CPSC 490 – Problem Solving in Computer Science

Lecture 19: Line Sweep II

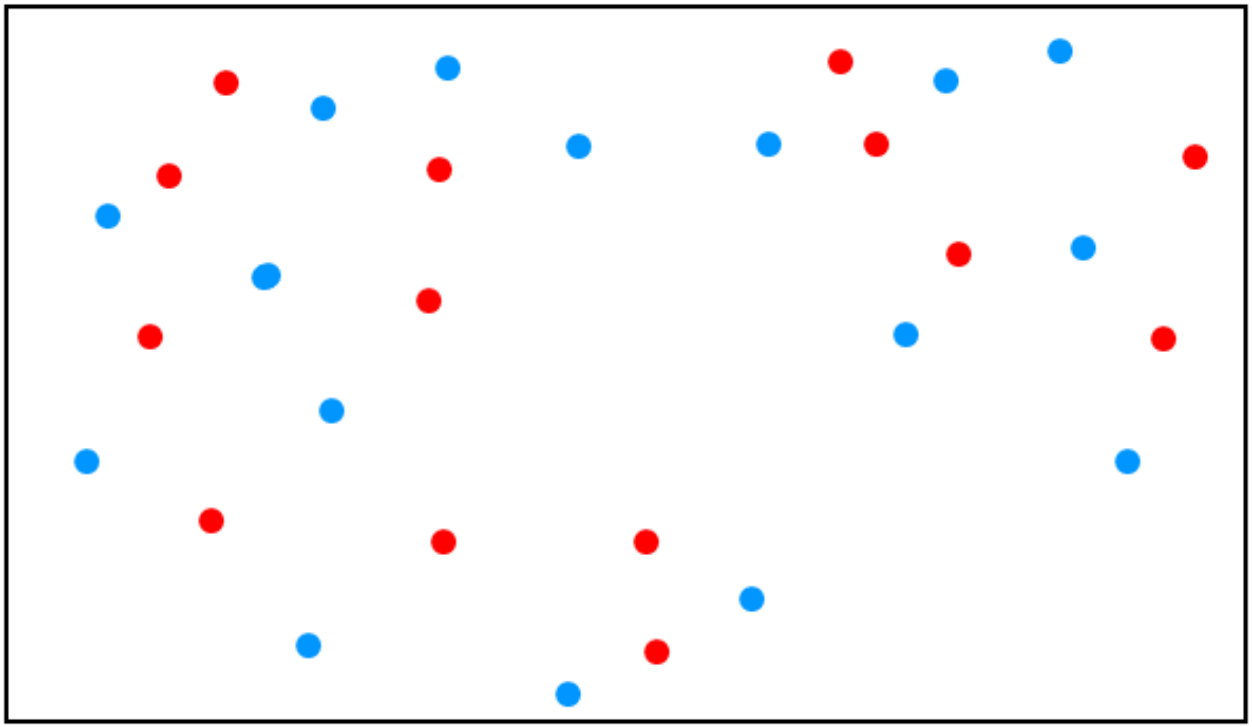
Jason Chiu and Raunak Kumar

2017/03/08

University of British Columbia

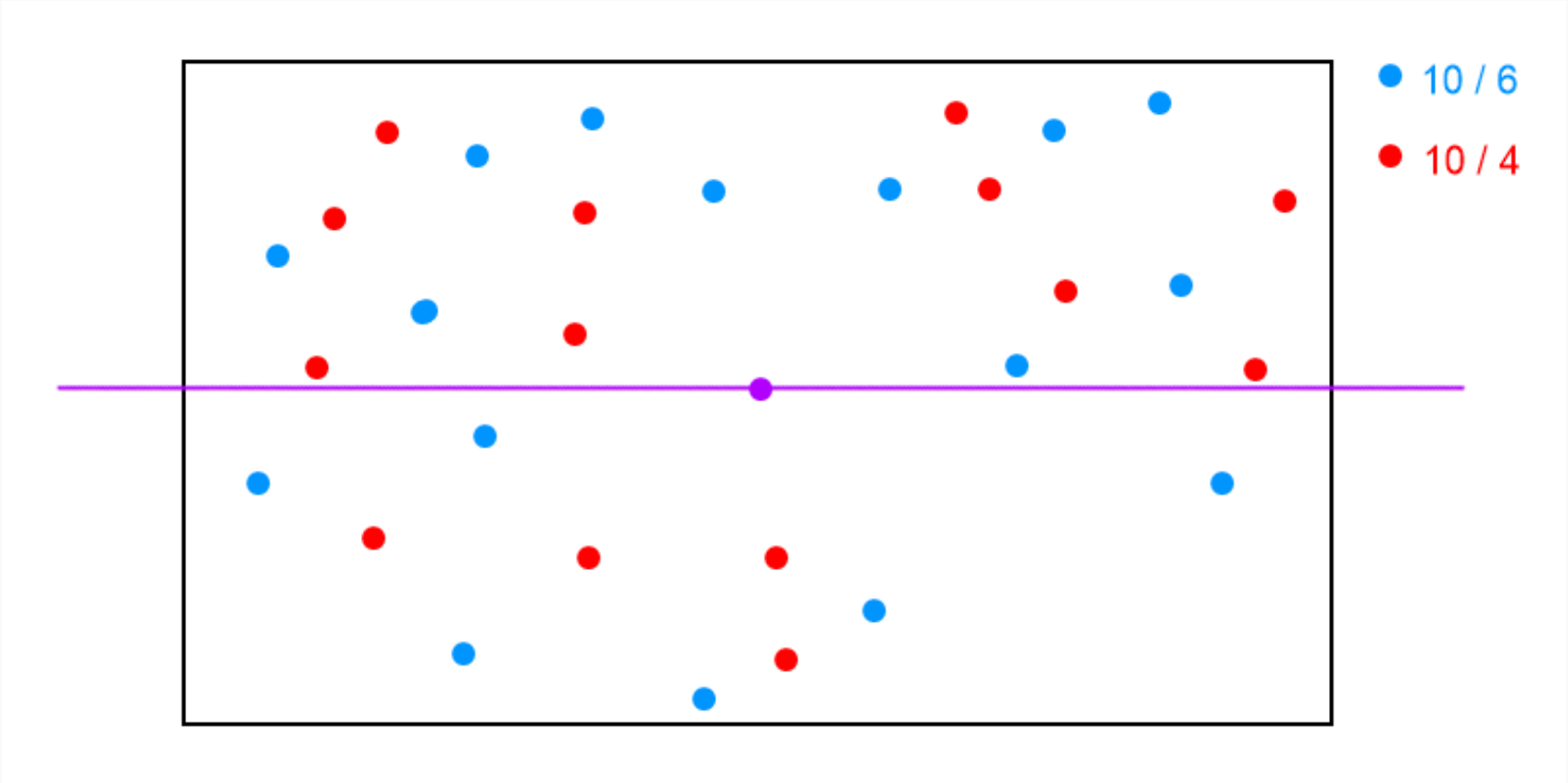
Problem 1: Angular Sweep

Divide the rectangle into two halves with equal area such that the two sides have an equal number of blue dots and red dots.



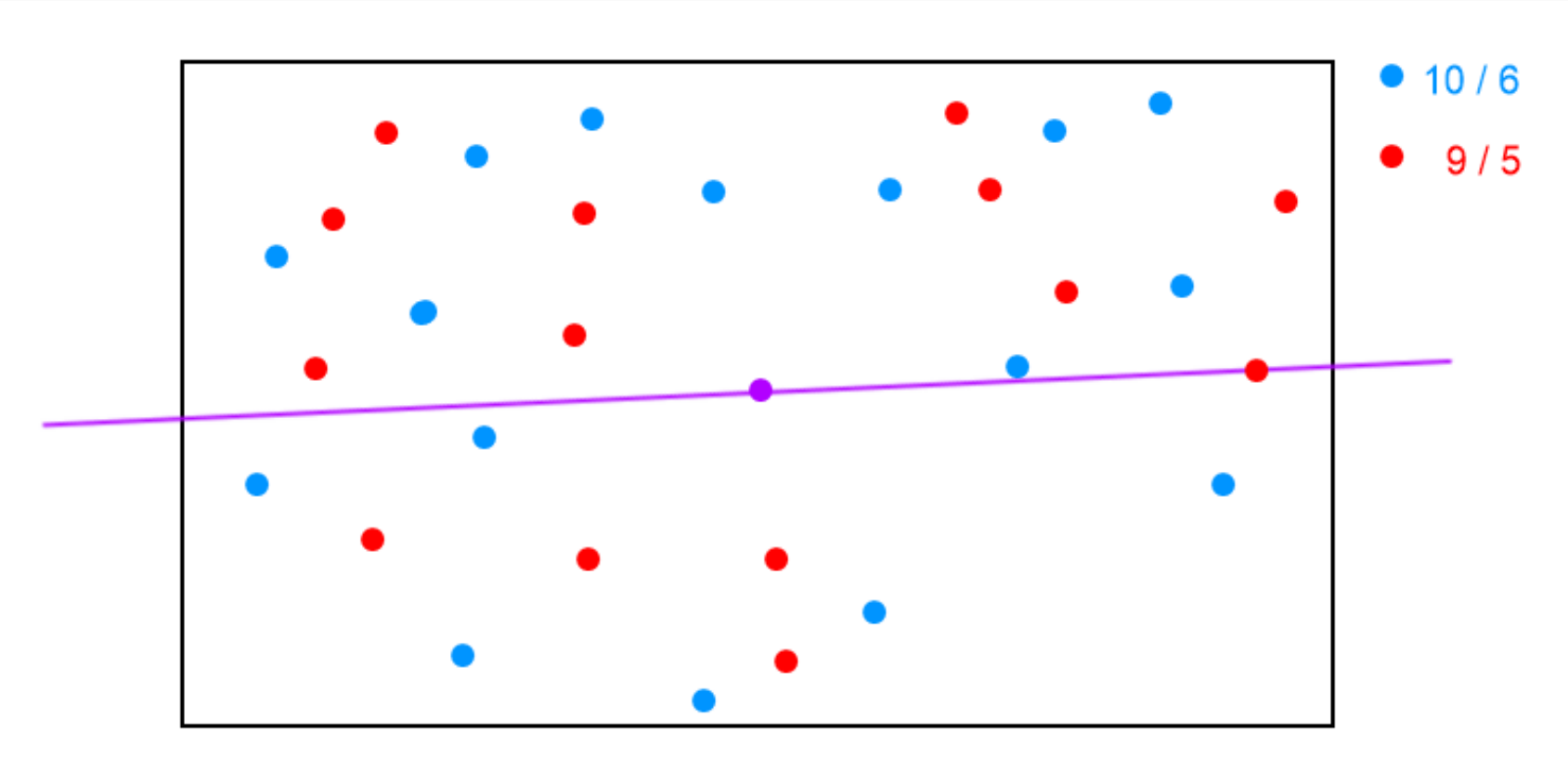
Problem 1: Solution

Observation: line must go through center $\Rightarrow$ do an "angular sweep" by rotating a line around the center of rectangle!



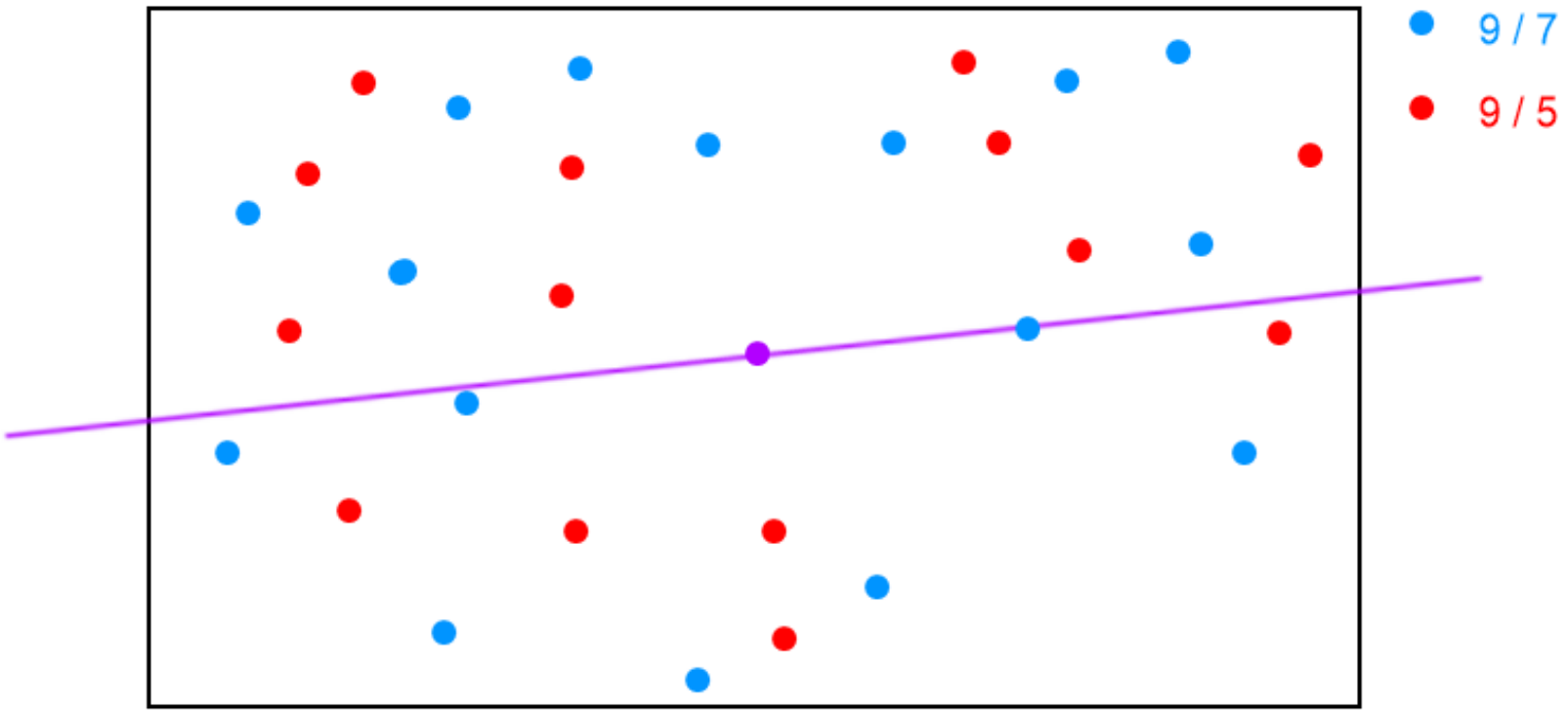
Problem 1: Solution

Observation: line must go through center $\Rightarrow$ do an "angular sweep" by rotating a line around the center of rectangle!



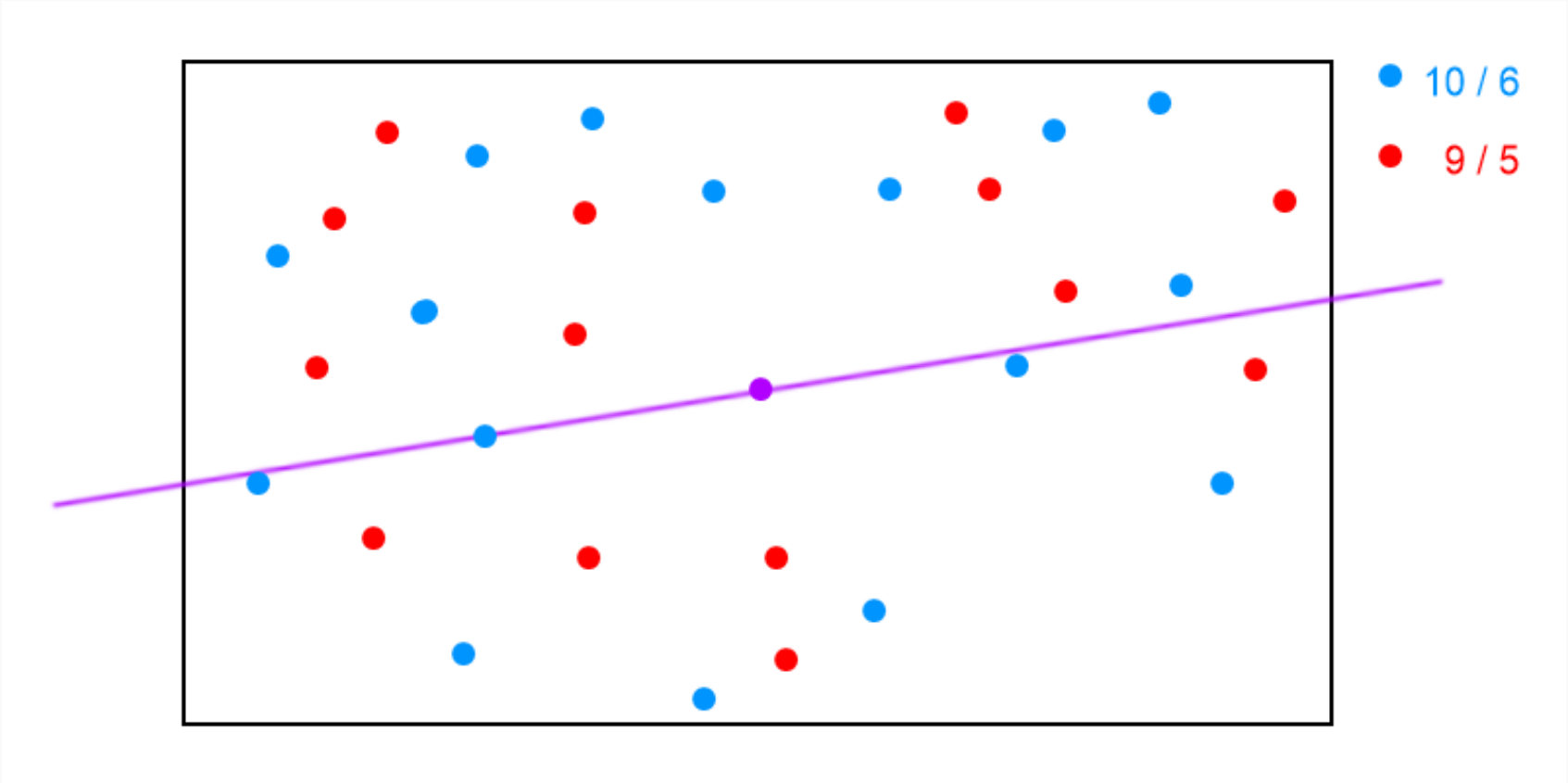
Problem 1: Solution

Observation: line must go through center $\Rightarrow$ do an "angular sweep" by rotating a line around the center of rectangle!



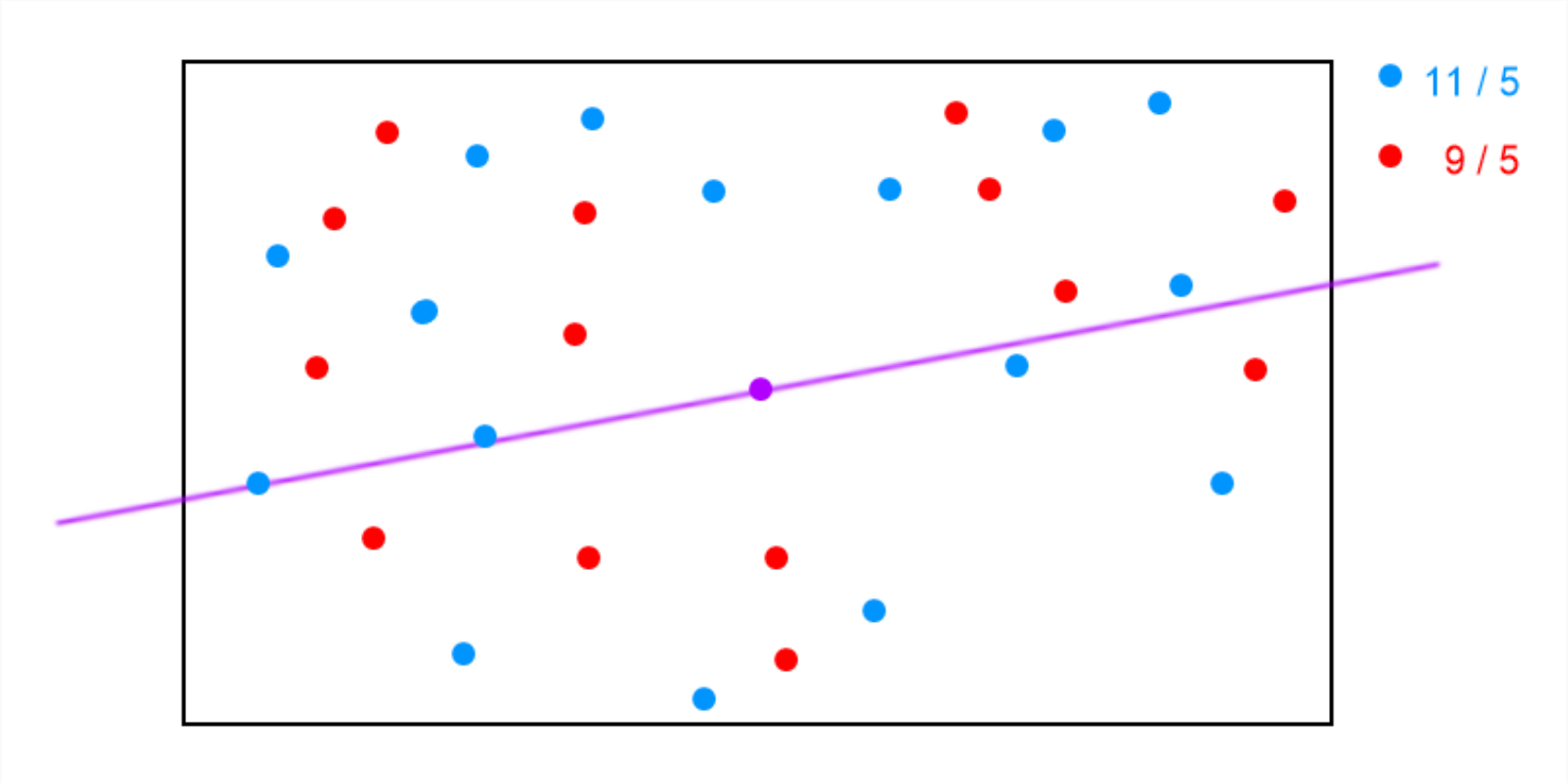
Problem 1: Solution

Observation: line must go through center $\Rightarrow$ do an "angular sweep" by rotating a line around the center of rectangle!



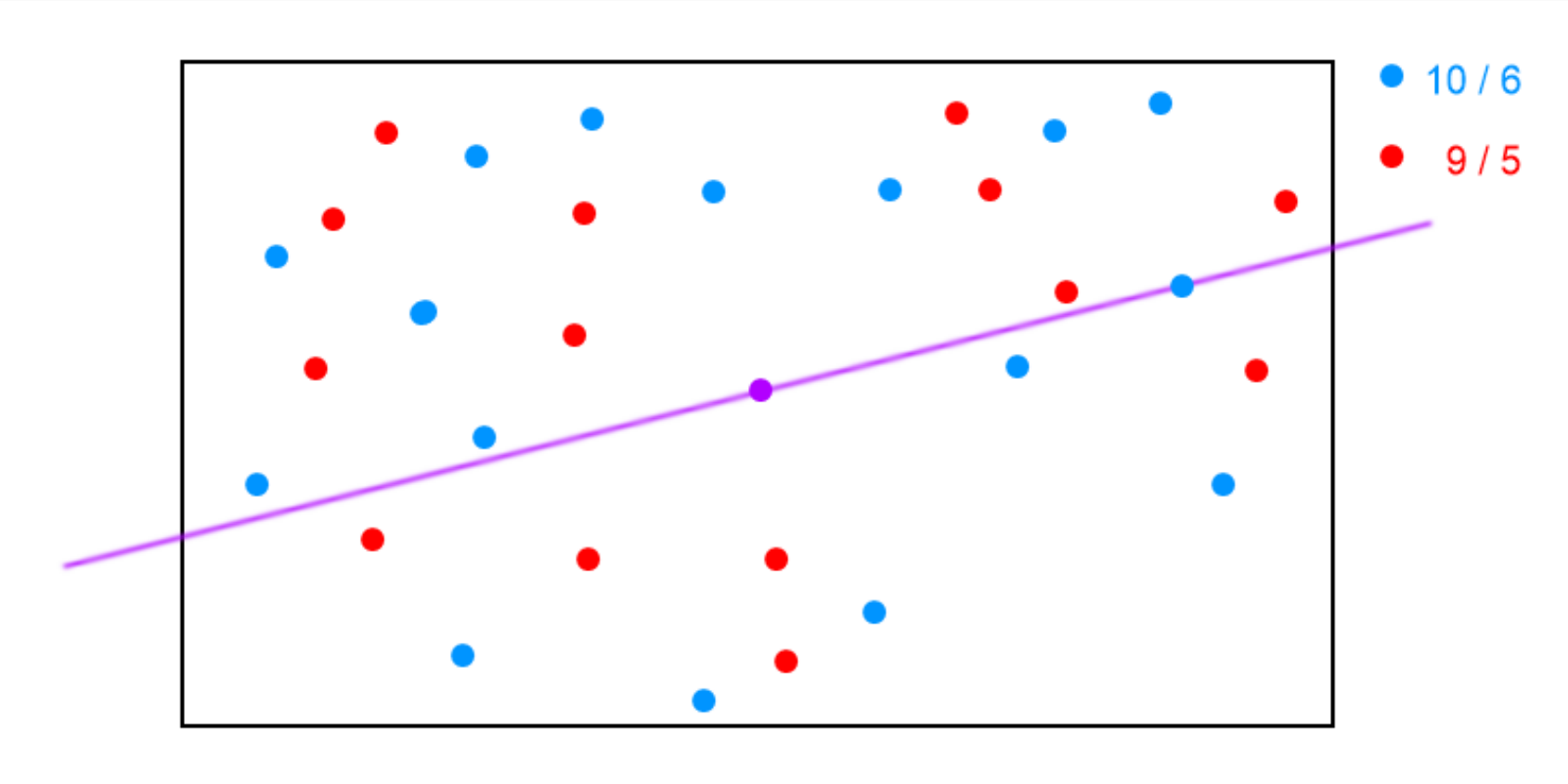
Problem 1: Solution

Observation: line must go through center $\Rightarrow$ do an "angular sweep" by rotating a line around the center of rectangle!



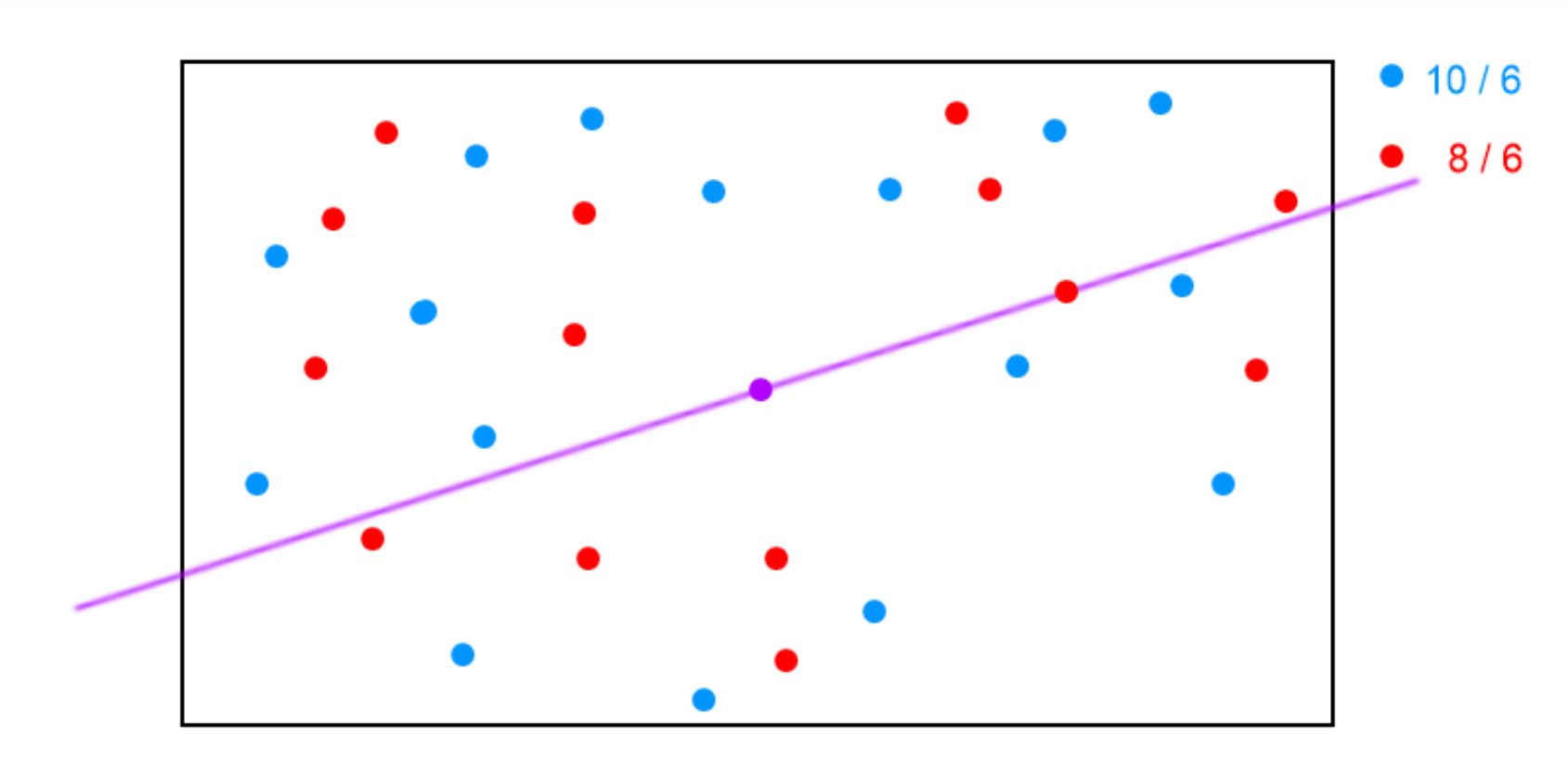
Problem 1: Solution

Observation: line must go through center ⇒ do an “angular sweep” by rotating a line around the center of rectangle!



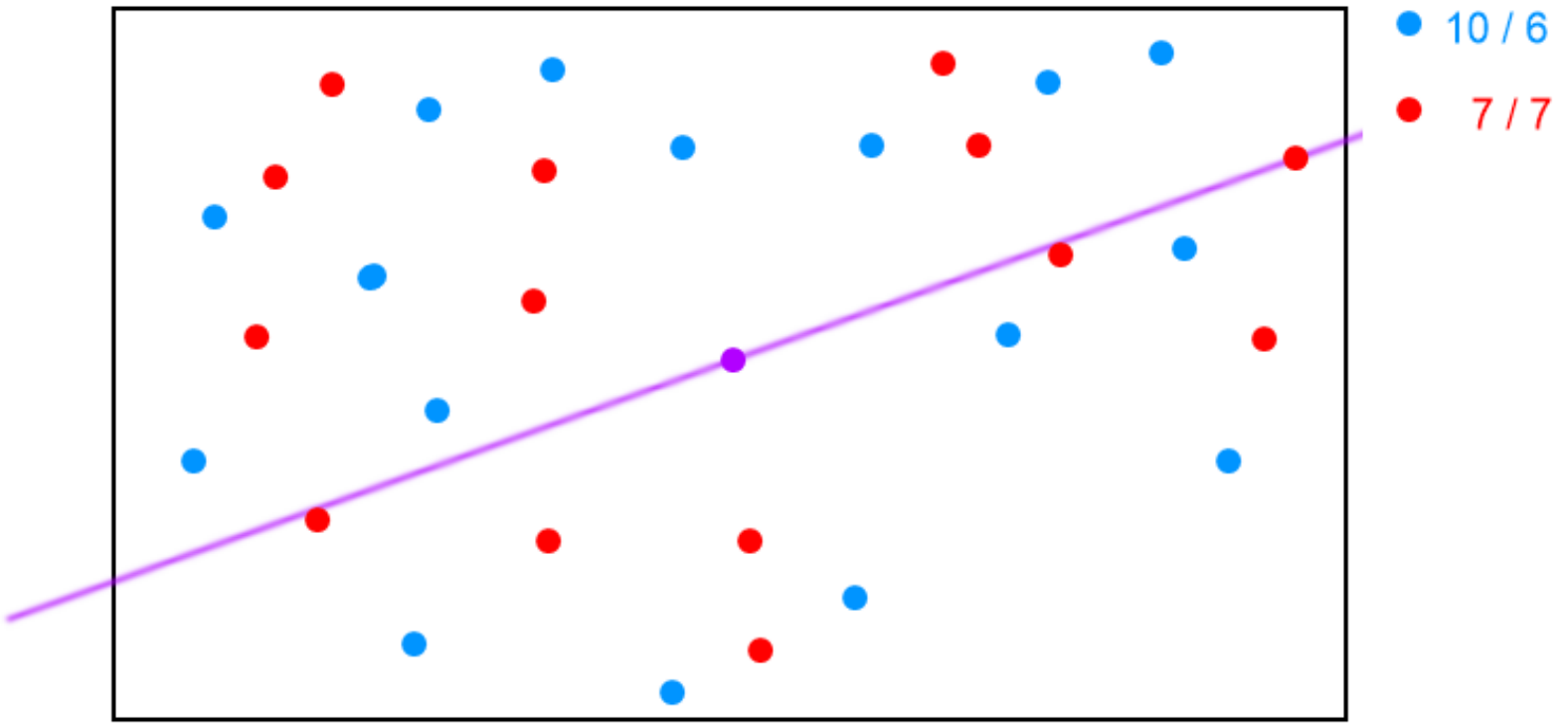
Problem 1: Solution

Observation: line must go through center $\Rightarrow$ do an "angular sweep" by rotating a line around the center of rectangle!



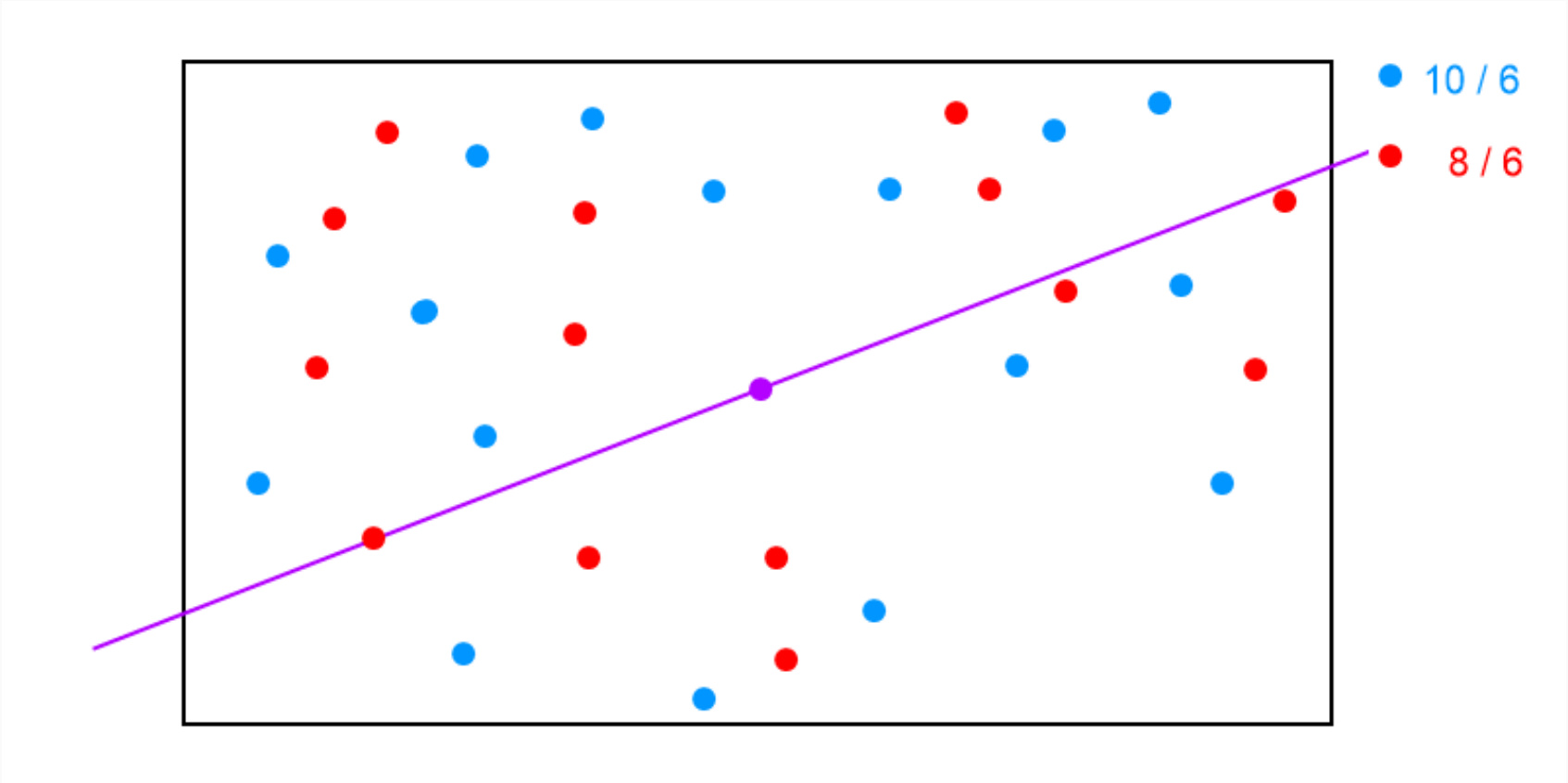
Problem 1: Solution

Observation: line must go through center ⇒ do an “angular sweep” by rotating a line around the center of rectangle!



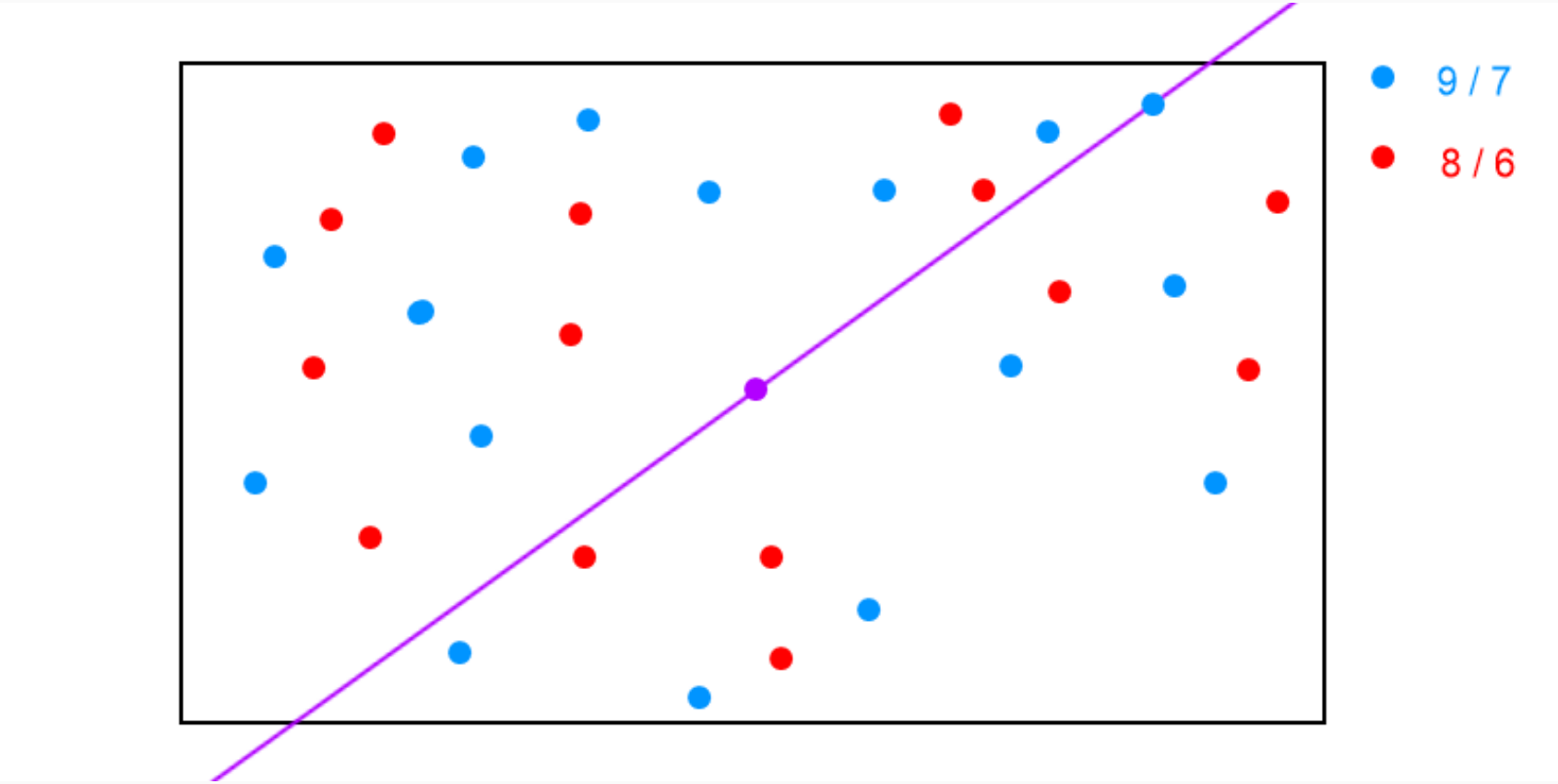
Problem 1: Solution

Observation: line must go through center $\Rightarrow$ do an "angular sweep" by rotating a line around the center of rectangle!



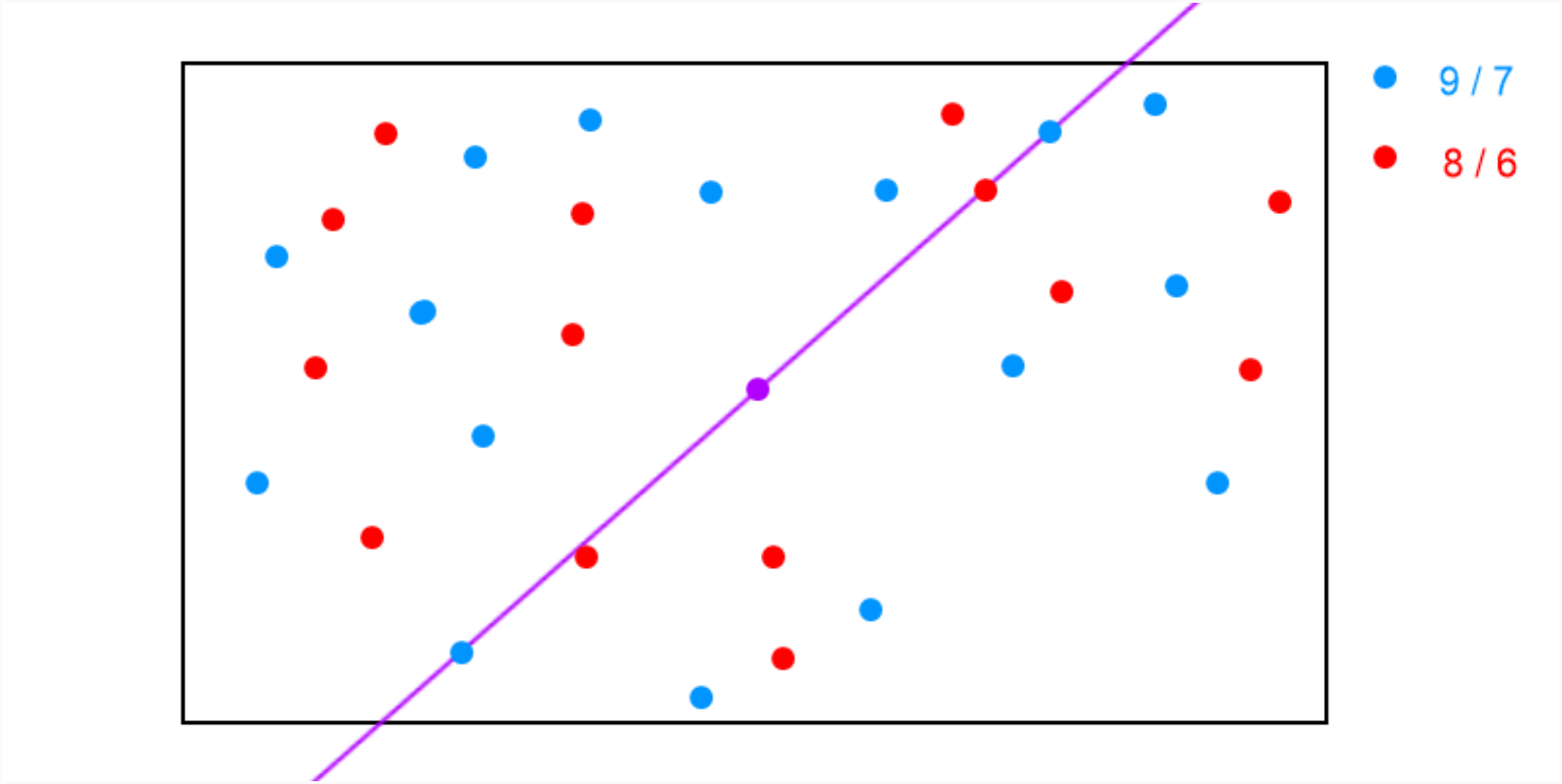
Problem 1: Solution

Observation: line must go through center $\Rightarrow$ do an "angular sweep" by rotating a line around the center of rectangle!



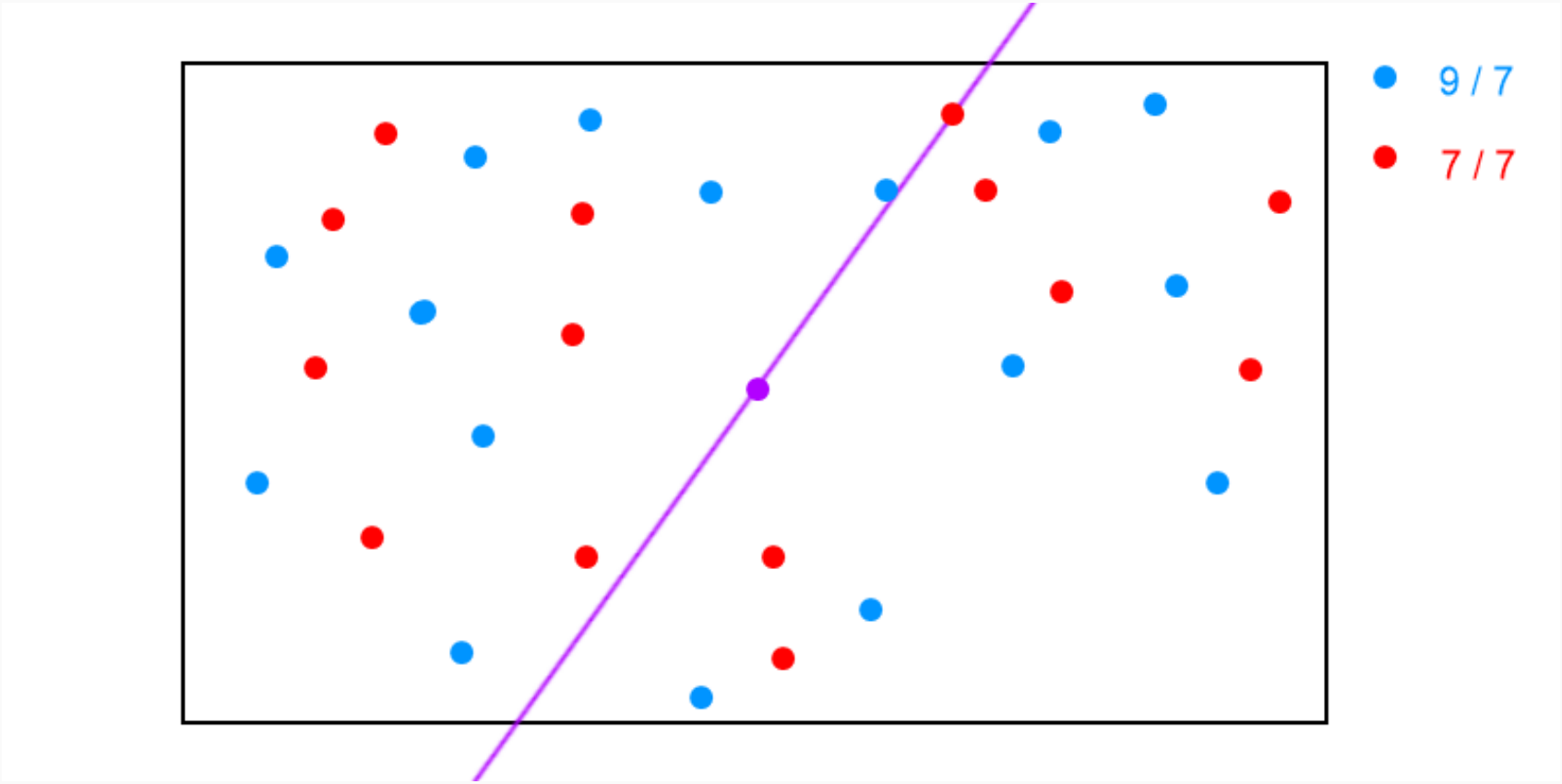
Problem 1: Solution

Observation: line must go through center $\Rightarrow$ do an "angular sweep" by rotating a line around the center of rectangle!



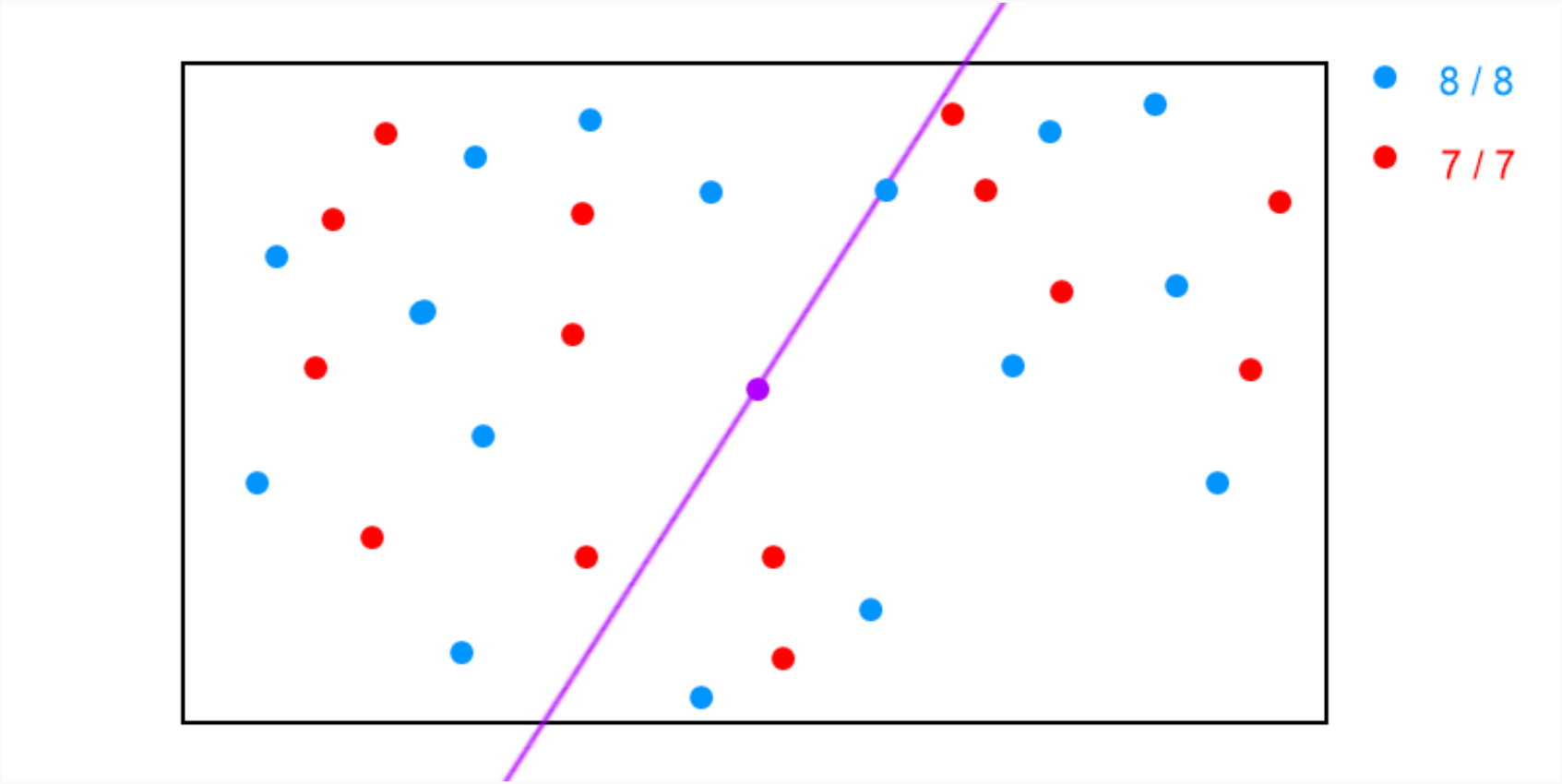
Problem 1: Solution

Observation: line must go through center $\Rightarrow$ do an "angular sweep" by rotating a line around the center of rectangle!



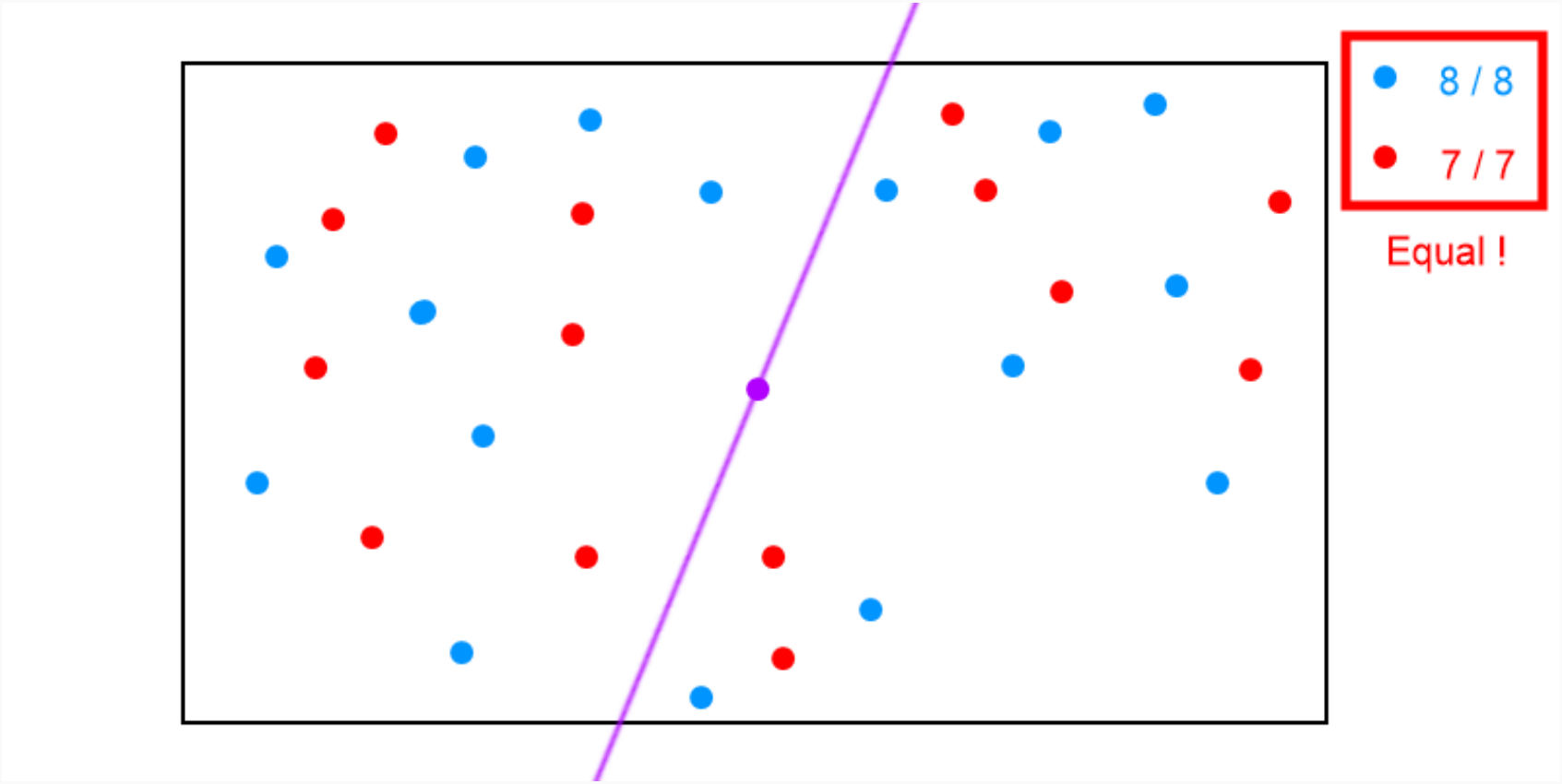
Problem 1: Solution

Observation: line must go through center $\Rightarrow$ do an "angular sweep" by rotating a line around the center of rectangle!



Problem 1: Solution

Observation: line must go through center ⇒ do an “angular sweep” by rotating a line around the center of rectangle!



Problem 1: Angular Sweep

State: how many of each kind of dot is on each side.

Event: line hits a dot on either side, sorted by angle around center. (i.e. must add a "phantom event" on opposite side for dots $\geq 180°$)

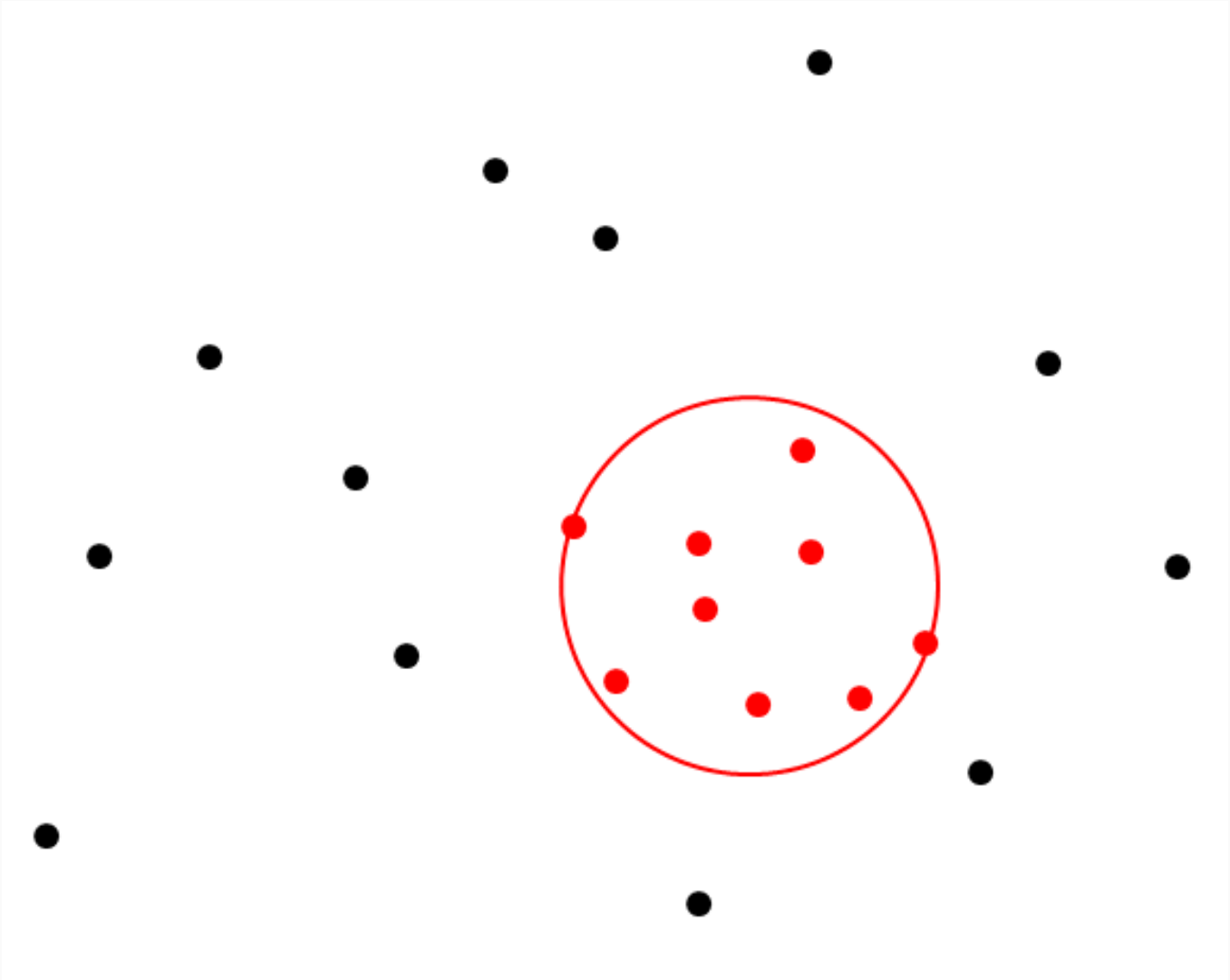
Action: when line hits a dot, $+1$ or -1 to the relevant state.

Time complexity: $O(n \log n)$

Problem 2: Angular Sweep with Circle

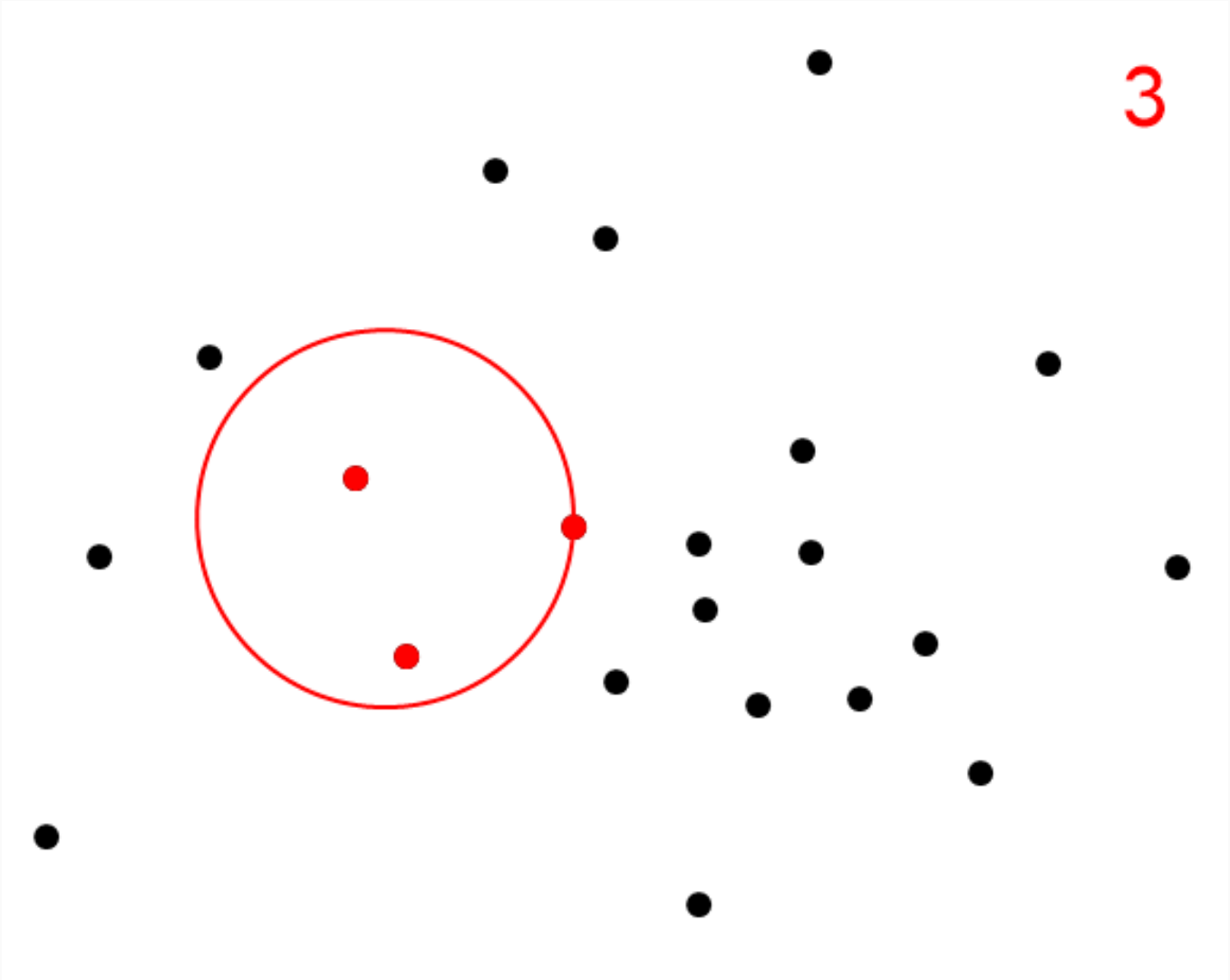
Find circle with radius R that covers the most points

Hint: rotate *what* around *where*? $O(n^2 \log n)$



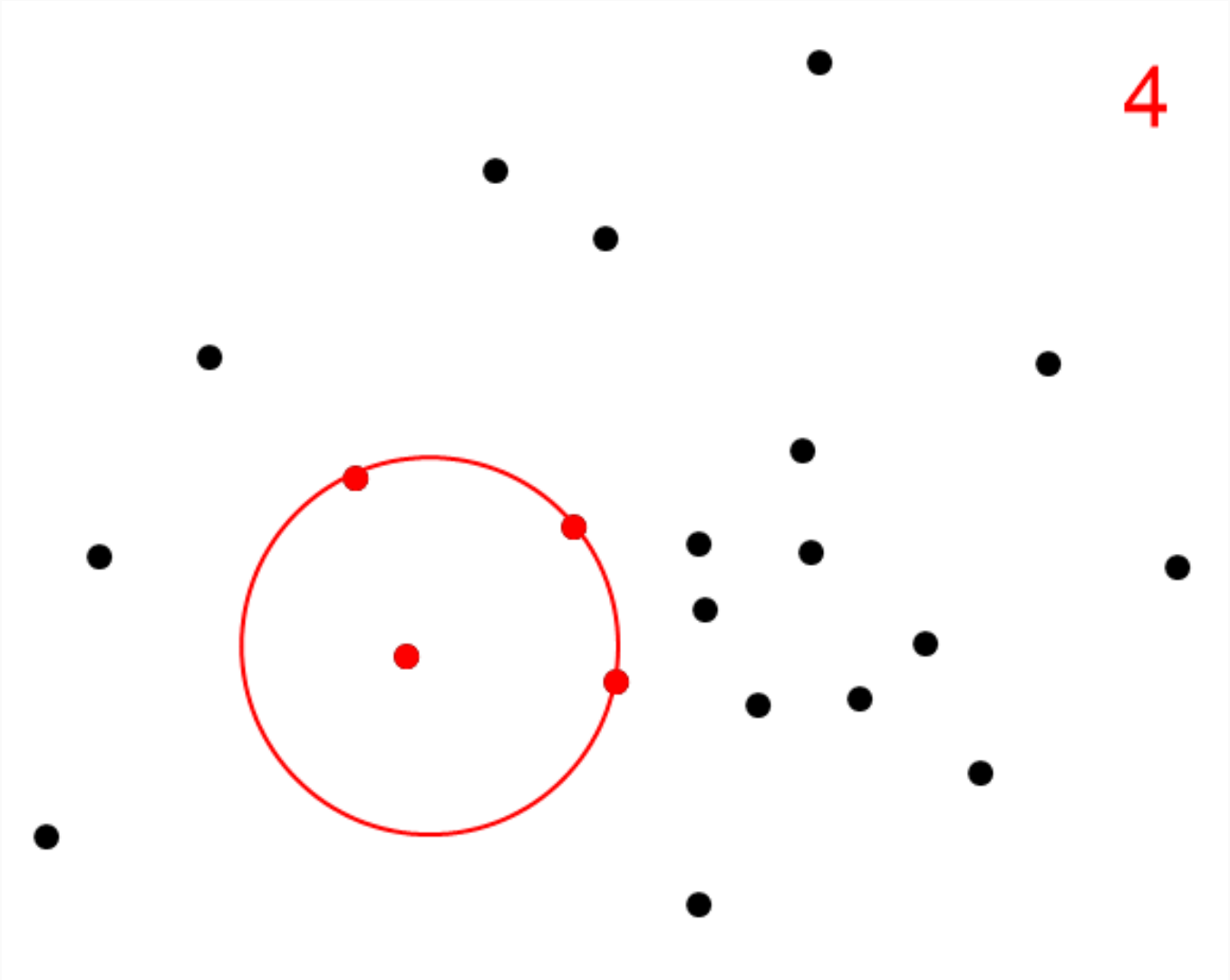
Problem 2: Solution

Observation: can wiggle circle so that it tangents one point $\Rightarrow$ try every single point, rotate a circle around that point.



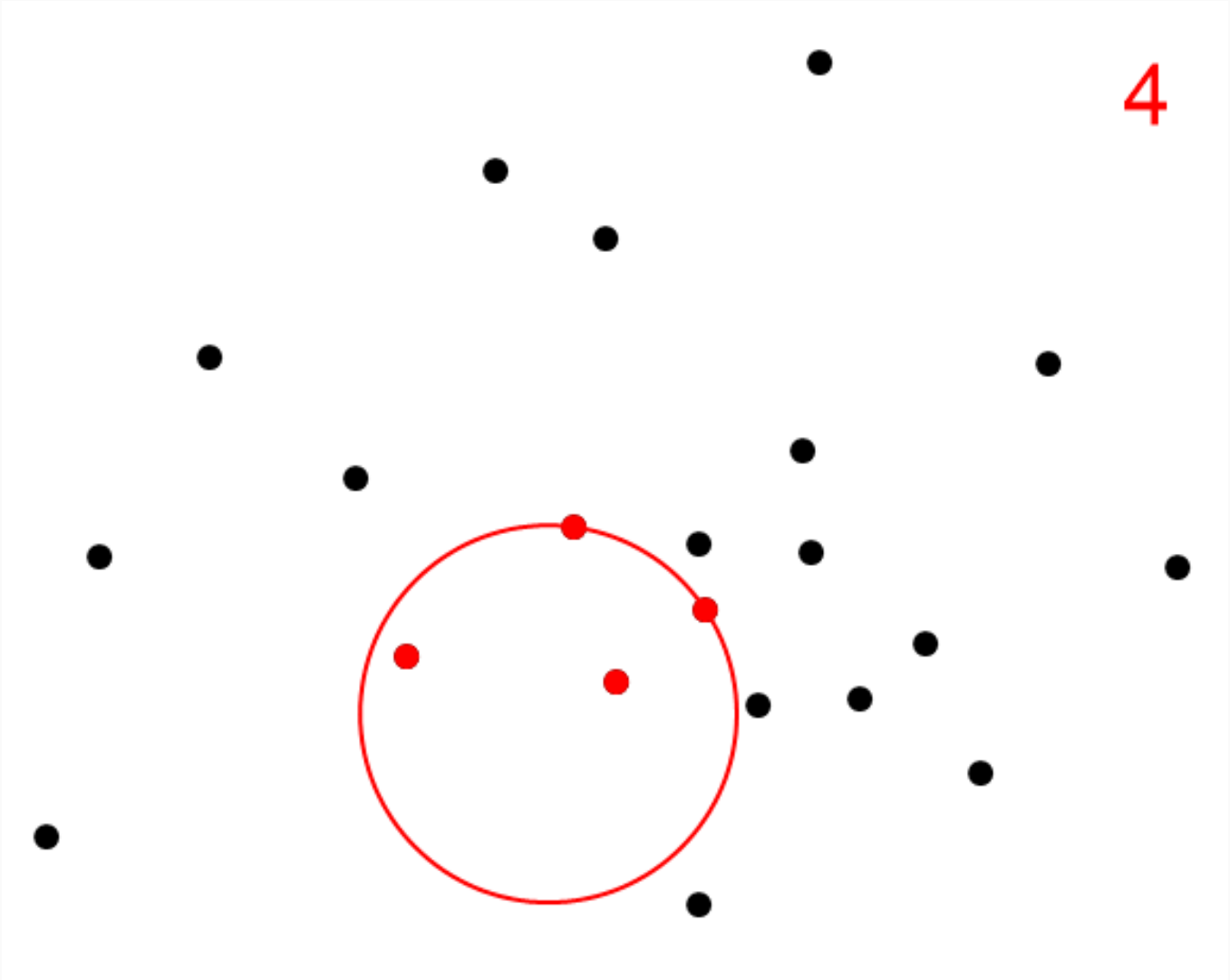
Problem 2: Solution

Observation: can wiggle circle so that it tangents one point $\Rightarrow$ try every single point, rotate a circle around that point.



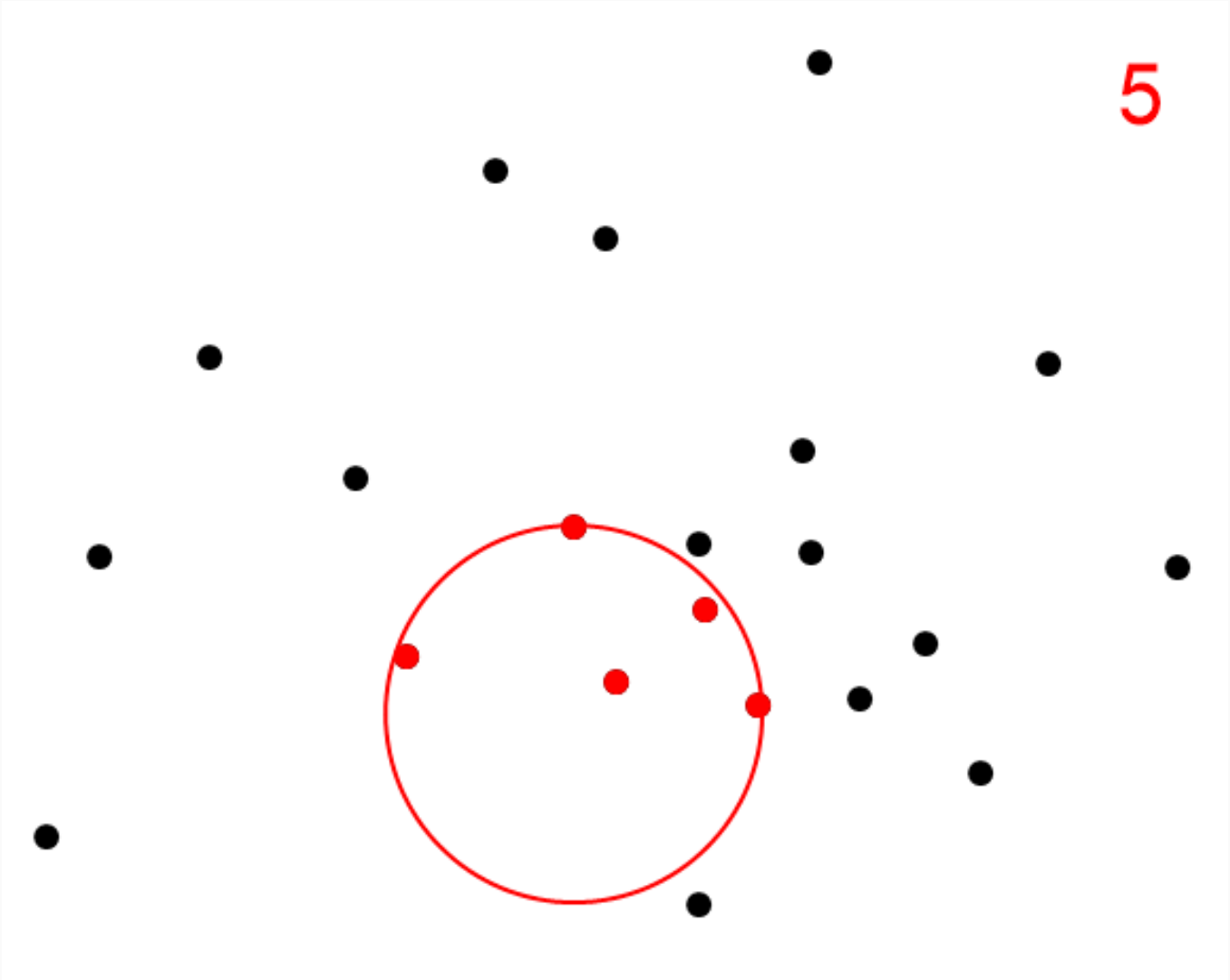
Problem 2: Solution

Observation: can wiggle circle so that it tangents one point $\Rightarrow$ try every single point, rotate a circle around that point.



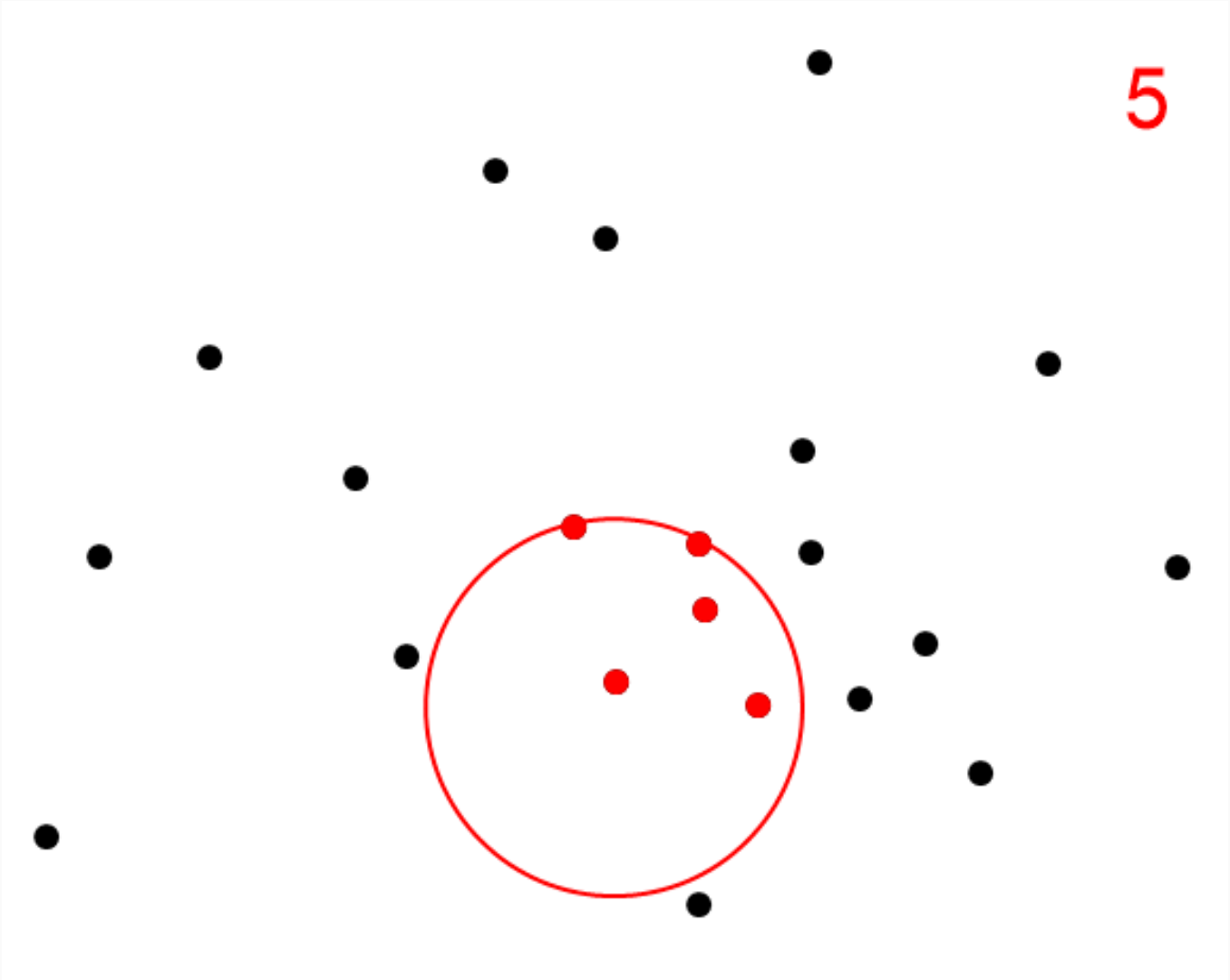
Problem 2: Solution

Observation: can wiggle circle so that it tangents one point $\Rightarrow$ try every single point, rotate a circle around that point.



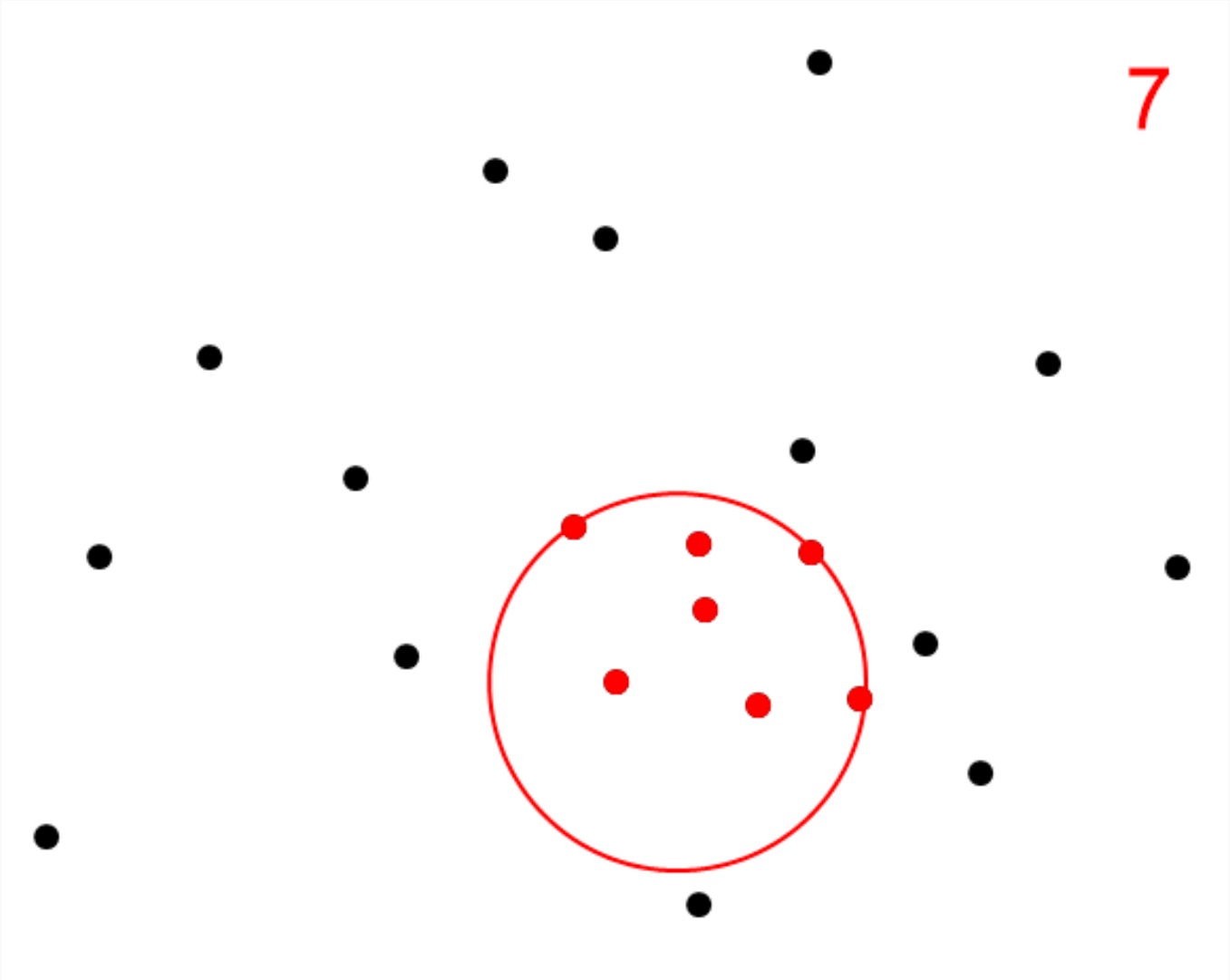
Problem 2: Solution

Observation: can wiggle circle so that it tangents one point $\Rightarrow$ try every single point, rotate a circle around that point.



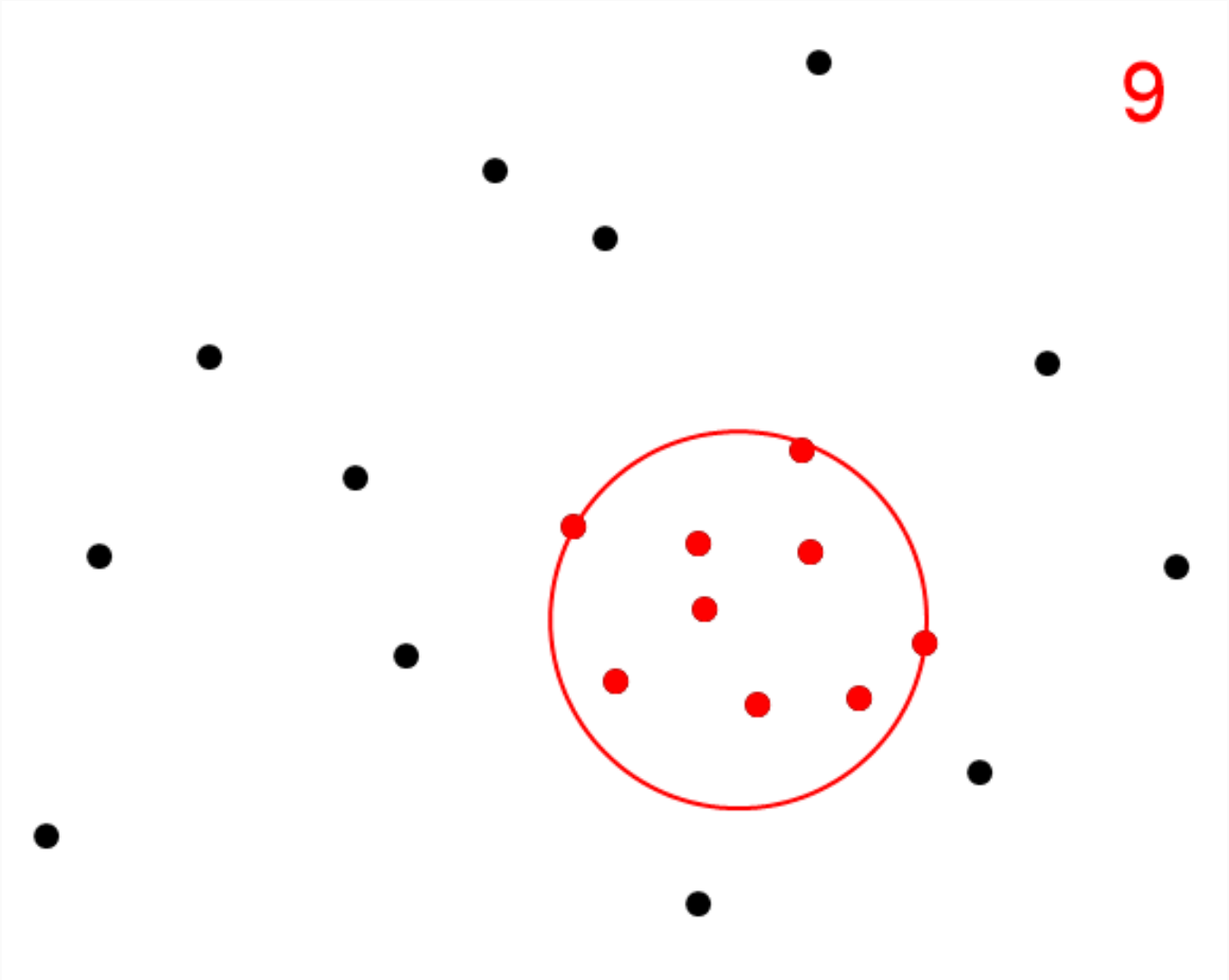
Problem 2: Solution

Observation: can wiggle circle so that it tangents one point $\Rightarrow$ try every single point, rotate a circle around that point.



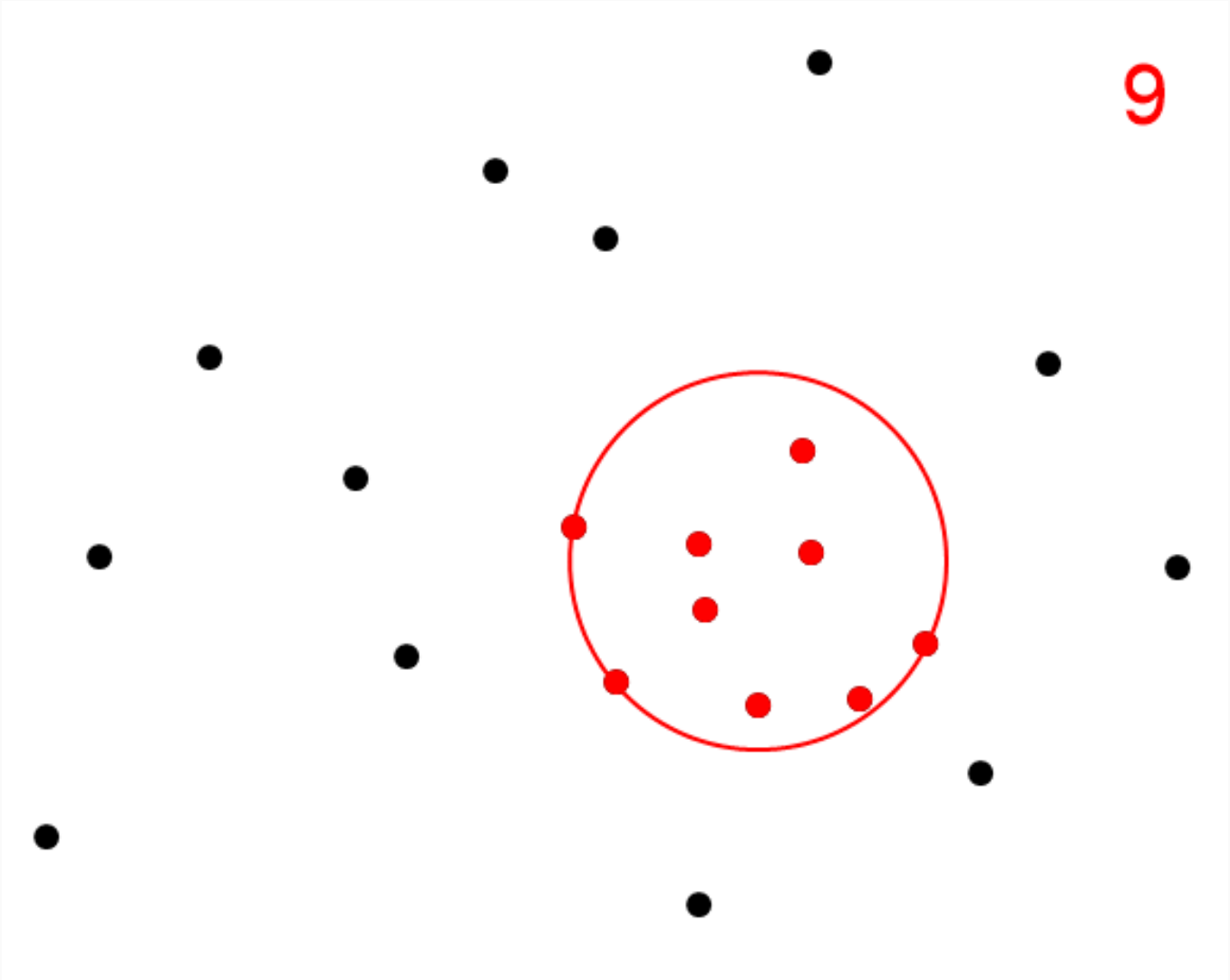
Problem 2: Solution

Observation: can wiggle circle so that it tangents one point $\Rightarrow$ try every single point, rotate a circle around that point.



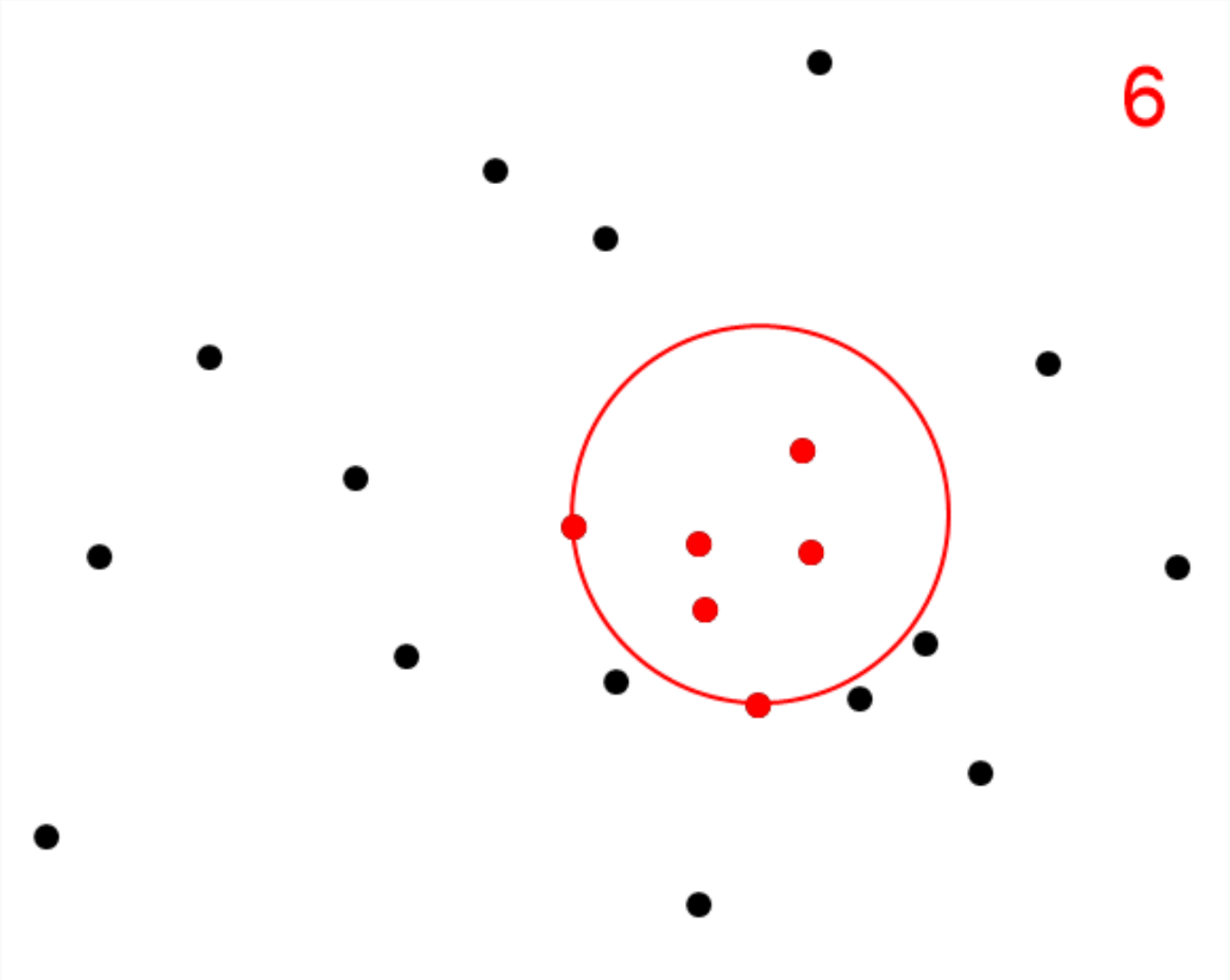
Problem 2: Solution

Observation: can wiggle circle so that it tangents one point $\Rightarrow$ try every single point, rotate a circle around that point.



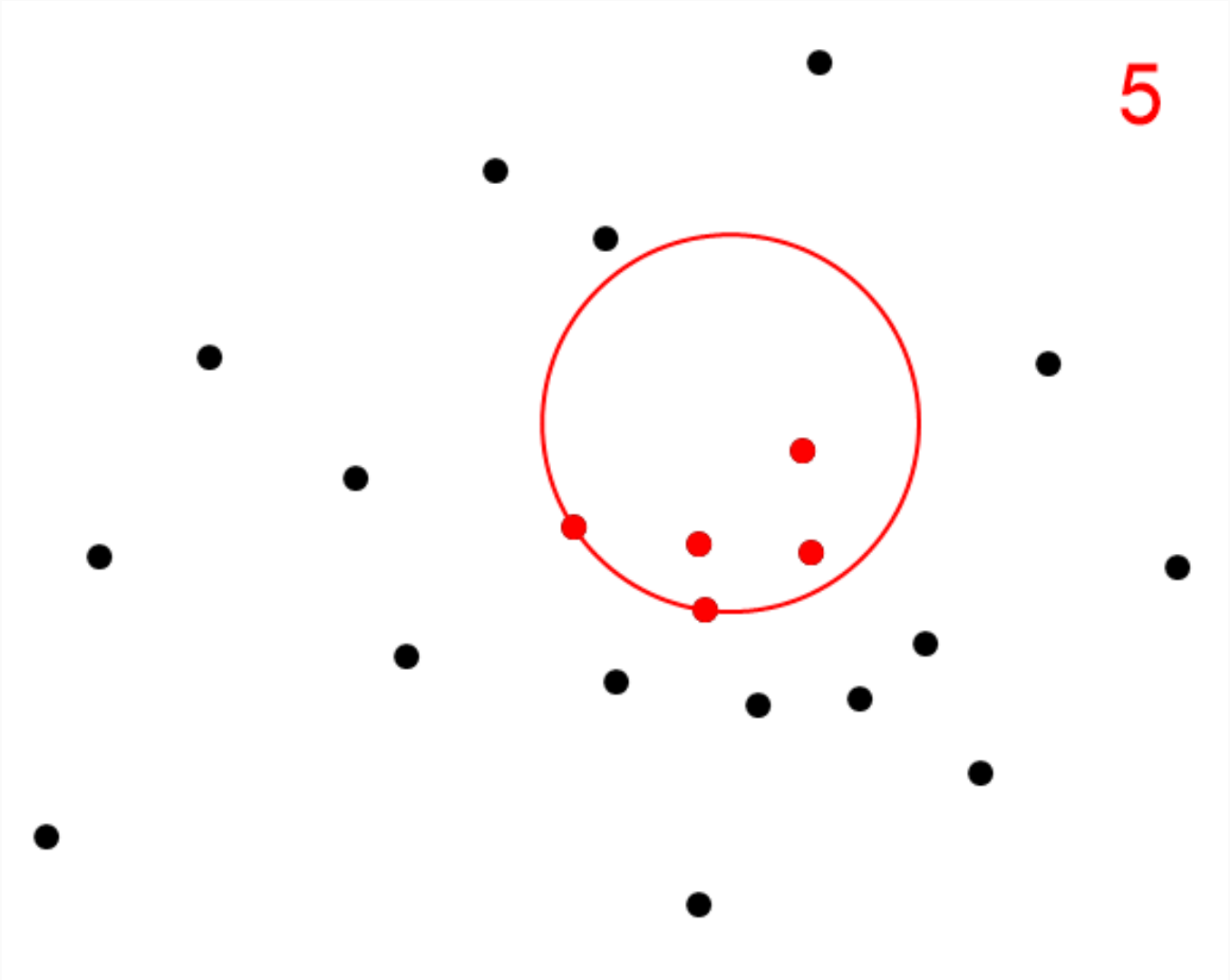
Problem 2: Solution

Observation: can wiggle circle so that it tangents one point $\Rightarrow$ try every single point, rotate a circle around that point.



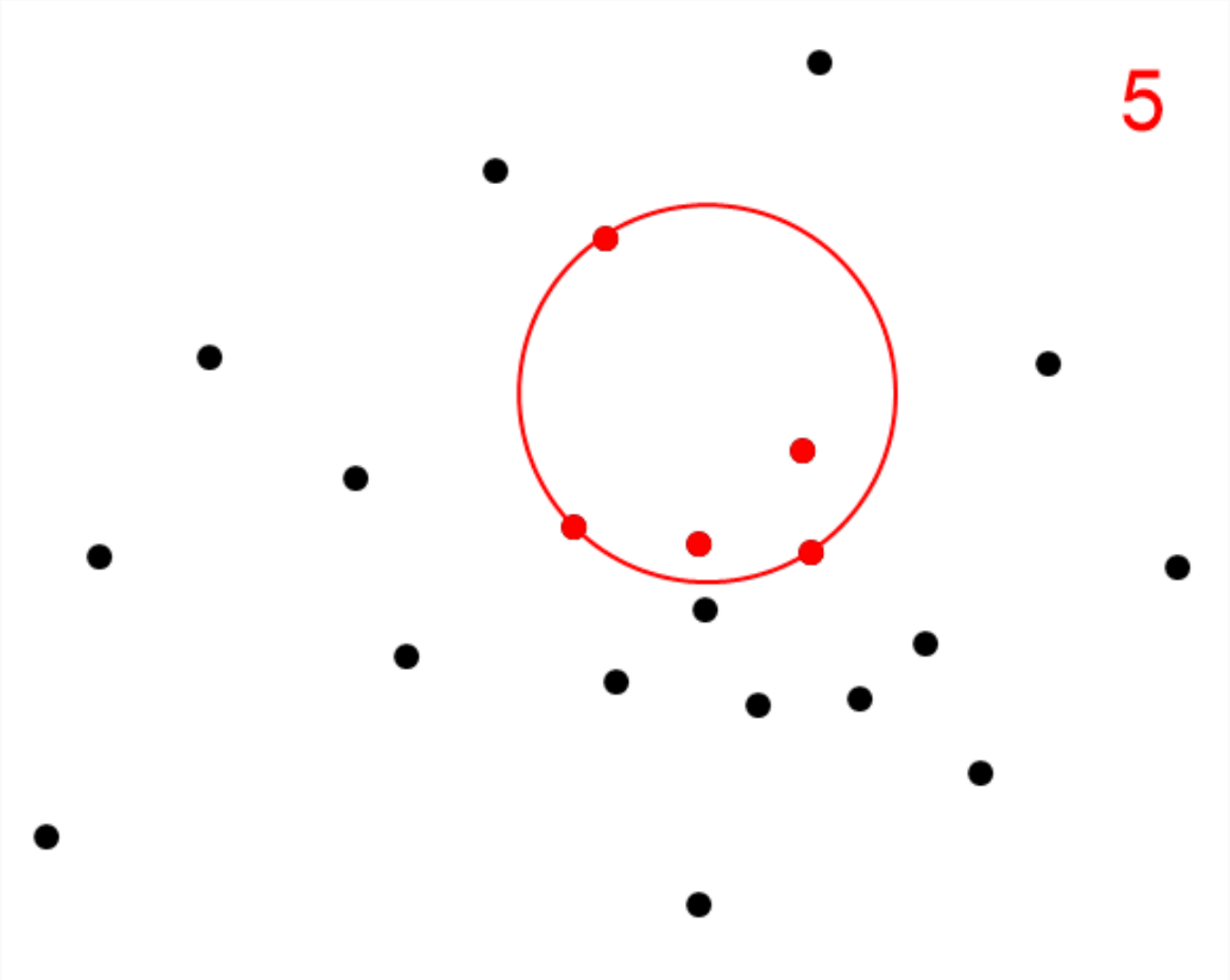
Problem 2: Solution

Observation: can wiggle circle so that it tangents one point $\Rightarrow$ try every single point, rotate a circle around that point.



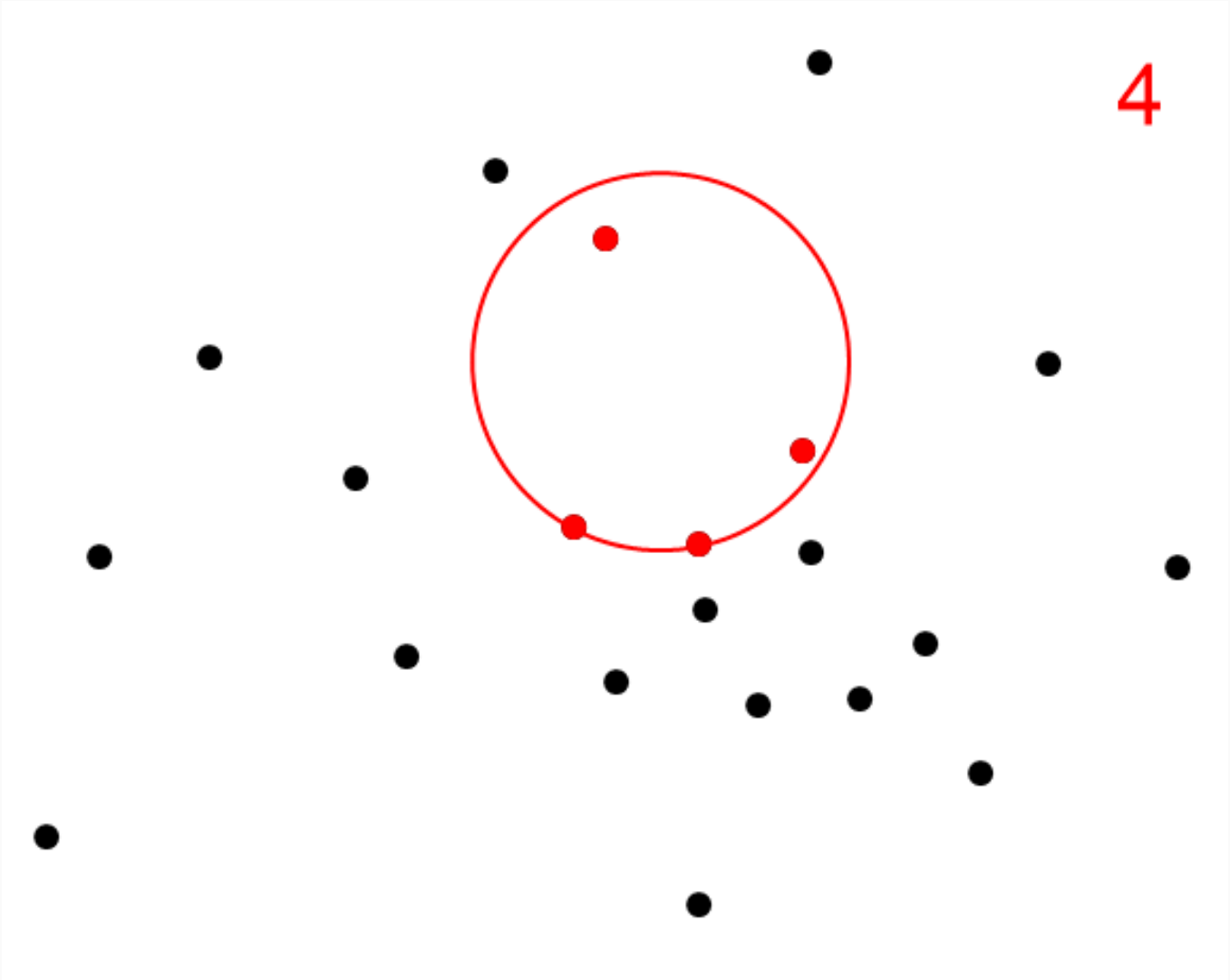
Problem 2: Solution

Observation: can wiggle circle so that it tangents one point $\Rightarrow$ try every single point, rotate a circle around that point.



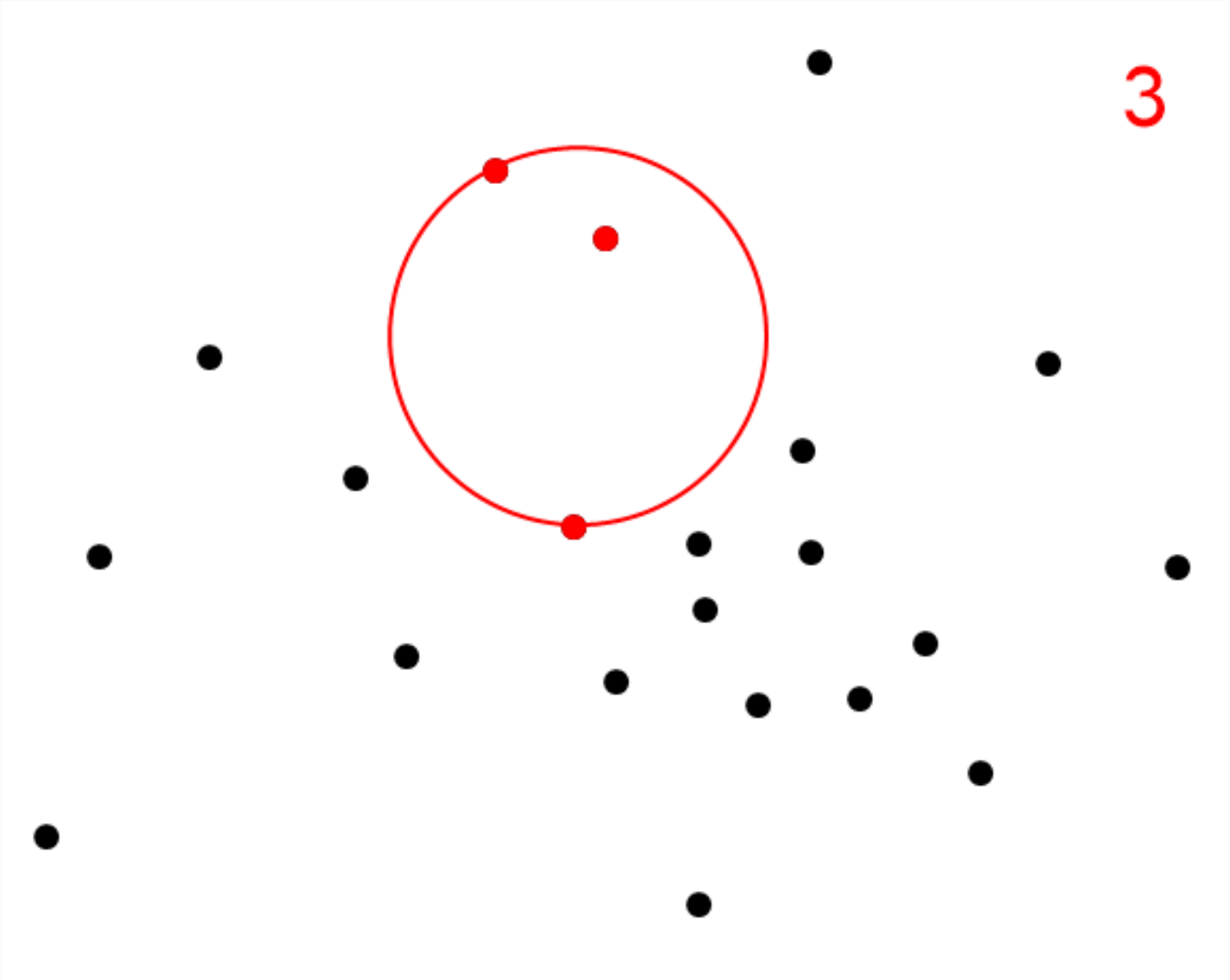
Problem 2: Solution

Observation: can wiggle circle so that it tangents one point $\Rightarrow$ try every single point, rotate a circle around that point.



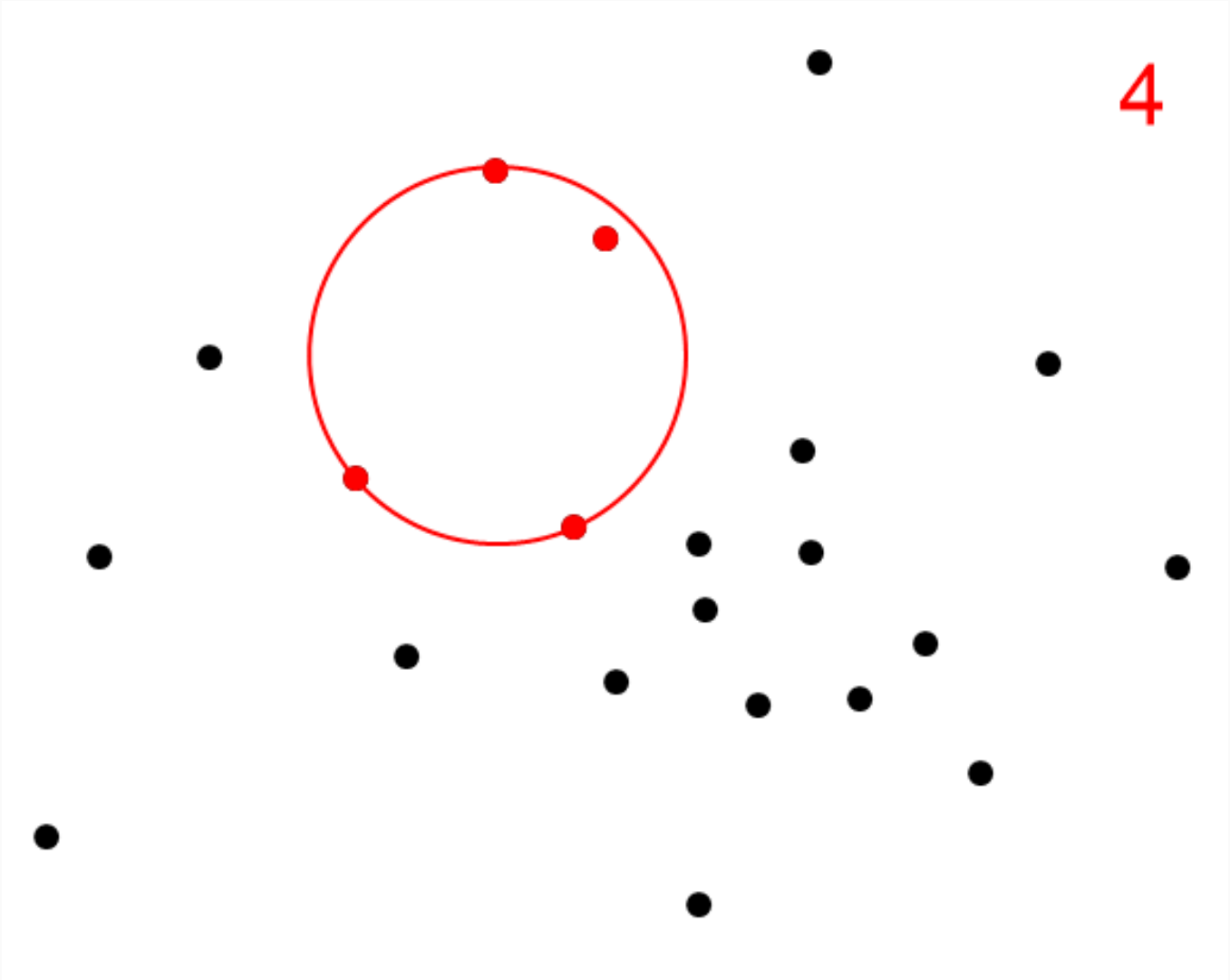
Problem 2: Solution

Observation: can wiggle circle so that it tangents one point $\Rightarrow$ try every single point, rotate a circle around that point.



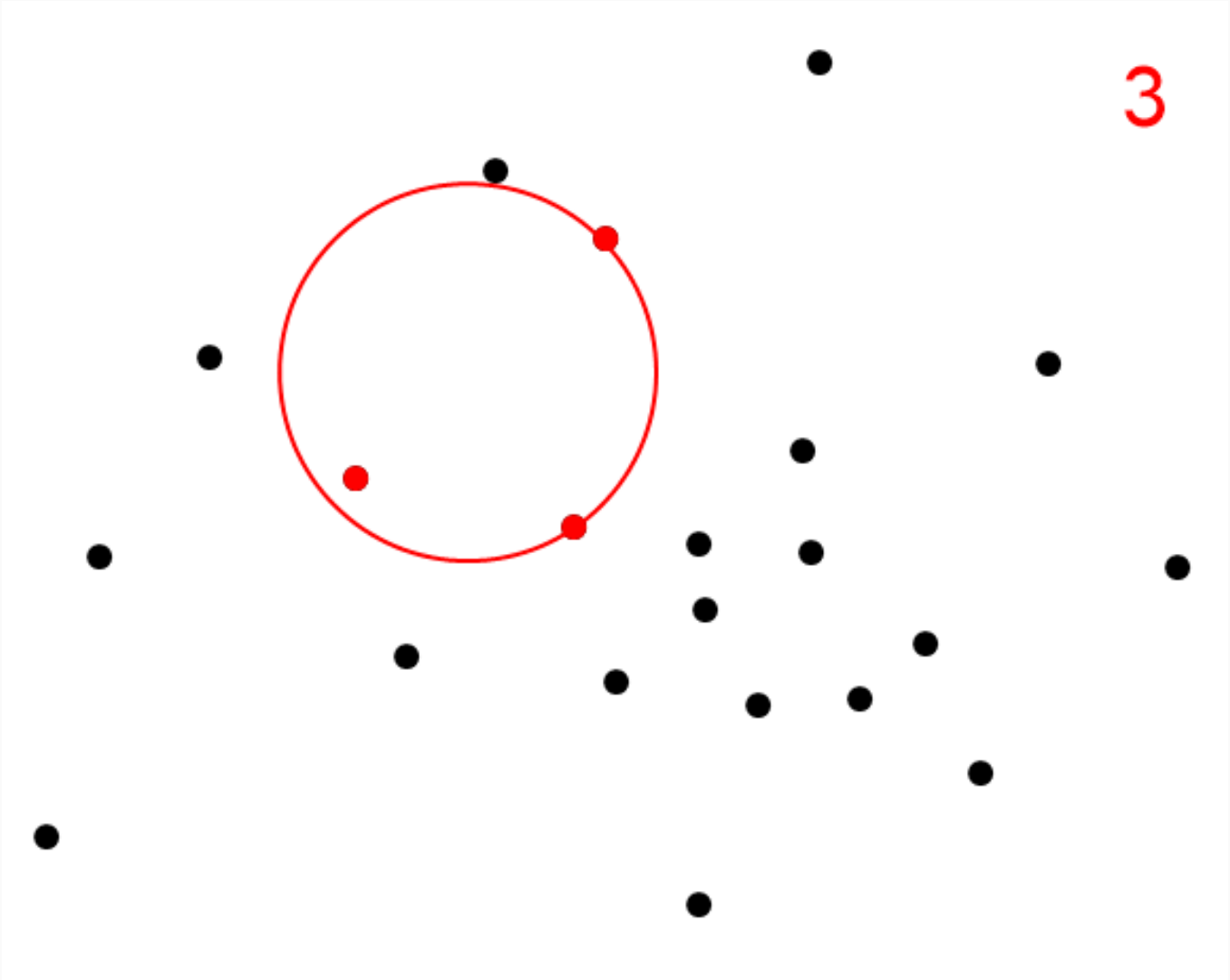
Problem 2: Solution

Observation: can wiggle circle so that it tangents one point $\Rightarrow$ try every single point, rotate a circle around that point.



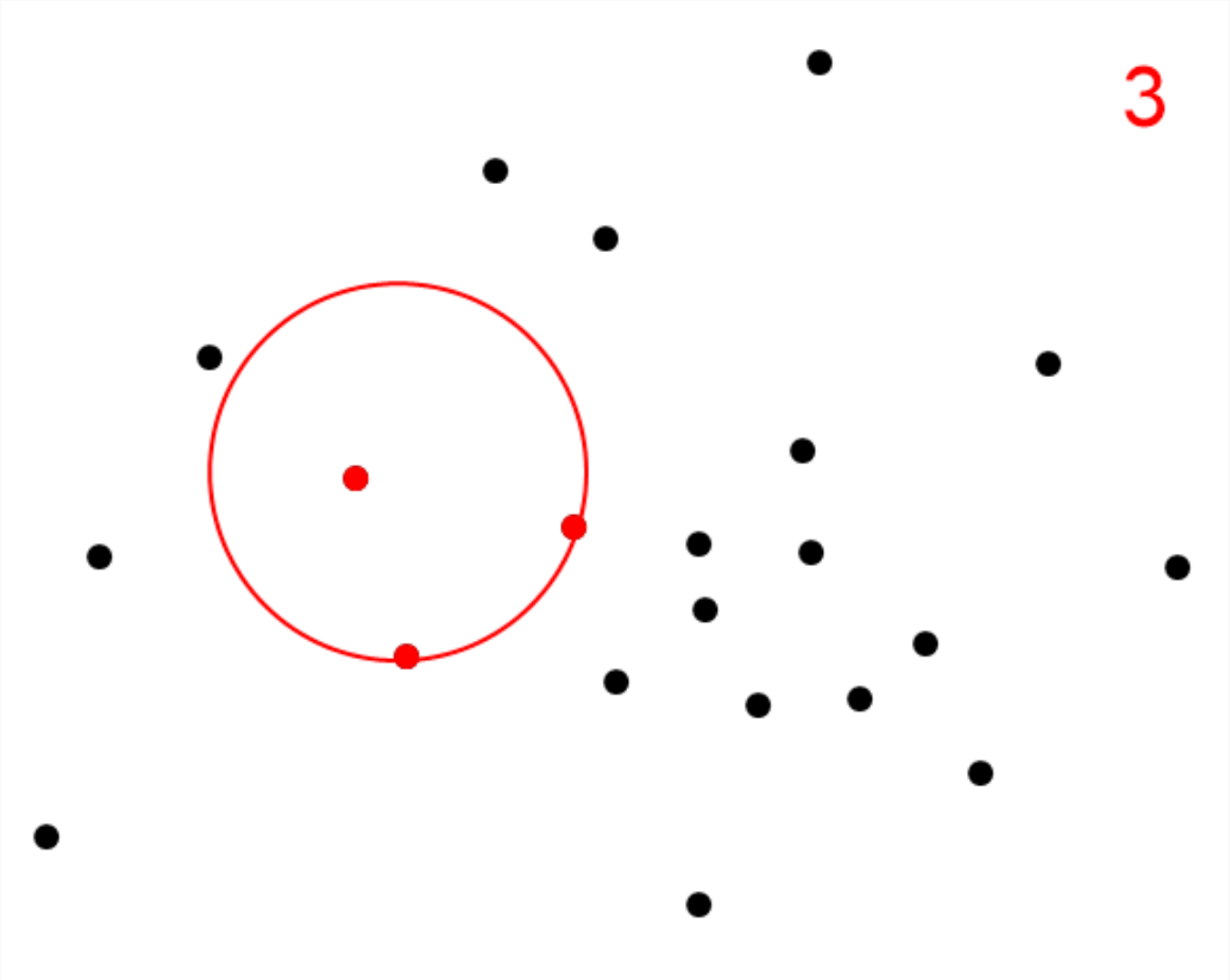
Problem 2: Solution

Observation: can wiggle circle so that it tangents one point $\Rightarrow$ try every single point, rotate a circle around that point.



Problem 2: Solution

Observation: can wiggle circle so that it tangents one point $\Rightarrow$ try every single point, rotate a circle around that point.



Problem 2: Angular Sweep with Circle

Idea: for every point, rotate a circle 360° around it

State: how many points are inside the rotating circle

Event: when point enters/exits circle, sorted by rotation angle of circle, be careful about "wrapping across" 360° and initial condition

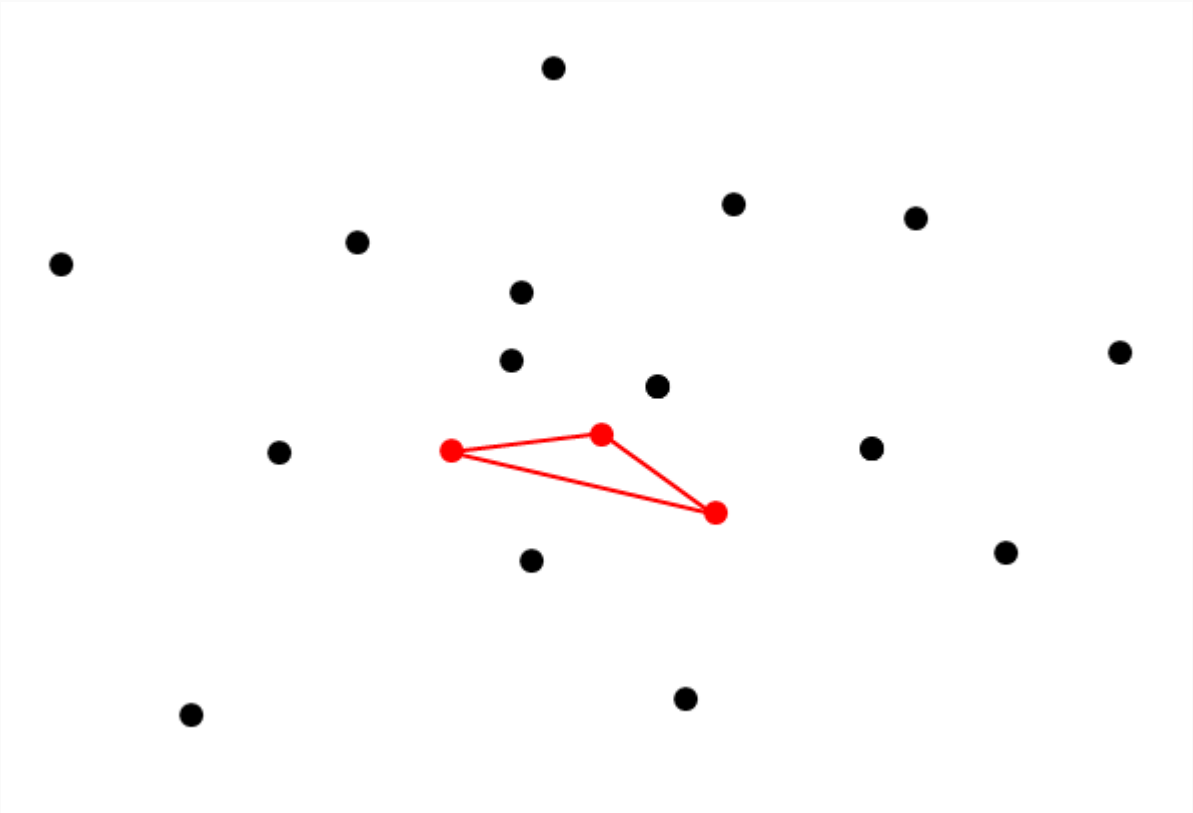
Action: when circle hits a point, $+1$ or -1 to the state.

Time complexity: $O(n^2 \log n)$

Problem 3: Duality

Find the smallest area triangle formed by using 3 of these points

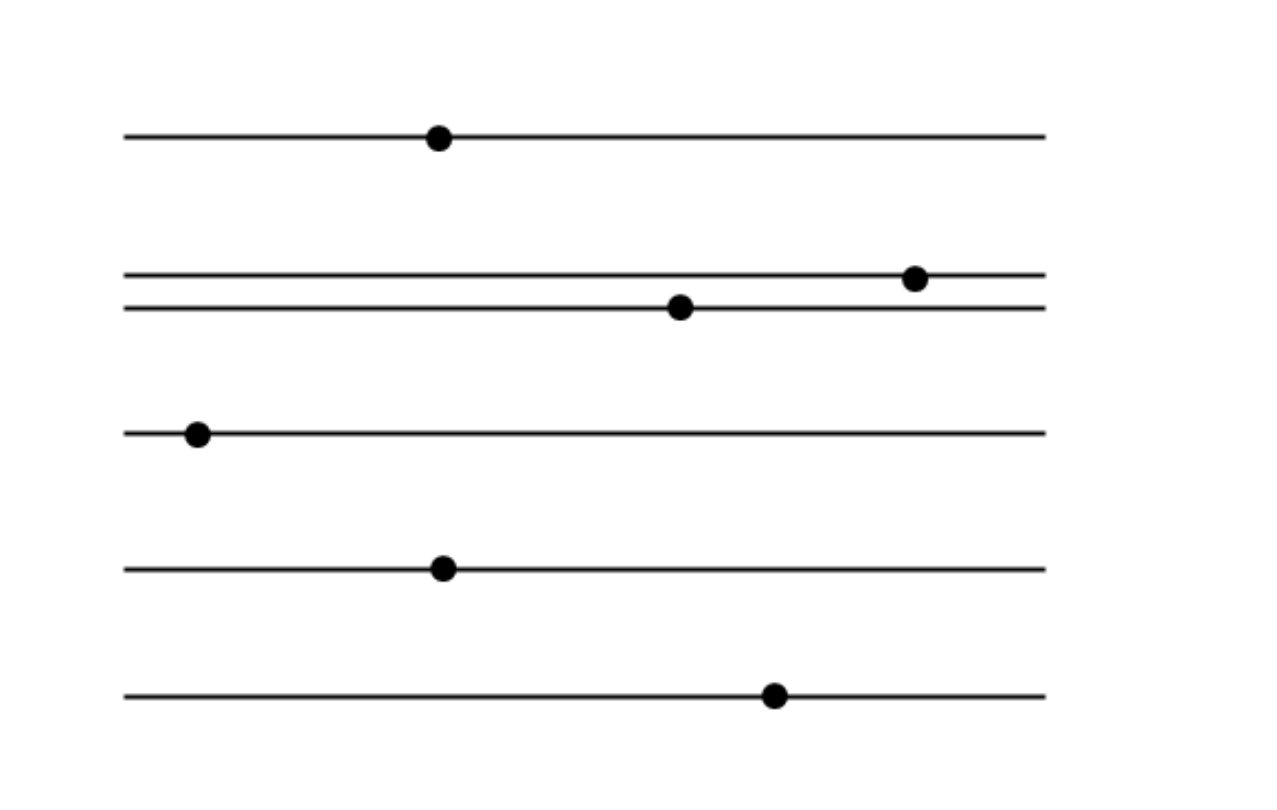
Hint: draw a set of parallel lines, one through every point, and rotate them all about their respective point at the same time.



Problem 3: Solution

Observation 1: only want to consider “adjacent points” in “vertical” ordering around any line segment “base”

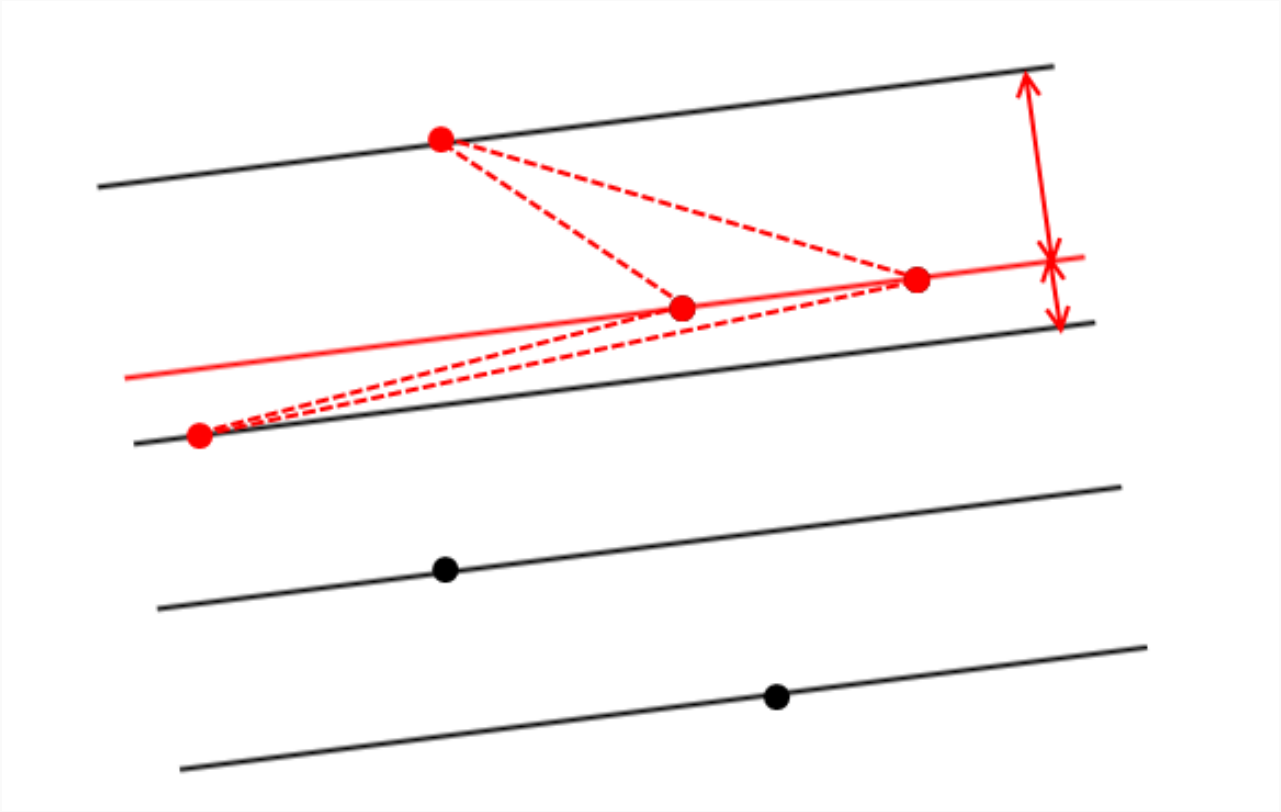
Observation 2: angular sweep simultaneously around all points $\Rightarrow$ ordering flip only when some line segment is flat



Problem 3: Solution

Observation 1: only want to consider “adjacent points” in “vertical” ordering around any line segment “base”

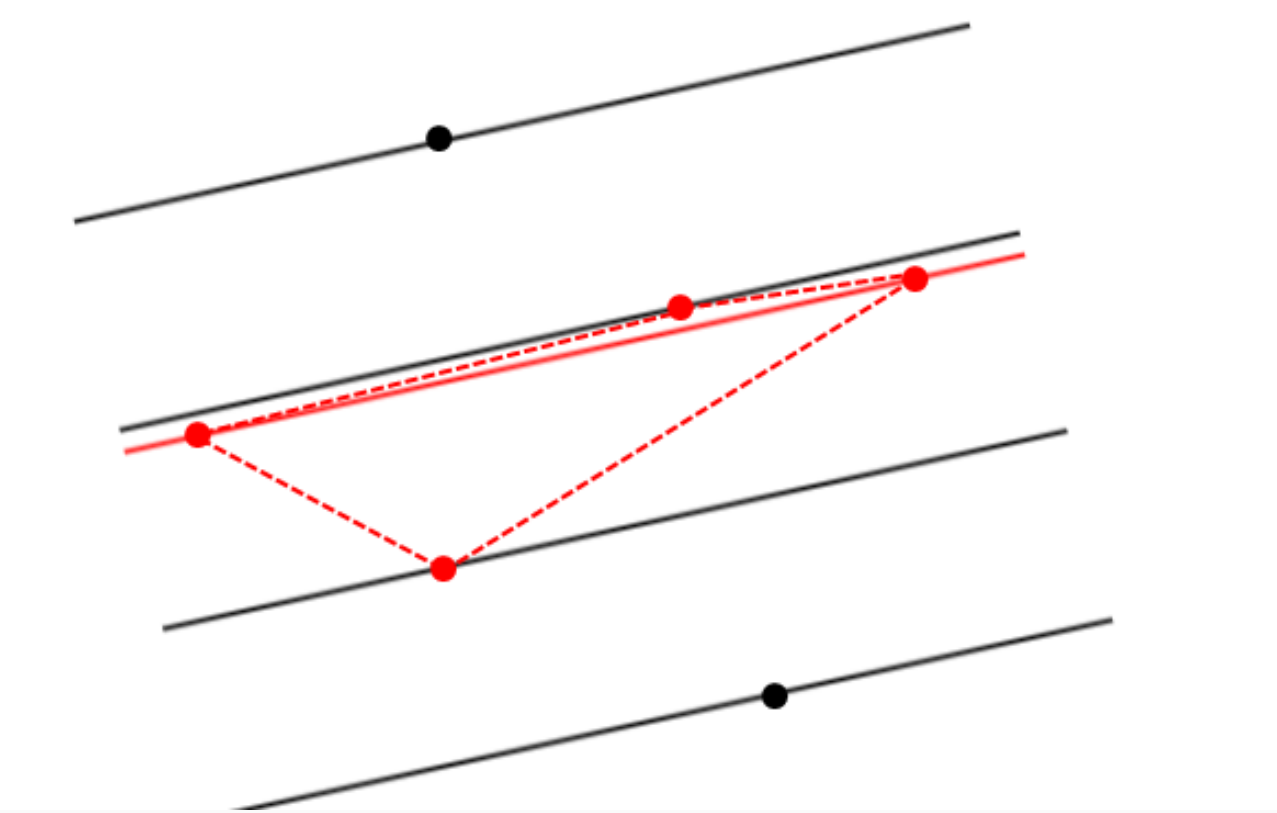
Observation 2: angular sweep simultaneously around all points $\Rightarrow$ ordering flip only when some line segment is flat



Problem 3: Solution

Observation 1: only want to consider “adjacent points” in “vertical” ordering around any line segment “base”

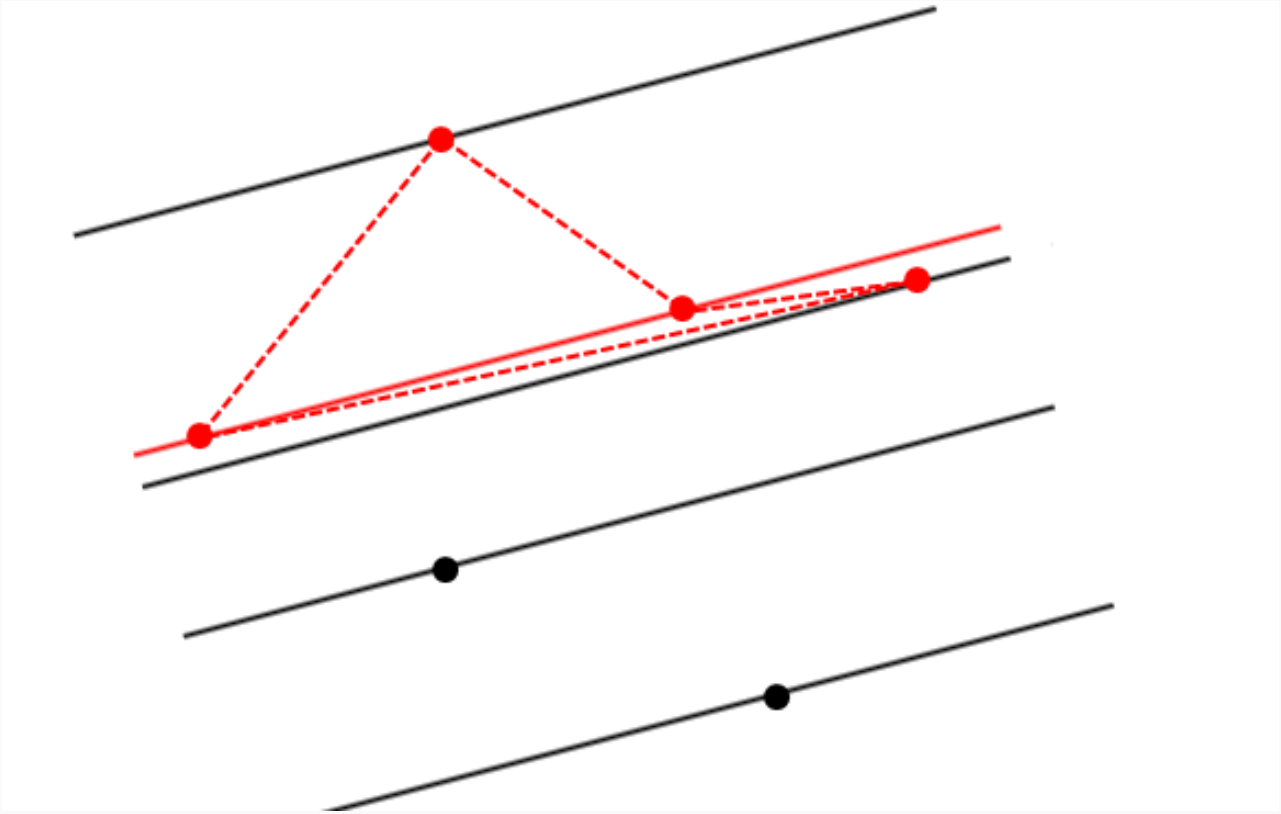
Observation 2: angular sweep simultaneously around all points $\Rightarrow$ ordering flip only when some line segment is flat



Problem 3: Solution

Observation 1: only want to consider “adjacent points” in “vertical” ordering around any line segment “base”

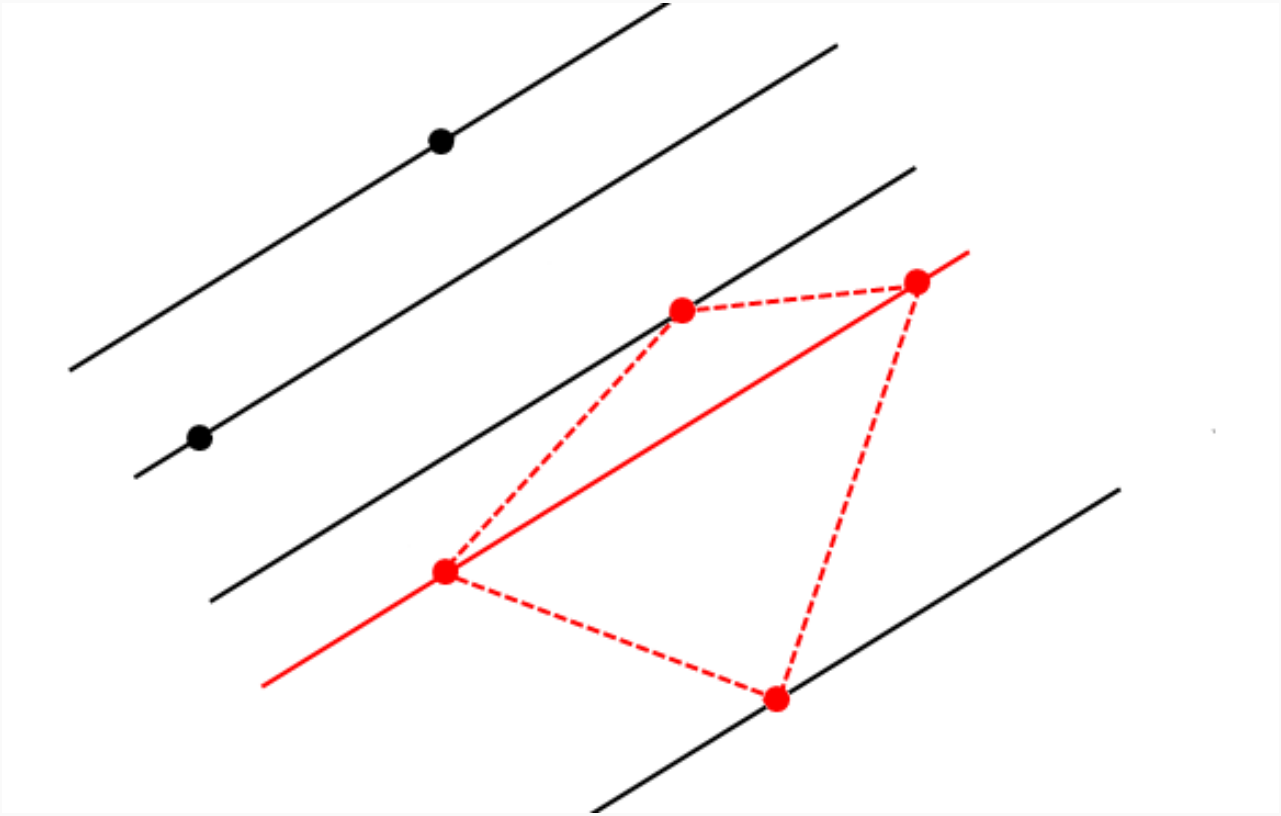
Observation 2: angular sweep simultaneously around all points $\Rightarrow$ ordering flip only when some line segment is flat



Problem 3: Solution

Observation 1: only want to consider “adjacent points” in “vertical” ordering around any line segment “base”

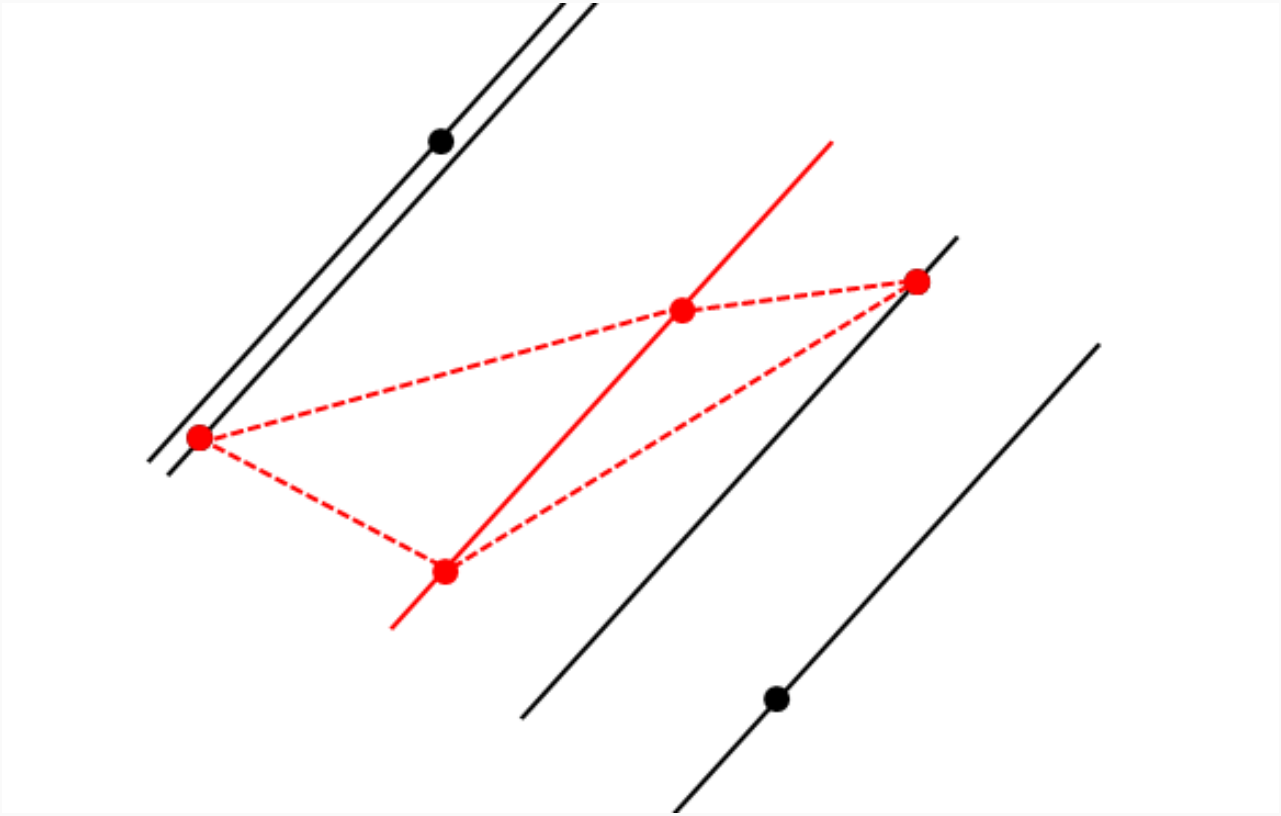
Observation 2: angular sweep simultaneously around all points $\Rightarrow$ ordering flip only when some line segment is flat



Problem 3: Solution

Observation 1: only want to consider “adjacent points” in “vertical” ordering around any line segment “base”

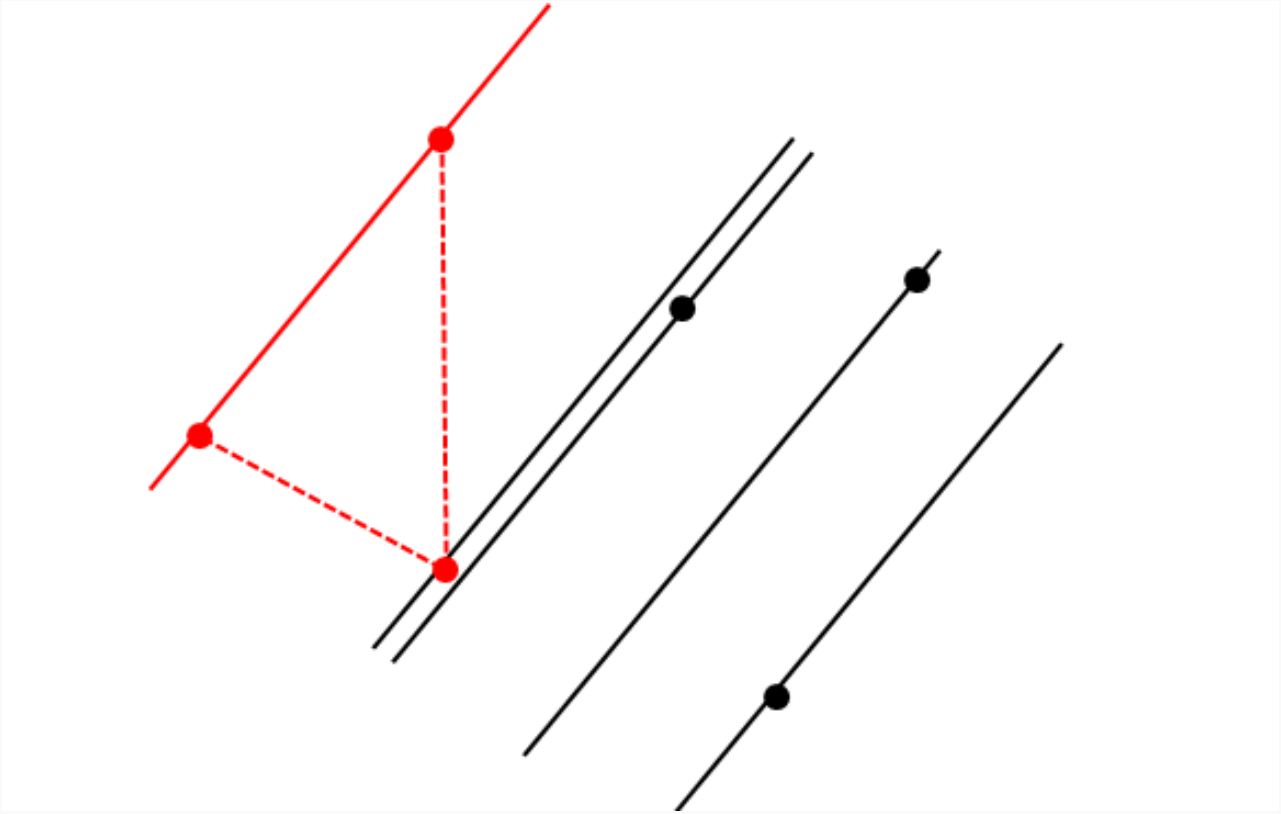
Observation 2: angular sweep simultaneously around all points $\Rightarrow$ ordering flip only when some line segment is flat



Problem 3: Solution

Observation 1: only want to consider “adjacent points” in “vertical” ordering around any line segment “base”

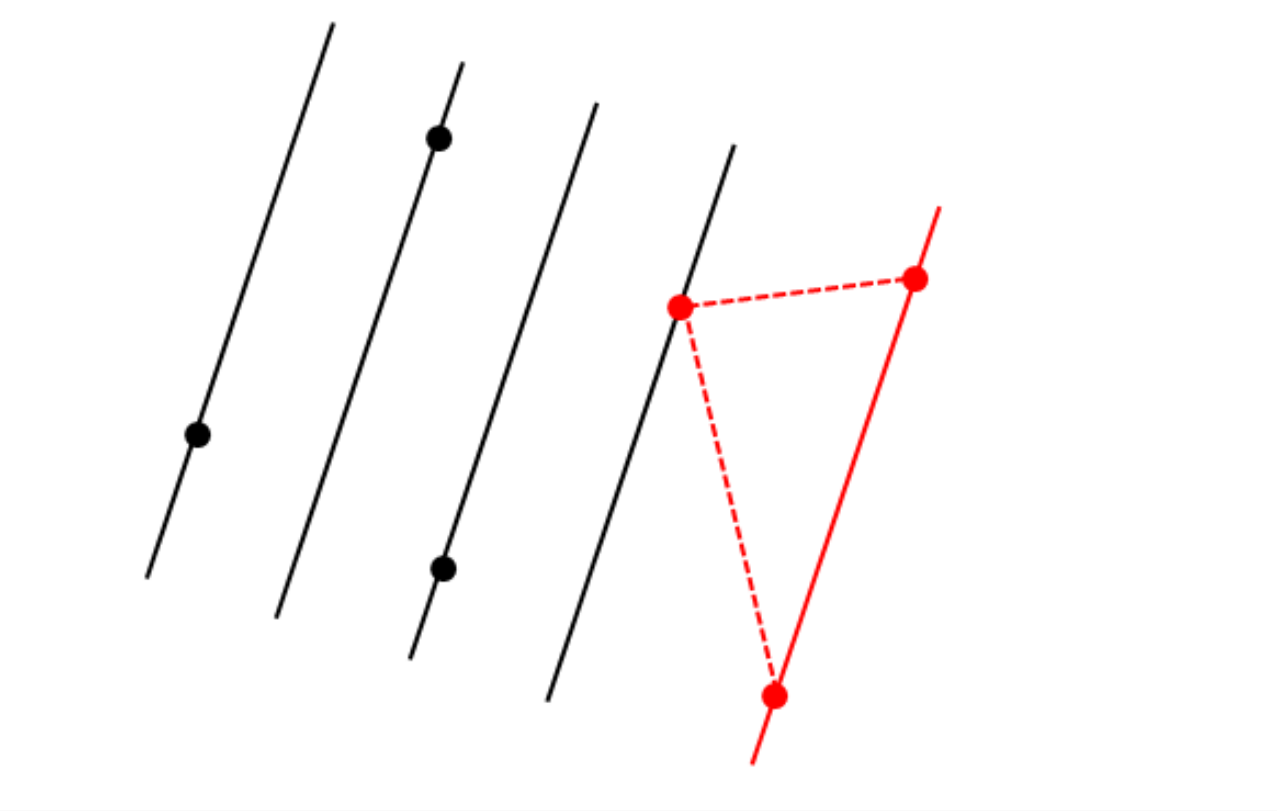
Observation 2: angular sweep simultaneously around all points $\Rightarrow$ ordering flip only when some line segment is flat



Problem 3: Solution

Observation 1: only want to consider “adjacent points” in “vertical” ordering around any line segment “base”

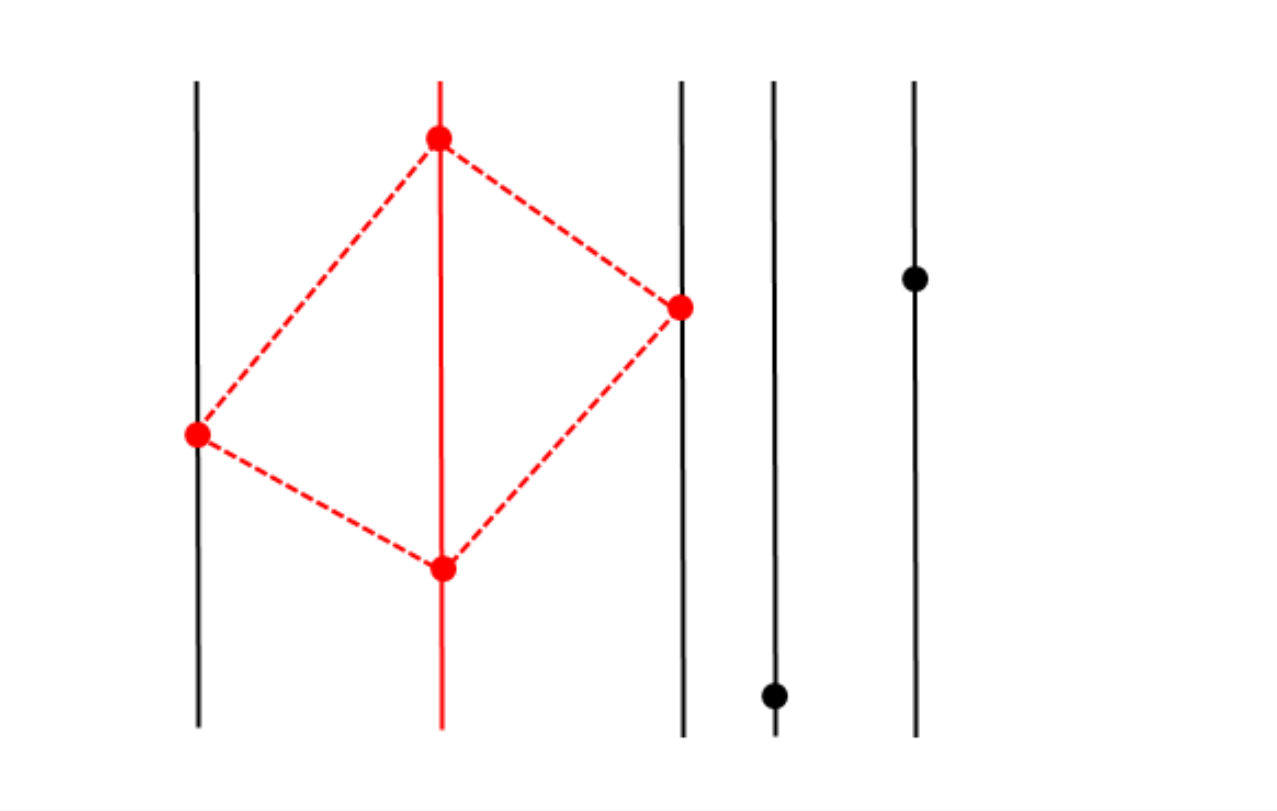
Observation 2: angular sweep simultaneously around all points $\Rightarrow$ ordering flip only when some line segment is flat



Problem 3: Solution

Observation 1: only want to consider “adjacent points” in “vertical” ordering around any line segment “base”

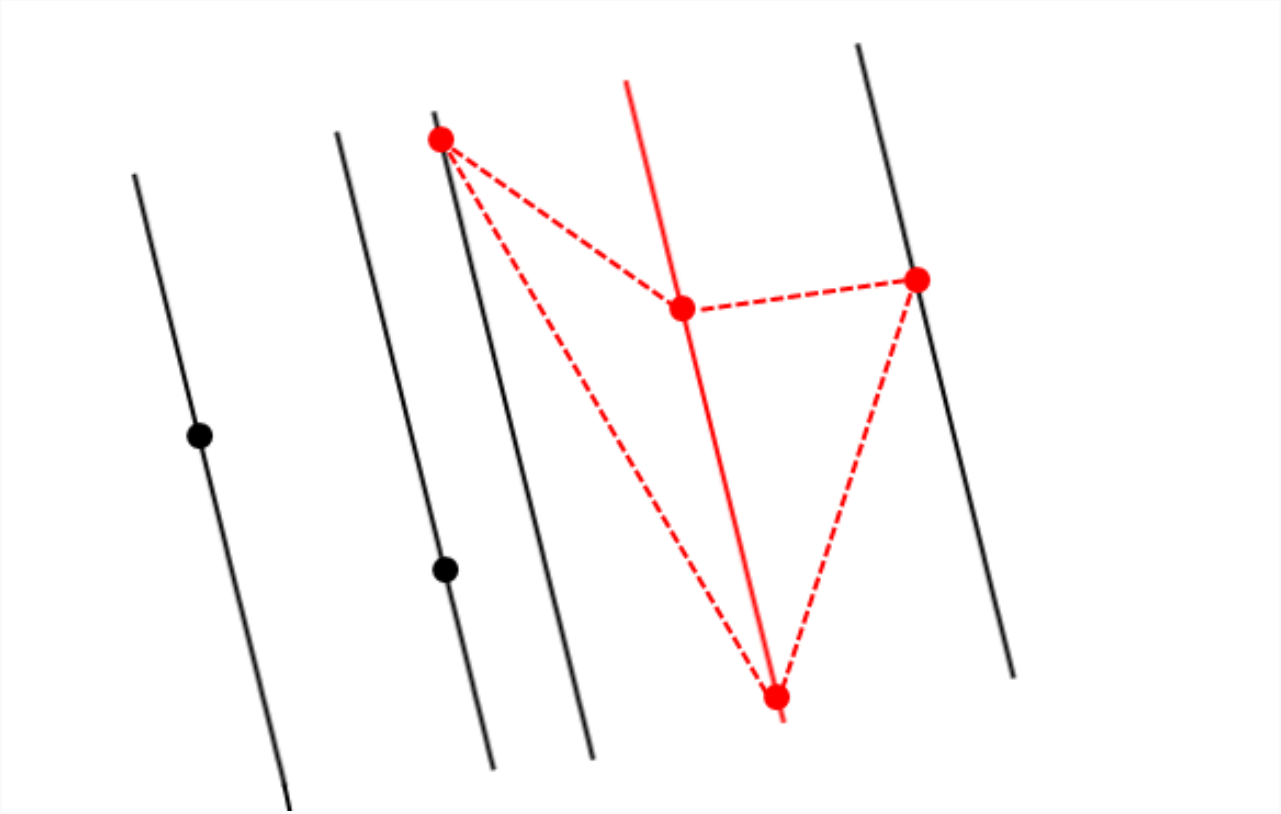
Observation 2: angular sweep simultaneously around all points $\Rightarrow$ ordering flip only when some line segment is flat



Problem 3: Solution

Observation 1: only want to consider “adjacent points” in “vertical” ordering around any line segment “base”

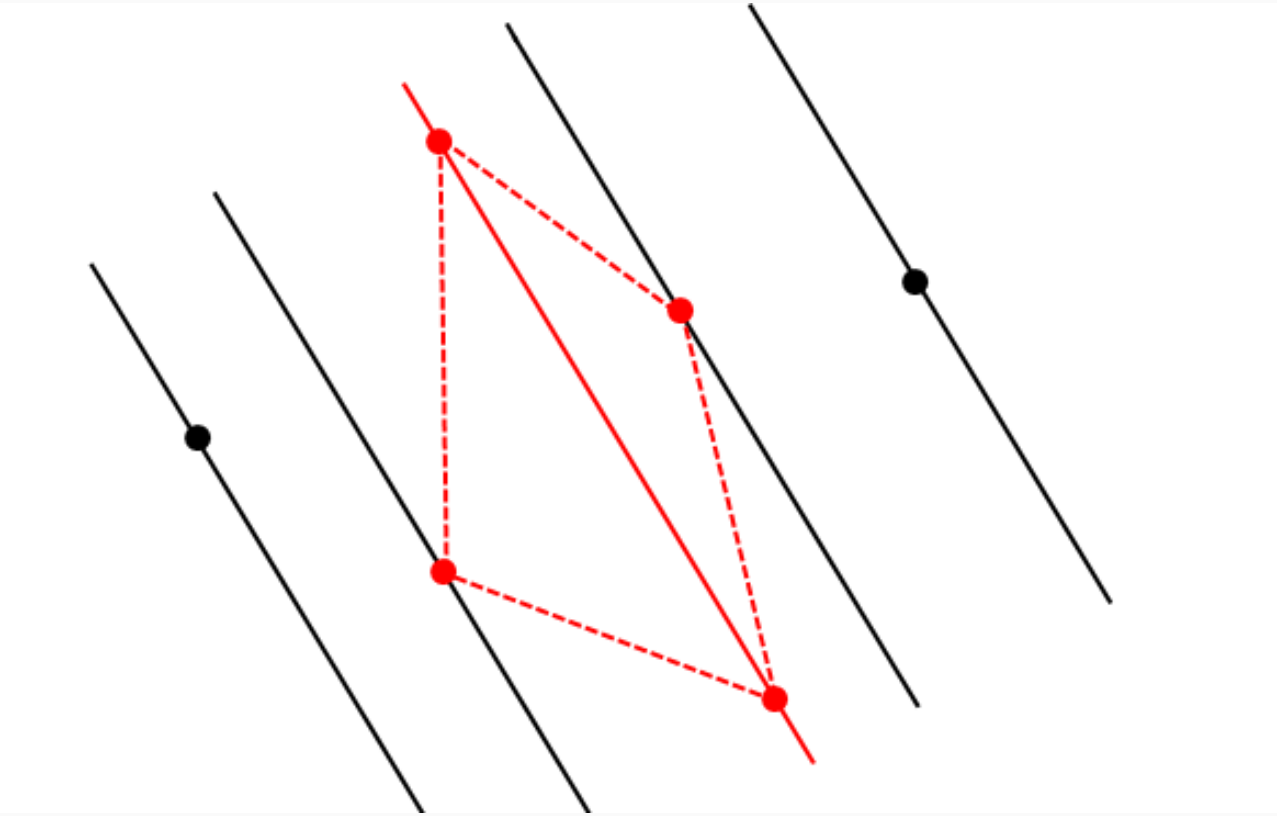
Observation 2: angular sweep simultaneously around all points $\Rightarrow$ ordering flip only when some line segment is flat



Problem 3: Solution

Observation 1: only want to consider “adjacent points” in “vertical” ordering around any line segment “base”

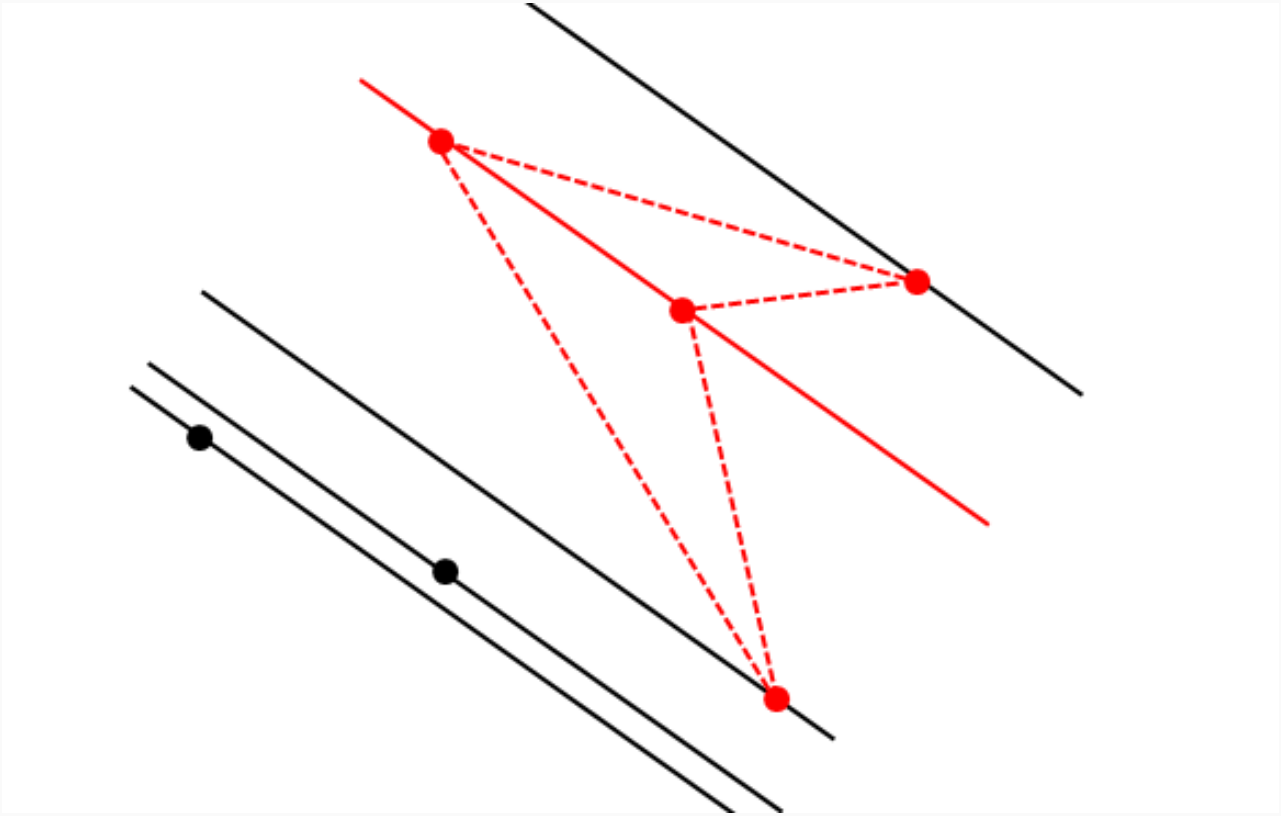
Observation 2: angular sweep simultaneously around all points $\Rightarrow$ ordering flip only when some line segment is flat



Problem 3: Solution

Observation 1: only want to consider “adjacent points” in “vertical” ordering around any line segment “base”

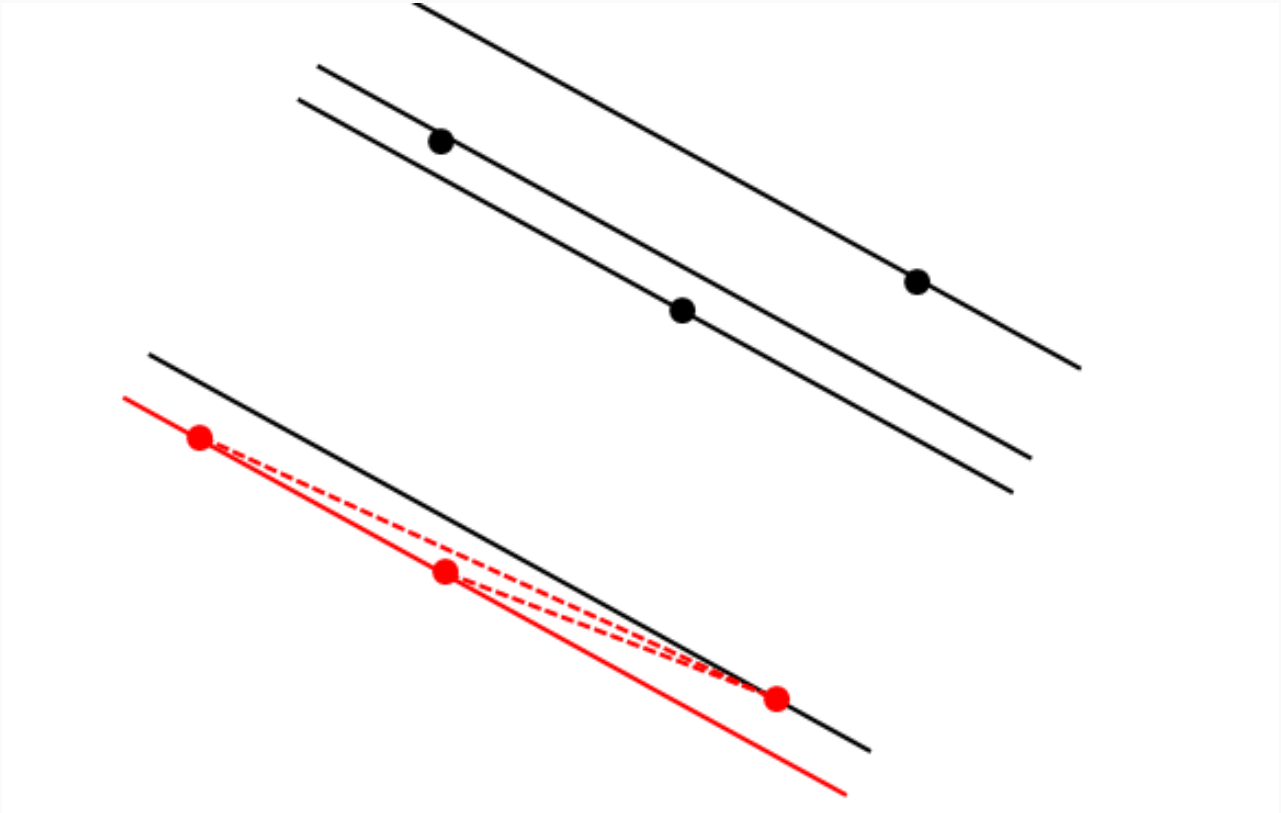
Observation 2: angular sweep simultaneously around all points $\Rightarrow$ ordering flip only when some line segment is flat



Problem 3: Solution

Observation 1: only want to consider “adjacent points” in “vertical” ordering around any line segment “base”

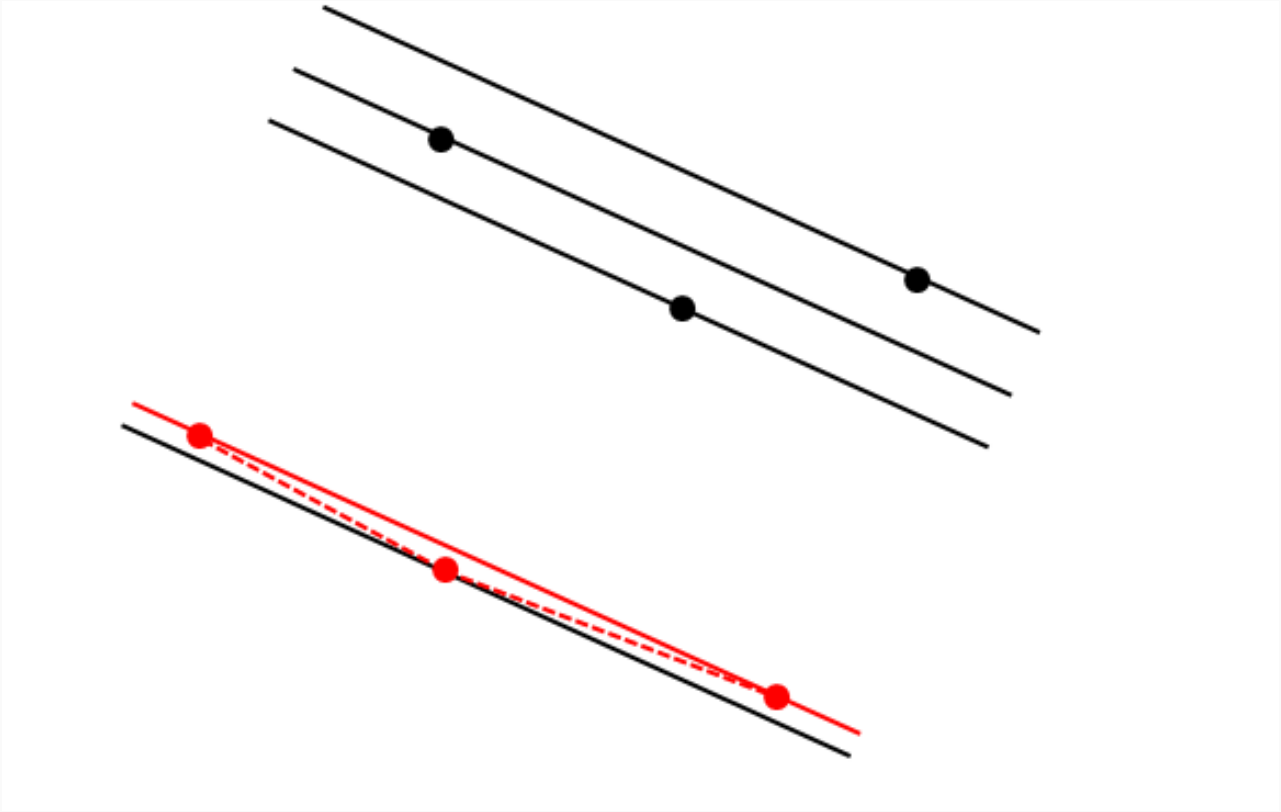
Observation 2: angular sweep simultaneously around all points $\Rightarrow$ ordering flip only when some line segment is flat



Problem 3: Solution

Observation 1: only want to consider “adjacent points” in “vertical” ordering around any line segment “base”

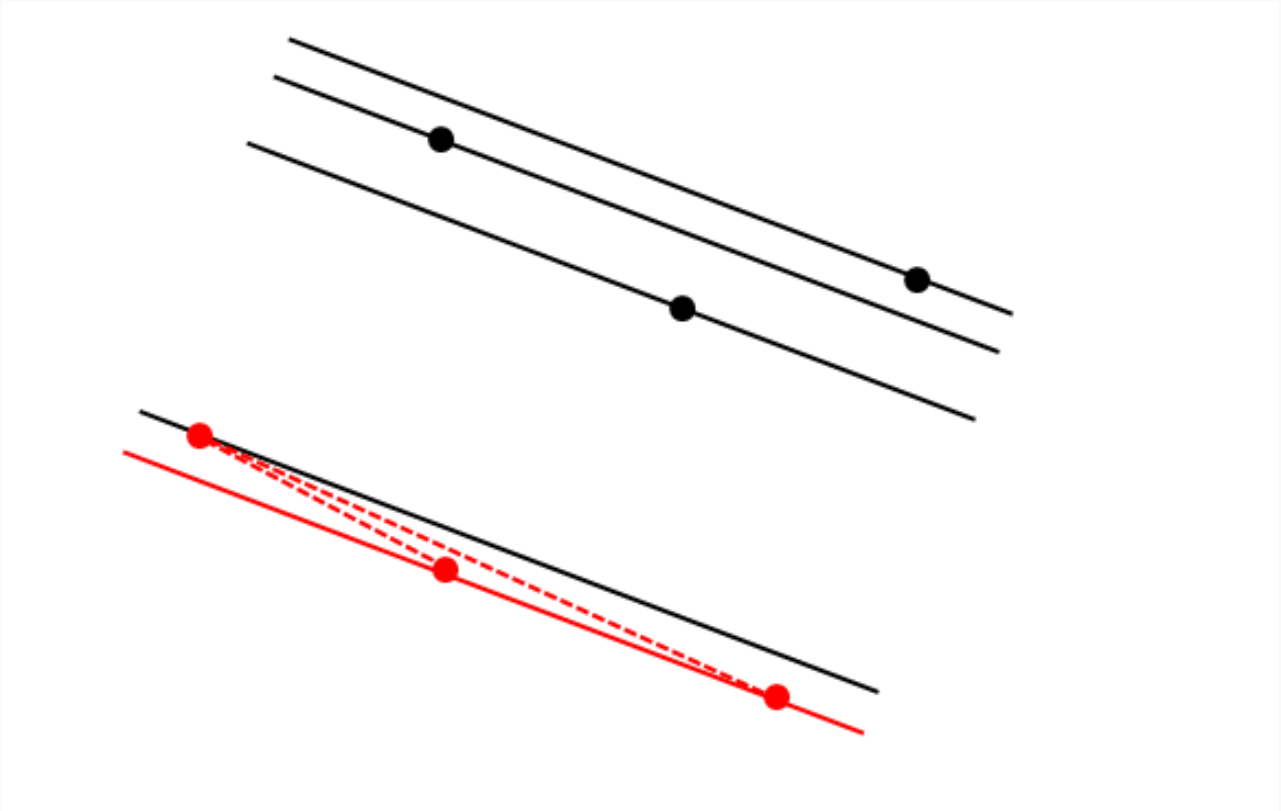
Observation 2: angular sweep simultaneously around all points $\Rightarrow$ ordering flip only when some line segment is flat



Problem 3: Solution

Observation 1: only want to consider “adjacent points” in “vertical” ordering around any line segment “base”

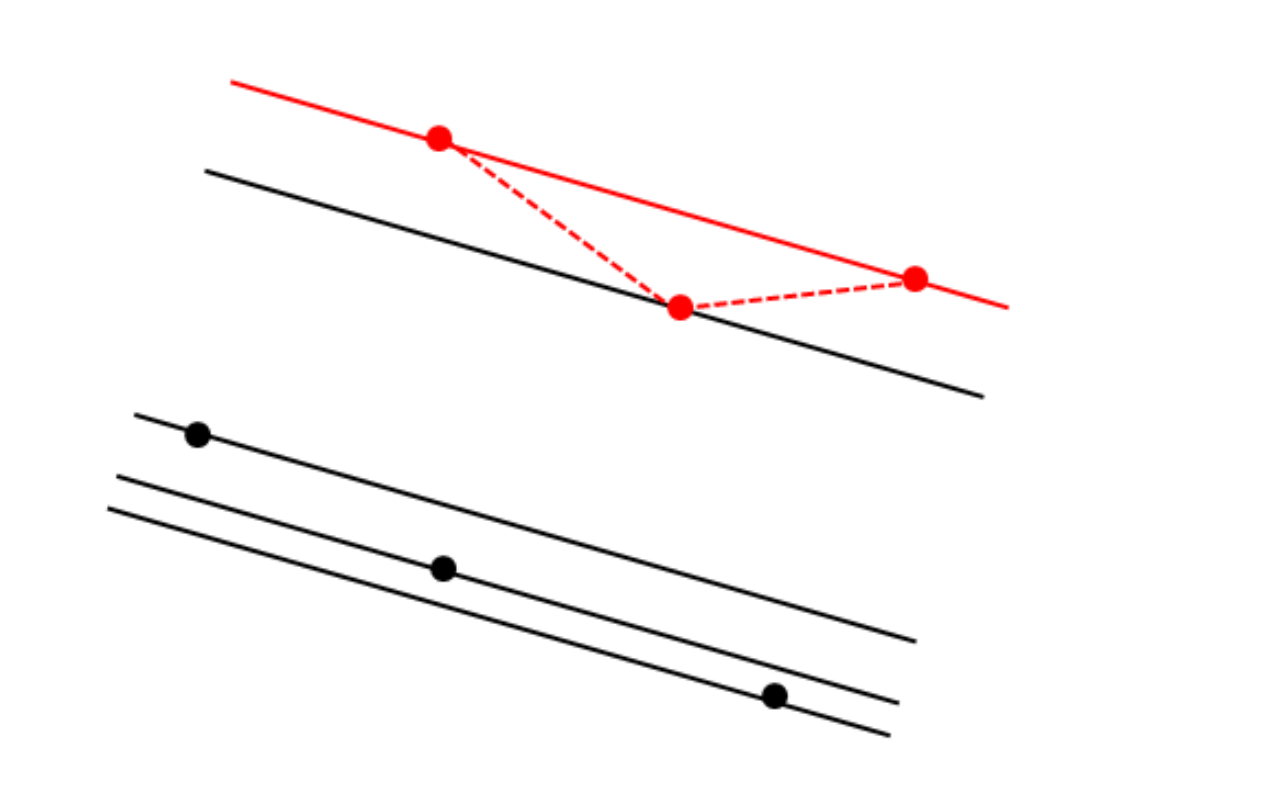
Observation 2: angular sweep simultaneously around all points $\Rightarrow$ ordering flip only when some line segment is flat



Problem 3: Solution

Observation 1: only want to consider “adjacent points” in “vertical” ordering around any line segment “base”

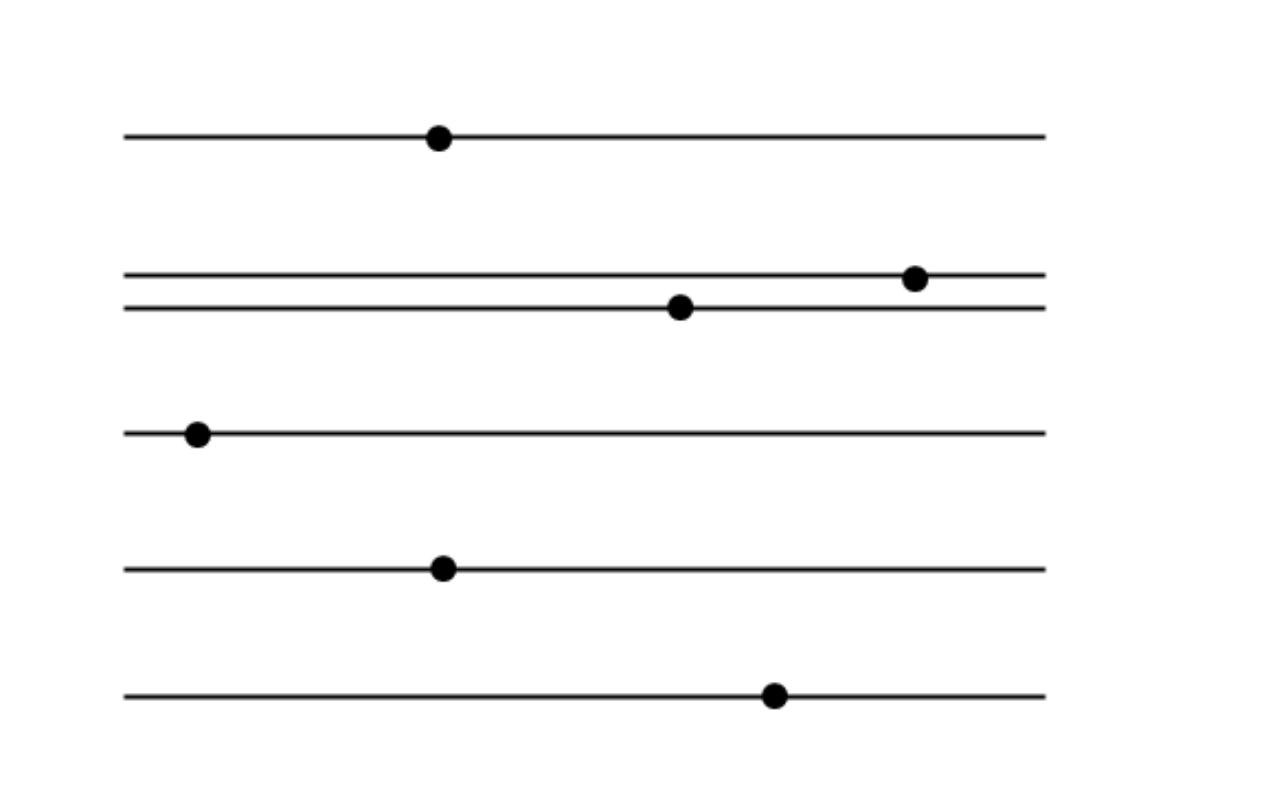
Observation 2: angular sweep simultaneously around all points $\Rightarrow$ ordering flip only when some line segment is flat



Problem 3: Solution

Observation 1: only want to consider “adjacent points” in “vertical” ordering around any line segment “base”

Observation 2: angular sweep simultaneously around all points $\Rightarrow$ ordering flip only when some line segment is flat



Problem 3: Solution

State: ordering of points w.r.t. current set of parallel sweep lines

Events: when slope matches that of some lines segment
(i.e. when two of these sweep lines coincide)

Action:

- Flip the ordering of the two points whose sweep lines coincide
- Compute area of the two triangles formed using two “adjacent” points to the line segment, in the current ordering of points

Time complexity: $O(n^2 \log n)$

Duality

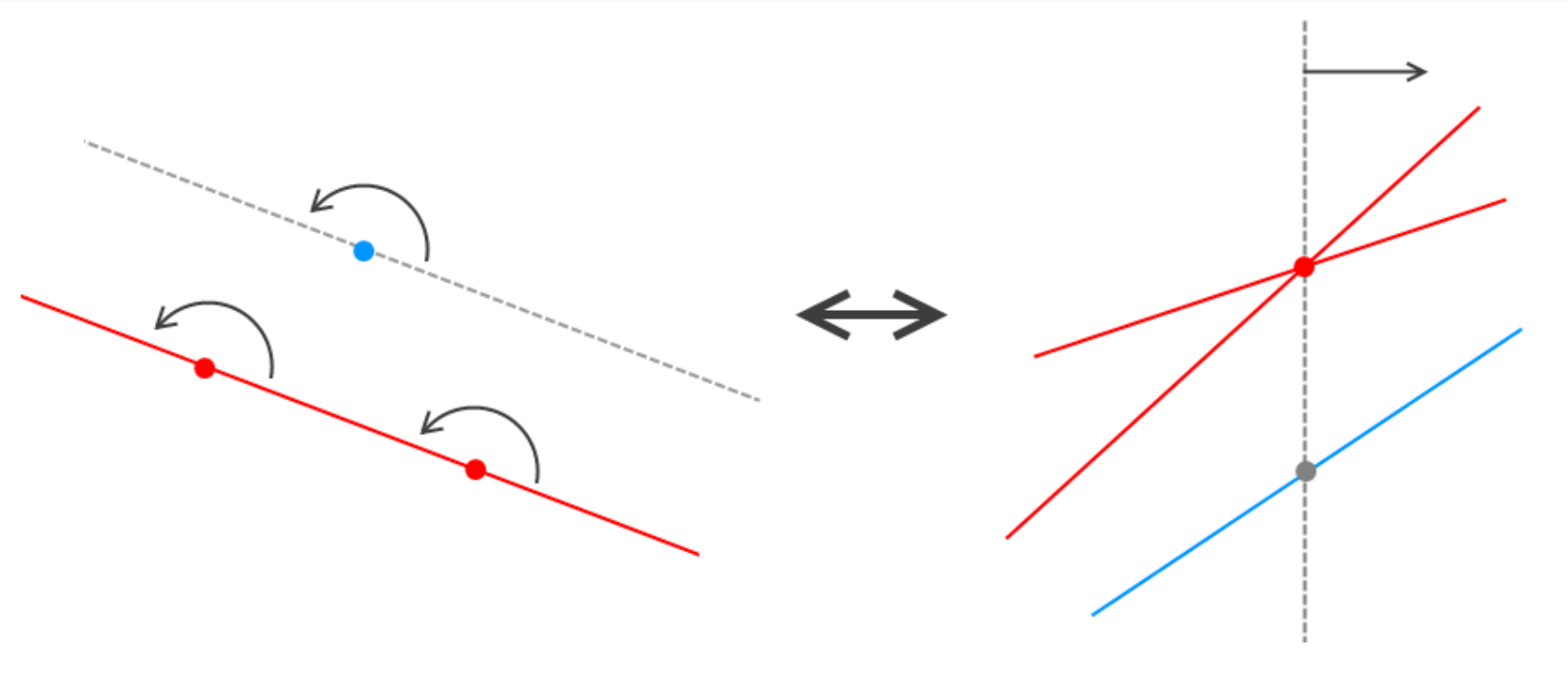
Map $(a, b) \mapsto y = ax - b$ then every point turns into a line, and every line turns into a point. This is the "dual space" representation.

Important properties

- Dual of dual of point is the same point
- Point above line $\Leftrightarrow$ dual of point below dual of line
- 2 lines intersect at point $\Leftrightarrow$ dual of point go thru dual of lines
- 3 points colinear $\Leftrightarrow$ dual of 3 points intersect at same point
- Points CCW on upper hull $\Leftrightarrow$ lines left to right on lower envelope

Smallest Triangle in Dual Space

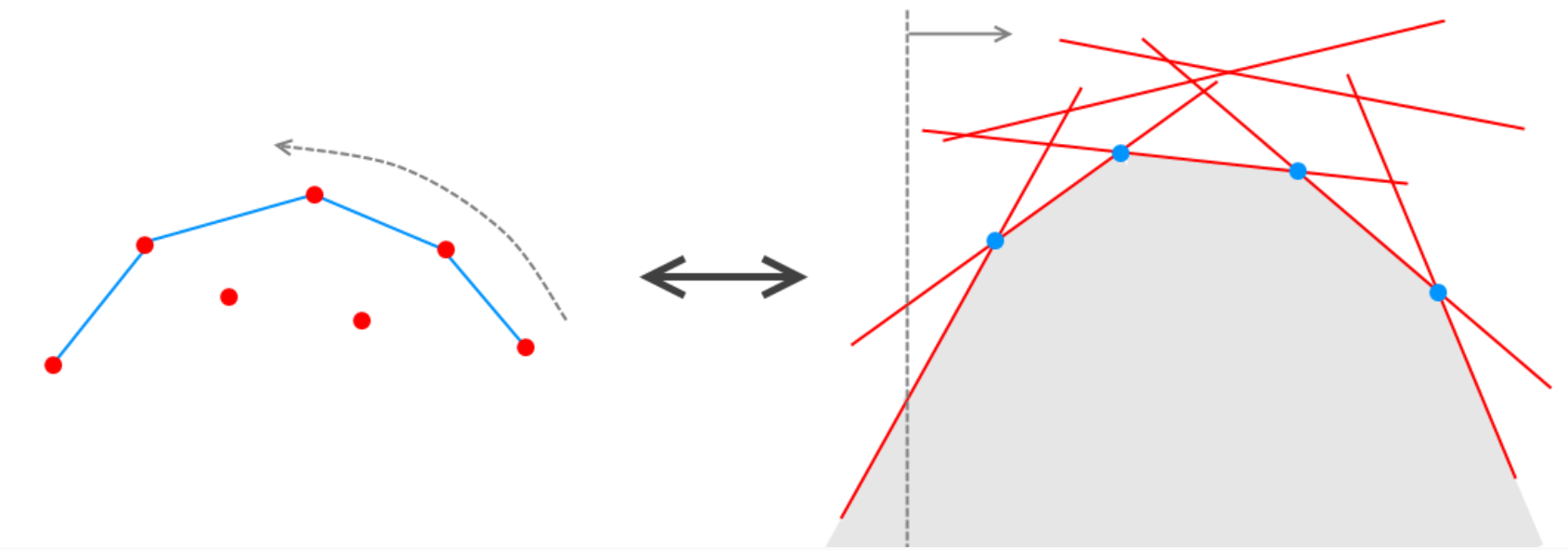
Point above line segment ⇔ Line below intersection of 2 lines.



Angular sweep in primal space ⇔ Line sweep in dual space

Convex Hull in Dual Space

Upper convex hull of points in CCW $\Leftrightarrow$ lower envelope of lines in CW

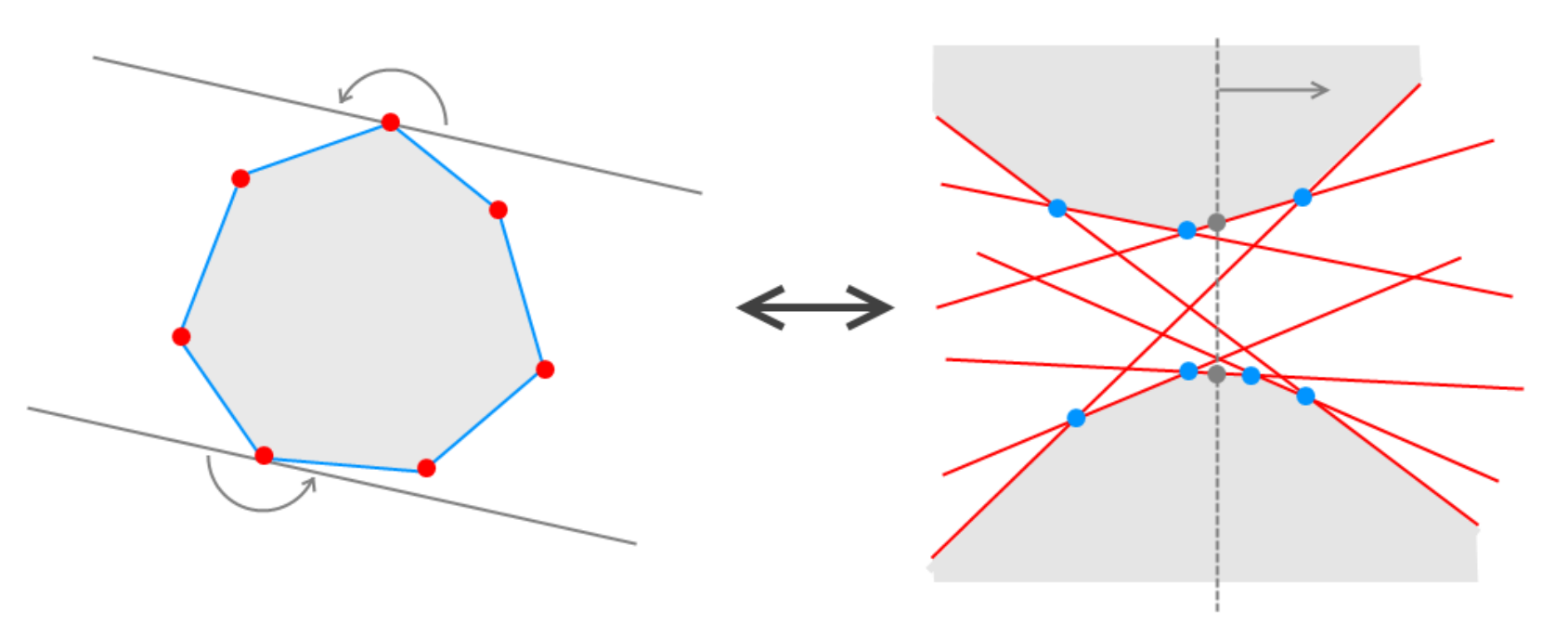


Gift wrapping in primal space $\Leftrightarrow$ Line sweep in dual space, $O(n^2)$

Can you formulate an algorithm similar to Monotone Chain to construct lower envelope in $O(n \log n)$?

Rotating Calipers in Dual Space

Upper convex hull of points in CCW ⇔ lower envelope of lines in CW
Lower convex hull of points in CCW ⇔ upper envelope of lines in CW



Rotating calipers in primal space ⇔ Line sweep in dual space

One sweep to rule them all!

Other interesting algorithms / data structures

You are encouraged to read up on these, but we will not cover them:

- Delaunay triangulation & Voronoi diagram
  - MST on 2D, divide regions by nearest neighbor
  - They are duals of each other!
- Line arrangement & topological line sweep
  - Many applications, including visibility graph in $O(n^2)$
- K-d trees, 2D segment trees
  - 2D range query, nearest neighbor query
- Doubly Connected Edge List (DCEL)
  - Represent a planar graph / 3D polytope / line arrangement
- Dynamic convex hull / lower envelope

Next Class

Divide and Conquer
(with some applications to geometry)