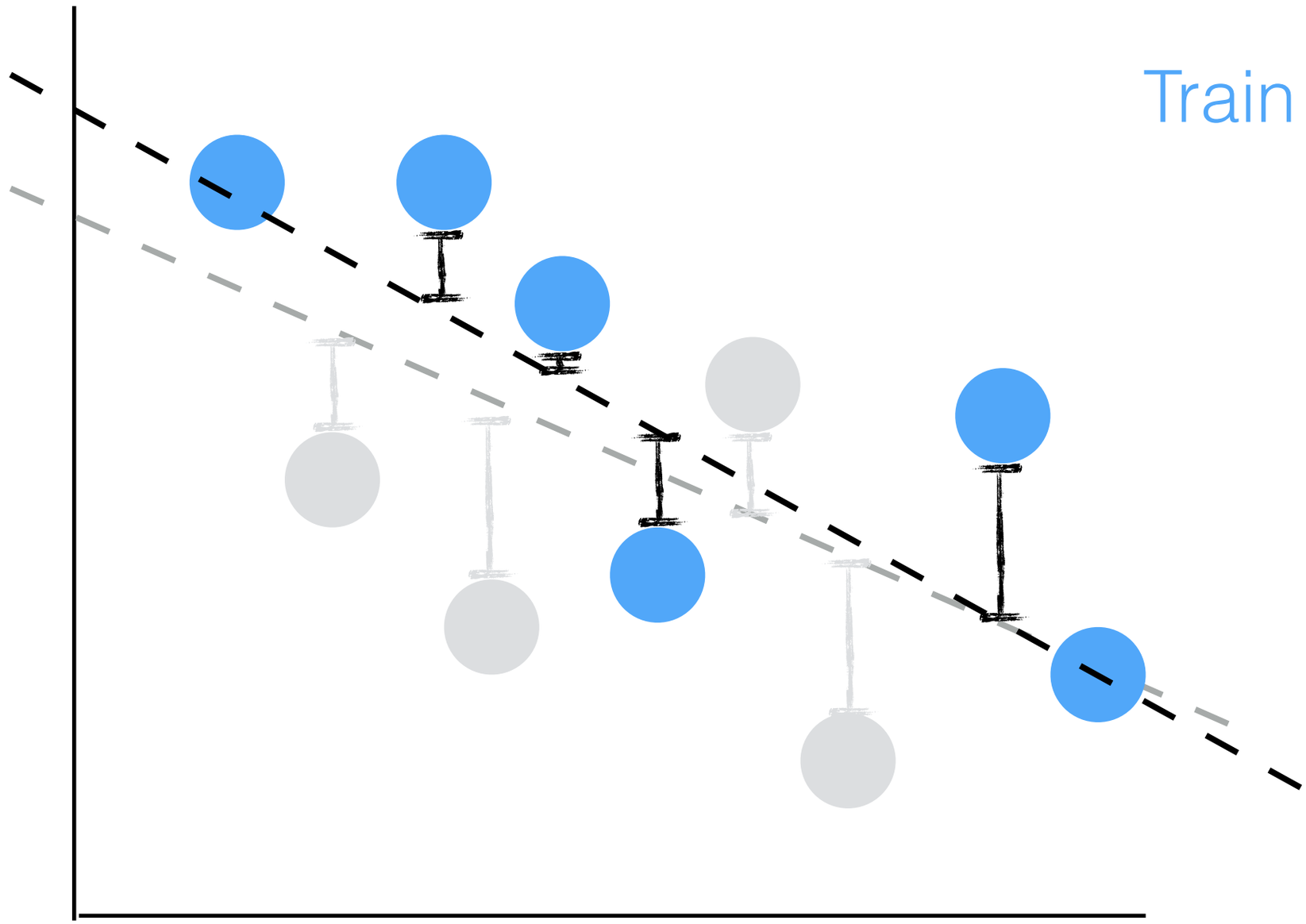


MSE = 10

Train/Test Splits

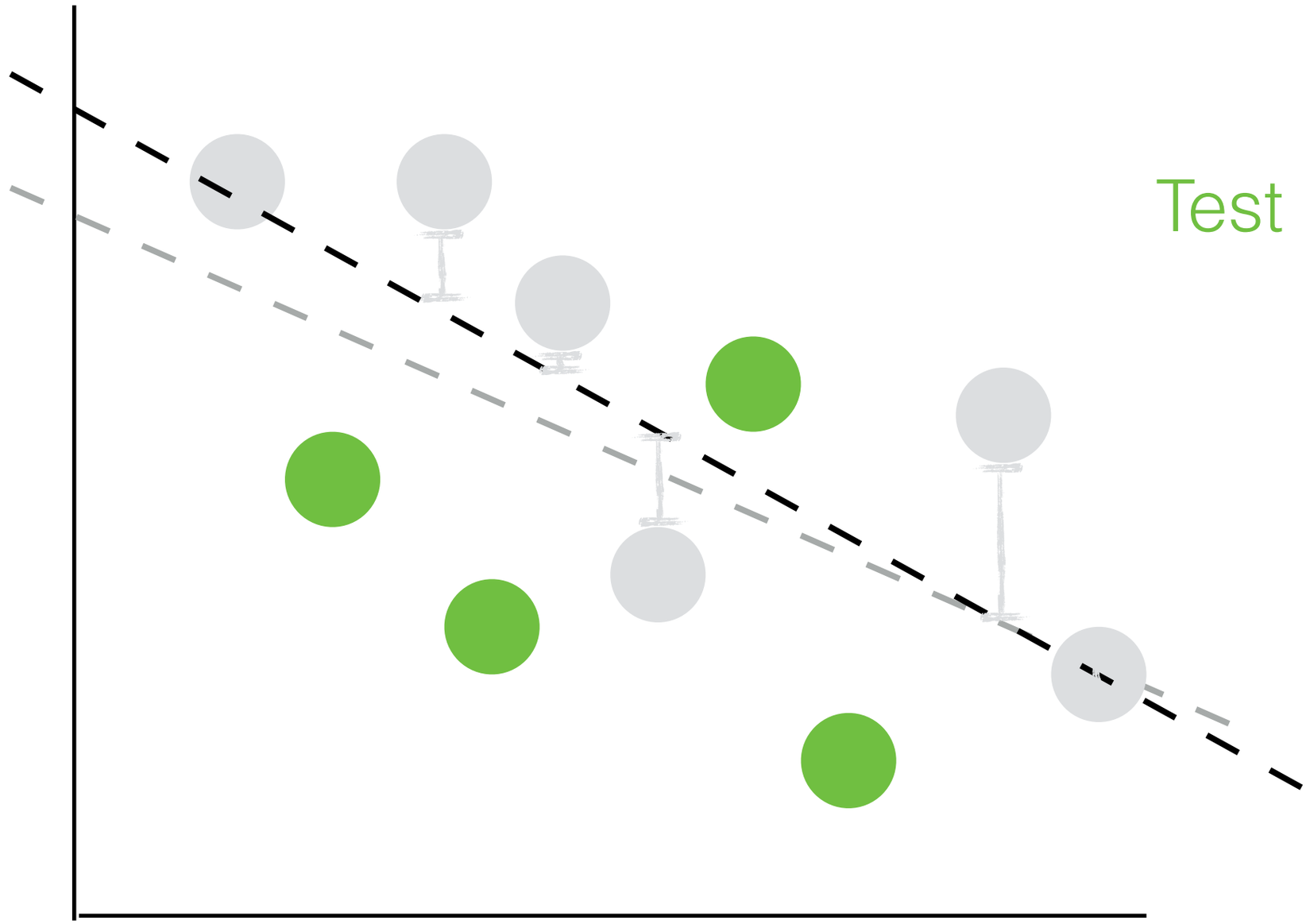


Train/Test Splits



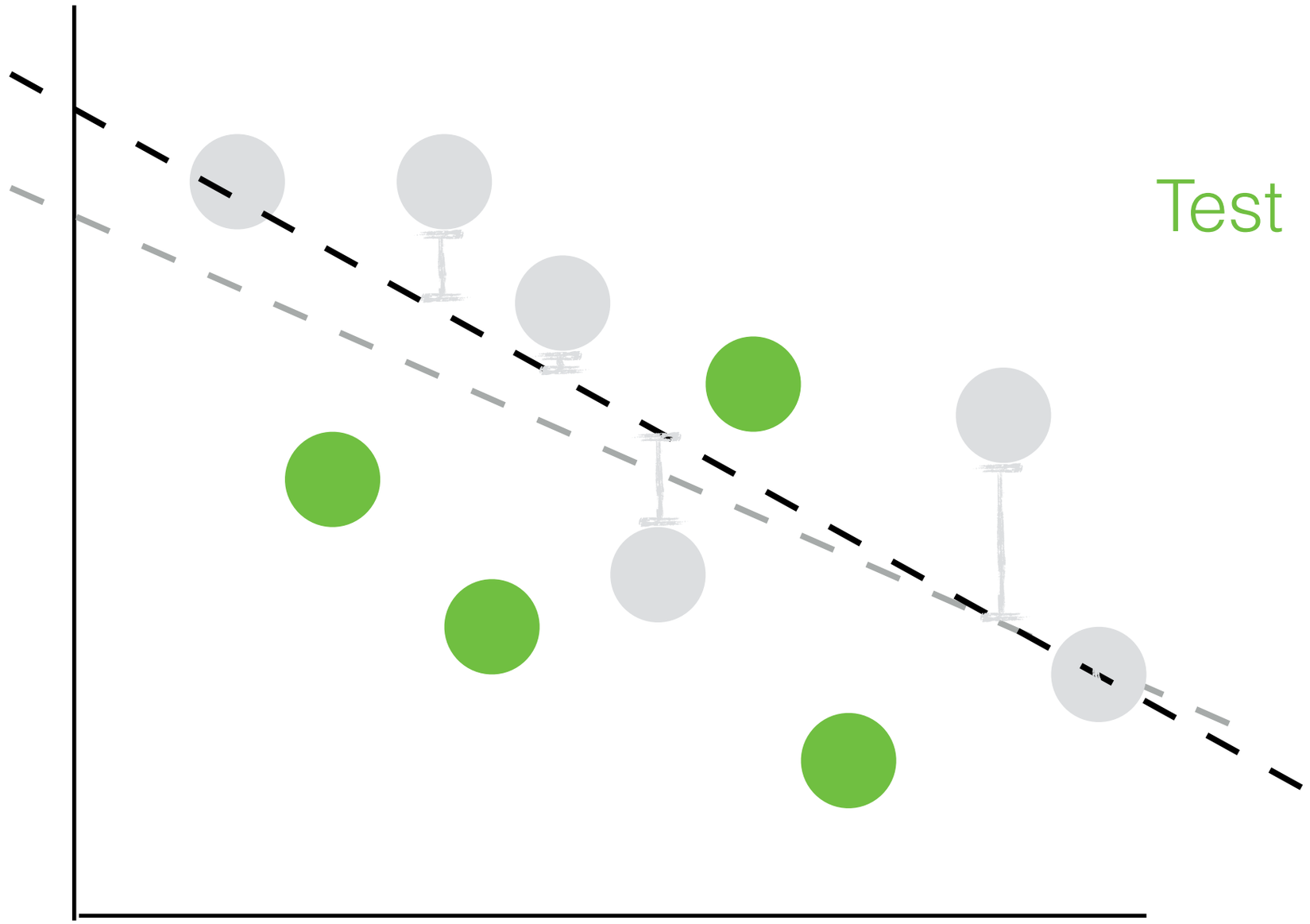
MSE = 6

Train/Test Splits

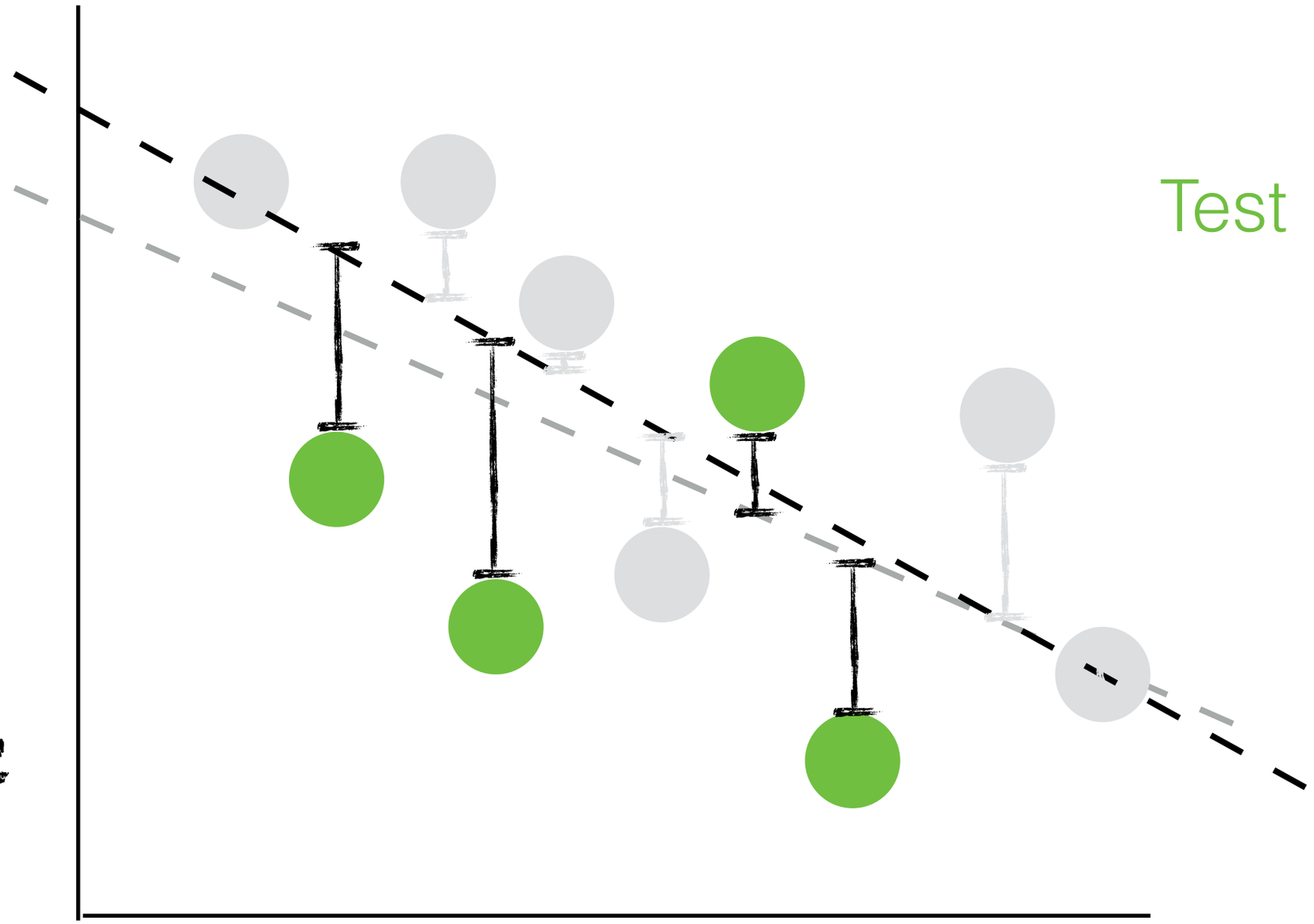


Clicker Question!

Train/Test Splits

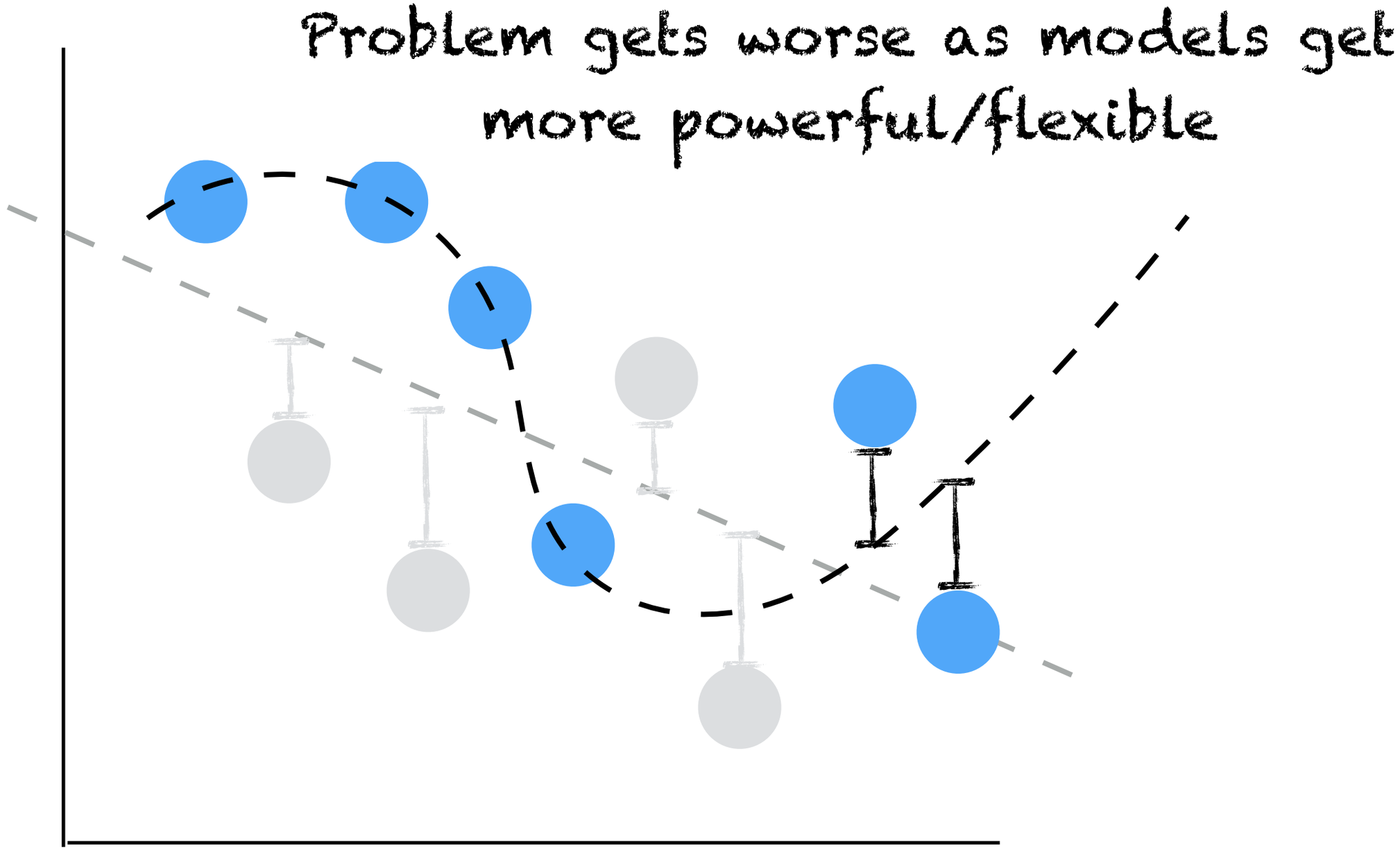


Train/Test Splits



MSE = 12

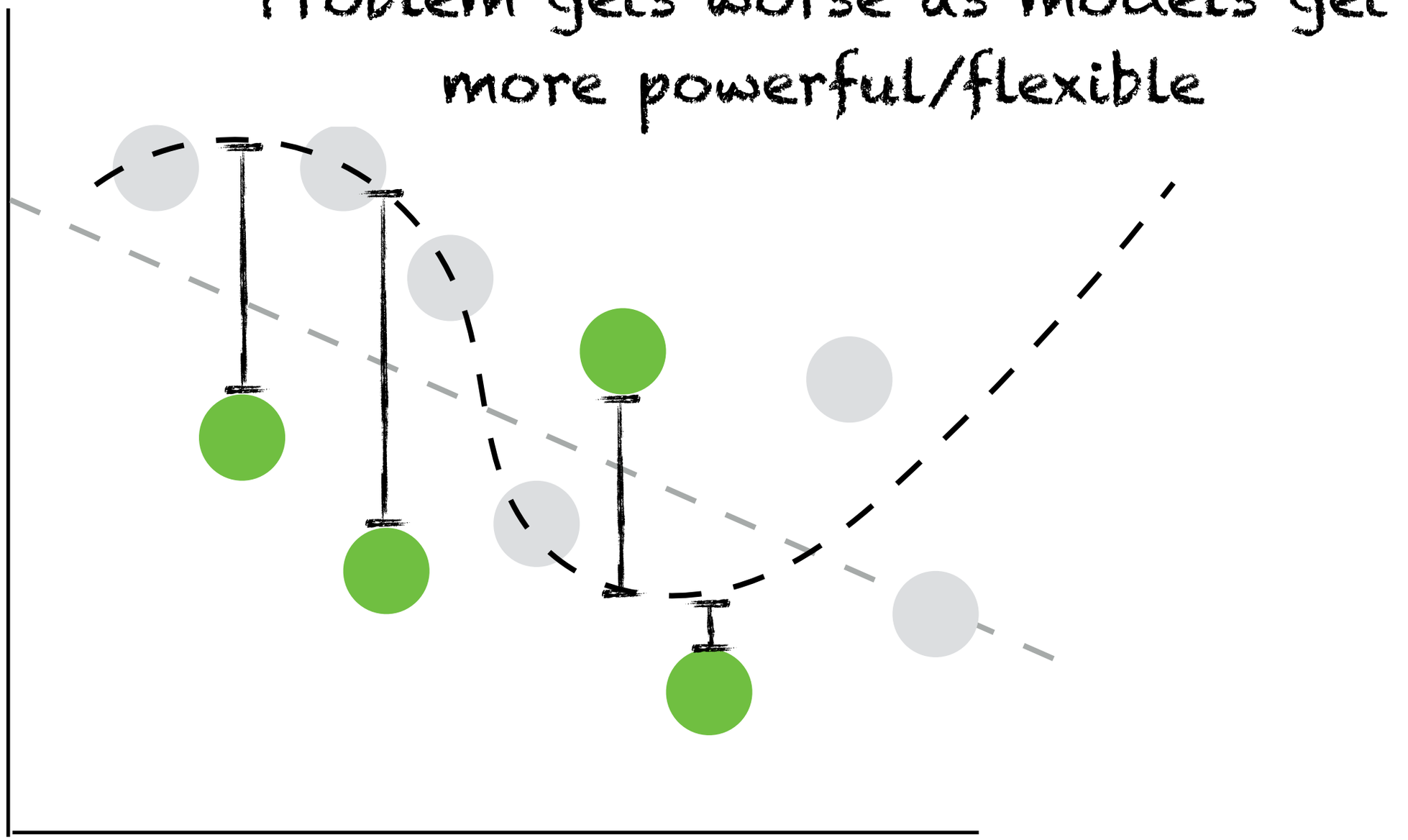
Train/Test Splits



MSE = 4

Train/Test Splits

Problem gets worse as models get more powerful/flexible



MSE = 14

Regression Analysis in Stats

- Make claims about whether there is a meaningful relationship between X and Y
- (Often) interested in causation; focus on controls and removing colinearity
- A "result" is typically in the form of a significant relationship and/or practically relevant effect size

Regression in ML

- Given X, predict Y; deploy a model to make predictions for new inputs
- Focused on prediction accuracy; exploiting correlation is totally fine
- A "result" is typically in the form of an improvement in prediction performance on a (held out) test set

Regression Analysis in Stats

- Make claims about whether there is a meaningful relationship between X and Y
- (Often) interested in causation; focus on controls and removing colinearity
- A "result" is typically in the form of a significant relationship and/or practically relevant effect size
- Avoid overfitting by preferring simple models; avoid overclaiming by accounting for "degrees of freedom" when computing p values

Regression in ML

- Given X, predict Y; deploy a model to make predictions for new inputs
- Focused on prediction accuracy; exploiting correlation is totally fine
- A "result" is typically in the form of an improvement in prediction performance on a (held out) test set

Regression Analysis in Stats

- Make claims about whether there is a meaningful relationship between X and Y
- (Often) interested in causation; focus on controls and removing colinearity
- A "result" is typically in the form of a significant relationship and/or practically relevant effect size
- Avoid overfitting by preferring simple models; avoid overclaiming by accounting for "degrees of freedom" when computing p values

Regression in ML

- Given X, predict Y; deploy a model to make predictions for new inputs
- Focused on prediction accuracy; exploiting correlation is totally fine
- A "result" is typically in the form of an improvement in prediction performance on a (held out) test set
- Avoid overfitting through regularization; avoid overclaiming by maintaining train/test splits and reporting test performance

Supervised vs. Unsupervised Learning

Supervised vs. Unsupervised Learning

- Supervised: Explicit data labels

Supervised vs. Unsupervised Learning

- Supervised: Explicit data labels
  - Sentiment analysis—review text -> star ratings

Supervised vs. Unsupervised Learning

- Supervised: Explicit data labels
  - Sentiment analysis—review text -> star ratings
  - Image tagging—image -> caption

Supervised vs. Unsupervised Learning

- Supervised: Explicit data labels
  - Sentiment analysis—review text -> star ratings
  - Image tagging—image -> caption
- Unsupervised: No explicit labels

Supervised vs. Unsupervised Learning

- Supervised: Explicit data labels
  - Sentiment analysis—review text -> star ratings
  - Image tagging—image -> caption
- Unsupervised: No explicit labels
  - Clustering—find groups similar customers

Supervised vs. Unsupervised Learning

- Supervised: Explicit data labels
  - Sentiment analysis—review text -> star ratings
  - Image tagging—image -> caption
- Unsupervised: No explicit labels
  - Clustering—find groups similar customers
  - Dimensionality Reduction—find features that differentiate individuals

Supervised vs. Unsupervised Learning

- Supervised: Explicit data labels
  - Sentiment analysis—review text -> star ratings
  - Image tagging—image -> caption
- Unsupervised: No explicit labels
  - Clustering—find groups similar customers
  - Dimensionality Reduction—find features that differentiate individuals

Supervised vs. Unsupervised Learning

- Supervised: Explicit data labels
  - Sentiment analysis—review text -> star ratings
  - Image tagging—image -> caption
- Unsupervised: No explicit labels
  - Clustering—find groups similar customers
  - Dimensionality Reduction—find features that differentiate individuals

Oh you thought it was that simple? How cute…

Oh you thought it was that simple? How cute…

- Semi Supervised—Combining large amounts of unlabelled with smaller amounts of labelled (pretraining)

Oh you thought it was that simple? How cute…

- Semi Supervised—Combining large amounts of unlabelled with smaller amounts of labelled (pretraining)

- Weakly/Distantly Supervised—using noisy labels or partial labels (bootstrapping, automatically-labeled data)

Oh you thought it was that simple? How cute…

- Semi Supervised—Combining large amounts of unlabelled with smaller amounts of labelled (pretraining)

- Weakly/Distantly Supervised—using noisy labels or partial labels (bootstrapping, automatically-labeled data)

- Reinforcement Learning—label on the result of a sequence of actions, but not on each action (games, robotics)

Oh you thought it was that simple? How cute…

- Semi Supervised—Combining large amounts of unlabelled with smaller amounts of labelled (pretraining)

- Weakly/Distantly Supervised—using noisy labels or partial labels (bootstrapping, automatically-labeled data)

- Reinforcement Learning—label on the result of a sequence of actions, but not on each action (games, robotics)

- “Found” Data… (?)

Unsupervised Learning

Unsupervised Learning

- “Finding structure in data” (vs. predicting labels)

Unsupervised Learning

- “Finding structure in data” (vs. predicting labels)
- In data science, this is typically for “exploratory analysis”. “What the $@%! is this data even?! Enlighten me.”

Unsupervised Learning

- “Finding structure in data” (vs. predicting labels)
- In data science, this is typically for “exploratory analysis”. “What the $@%! is this data even?! Enlighten me.”
- Or for preprocessing/featurizing—e.g. so you can use article “topics” to predict clicks.

Unsupervised Learning

- “Finding structure in data” (vs. predicting labels)
- In data science, this is typically for “exploratory analysis”. “What the $@%! is this data even?! Enlighten me.”
- Or for preprocessing/featurizing—e.g. so you can use article “topics” to predict clicks.
- In ML, right now, used extensively for “pretraining” (e.g. autoencoding, dimensionality reduction, language modeling*)

Clustering

Discussion Question!

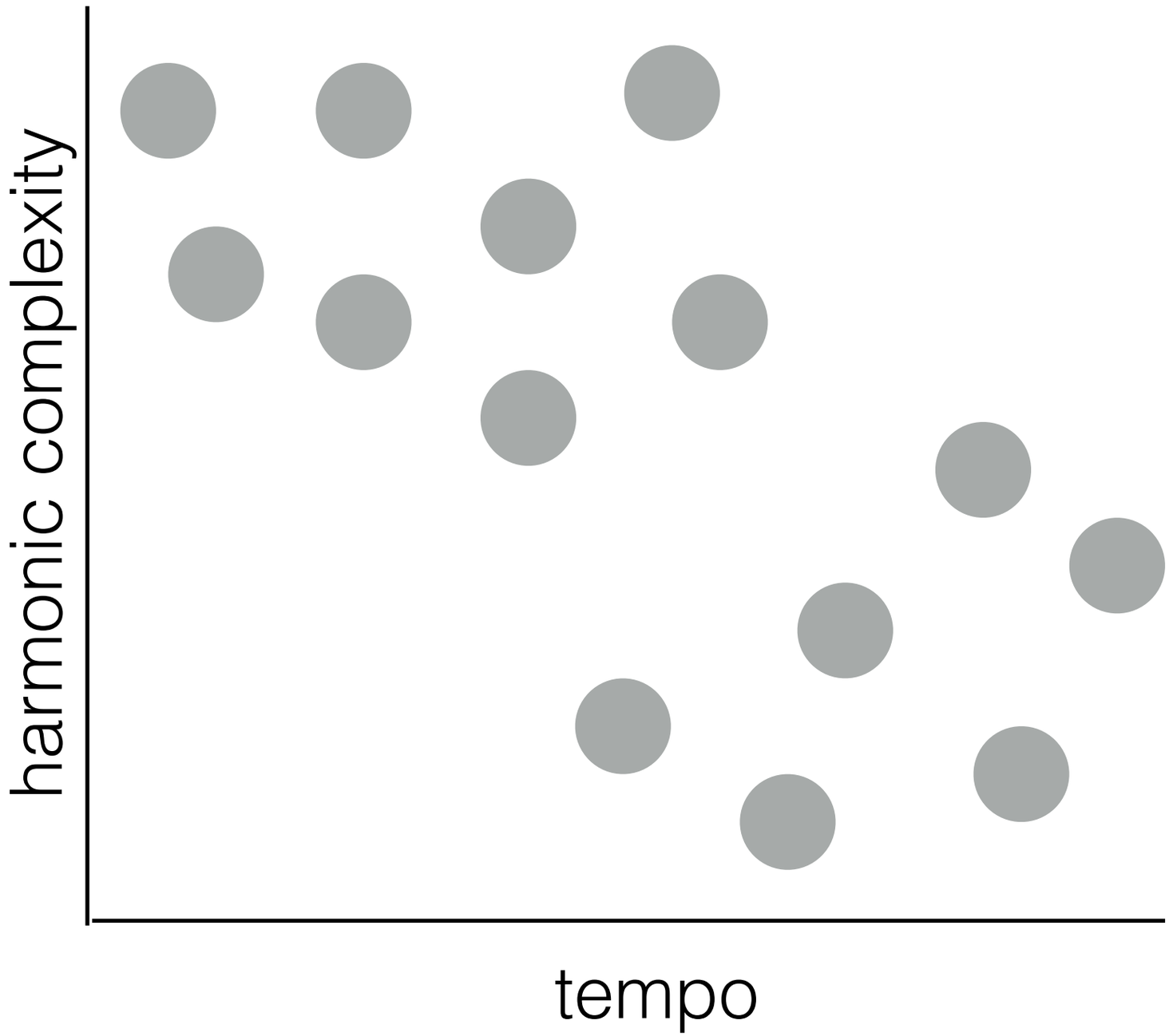
What is it good for…?

(…I’ve been talking a lot. Talk to meeee! :)…)

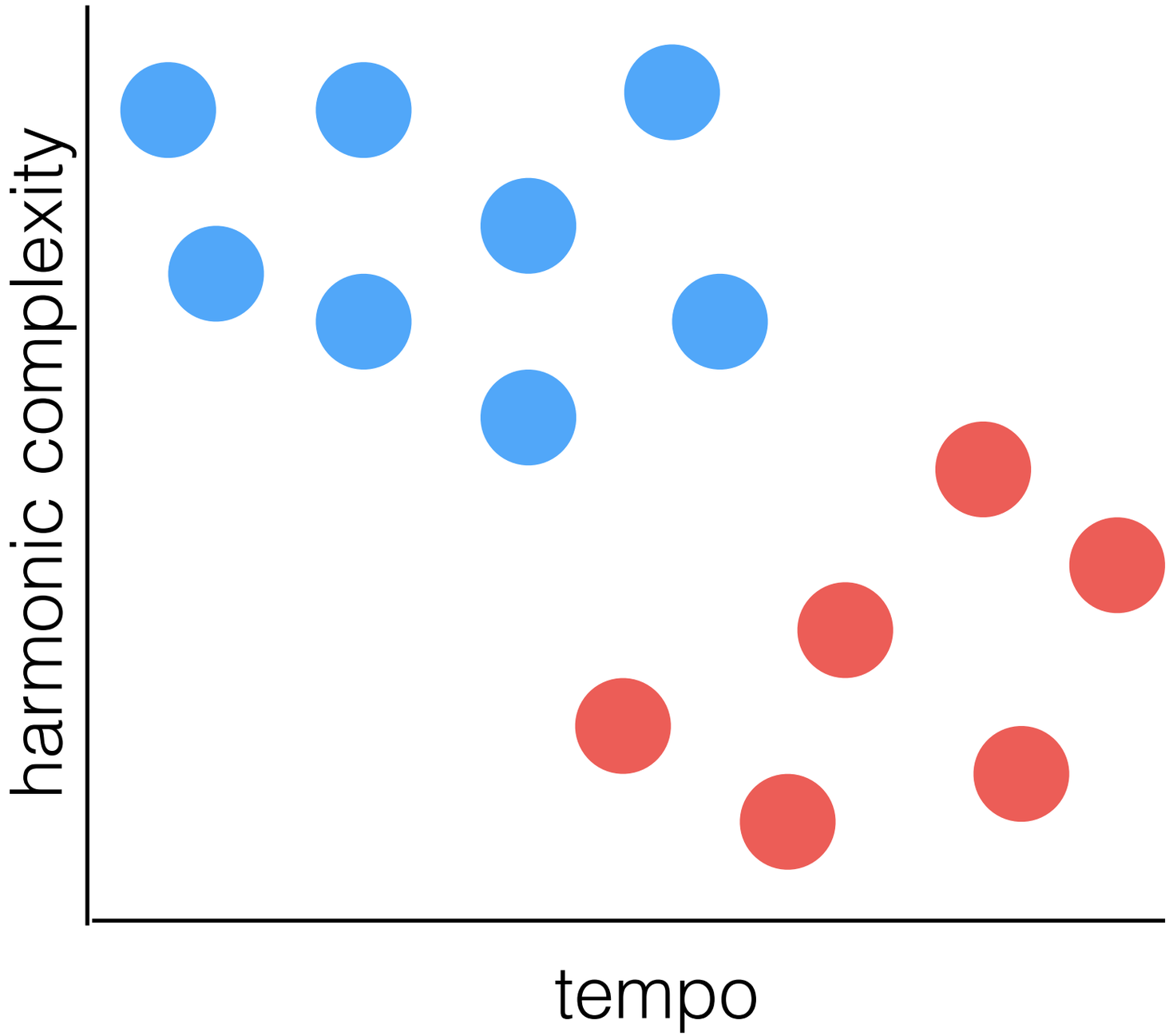
Clustering

- Find groups of customers with similar tastes
- Find topics within a set of news articles
- Find genres within a library of music
- Extrapolating—make predictions about your new business based on behavior of similar old businesses

Clustering



Clustering



K Means

```
define parameters: K, max_iter, min_diff

iter = 0
change = inf
means = [random() for _ in range(K)]
while iter < max_iter and change > min_diff:
    update_assignments()
    compute_new_means()
    change = max_i(dist(new_mean_i, old_mean_i))
    iter += 1
```

K Means

"Hyperparameters" (i.e. not model parameters)

```
define parameters: K, max_iter, min_diff

iter = 0
change = inf
means = [random() for _ in range(K)]
while iter < max_iter and change > min_diff:
    update_assignments()
    compute_new_means()
    change = max_i(dist(new_mean_i, old_mean_i))
    iter += 1
```

K Means

How many clusters we want to find

```
define parameters: K, max_iter, min_diff

iter = 0
change = inf
means = [random() for _ in range(K)]
while iter < max_iter and change > min_diff:
    update_assignments()
    compute_new_means()
    change = max_i(dist(new_mean_i, old_mean_i))
    iter += 1
```

K Means

When to quit. Things aren't changing, or we have gotten bored.

```
define parameters: K, max_iter, min_diff

iter = 0
change = inf
means = [random() for _ in range(K)]
while iter < max_iter and change > min_diff:
    update_assignments()
    compute_new_means()
    change = max_i(dist(new_mean_i, old_mean_i))
    iter += 1
```

K Means

Randomly guess what the means are (lots of ways to do this)

```
define parameters: K, max_iter, min_diff

iter = 0
change = inf
means = [random() for _ in range(K)]
while iter < max_iter and change > min_diff:
    update_assignments()
    compute_new_means()
    change = max_i(dist(new_mean_i, old_mean_i))
    iter += 1
```

K Means

Repeat until your hyperparameters say to stop

```
define parameters: K, max_iter, min_diff

iter = 0
change = inf
means = [random() for _ in range(K)]
while iter < max_iter and change > min_diff:
    update_assignments()
    compute_new_means()
    change = max_i(dist(new_mean_i, old_mean_i))
    iter += 1
```

K Means

Assign each point to its closest mean

```
define parameters: K, max_iter, min_diff

iter = 0
change = inf
means = [random() for _ in range(K)]
while iter < max_iter and change > min_diff:
    update_assignments()
    compute_new_means()
    change = max_i(dist(new_mean_i, old_mean_i))
    iter += 1
```

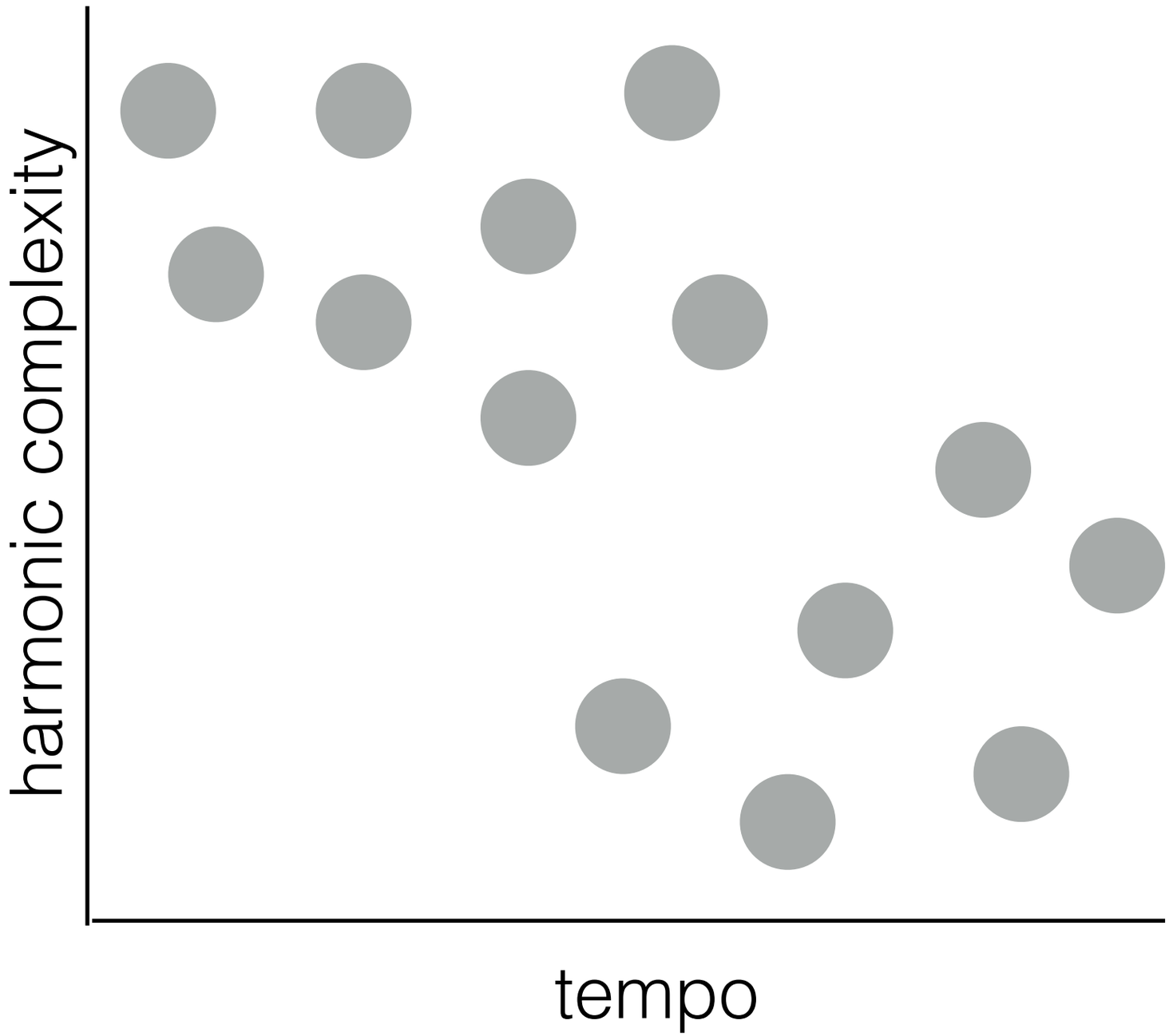
K Means

Recompute the means to be the mean of the points assigned to each cluster

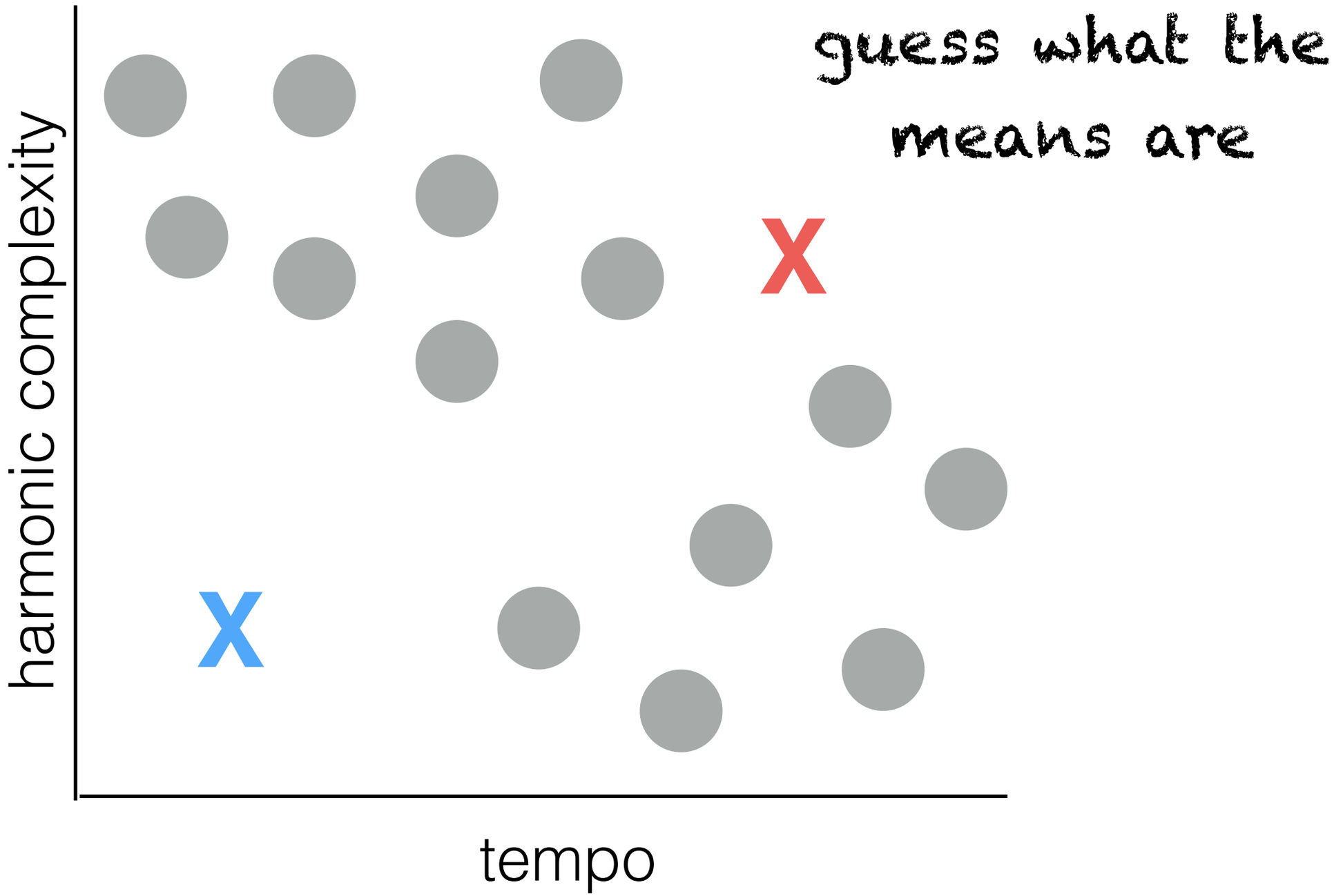
```
define parameters: K, max_iter, min_diff

iter = 0
change = inf
means = [random() for _ in range(K)]
while iter < max_iter and change > min_diff:
    update_assignments()
    compute_new_means()
    change = max_i(dist(new_mean_i, old_mean_i))
    iter += 1
```

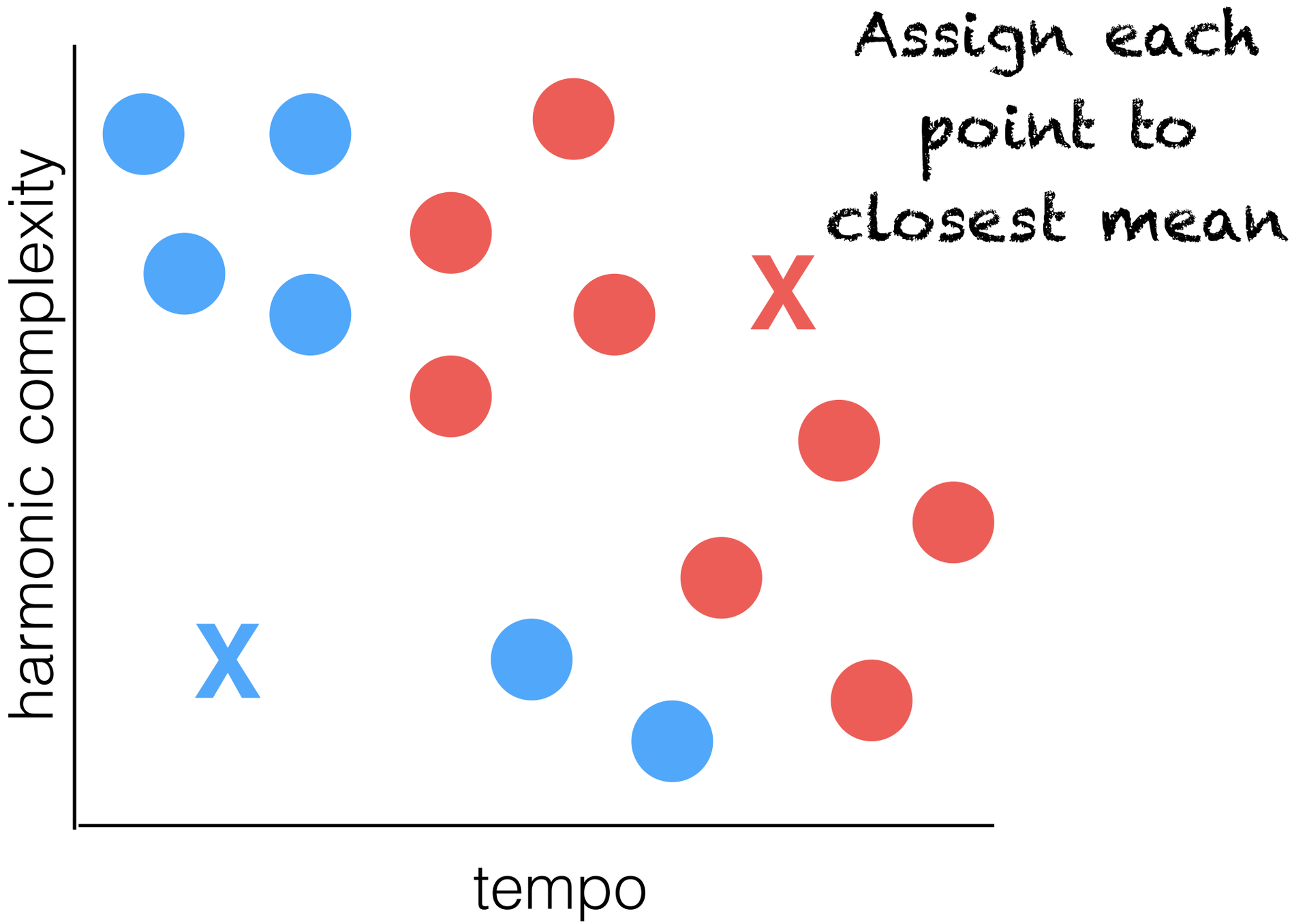
K Means



K Means

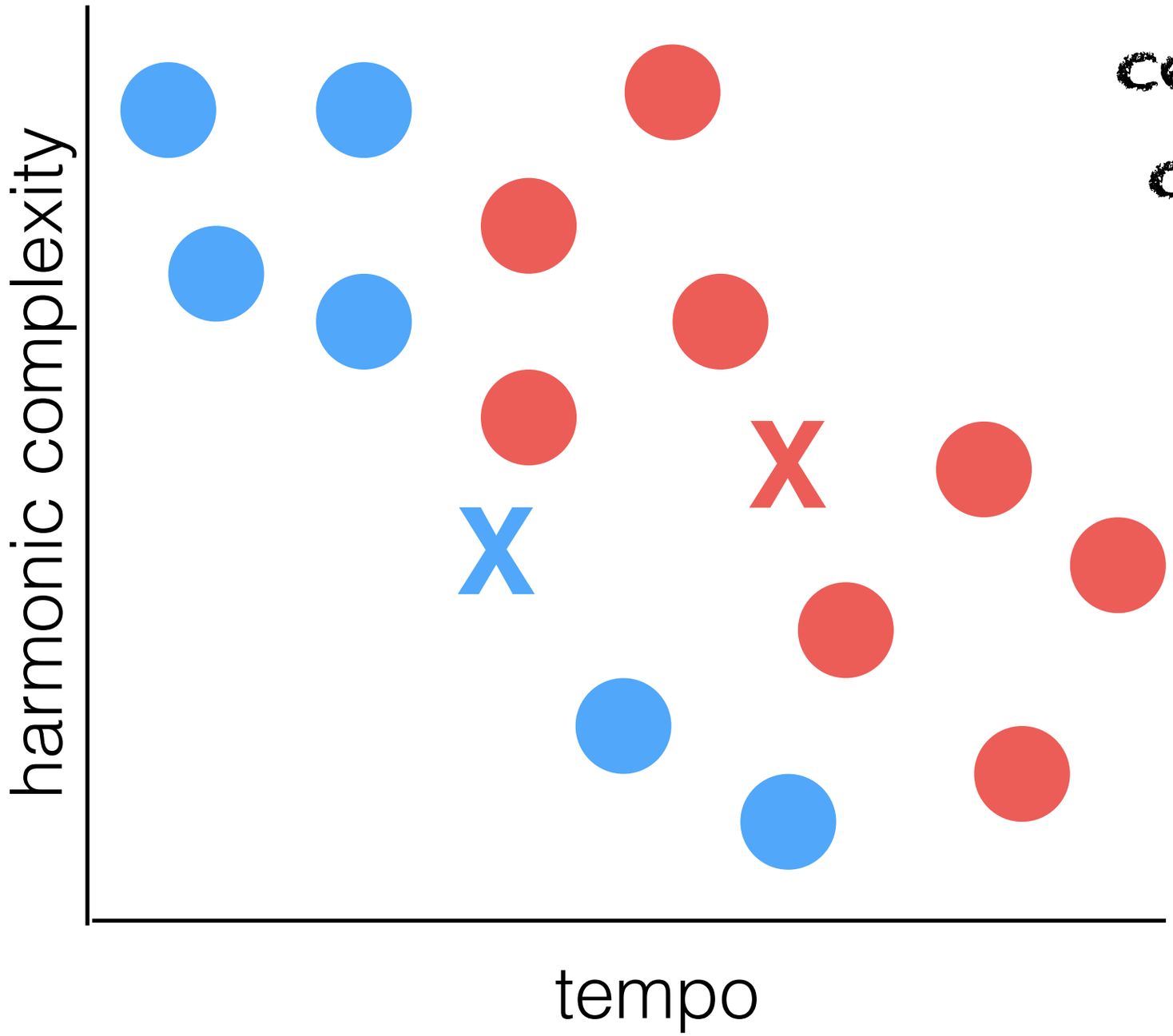


K Means

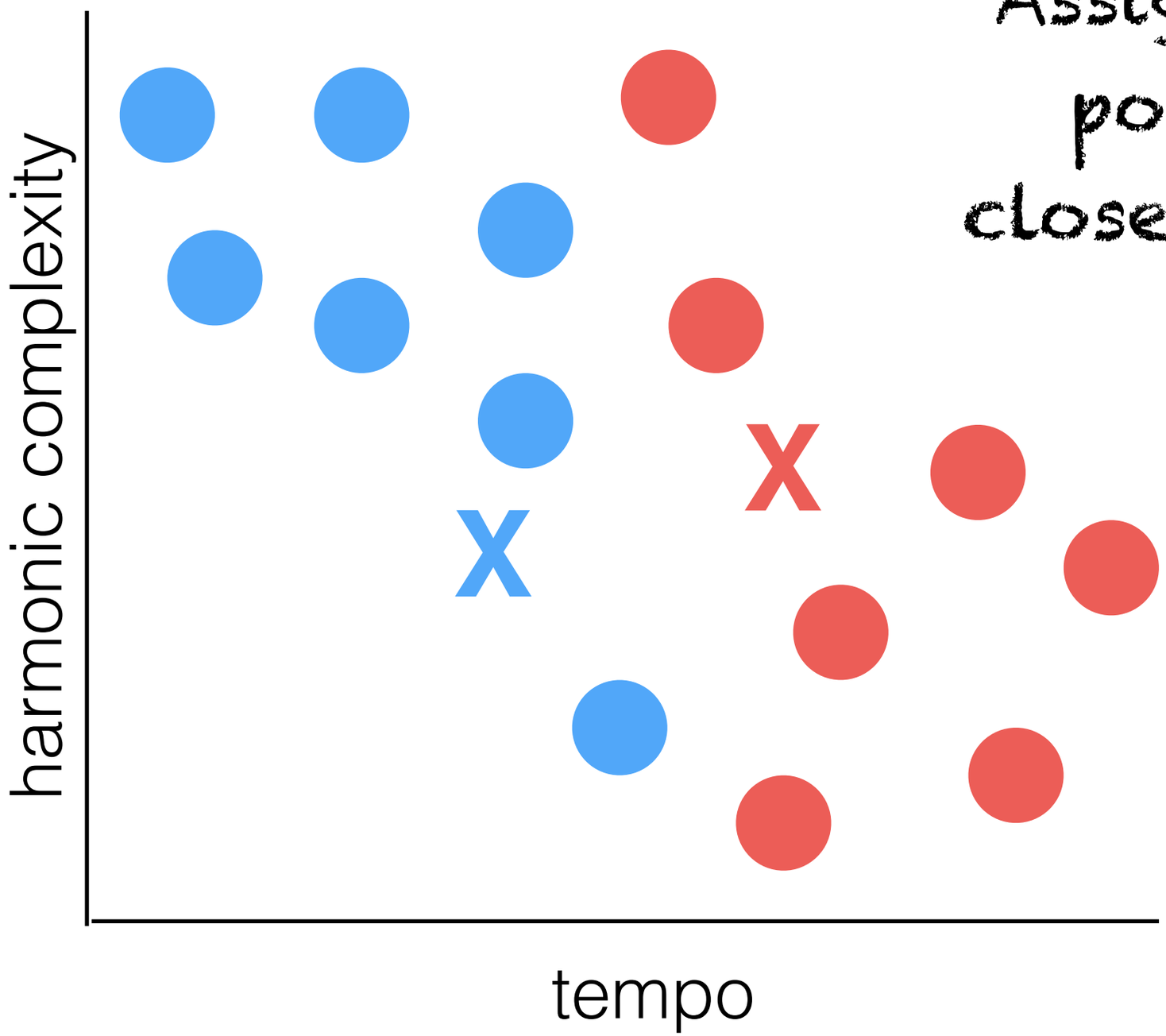

Assign each point to closest mean
harmonic complexity
tempo

K Means

re-compute means to be center of clusters



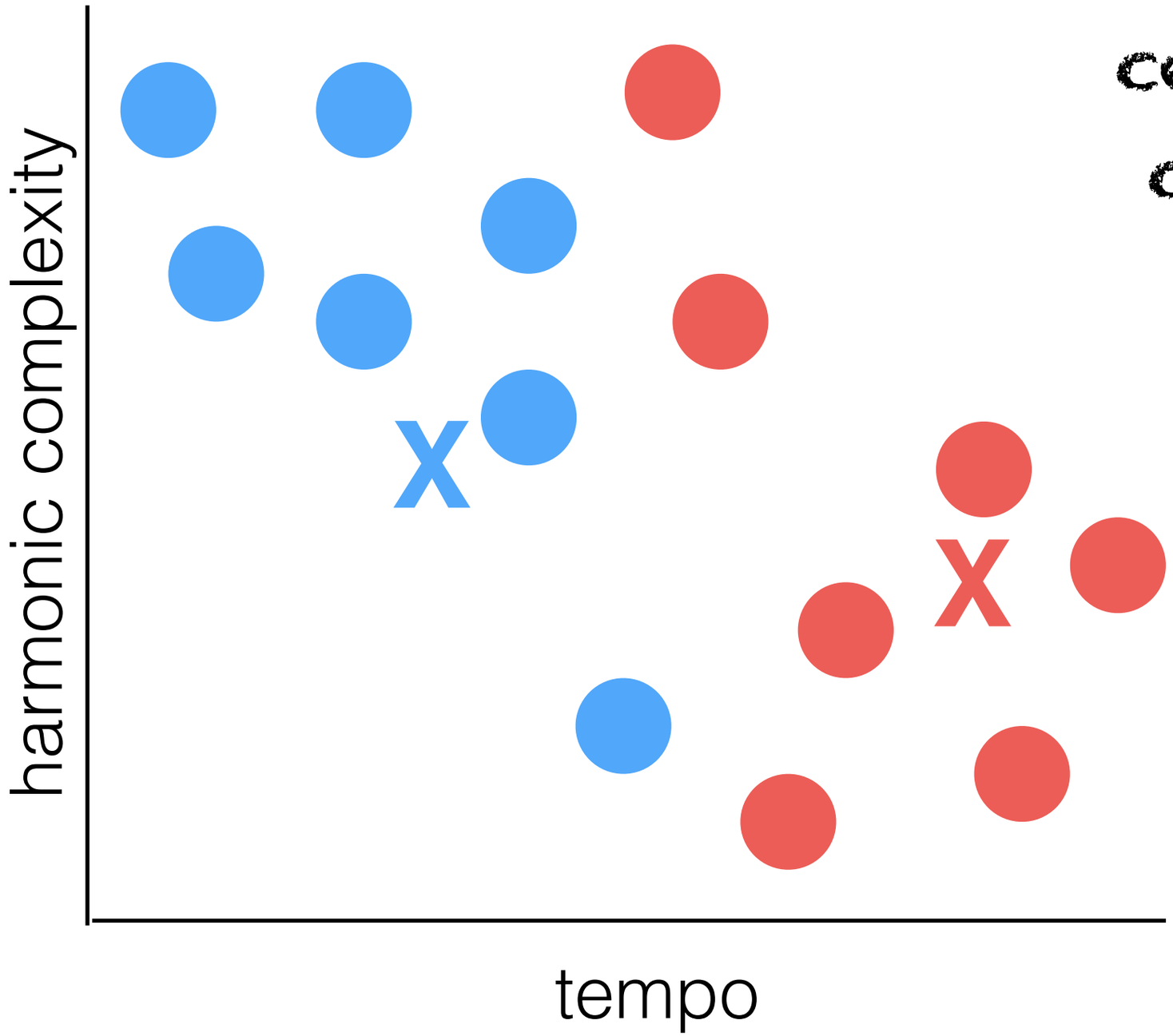
K Means



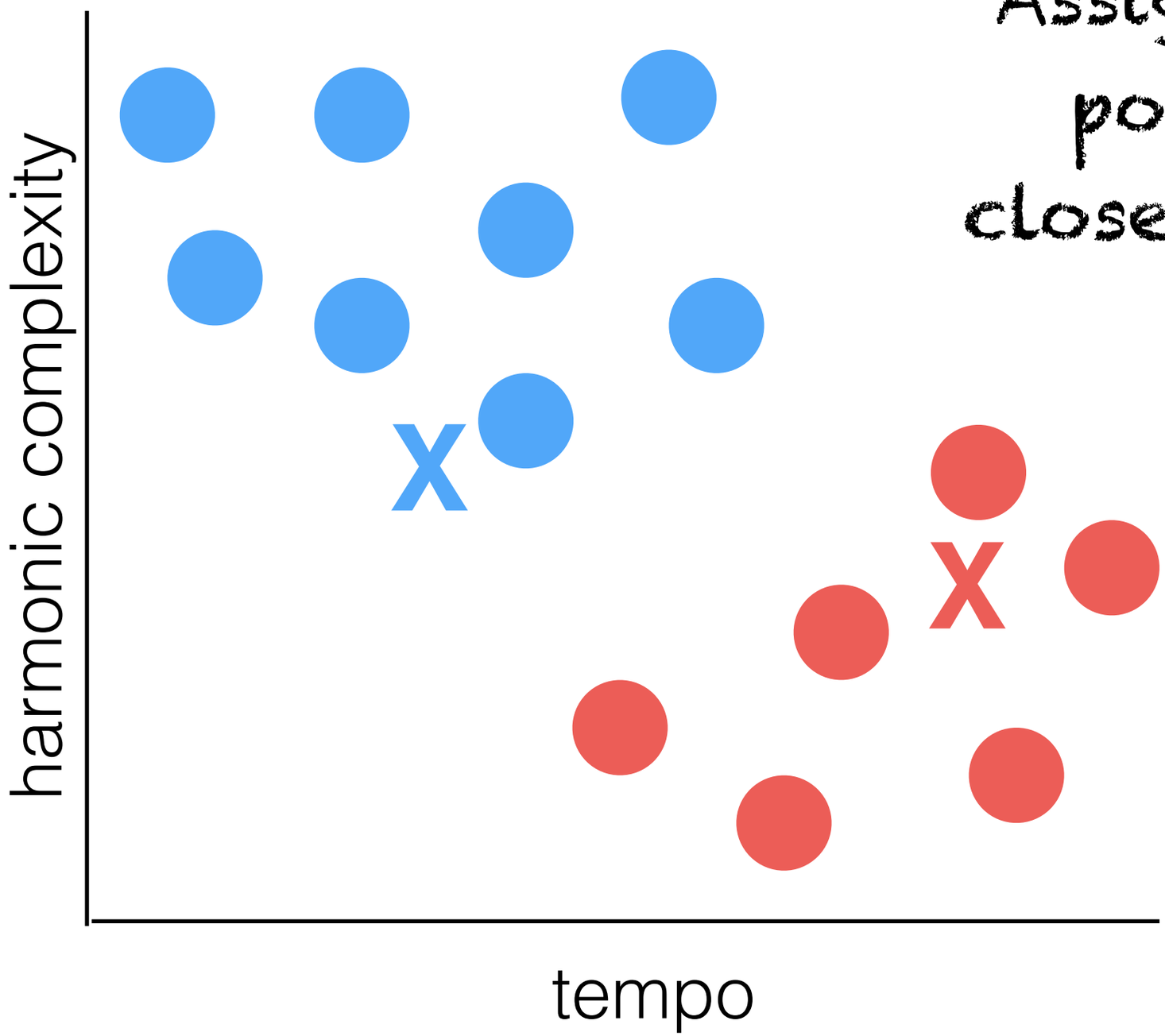
Assign each point to closest mean

K Means

re-compute means to be center of clusters



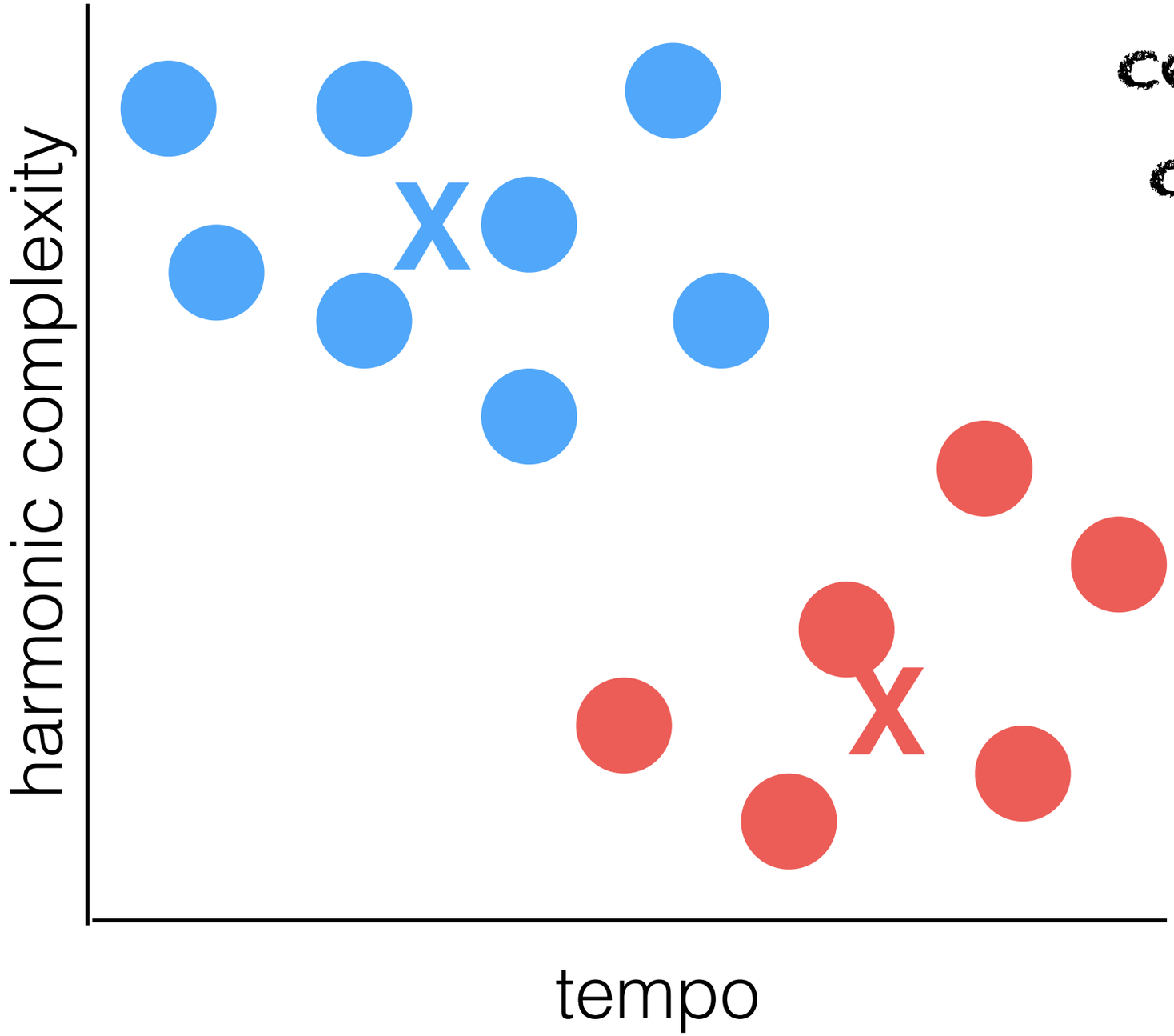
K Means



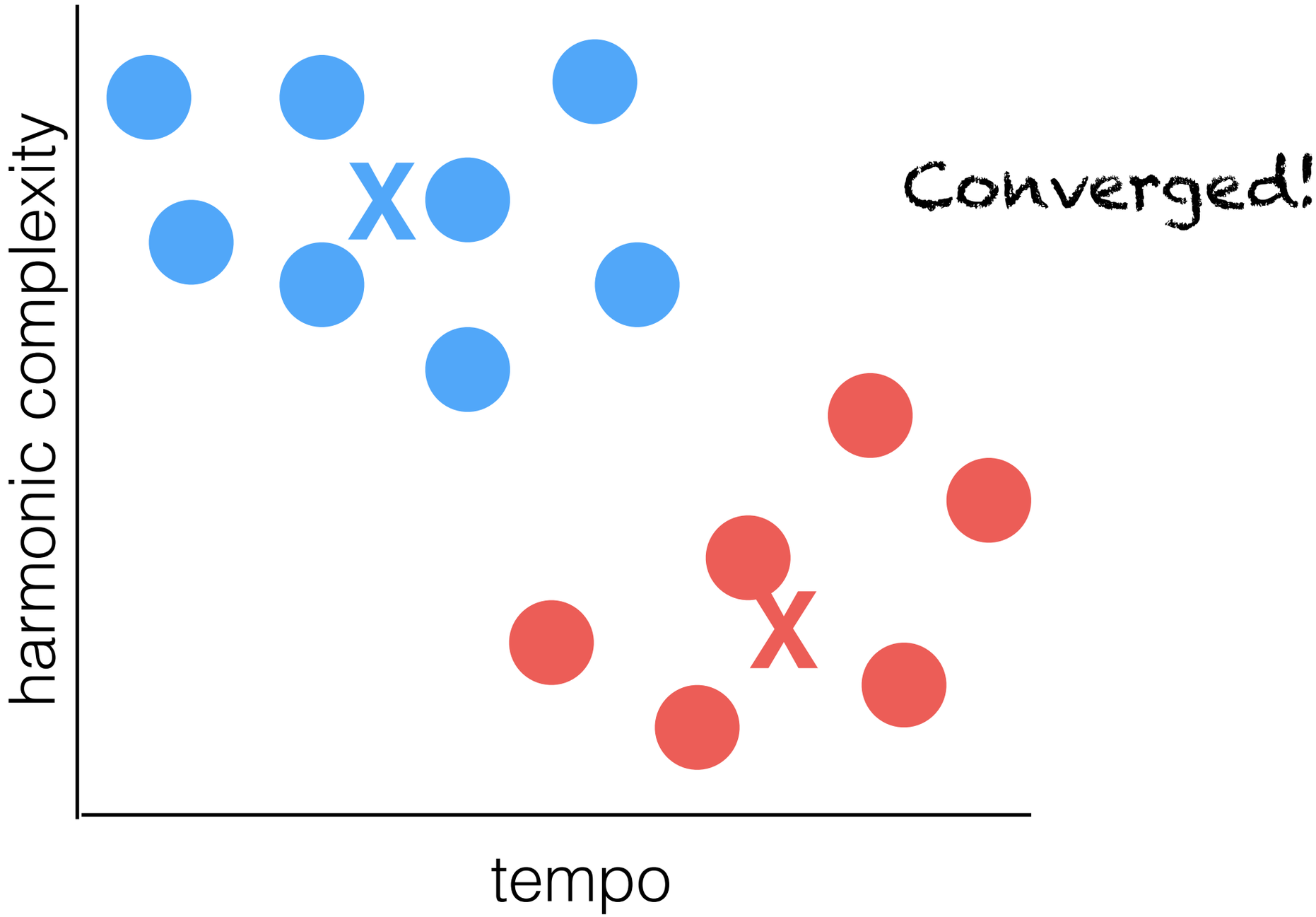
Assign each point to closest mean

K Means

re-compute means to be center of clusters

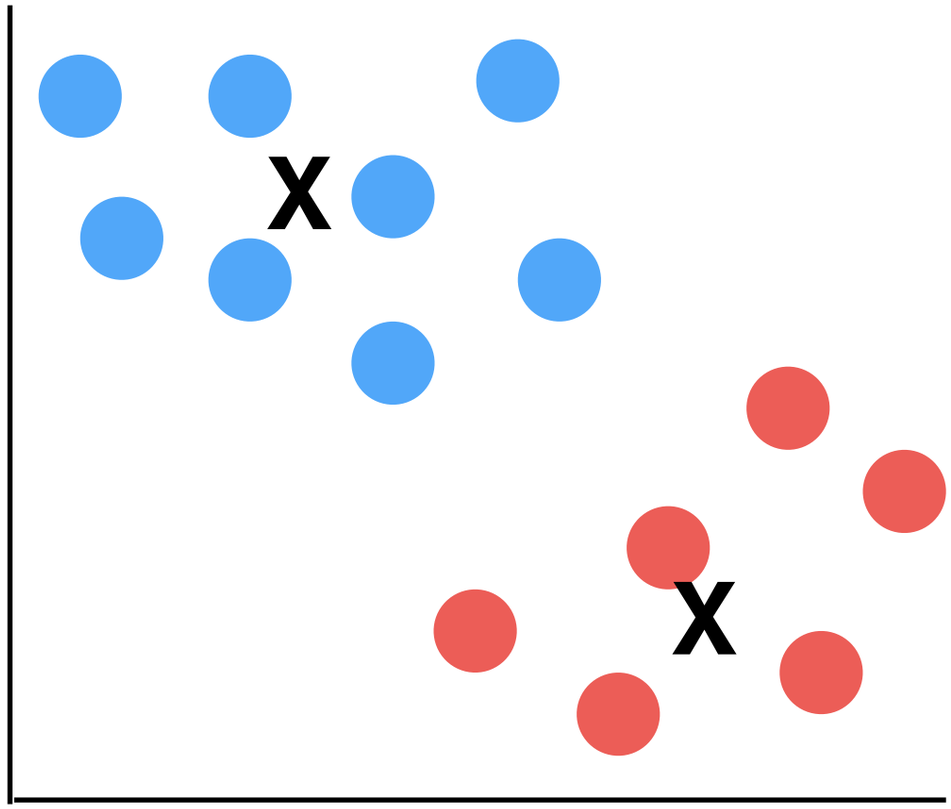


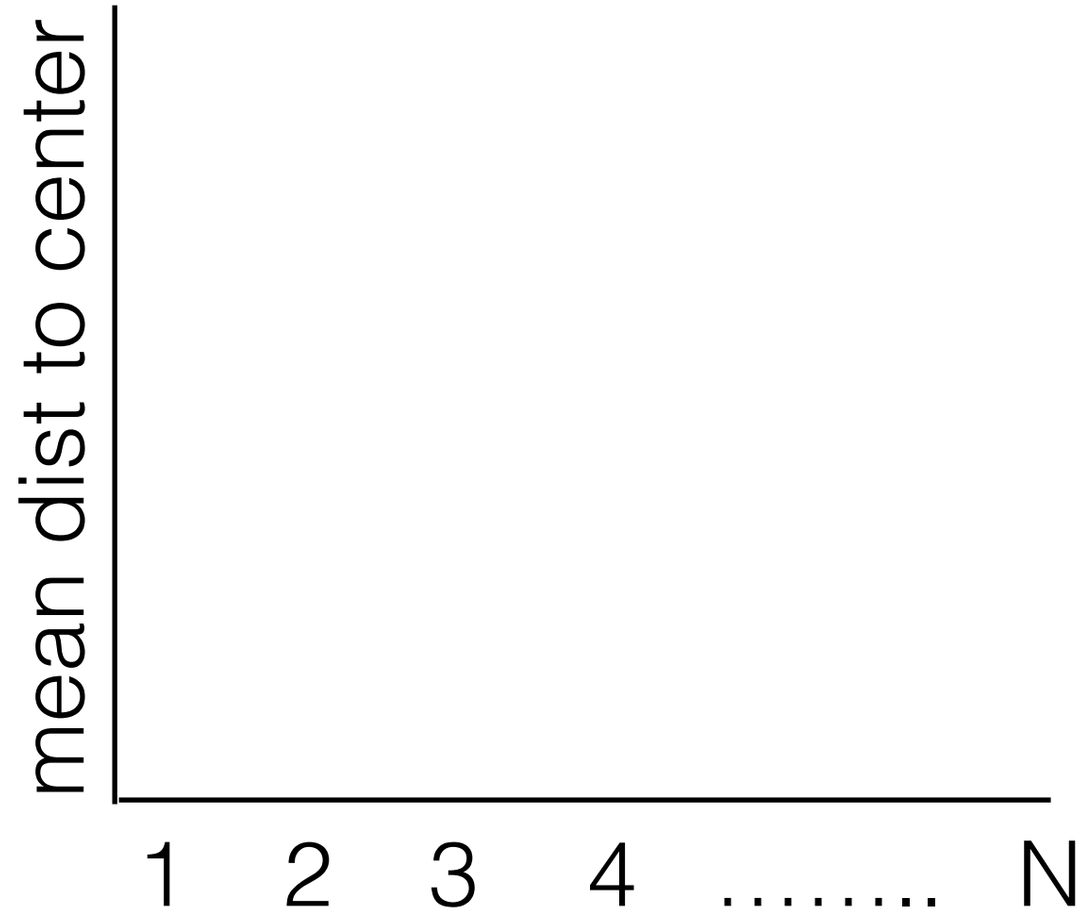
K Means



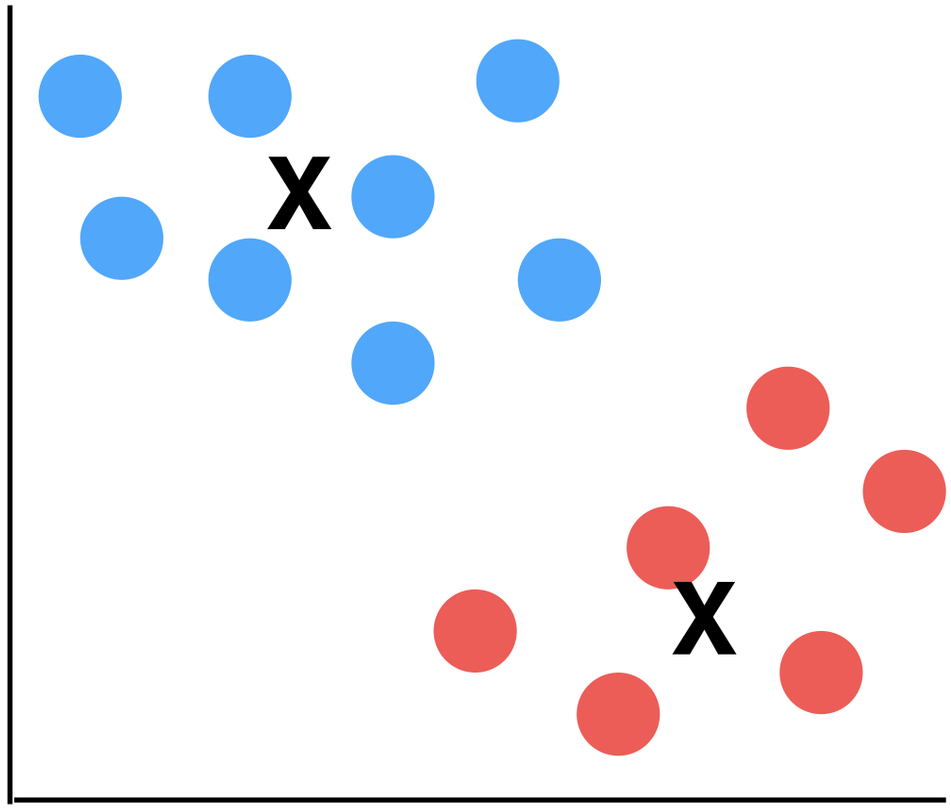
Clicker Question!

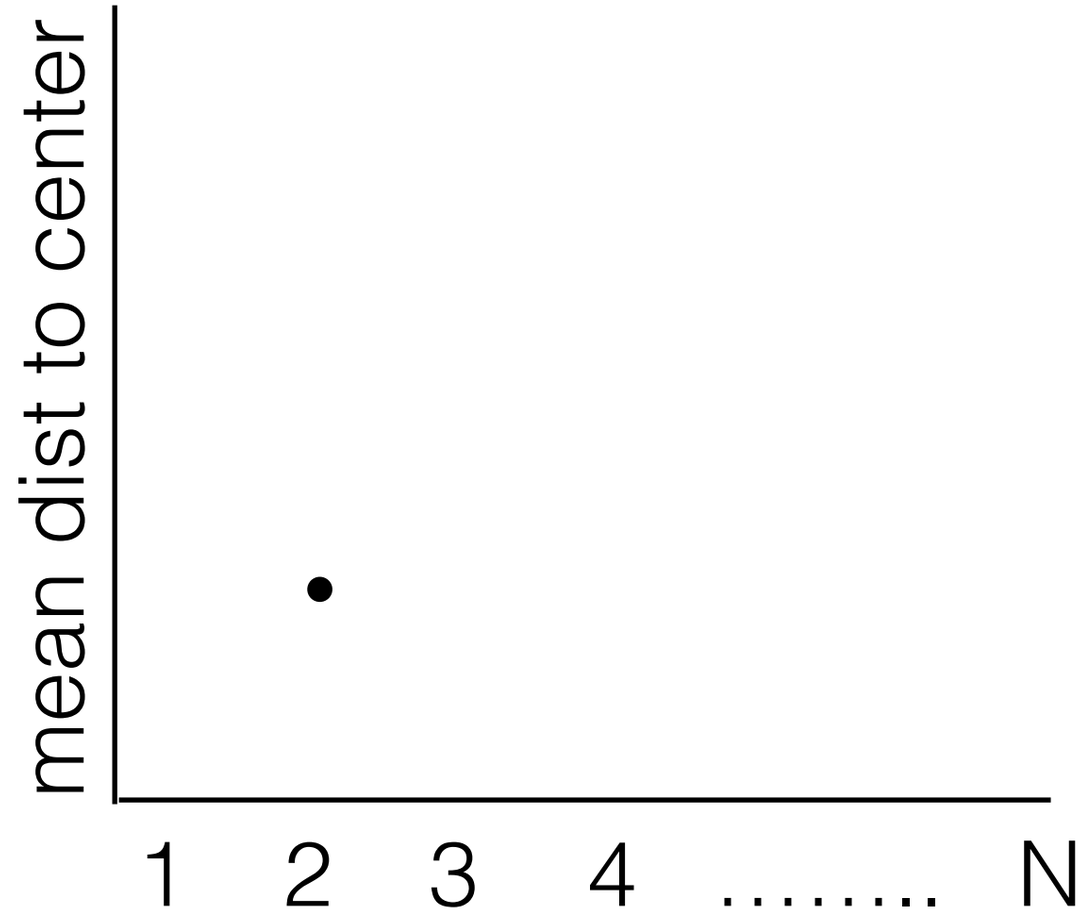
How many clusters?



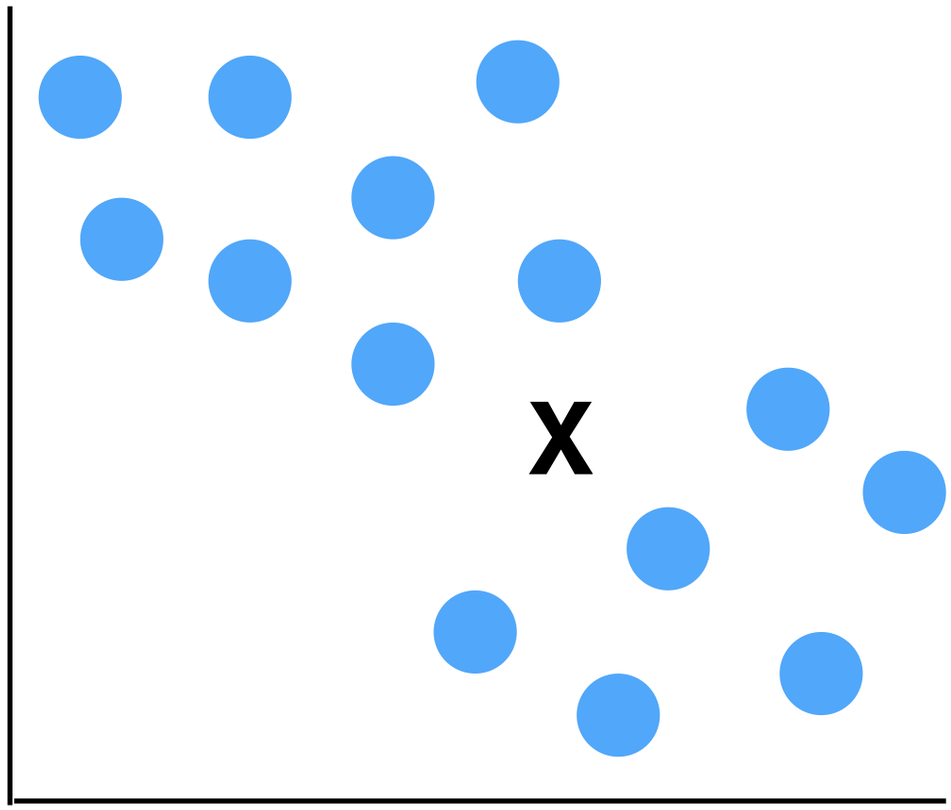


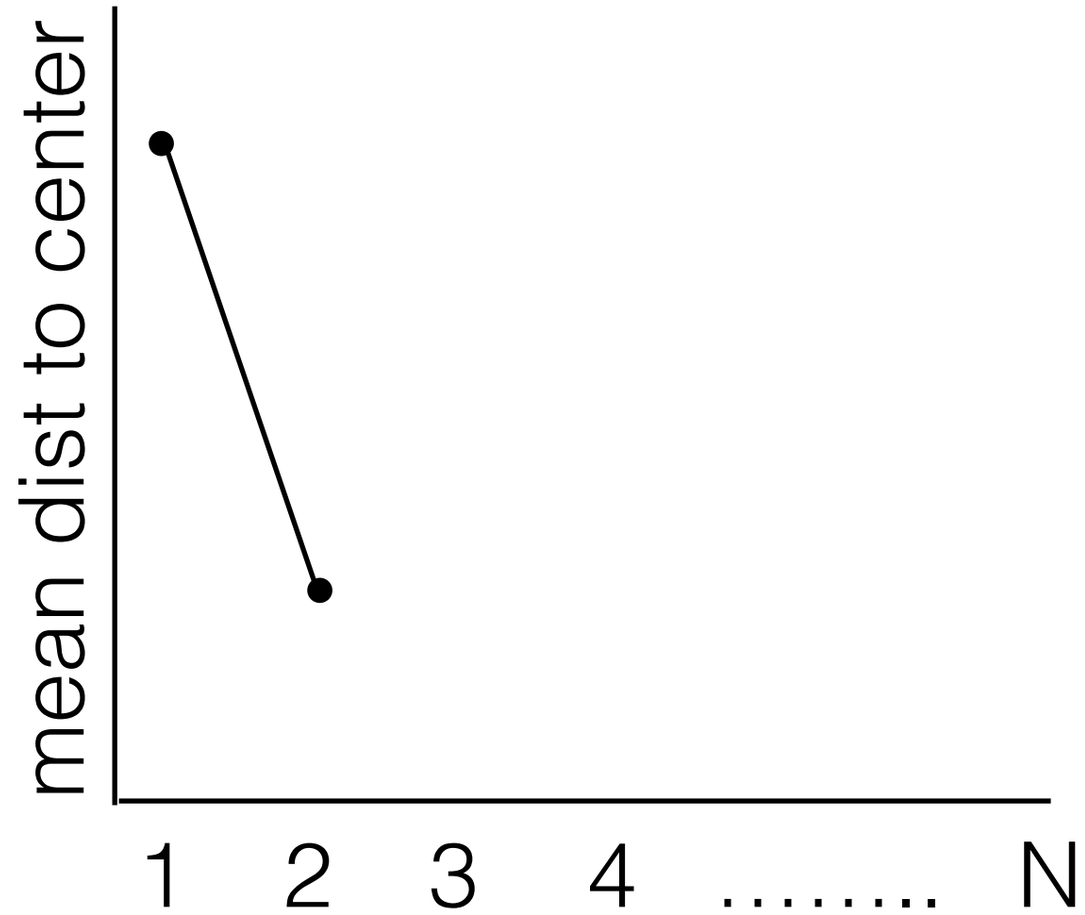
How many clusters?



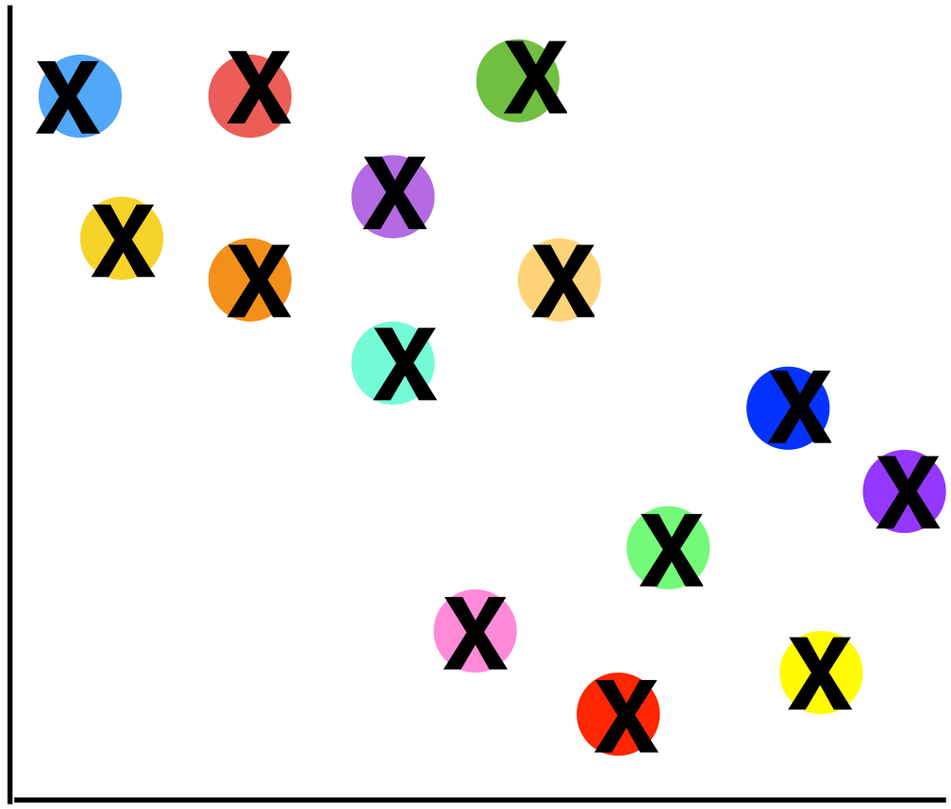


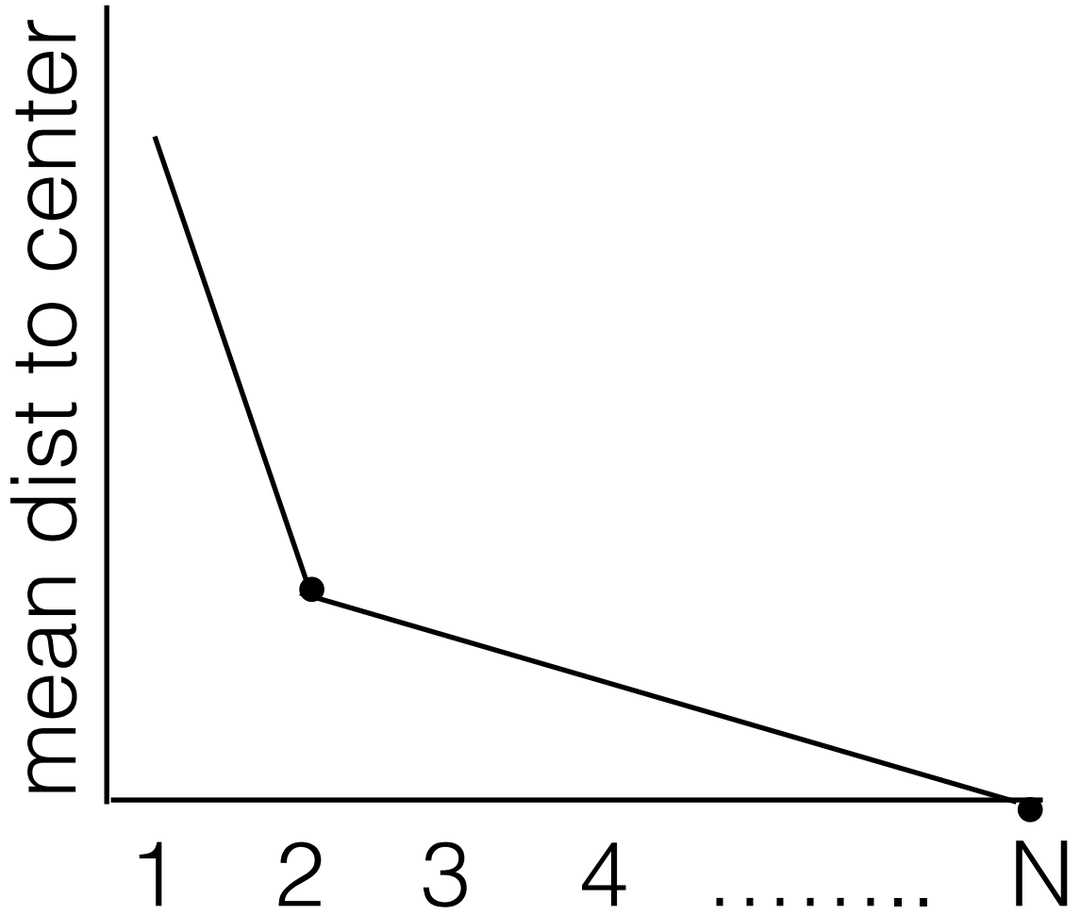
How many clusters?



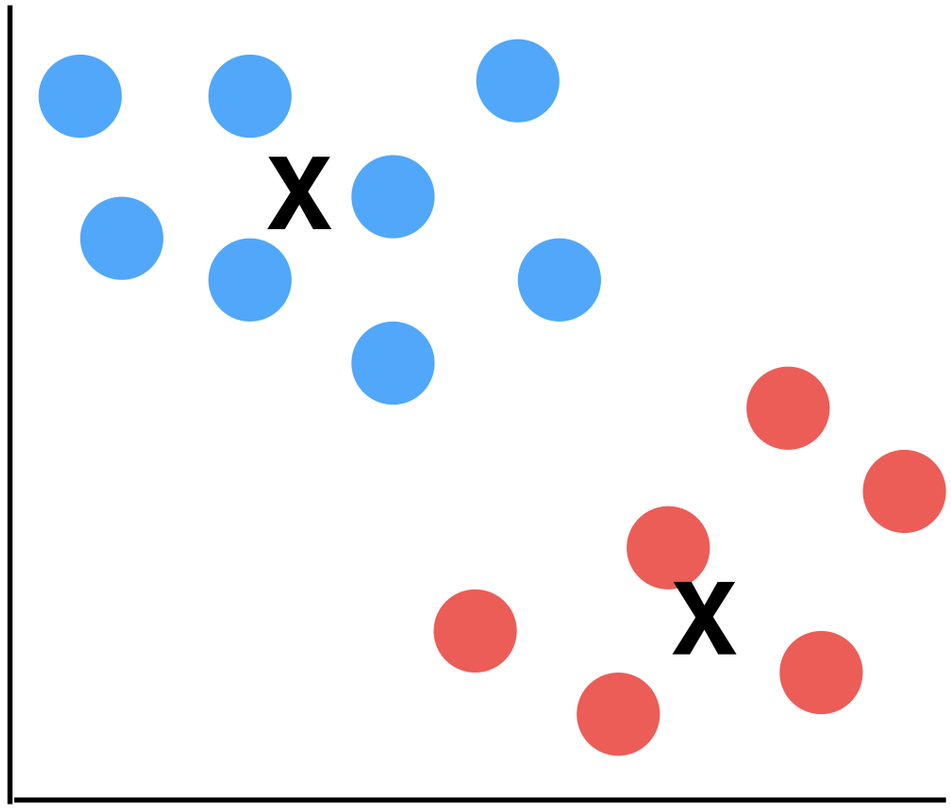


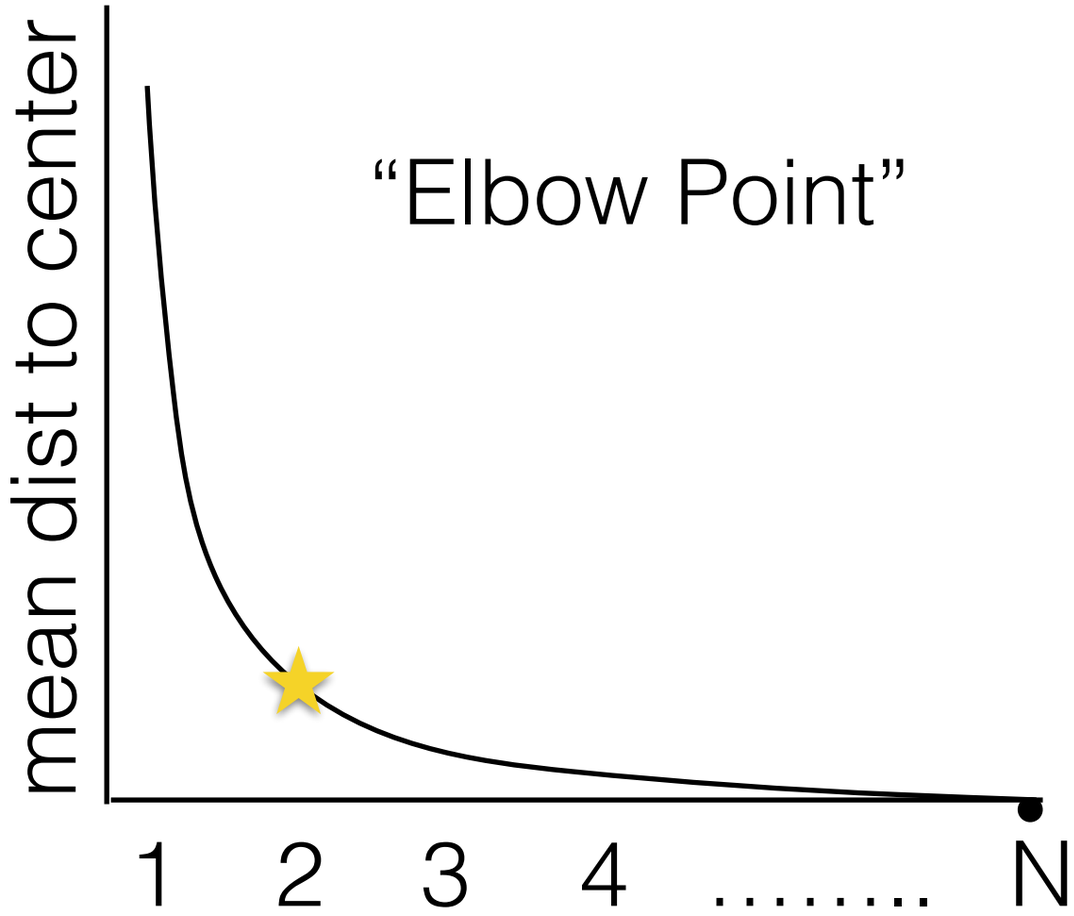
How many clusters?





How many clusters?



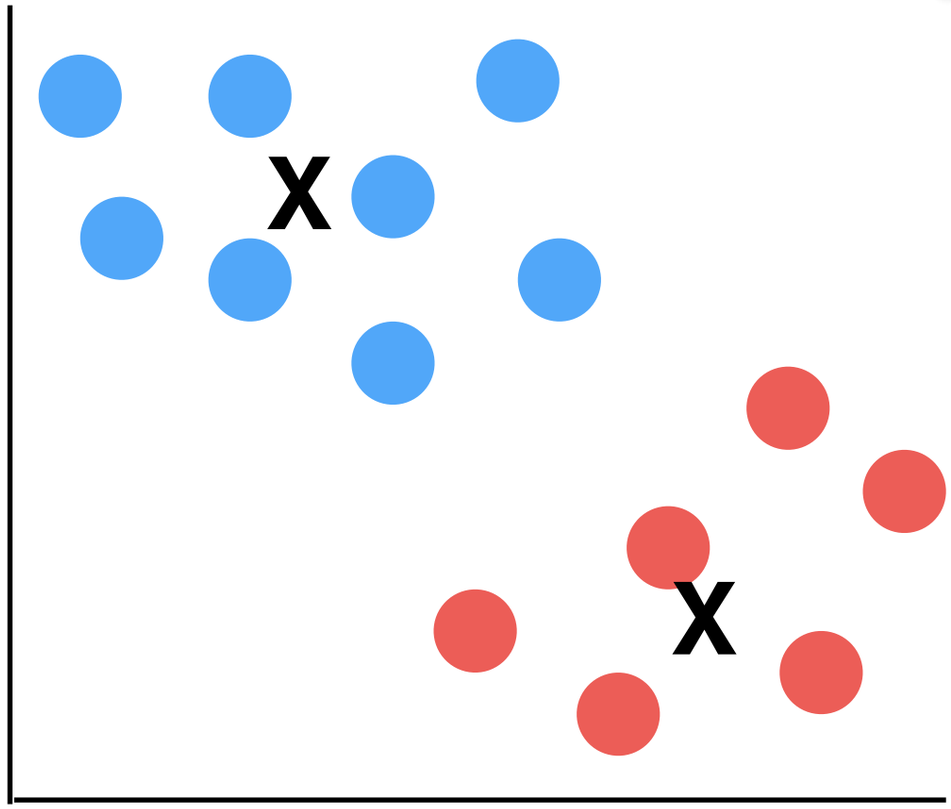


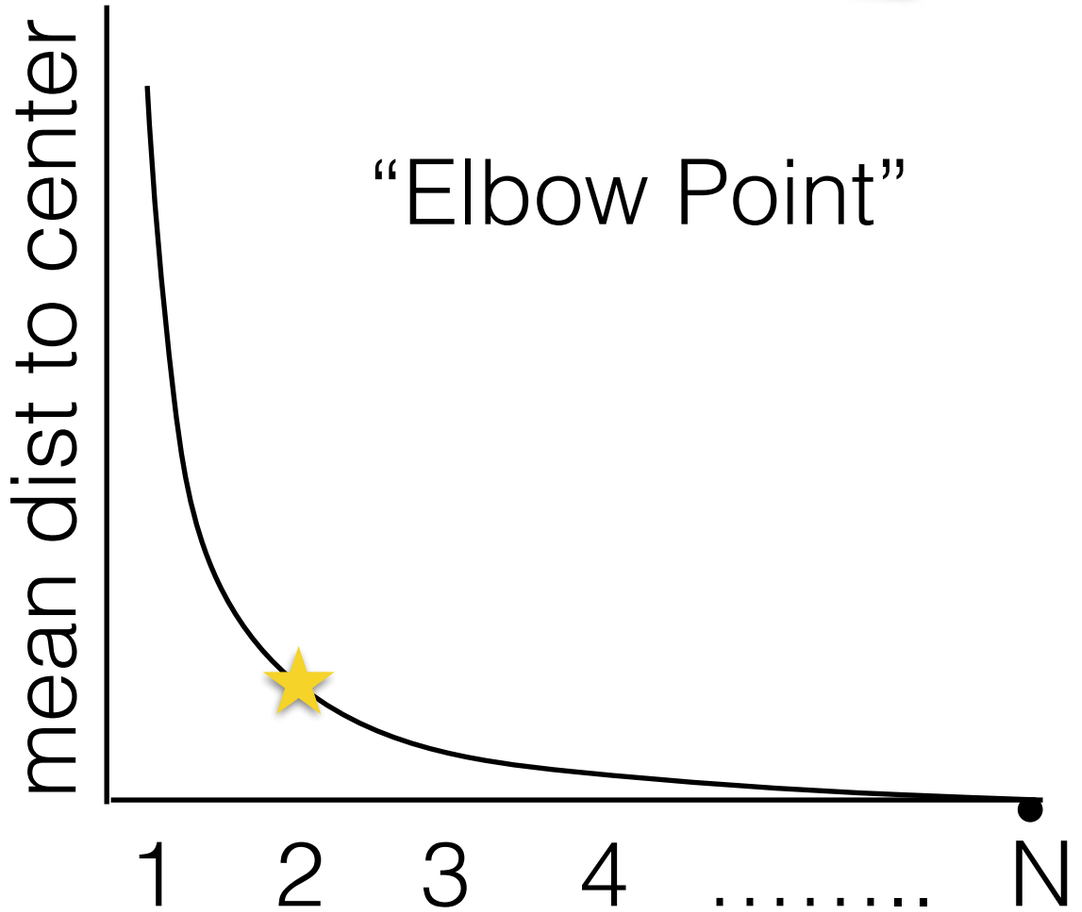
"Elbow Point"

How ma

Other techniques:

- Silhouette
- Intuition/Divine Intervention
- LGTM





"Elbow Point"

Expectation Maximization (EM)

Expectation Maximization (EM)

```
define parameters: K, max_iter, min_diff

iter = 0
change = inf
means = [random() for _ in range(K)]
while iter < max_iter and change > min_diff:
    update_assignments()
    compute_new_means()
    change = max_i(dist(new_mean_i, old_mean_i))
    iter += 1
```

Expectation Maximization (EM)

```
randomly initialize params
while not converged:
    data = estimate_likelihood(params)
    params = maximize_likelihood(data)
```

Expectation Maximization (EM)

E Step: estimate the likelihood of data under current parameter setting

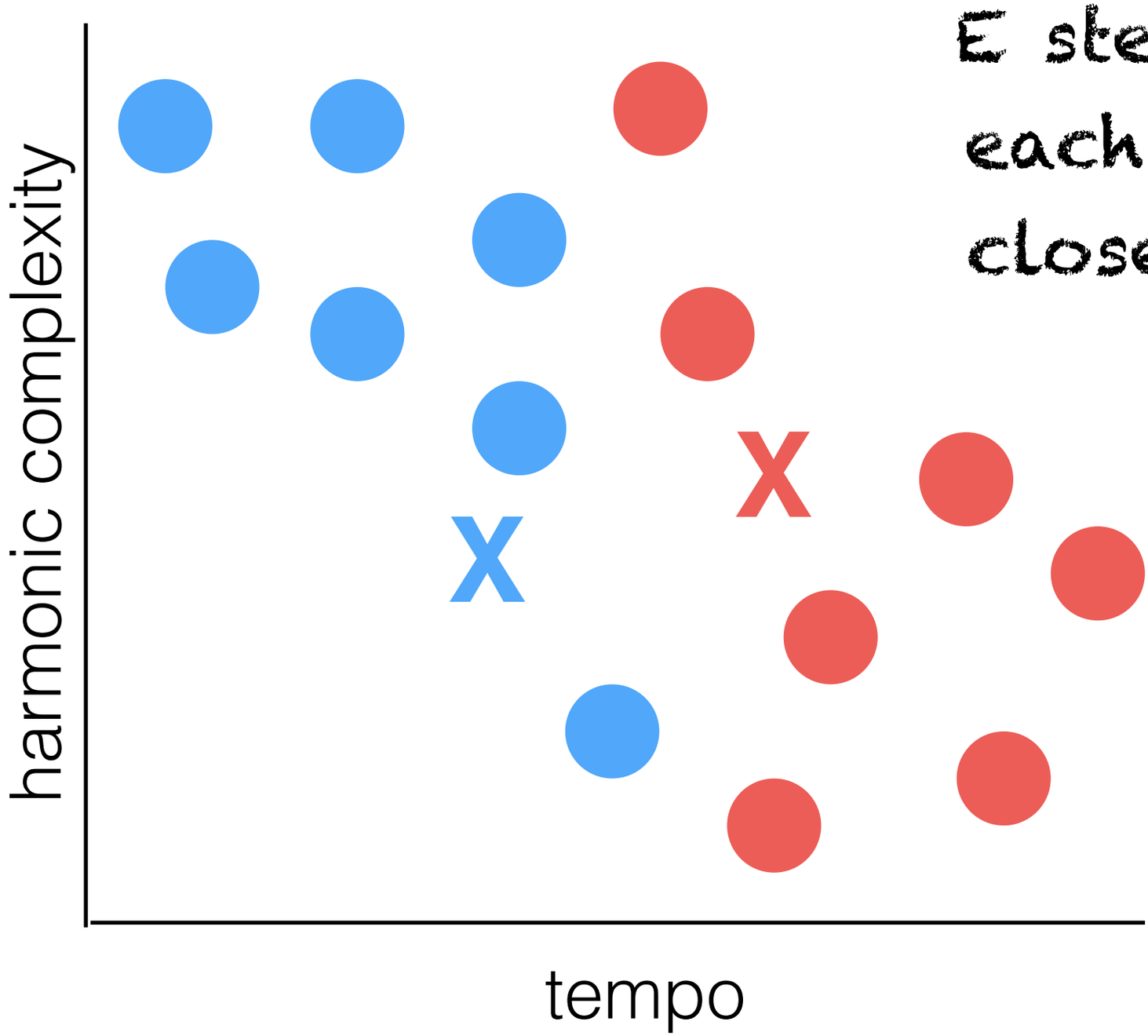
```
randomly initialize params
while not converged:
    data = estimate_likelihood(params)
    params = maximize_likelihood(data)
```

Expectation Maximization (EM)

M Step: adjust the the parameters so as to maximize the expectation of the data

```
randomly initialize params
while not converged:
    data = estimate_likelihood(params)
    params = maximize_likelihood(data)
```

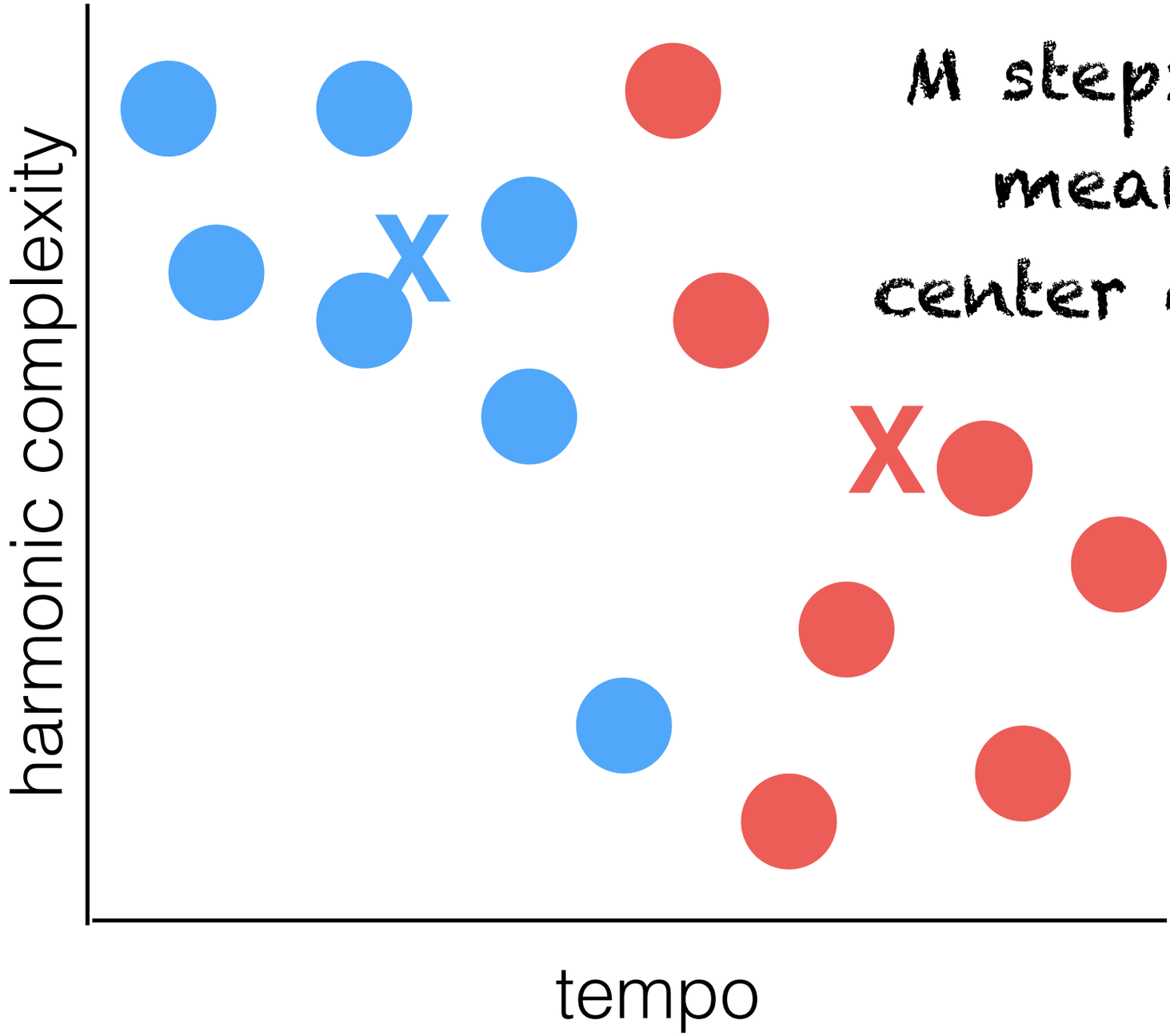
Expectation Maximization (EM)



E step: Assign each point to closest mean

http://www.dirkhovy.com/portfolio/papers/download/an_evening_with_EM.pdf

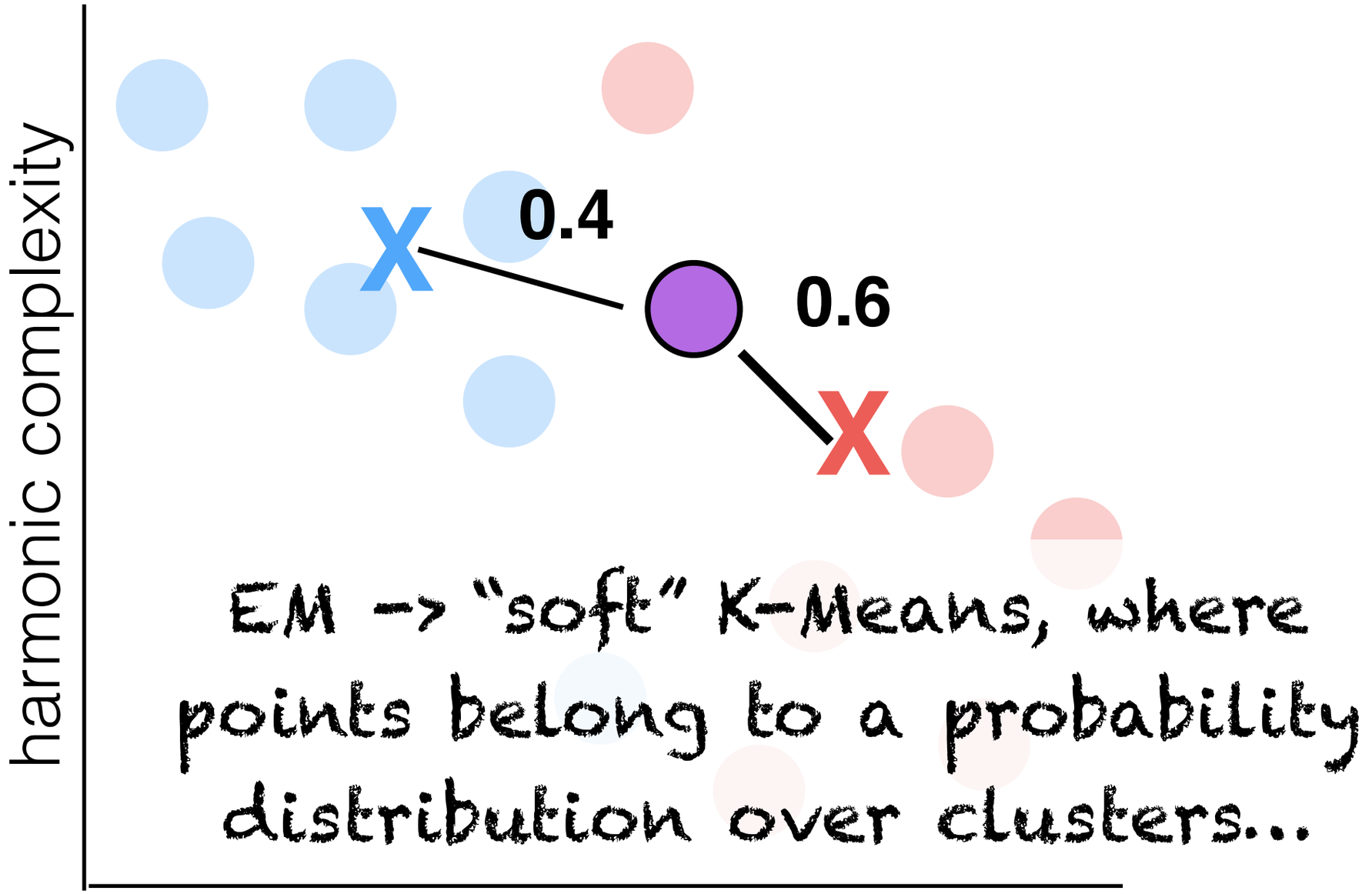
Expectation Maximization (EM)



M step: Compute means to be center of clusters

http://www.dirkhovy.com/portfolio/papers/download/an_evening_with_EM.pdf

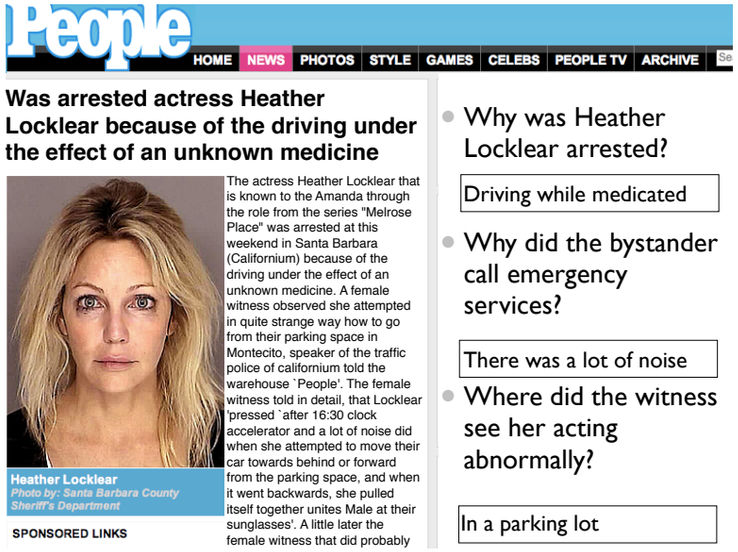
Expectation Maximization (EM)



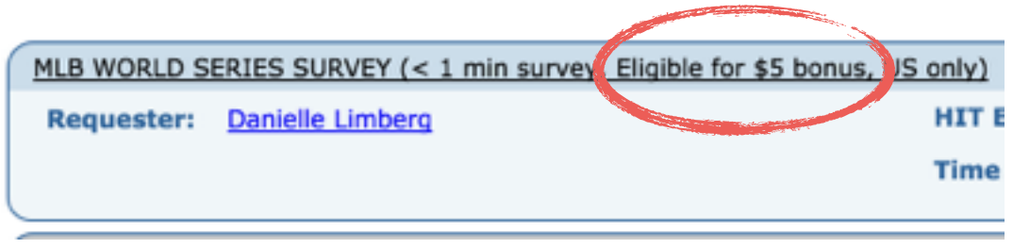
http://www.dirkhovy.com/portfolio/papers/download/an_evening_with_EM.pdf

#tbt slide from crowdsourcing lecture

Quality Control



Second-Pass HIT



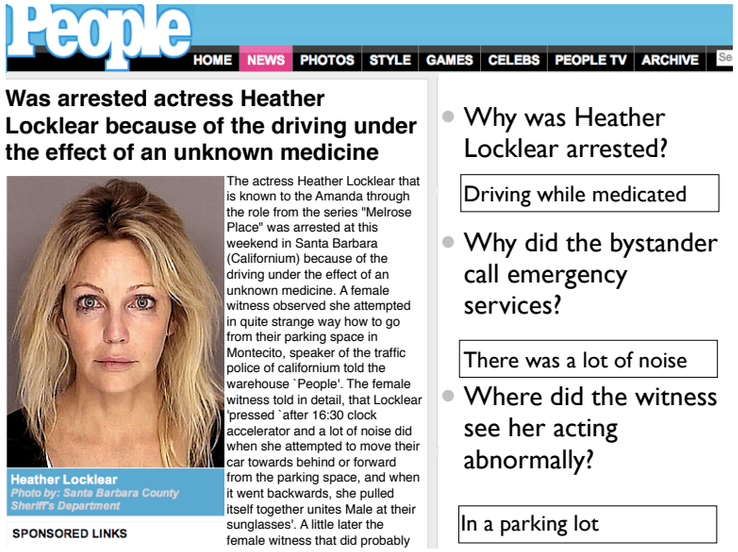
Incentive Pay

$$L(\boldsymbol{\theta}; \mathbf{X}) = p(\mathbf{X}|\boldsymbol{\theta}) = \sum_{\mathbf{Z}} p(\mathbf{X}, \mathbf{Z}|\boldsymbol{\theta})$$

Statistical Models

#tbt

Quality Control


People

HOME NEWS PHOTOS STYLE GAMES CELEBS PEOPLE TV ARCHIVE

Was arrested actress Heather Locklear because of the driving under the effect of an unknown medicine

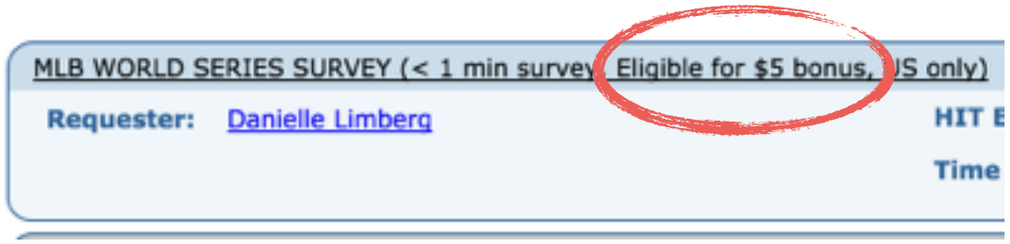
The actress Heather Locklear that is known to the Amanda through the role from the series "Melrose Place" was arrested at this weekend in Santa Barbara (Californium) because of the driving under the effect of an unknown medicine. A female witness observed she attempted in quite strange way how to go from their parking space in Montecito, speaker of the traffic police of californium told the warehouse `People'. The female witness told in detail, that Locklear 'pressed `after 16:30 clock accelerator and a lot of noise did when she attempted to move their car towards behind or forward from the parking space, and when it went backwards, she pulled itself together unites Male at their sunglasses'. A little later the female witness that did probably

Heather Locklear
Photo by: Santa Barbara County Sheriff's Department

SPONSORED LINKS

- Why was Heather Locklear arrested?
  - Driving while medicated
- Why did the bystander call emergency services?
  - There was a lot of noise
- Where did the witness see her acting abnormally?
  - In a parking lot

Second-Pass HIT


MLB WORLD SERIES SURVEY (< 1 min survey, Eligible for $5 bonus, US only)

Requester: Danielle Limberg

HIT E

Time

Incentive Pay

$$L(\boldsymbol{\theta}; \mathbf{X}) = p(\mathbf{X}|\boldsymbol{\theta}) = \sum_{\mathbf{Z}} p(\mathbf{X}, \mathbf{Z}|\boldsymbol{\theta})$$

Statistical Models

Soooo obnoxious!

Goal: Find “true” labels despite noisy annotations from workers…

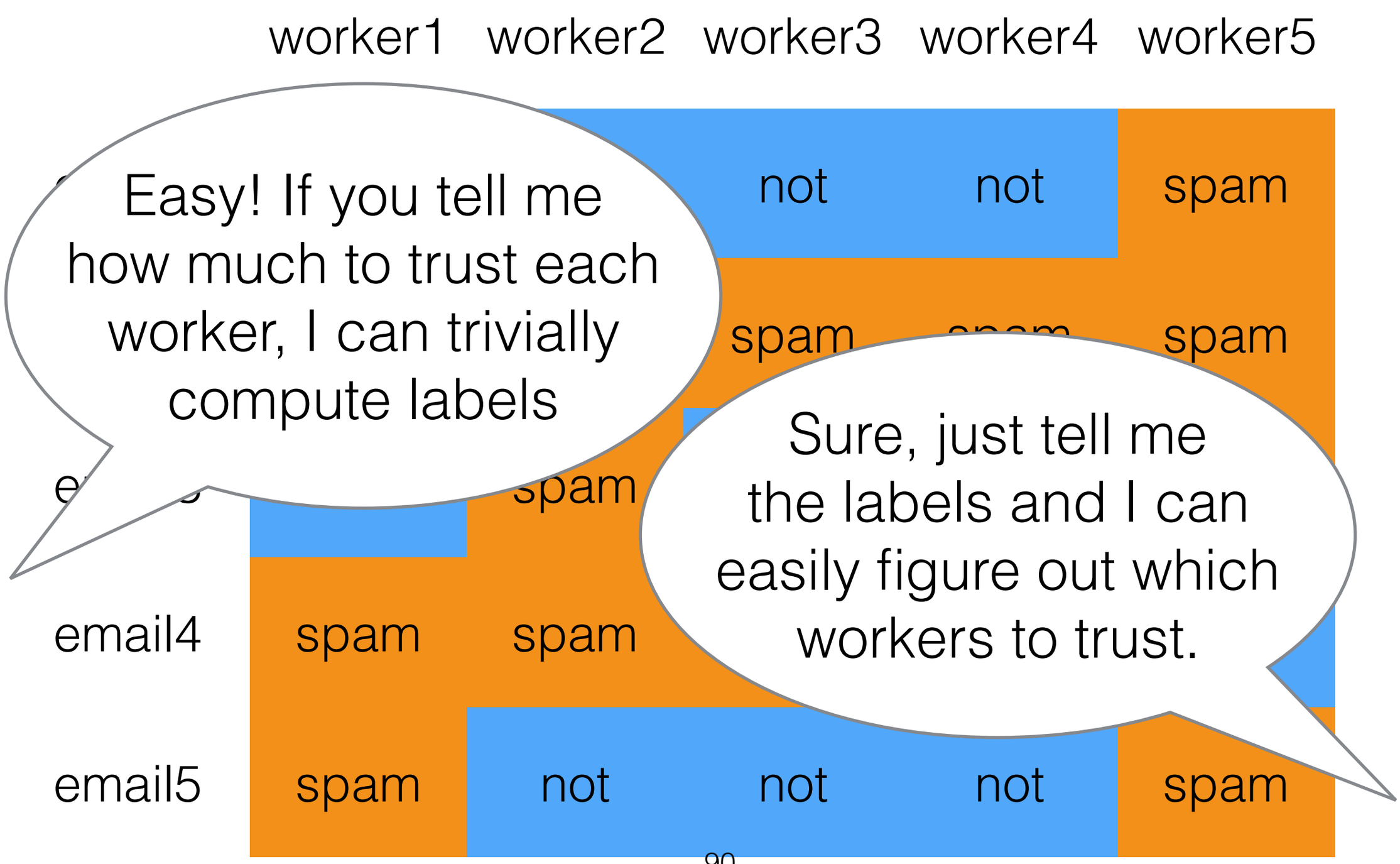
	worker1	worker2	worker3	worker4	worker5
email1	spam	not	not	not	spam
email2	spam	spam	spam	spam	spam
email3	not	spam	not	not	spam
email4	spam	spam	spam	spam	not
email5	spam	not	not	not	spam

Goal: Find “true” labels despite noisy annotations from workers…

	worker1	worker2	worker3	worker4	worker5
email1	not	not	not	not	spam
email2	spam	spam	spam	spam	spam
email3	not	spam	not	not	spam
email4	spam	spam	spam	spam	not
email5	spam	not	not	not	spam

Easy! If you tell me how much to trust each worker, I can trivially compute labels

Goal: Find “true” labels despite noisy annotations from workers…



Goal: Find "true" noisy annotation

EM EVERYTHING!!!!

	worker1	worker2	worker3	worker4	worker5
email1	spam	spam	spam	spam	spam
email2	not	not	not	not	not
email3	not	spam	not	not	not
email4	spam	spam	not	not	not
email5	spam	not	not	not	spam

Easy! If you tell me how much to trust each worker, I can trivially compute labels

Sure, just tell me the labels and I can easily figure out which workers to trust.

	w1	w2	w3	w4	w5
email1	spam	not	not	not	spam
email2	spam	spam	spam	spam	spam
email3	not	spam	not	not	spam
email4	spam	spam	spam	spam	not
email5	spam	not	not	not	spam

	spam	not
email1	?	?
email2	?	?
email3	?	?
email4	?	?
email5	?	?

w1	spam	not
spam	?	?
not	?	?

w2	spam	not
spam	?	?
not	?	?

w3	spam	not
spam	?	?
not	?	?

w4	spam	not
spam	?	?
not	?	?

w5	spam	not
spam	?	?
not	?	?

	w1	w2	w3	w4	w5
email1	spam	not	not	not	spam
email2	spam	spam	spam	spam	spam
email3	not	spam	not	not	spam
email4	spam	spam	spam	spam	not
email5	spam	not	not	not	spam

P(email1 is spam)

	spam	not
email1	?	?
email2	?	?
email3	?	?
email4	?	?
email5	?	?

w1	spam	not
spam	?	?
not	?	?

w2	spam	not
spam	?	?
not	?	?

w3	spam	not
spam	?	?
not	?	?

w4	spam	not
spam	?	?
not	?	?

w5	spam	not
spam	?	?
not	?	?

	w1	w2	w3	w4	w5
email1	spam	not	not	not	spam
email2	spam	spam	spam	spam	spam
email4	spam	spam	spam	spam	not
email5	spam	not	not	not	spam

P(w1 says spam | not spam)

	spam	not
email1	?	?
email2	?	?
email3	?	?
email4	?	?
email5	?	?

w1	spam	not
spam	?	?
not	?	?

w2	spam	not
spam	?	?
not	?	?

w3	spam	not
spam	?	?
not	?	?

w4	spam	not
spam	?	?
not	?	?

w5	spam	not
spam	?	?
not	?	?

	w1	w2	w3	w4	w5
email1	spam	not	not	not	spam
email2	spam	spam	spam	spam	spam
email3	not	spam	not	not	spam
email4	spam	spam	spam	spam	not
email5	spam	not	not	not	spam

Assume all workers are perfect

	spam	not
email1	?	?
email2	?	?
email3	?	?
email4	?	?
email5	?	?

w1	spam	not
spam	1	0
not	0	1

w2	spam	not
spam	1	0
not	0	1

w3	spam	not
spam	1	0
not	0	1

w4	spam	not
spam	1	0
not	0	1

w5	spam	not
spam	1	0
not	0	1

	w1	w2	w3	w4	w5
email1	spam	not	not	not	spam
email2	spam	spam	spam	spam	spam
email3	not	spam	not	not	spam
email4	spam	spam	spam	spam	not
email5	spam	not	not	not	spam

Compute labels using majority vote

	spam	not
email1	?	?
email2	?	?
email3	?	?
email4	?	?
email5	?	?

w1	spam	not
spam	1	0
not	0	1

w2	spam	not
spam	1	0
not	0	1

w3	spam	not
spam	1	0
not	0	1

w4	spam	not
spam	1	0
not	0	1

w5	spam	not
spam	1	0
not	0	1

Clicker Question!

	w1	w2	w3	w4	w5
email1	spam	not	not	not	spam
email2	spam	spam	spam	spam	spam
email3	not	spam	not	not	spam
email4	spam	spam	spam	spam	not
email5	spam	not	not	not	spam

Compute labels using majority vote

	spam	not
email1	0.4	0.6
email2	1	0
email3	0.4	0.6
email4	0.8	0.2
email5	0.4	0.6

w1	spam	not
spam	1	0
not	0	1

w2	spam	not
spam	1	0
not	0	1

w3	spam	not
spam	1	0
not	0	1

w4	spam	not
spam	1	0
not	0	1

w5	spam	not
spam	1	0
not	0	1

	w1	w2	w3	w4	w5
email1	spam	not	not	not	spam
email2	spam	spam	spam	spam	spam
email3	not	spam	not	not	spam
email4	spam	spam	spam	spam	not
email5	spam	not	not	not	spam

Assume these labels, and recompute confusion matrices

	spam	not
email1	0.4	0.6
email2	1	0
email3	0.4	0.6
email4	0.8	0.2
email5	0.4	0.6

w1	spam	not
spam	1	0
not	0	1

w2	spam	not
spam	1	0
not	0	1

w3	spam	not
spam	1	0
not	0	1

w4	spam	not
spam	1	0
not	0	1

w5	spam	not
spam	1	0
not	0	1

	w1	w2	w3	w4	w5
email1	spam	not	not	not	spam
email2	spam	spam	spam	spam	spam
email3	not	spam	not	not	spam
email4	spam	spam	spam	spam	not
email5	spam	not	not	not	spam

Assume these labels, and recompute confusion matrices

	spam	not
email1	0.4	0.6
email2	1	0
email3	0.4	0.6
email4	0.8	0.2
email5		0.6

w1	spam	not
spam		
not		

w2	spam	not
spam		
not		

w3	spam	not
spam		
not		

w4	spam	not
spam		
not		

w5	spam	not
spam		
not		

Clicker Question!

	w1	w2	w3	w4	w5
email1	spam	not	not	not	spam
email2	spam	spam	spam	spam	spam
email3	not	spam	not	not	spam
email4	spam	spam	spam	spam	not
email5	spam	not	not	not	spam

Assume these labels, and recompute confusion matrices

	spam	not
email1	0.4	0.6
email2	1	0
email3	0.4	0.6
email4	0.8	0.2
email5		0.6

w1	spam	not
spam	1	0
not	0.67	0.33

w2	spam	not
spam	1	0
not	0.33	0.67

w3	spam	not
spam	1	0
not	0	1

w4	spam	not
spam	1	0
not	0	1

w5	spam	not
spam	0.5	0.5
not	1	0

	w1	w2	w3	w4	w5
email1	spam	not	not	not	spam
email2	spam	spam	spam	spam	spam
email3	not	spam	not	not	spam
email4	spam	spam	spam	spam	not
email5	spam	not	not	not	spam

Recompute labels using (weighted) majority vote

	spam	not
email1	1.5	
email2		
email3		
email4		
email5		

w1	spam	not
spam	**1**	0
not	0.67	0.33

w2	spam	not
spam	1	**0**
not	0.33	0.67

w3	spam	not
spam	1	**0**
not	0	1

w4	spam	not
spam	1	**0**
not	0	1

w5	spam	not
spam	**0.5**	0.5
not	1	0

	w1	w2	w3	w4	w5
email1	spam	not	not	not	spam
email2	spam	spam	spam	spam	spam
email3	not	spam	not	not	spam
email4	spam	spam	spam	spam	not
email5	spam	not	not	not	spam

Recompute labels using (weighted) majority vote

	spam	not
email1	1.5	4.34
email2		
email3		
email4		
email5		

w1	spam	not
spam	1	0
not	**0.67**	0.33

w2	spam	not
spam	1	0
not	0.33	**0.67**

w3	spam	not
spam	1	0
not	0	**1**

w4	spam	not
spam	1	0
not	0	**1**

w5	spam	not
spam	0.5	0.5
not	**1**	0

	w1	w2	w3	w4	w5
email1	spam	not	not	not	spam
email2	spam	spam	spam	spam	spam
email3	not	spam	not	not	spam
email4	spam	spam	spam	spam	not
email5	spam	not	not	not	spam

Renormalize

	spam	not
email1	0.26	0.74
email2	0.69	0.31
email3	0.29	0.71
email4	0.82	0.18
email5	0.26	0.74

w1	spam	not
spam	1	0
not	0.67	0.33

w2	spam	not
spam	1	0
not	0.33	0.67

w3	spam	not
spam	1	0
not	0	1

w4	spam	not
spam	1	0
not	0	1

w5	spam	not
spam	0.5	0.5
not	1	0

	w1	w2	w3	w4	w5
email1	spam	not	not	not	spam
email2	spam	spam	spam	spam	spam
email3	not	spam	not	not	spam
email4	spam	spam	spam	spam	not
email5	spam	not	not	not	spam

Iterate until convergence!

	spam	not
email1	0.26	**0.74**
email2	**0.69**	0.31
email3	0.29	**0.71**
email4	**0.82**	0.18
email5	0.26	**0.74**

w1	spam	not
spam	1	
not		

w2	spam	not
spam	1	0
not	0.33	0.67

w3	spam	not
spam	1	0
not	0	1

w4	spam	not
spam	1	0
not	0	1

w5	spam	not
spam	0.5	0.5
not	1	0

	w1	w2	w3	w4	w5
email1	spam	not	not	not	spam
email2	spam	spam	spam	spam	spam
email3	not	spam	not	not	spam
email4	spam	spam	spam	spam	not
email5	spam	not	not	not	spam

(This example converges after 1 iteration)

	spam	not
email1	0.26	0.74
email2	0.69	0.31
email3	0.29	0.71
email4	0.82	0.18
email5	0.26	0.74

w1	spam	not
spam	1	0
not	0.67	0.33

w2	spam	not
spam	1	0
not	0.33	0.67

w3	spam	not
spam	1	0
not	0	1

w4	spam	not
spam	1	0
not	0	1

w5	spam	not
spam	0.5	0.5
not	1	0

```
iter == max_iter or
change == min_diff
```