A Polynomial Time Algorithm for Lossy Population Recovery

Ankur Moitra
Massachusetts Institute of Technology

joint work with Mike Saks

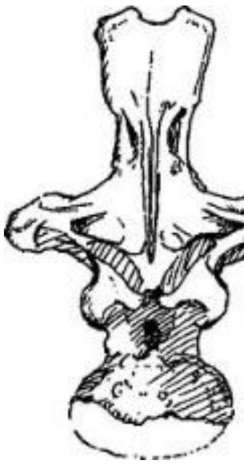
A Story…

A Story…

A Story…

A Story…

A Story…



3

Can you reconstruct a description of the population from these **fragments**?

Can you reconstruct a description of the population from these **fragments**?

features (n)

species (k)

Can you reconstruct a description of the population from these **fragments**?

features (n)

species (k)

1	0	1	1	0	1	0	1	0	0
0	1	1	1	0	0	1	1	0	0
	⋮				⋮			⋮	
1	1	1	1	0	0	0	1	0	0

Can you reconstruct a description of the population from these **fragments**?

features (n)

species (k)

p_1	1	0	1	1	0	1	0	1	0	0
p_2	0	1	1	1	0	0	1	1	0	0
$\vdots$		$\vdots$				$\vdots$			$\vdots$	
p_k	1	1	1	1	0	0	0	1	0	0

Can you reconstruct a description of the population from these **fragments**?

features (n)

species (k)

p_1	1	0	1	1	0	1	0	1	0	0
p_2	0	1	1	1	0	0	1	1	0	0
⋮		⋮				⋮			⋮	
p_k	1	1	1	1	0	0	0	1	0	0

Can you reconstruct a description of the population from these **fragments**?

features (n)

species (k)

p_1	1	0	1	1	0	1	0	1	0	0
p_2	0	1	1	1	0	0	1	1	0	0
⋮		⋮				⋮			⋮	
p_k	1	1	1	1	0	0	0	1	0	0

samples:	?	1	?	?	?	0	?	?	?	?

The Model (Dvir, Rao, Wigderson, Yehudayoff)

The Model (Dvir, Rao, Wigderson, Yehudayoff)

Lossy Population Recovery:

- **Unknown** set of k strings, a_1, a_2, …, a_k and probabilities p_1, p_2, …, p_k

The Model (Dvir, Rao, Wigderson, Yehudayoff)

Lossy Population Recovery:

- **Unknown** set of k strings, a_1, a_2, …, a_k and probabilities p_1, p_2, …, p_k

- Given samples (chosen according to p_i), but every bit is deleted independently with probability 1-μ and replaced with a '**?**'

The Model (Dvir, Rao, Wigderson, Yehudayoff)

Lossy Population Recovery:

- **Unknown** set of k strings, a_1, a_2, …, a_k and probabilities p_1, p_2, …, p_k

- Given samples (chosen according to p_i), but every bit is deleted independently with probability 1-μ and replaced with a '**?**'

Is there a polynomial time algorithm for any fixed μ>0?

Another Model (Wigderson, Yehudayoff)

Noisy Population Recovery:

- **Unknown** set of k strings, a_1, a_2, …, a_k and probabilities p_1, p_2, …, p_k

Another Model (Wigderson, Yehudayoff)

Noisy Population Recovery:

- **Unknown** set of k strings, a_1, a_2, …, a_k and probabilities p_1, p_2, …, p_k
- Given samples (chosen according to p_i), but every bit is flipped independently with probability $1/2-\eta$

Another Model (Wigderson, Yehudayoff)

Noisy Population Recovery:

- **Unknown** set of k strings, $a_1, a_2, \ldots, a_k$ and probabilities $p_1, p_2, \ldots, p_k$

- Given samples (chosen according to p_i), but every bit is flipped independently with probability $1/2-\eta$

Is there a polynomial time algorithm for any fixed $\eta>0$?

Theorem [Dvir et al ITCS 2012]: There is a polynomial time algorithm for any $\mu > 0.36$ **(lossy)**

Theorem [Dvir et al ITCS 2012]: There is a polynomial time algorithm for any $\mu > 0.36$ **(lossy)**

Theorem [Wigderson, Yehudayoff FOCS 2012]: There is a quasi-polynomial time algorithm for any μ, $\eta > 0$ **(lossy, noisy)**

Theorem [Dvir et al ITCS 2012]: There is a polynomial time algorithm for any $\mu > 0.36$ **(lossy)**

Theorem [Wigderson, Yehudayoff FOCS 2012]: There is a quasi-polynomial time algorithm for any μ, $\eta > 0$ **(lossy, noisy)**

However, their framework provably cannot yield a polynomial time algorithm!

Theorem [Dvir et al ITCS 2012]: There is a polynomial time algorithm for any $\mu > 0.36$ **(lossy)**

Theorem [Wigderson, Yehudayoff FOCS 2012]: There is a quasi-polynomial time algorithm for any μ, $\eta > 0$ **(lossy, noisy)**

However, their framework provably cannot yield a polynomial time algorithm!

Theorem [Batman et al RANDOM 2013]: There is a polynomial time algorithm for any $\mu > 0.30$ **(lossy)**

Theorem [Dvir et al ITCS 2012]: There is a polynomial time algorithm for any $\mu > 0.36$ **(lossy)**

Theorem [Wigderson, Yehudayoff FOCS 2012]: There is a quasi-polynomial time algorithm for any μ, $\eta > 0$ **(lossy, noisy)**

However, their framework provably cannot yield a polynomial time algorithm!

Theorem [Batman et al RANDOM 2013]: There is a polynomial time algorithm for any $\mu > 0.30$ **(lossy)**

Theorem [Moitra, Saks FOCS 2013]: There is a polynomial time algorithm for any $\mu > 0$ **(lossy)**

An Application

An Application

DNF: $(x_1 \wedge x_3 \wedge \bar{x}_5) \vee (\bar{x}_2 \wedge \bar{x}_3 \wedge x_8) \ldots$

An Application

DNF: $(x_1 \wedge x_3 \wedge \bar{x}_5) \vee (\bar{x}_2 \wedge \bar{x}_3 \wedge x_8) \ldots$

PAC Model: distribution **D** on examples, given the example and its evaluation

An Application

DNF: $(x_1 \wedge x_3 \wedge \bar{x}_5) \vee (\bar{x}_2 \wedge \bar{x}_3 \wedge x_8) \ldots$

PAC Model: distribution **D** on examples, given the example and its evaluation

---

Theorem [Klivans, Servedio STOC 2001]: There is a $2^{O(n^{1/3})}$ time algorithm to PAC learn DNFs

An Application

DNF: $(x_1 \wedge x_3 \wedge \bar{x}_5) \vee (\bar{x}_2 \wedge \bar{x}_3 \wedge x_8) \ldots$

PAC Model: distribution **D** on examples, given the example and its evaluation

---

Theorem [Klivans, Servedio STOC 2001]: There is a $2^{O(n^{1/3})}$ time algorithm to PAC learn DNFs

Theorem [folk]: There is a quasi-polynomial time algorithm to PAC learn DNFs under the **uniform distribution**

An Application

DNF: $(x_1 \wedge x_3 \wedge \bar{x}_5) \vee (\bar{x}_2 \wedge \bar{x}_3 \wedge x_8) \ldots$

PAC Model: distribution **D** on examples, given the example and its evaluation

---

This is **black-box** access to the formula

An Application

DNF: $(x_1 \wedge x_3 \wedge \bar{x}_5) \vee (\bar{x}_2 \wedge \bar{x}_3 \wedge x_8) \ldots$

PAC Model: distribution **D** on examples, given the example and its evaluation

---

This is **black-box** access to the formula

Is there a natural **grey-box** model? Can we design better algorithms?

Restriction Access (Dvir et al)

DNF: $(x_1 \wedge x_3 \wedge \bar{x}_5) \vee (\bar{x}_2 \wedge \bar{x}_3 \wedge x_8) \ldots$

Restriction Access (Dvir et al)

DNF: $(x_1 \wedge x_3 \wedge \bar{x}_5) \vee (\bar{x}_2 \wedge \bar{x}_3 \wedge x_8) \ldots$

New Model: Set each bit independently with prob 1-μ, given the restricted formula

Restriction Access (Dvir et al)

DNF: $(x_1 \wedge x_3 \wedge \bar{x}_5) \vee (\bar{x}_2 \wedge \bar{x}_3 \wedge x_8) \ldots$

New Model: Set each bit independently with prob 1-μ, given the restricted formula

Each clause that survives, we get a fragment of its variables

Restriction Access (Dvir et al)

DNF: $(x_1 \wedge x_3 \wedge \bar{x}_5) \vee (\bar{x}_2 \wedge \bar{x}_3 \wedge x_8)\ldots$

New Model: Set each bit independently with prob 1-μ, given the restricted formula

Each clause that survives, we get a fragment of its variables

Population Recovery → Learning DNFs in Restriction Access

Restriction Access (Dvir et al)

DNF: $(x_1 \wedge x_3 \wedge \bar{x}_5) \vee (\bar{x}_2 \wedge \bar{x}_3 \wedge x_8) \ldots$

New Model: Set each bit independently with prob 1-μ, given the restricted formula

Each clause that survives, we get a fragment of its variables

Restriction Access (Dvir et al)

DNF: $(x_1 \wedge x_3 \wedge \bar{x}_5) \vee (\bar{x}_2 \wedge \bar{x}_3 \wedge x_8)\ldots$

New Model: Set each bit independently with prob 1-μ, given the restricted formula

Each clause that survives, we get a fragment of its variables

Corollary: There is a polynomial time algorithm for learning DNFs in the Restriction Access Model for any $\mu > 0$.

What is this talk about?

What is this talk about?

Inverse Problems:

What is this talk about?

Inverse Problems:

Complex Analysis:

What is this talk about?

Inverse Problems:

Given $\mathbf{Ax} \approx \mathbf{b}$, can we do better than $\mathbf{x} \approx \mathbf{A}^{-1}\mathbf{b}$?

Complex Analysis:

What is this talk about?

Inverse Problems:

Given $\mathbf{Ax} \approx \mathbf{b}$, can we do better than $\mathbf{x} \approx \mathbf{A}^{-1}\mathbf{b}$?

Even though the condition number of **A** is **exponentially** large, we will find ways around it…

Complex Analysis:

What is this talk about?

Inverse Problems:

Given $\mathbf{Ax} \approx \mathbf{b}$, can we do better than $\mathbf{x} \approx \mathbf{A}^{-1}\mathbf{b}$?

Even though the condition number of **A** is **exponentially** large, we will find ways around it…

Complex Analysis:

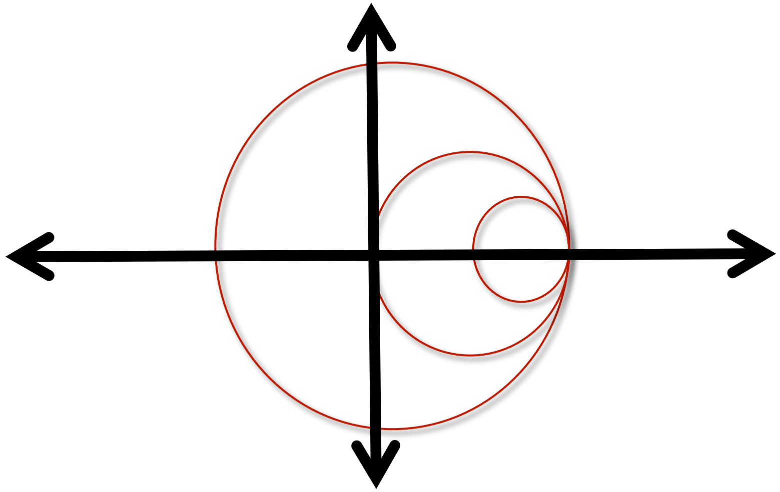
uncertainty principles…

What is this talk about?

Inverse Problems:

Given $\mathbf{Ax} \approx \mathbf{b}$, can we do better than $\mathbf{x} \approx \mathbf{A}^{-1}\mathbf{b}$?

Even though the condition number of **A** is **exponentially** large, we will find ways around it…



Complex Analysis:

uncertainty principles…

Two Reductions (Dvir et al)

Claim: We can assume we know the strings a_1, a_2, …, a_k (all we need is to find p_1, p_2, …, p_k)

Two Reductions (Dvir et al)

Suppose we had an algorithm for population recovery when the strings $a_1, a_2, \ldots, a_k$ are known:

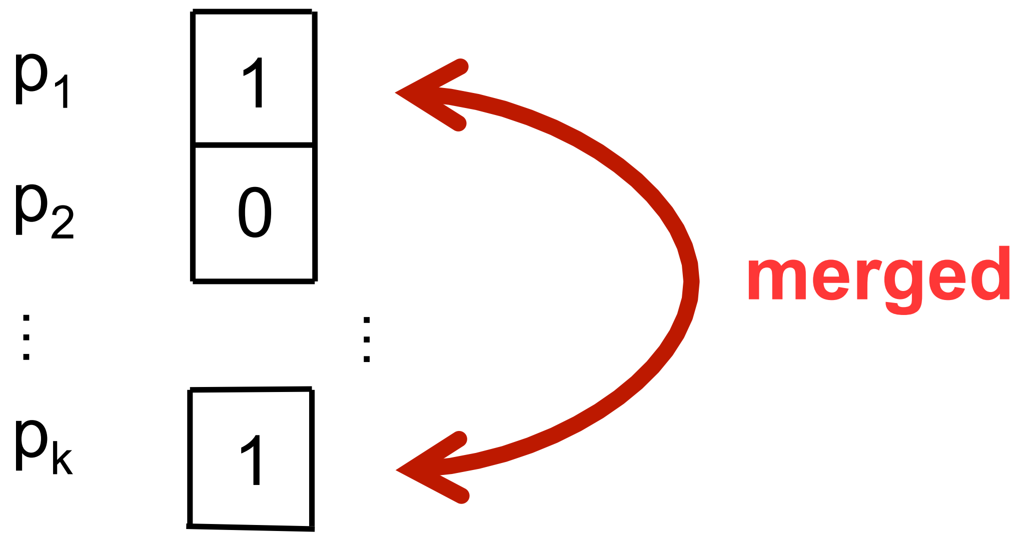
Two Reductions (Dvir et al)

Suppose we had an algorithm for population recovery when the strings $a_1, a_2, \ldots, a_k$ are known:

p_1	$\boxed{1}$
p_2	$\boxed{0}$
$\vdots$	$\vdots$
p_k	$\boxed{1}$

Two Reductions (Dvir et al)

Suppose we had an algorithm for population recovery when the strings $a_1, a_2, \ldots, a_k$ are known:



Two Reductions (Dvir et al)

Suppose we had an algorithm for population recovery when the strings a_1, a_2, …, a_k are known:

p'_1	1
p'_2	0

Two Reductions (Dvir et al)

Suppose we had an algorithm for population recovery when the strings $a_1, a_2, \ldots, a_k$ are known:

p'_1	1	0 1
p'_2	0	0 1

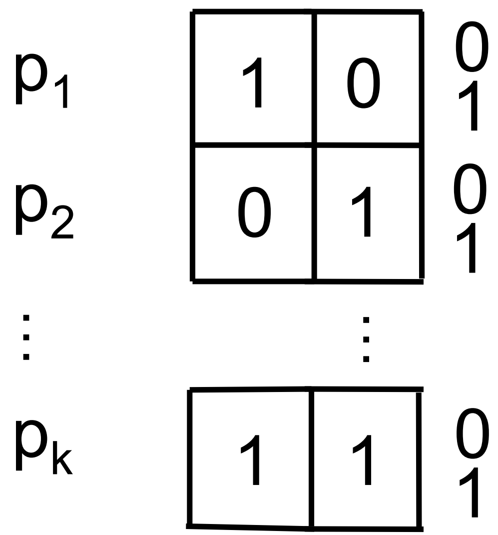
Two Reductions (Dvir et al)

Suppose we had an algorithm for population recovery when the strings a_1, a_2, …, a_k are known:

p_1	1	0
p_2	0	1
$\vdots$		$\vdots$
p_k	1	1

Two Reductions (Dvir et al)

Suppose we had an algorithm for population recovery when the strings a_1, a_2, …, a_k are known:



and so on…

Two Reductions (Dvir et al)

p_1	1	0	1	1	0	1	0	1	0	0
p_2	0	1	1	1	0	0	1	1	0	0
$\vdots$		$\vdots$				$\vdots$			$\vdots$	
p_k	1	1	1	1	0	0	0	1	0	0

Two Reductions (Dvir et al)

Claim: We just need to learn p_i for the all zero string

p_1	1	0	1	1	0	1	0	1	0	0
p_2	0	1	1	1	0	0	1	1	0	0
⋮		⋮			⋮				⋮	
p_k	1	1	1	1	0	0	0	1	0	0

Two Reductions (Dvir et al)

Claim: We just need to learn p_i for the all zero string

p_1	1	0	1	1	0	1	0	1	0	0
p_2	0	1	1	1	0	0	1	1	0	0
⋮		⋮			⋮				⋮	
p_k	1	1	1	1	0	0	0	1	0	0

E.g. we can **XOR** with a_1

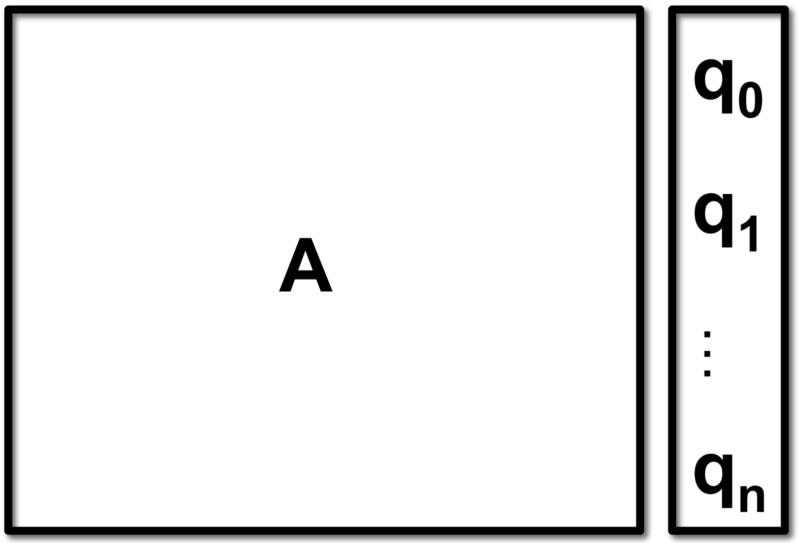
Two Reductions (Dvir et al)

Claim: We just need to learn p_i for the all zero string

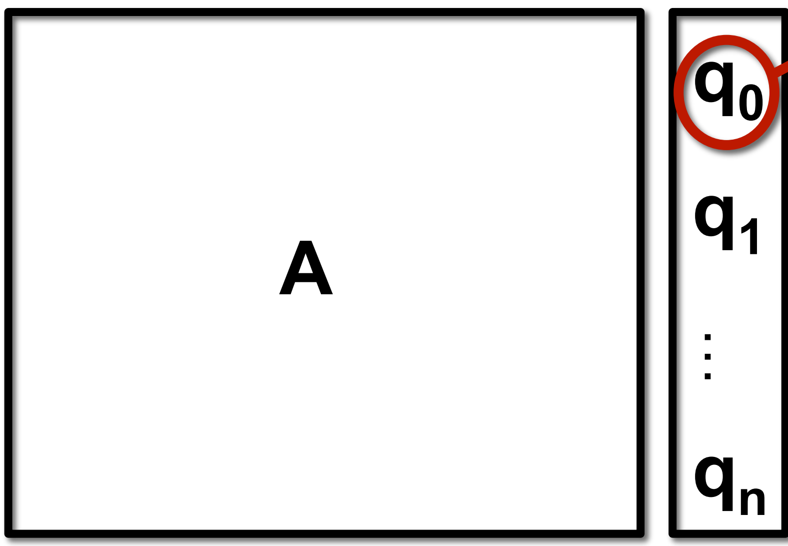
p_1	0	0	0	0	0	0	0	0	0	0
p_2	1	1	0	0	0	1	1	0	0	0
$\vdots$		$\vdots$				$\vdots$			$\vdots$	
p_k	0	1	0	0	0	1	0	0	0	0

E.g. we can **XOR** with a_1

The Setup

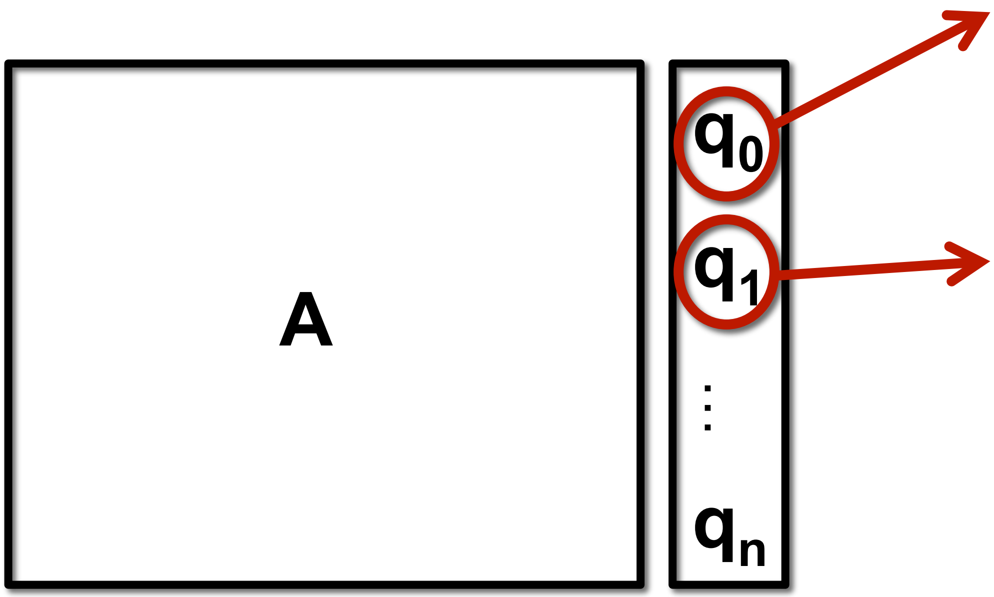


The Setup



probability of all zero string

The Setup



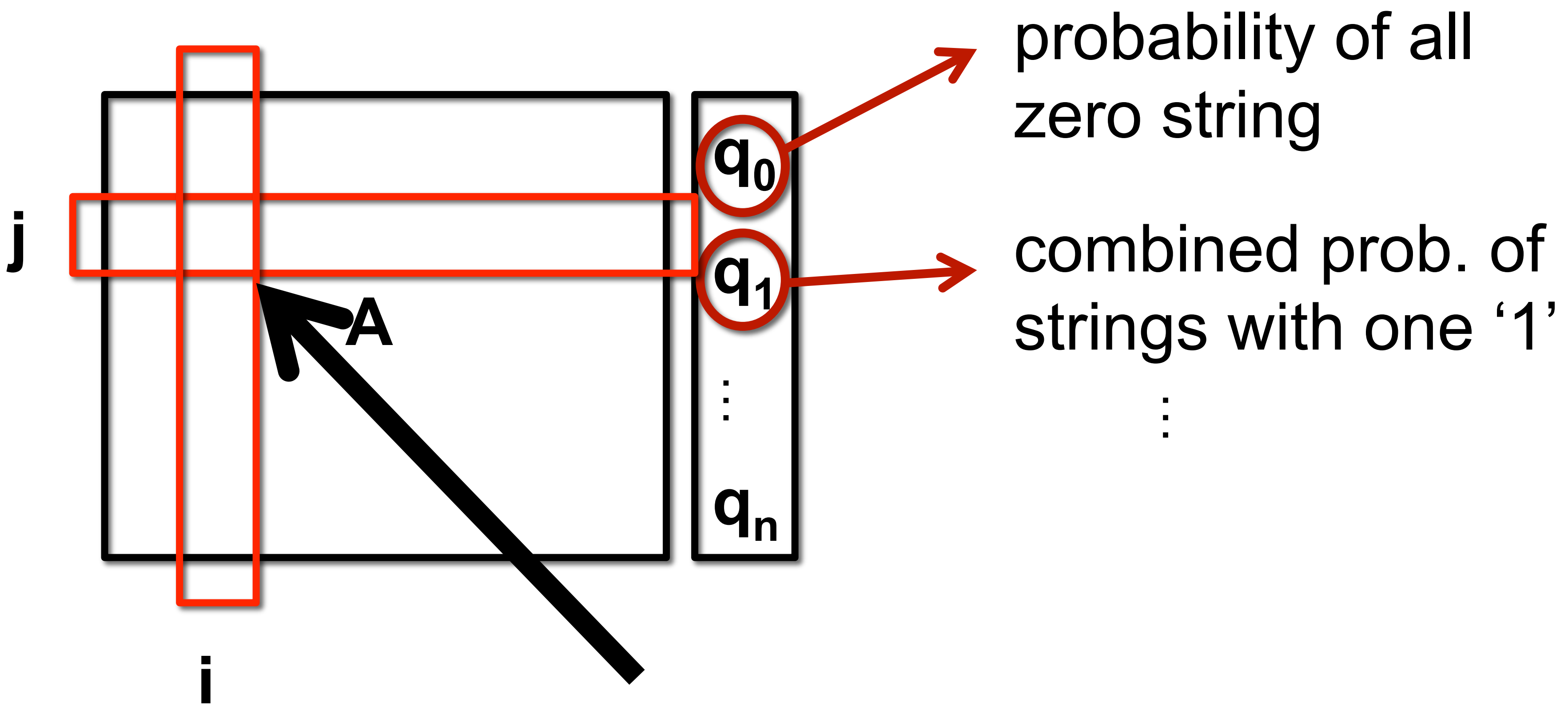
probability of all zero string

total probability of strings with one ‘1’

⋮

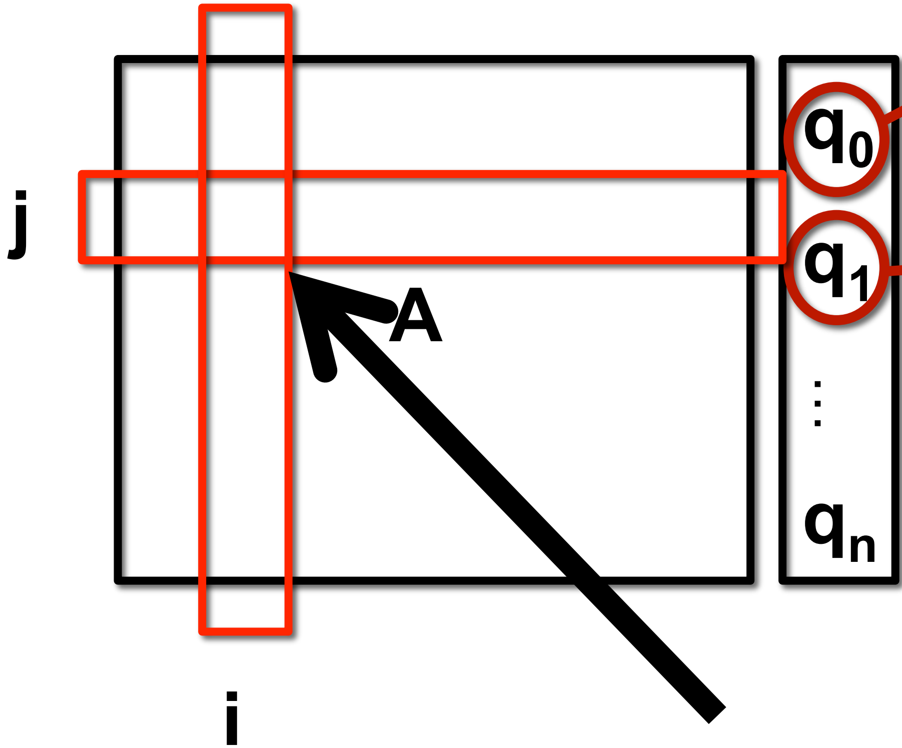
i.e. $q_1 = \Sigma_{i \text{ in } S}\, p_i$ for $S = \{i \mid a_i$ has one ‘1’$\}$

The Setup



“probability that if there are **i** ones, **j** remain”

The Setup



probability of all zero string

combined prob. of strings with one '1'

⋮

$$\binom{i}{j}\mu^j(1-\mu)^{i-j}$$

"probability that if there are **i** ones, **j** remain"

The Setup

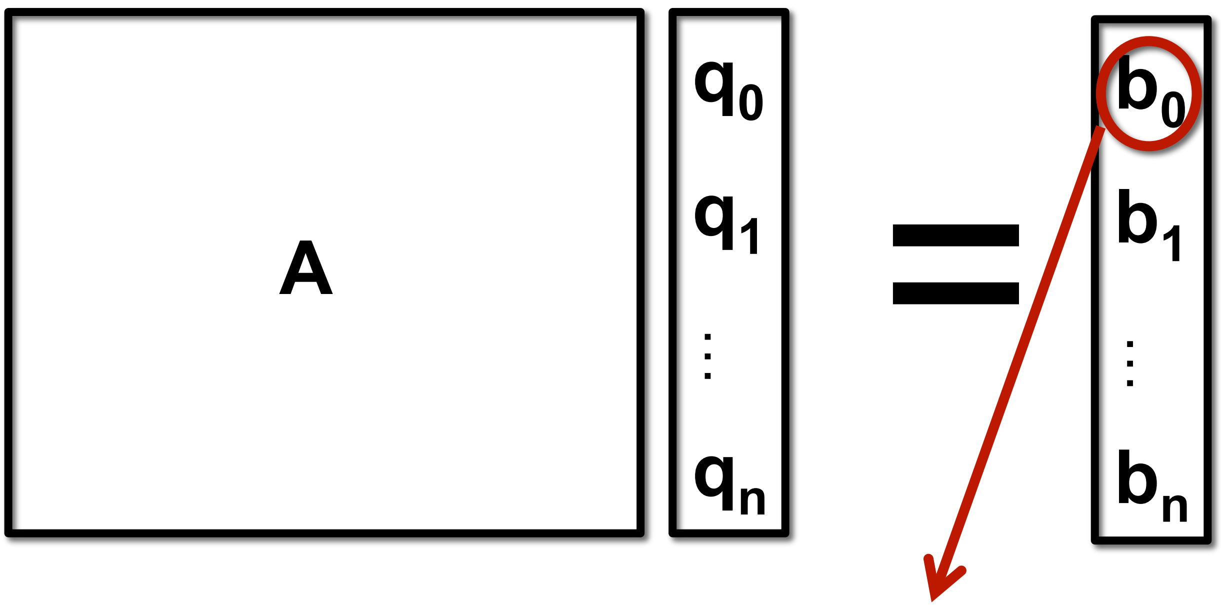
$$
\begin{bmatrix} & & \\ & \mathbf{A} & \\ & & \end{bmatrix}
\begin{bmatrix} \mathbf{q_0} \\ \mathbf{q_1} \\ \vdots \\ \mathbf{q_n} \end{bmatrix}
=
\begin{bmatrix} \mathbf{b_0} \\ \mathbf{b_1} \\ \vdots \\ \mathbf{b_n} \end{bmatrix}
$$

The Setup

$$\boxed{\quad A \quad} \begin{bmatrix} q_0 \\ q_1 \\ \vdots \\ q_n \end{bmatrix} = \begin{bmatrix} b_0 \\ b_1 \\ \vdots \\ b_n \end{bmatrix}$$

probability of all ‘0’s and ‘?’s in the sample

The Setup

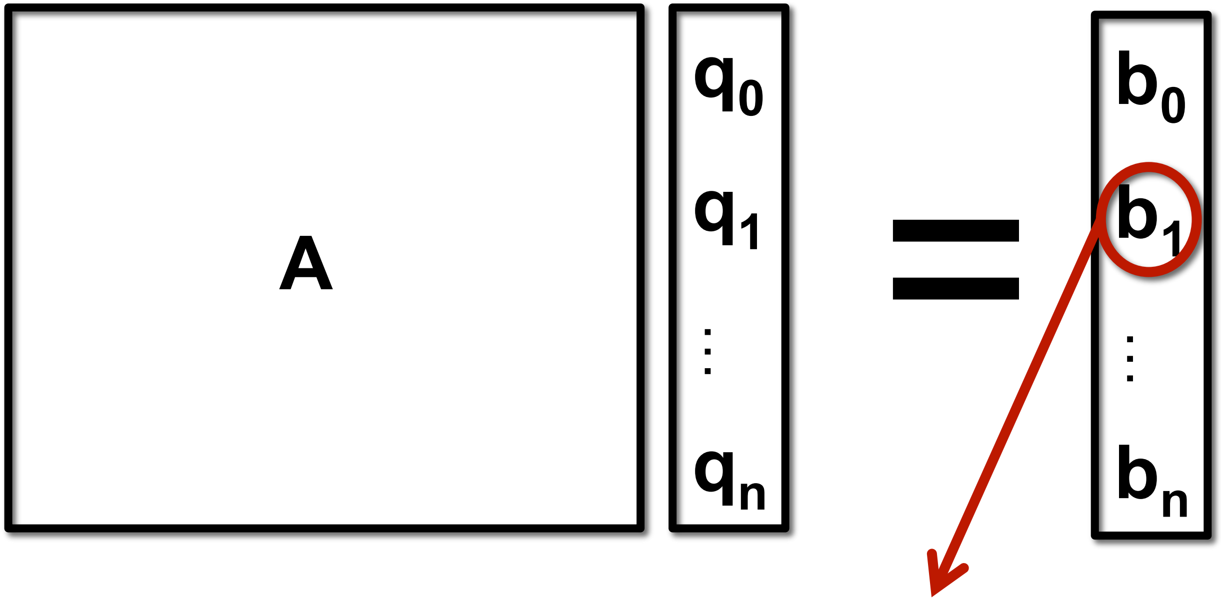


probability of all ‘0’s and ‘?’s in the sample

e.g.

?	0	?	?	?	0	?	?	?	?

The Setup



probability of one ‘1’
in the sample

The Setup

$$\mathbf{A} \begin{bmatrix} \mathbf{q_0} \\ \mathbf{q_1} \\ \vdots \\ \mathbf{q_n} \end{bmatrix} = \begin{bmatrix} \mathbf{b_0} \\ \mathbf{b_1} \\ \vdots \\ \mathbf{b_n} \end{bmatrix}$$

probability of one ‘1’
in the sample

e.g.

0	?	0	?	?	?	?	1	?	0

The Issue…

$$\begin{bmatrix} & & \\ & A & \\ & & \end{bmatrix} \begin{bmatrix} q_0 \\ q_1 \\ \vdots \\ q_n \end{bmatrix} = \begin{bmatrix} b_0 \\ b_1 \\ \vdots \\ b_n \end{bmatrix}$$

The Issue…

$$\mathbf{A} \begin{bmatrix} \mathbf{q_0} \\ \mathbf{q_1} \\ \vdots \\ \mathbf{q_n} \end{bmatrix} = \begin{bmatrix} \mathbf{b_0} \\ \mathbf{b_1} \\ \vdots \\ \mathbf{b_n} \end{bmatrix}$$

If we are given an approx $\overline{b}$, can we just compute $A^{-1}\overline{b}$ and take its first coordinate? (i.e. $e_0 A^{-1}\overline{b}$)

The Issue…

$$\begin{bmatrix} & & \\ & \mathbf{A} & \\ & & \end{bmatrix} \begin{bmatrix} \mathbf{q_0} \\ \mathbf{q_1} \\ \vdots \\ \mathbf{q_n} \end{bmatrix} = \begin{bmatrix} \mathbf{b_0} \\ \mathbf{b_1} \\ \vdots \\ \mathbf{b_n} \end{bmatrix}$$

If we are given an approx $\bar{b}$, can we just compute $A^{-1}\bar{b}$ and take its first coordinate? (i.e. $e_0 A^{-1}\bar{b}$)

No, condition number of **A** is **exponentially** large!

Robust Local Inverse (Dvir et al)

Robust Local Inverse (Dvir et al)

Set $x = e_0A^{-1}$, then $xb = e_0A^{-1}Aq = q_0$

Robust Local Inverse (Dvir et al)

Set $x = e_0A^{-1}$, then $xb = e_0A^{-1}Aq = q_0$

But x has exponentially large norm, so we'd need to know b within exponentially small error

Robust Local Inverse (Dvir et al)

Set $x = e_0A^{-1}$, then $xb = e_0A^{-1}Aq = q_0$

But x has exponentially large norm, so we'd need to know b within exponentially small error

Idea: Add a perturbation (vector) η so that

Robust Local Inverse (Dvir et al)

Set $x = e_0A^{-1}$, then $xb = e_0A^{-1}Aq = q_0$

But x has exponentially large norm, so we'd need to know b within exponentially small error

Idea: Add a perturbation (vector) η so that

Set $\overline{x} = (e_0+\eta)A^{-1}$, then $\overline{x}\, b = (e_0+\eta)A^{-1}Aq = q_0+\eta q$

Robust Local Inverse (Dvir et al)

Set $x = e_0A^{-1}$, then $xb = e_0A^{-1}Aq = q_0$

But x has exponentially large norm, so we'd need to know b within exponentially small error

Idea: Add a perturbation (vector) η so that

Set $\bar{x} = (e_0+\eta)A^{-1}$, then $\bar{x}\, b = (e_0+\eta)A^{-1}Aq = q_0+\eta q$

Can we perturb e_0 s.t. $(e_0+\eta)A^{-1}$ has bdd norm?

Theorem [Dvir et al]: There is a robust local inverse for any $\mu > 0.36$

Theorem [Dvir et al]: There is a robust local inverse for any $\mu > 0.36$

Theorem [Batman et al]: The same robust local inverse works for any $\mu > 0.30$, conjectured it doesn't work for $\mu < 1/4$

Theorem [Dvir et al]: There is a robust local inverse for any $\mu > 0.36$

Theorem [Batman et al]: The same robust local inverse works for any $\mu > 0.30$, conjectured it doesn't work for $\mu < 1/4$

Theorem [Moitra, Saks]: There is a robust local inverse for any $\mu > 0$

Theorem [Dvir et al]: There is a robust local inverse for any $\mu > 0.36$

Theorem [Batman et al]: The same robust local inverse works for any $\mu > 0.30$, conjectured it doesn't work for $\mu < 1/4$

Theorem [Moitra, Saks]: There is a robust local inverse for any $\mu > 0$

What does this robust local inverse look like??

Theorem [Dvir et al]: There is a robust local inverse for any $\mu > 0.36$

Theorem [Batman et al]: The same robust local inverse works for any $\mu > 0.30$, conjectured it doesn't work for $\mu < 1/4$

Theorem [Moitra, Saks]: There is a robust local inverse for any $\mu > 0$

What does this robust local inverse look like??

Idea: Write a linear program for computing a good RLI, and prove that the dual has no solution

We can write an LP for finding a RLI:

We can write an LP for finding a RLI:

(accuracy) $||\mathbf{x} A - e_0||_\infty \le \varepsilon$

We can write an LP for finding a RLI:

(accuracy) $\|\mathbf{x} A - e_0\|_\infty \le \varepsilon$

(insensitivity) $\| \mathbf{x} \|_\infty \le C = \text{poly}(n, 1/\varepsilon)$

We can write an LP for finding a RLI:

(accuracy) $\|\mathbf{x} A - e_0\|_\infty \le \varepsilon$

(insensitivity) $\| \mathbf{x} \|_\infty \le C = \mathrm{poly}(n, 1/\varepsilon)$

Instead, use a natural **basis** of estimators:

We can write an LP for finding a RLI:

(accuracy) $||\mathbf{x} A - e_0||_\infty \leq \varepsilon$

(insensitivity) $|| \mathbf{x} ||_\infty \leq C = \text{poly}(n, 1/\varepsilon)$

Instead, use a natural **basis** of estimators:

i.e. can we find a good RLI as a linear combination of estimators of the form:

$$[1, \alpha, \alpha^2, \alpha^3, \ldots \alpha^{n-1}]$$

We can write an LP for finding a RLI:

(accuracy) $\|\mathbf{x} A - e_0\|_\infty \leq \varepsilon$

(insensitivity) $\| \mathbf{x} \|_\infty \leq C = \mathrm{poly}(n, 1/\varepsilon)$

Instead, use a natural **basis** of estimators:

i.e. can we find a good RLI as a linear combination of estimators of the form:

$$[1, \alpha, \alpha^2, \alpha^3, \ldots \alpha^{n-1}]$$

Why is this basis natural for population recovery?

Basis: $[1, \alpha, \alpha^2, \alpha^3, \ldots \alpha^{n-1}]$

Basis: $[1, \alpha, \alpha^2, \alpha^3, \ldots \alpha^{n-1}]$

This transforms the constraints of the LP to be monomials of a polynomial

Basis: $[1, \alpha, \alpha^2, \alpha^3, \ldots \alpha^{n-1}]$

This transforms the constraints of the LP to be monomials of a polynomial

Hence the dual program wants to construct a certain type of polynomial

Basis: $[1, \alpha, \alpha^2, \alpha^3, \ldots \alpha^{n-1}]$

This transforms the constraints of the LP to be monomials of a polynomial

Hence the dual program wants to construct a certain type of polynomial

---

If we can prove no such polynomial exists

⬇

There is a good RLI, which we can find via an LP

An Uncertainty Principle?

An Uncertainty Principle?

The dual program wants to construct **p(x)** s.t.

$$p(0) \geq \varepsilon \, ||p||_{coeff} + C \, ||q||_{coeff}$$

where $||p||_{coeff} = \Sigma_i \, |p_i|$ for $p(x) = \Sigma_i p_i x^i$

An Uncertainty Principle?

The dual program wants to construct **p(x)** s.t.

$$p(0) \geq \varepsilon \, ||p||_{coeff} + C \, ||q||_{coeff}$$

where $||p||_{coeff} = \Sigma_i \, |p_i|$ for $p(x) = \Sigma_i p_i x^i$

and $q(x) \cong p(1-\mu + \mu x)$

An Uncertainty Principle?

The dual program wants to construct **p(x)** s.t.

$$p(0) \geq \varepsilon \, ||p||_{coeff} + C \, ||q||_{coeff}$$

where $||p||_{coeff} = \Sigma_i \, |p_i|$ for $p(x) = \Sigma_i p_i x^i$

and $q(x) \cong p(1-\mu + \mu x)$

Conversely, for a polynomial are its coefficients large in at least one of the two representations?

Relaxation #1

Relaxation #1

Claim: $||p||_{coeff} \geq \sup_{x \text{ in } [-1,1]} |p(x)|$

Relaxation #1

Claim: $||p||_{coeff} \geq \sup_{x \text{ in } [-1,1]} |p(x)|$

Proof: Consider x in [-1,1]:

$$|p(x)| \leq \Sigma_i |p_i| \, |x^i| \leq \Sigma_i |p_i| = ||p||_{coeff}$$

Relaxation #1

Claim: $||p||_{coeff} \geq \sup_{x \text{ in } [-1,1]} |p(x)|$

Proof: Consider x in [-1,1]:

$$|p(x)| \leq \Sigma_i |p_i| \, |x^i| \leq \Sigma_i |p_i| = ||p||_{coeff}$$

New Question:

For all polynomials is it true that:

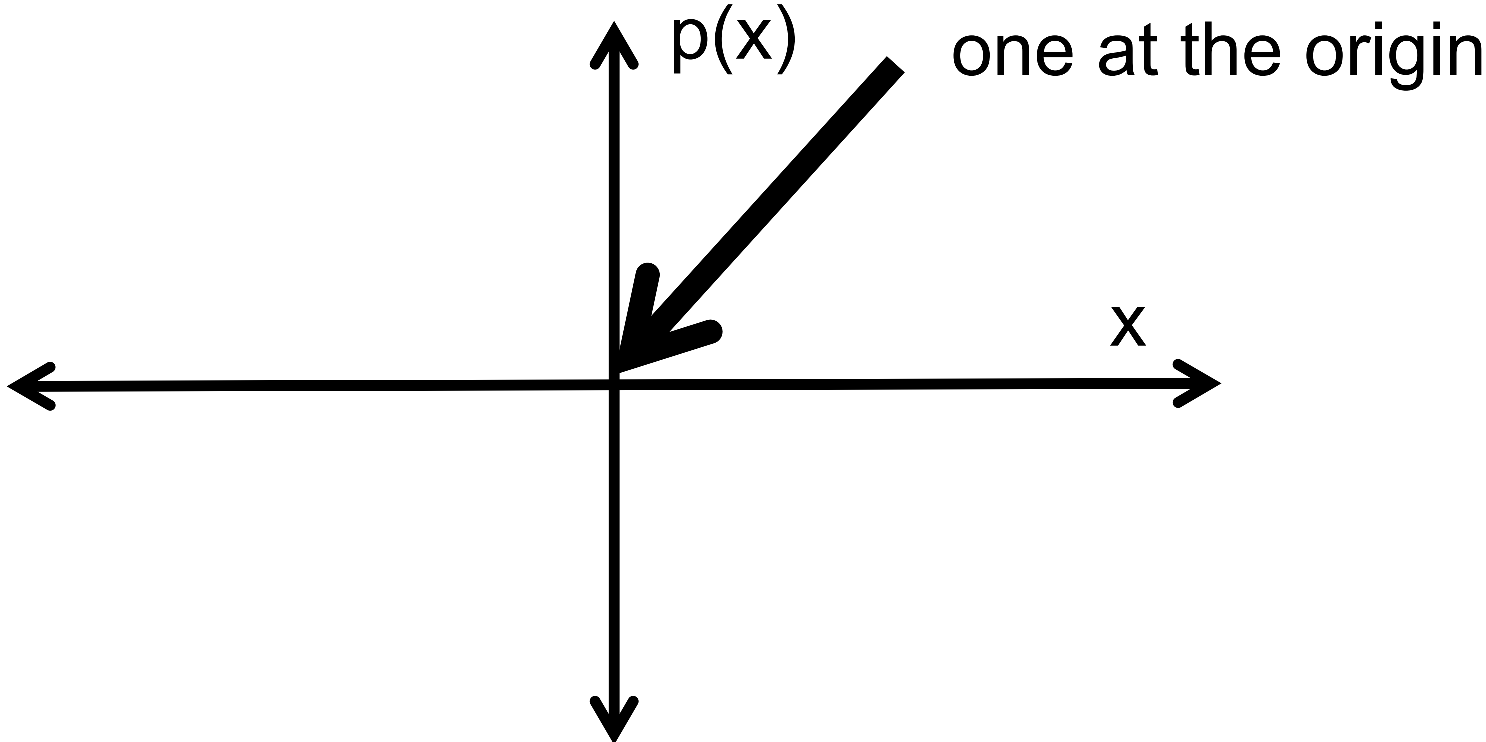
$$p(0) < \varepsilon \sup_{x \text{ in } [-1,1]} |p(x)| + C \sup_{x \text{ in } [-1,1]} |p(1-\mu+\mu x)| \ ?$$

For all polynomials with p(0) =1 is it true that:

$1 < \varepsilon \sup_{x \text{ in } [-1,1]}|p(x)| + C \sup_{x \text{ in } [-1,1]}|p(1- \mu+\mu x)|$?

For all polynomials with p(0) =1 is it true that:

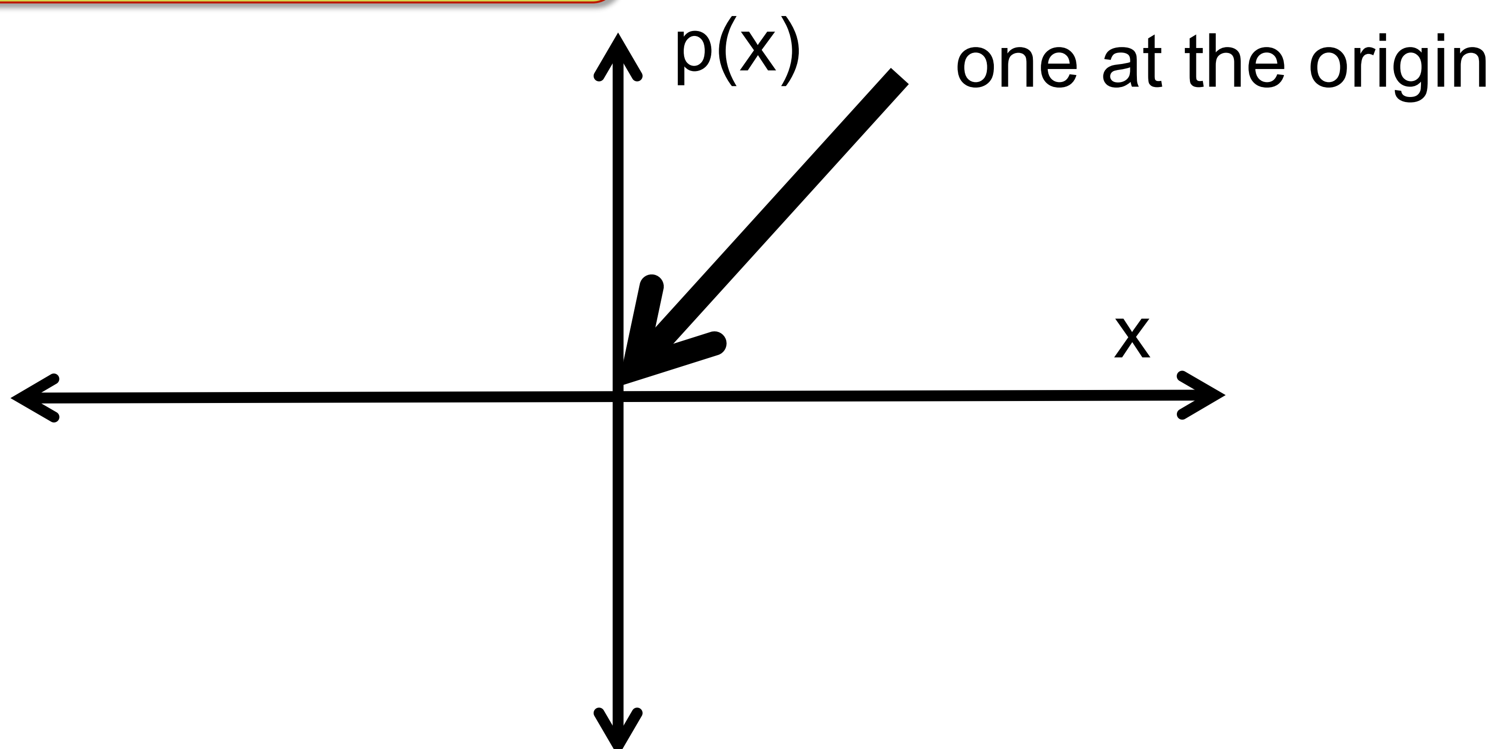
$$1 < \varepsilon \sup_{x \text{ in } [-1,1]}|p(x)| + C \sup_{x \text{ in } [-1,1]}|p(1-\mu+\mu x)| \ ?$$



For all polynomials with p(0) =1 is it true that:

$$1 < \varepsilon \sup_{x \text{ in } [-1,1]} |p(x)| + C \sup_{x \text{ in } [-1,1]} |p(1-\mu+\mu x)| \ ?$$

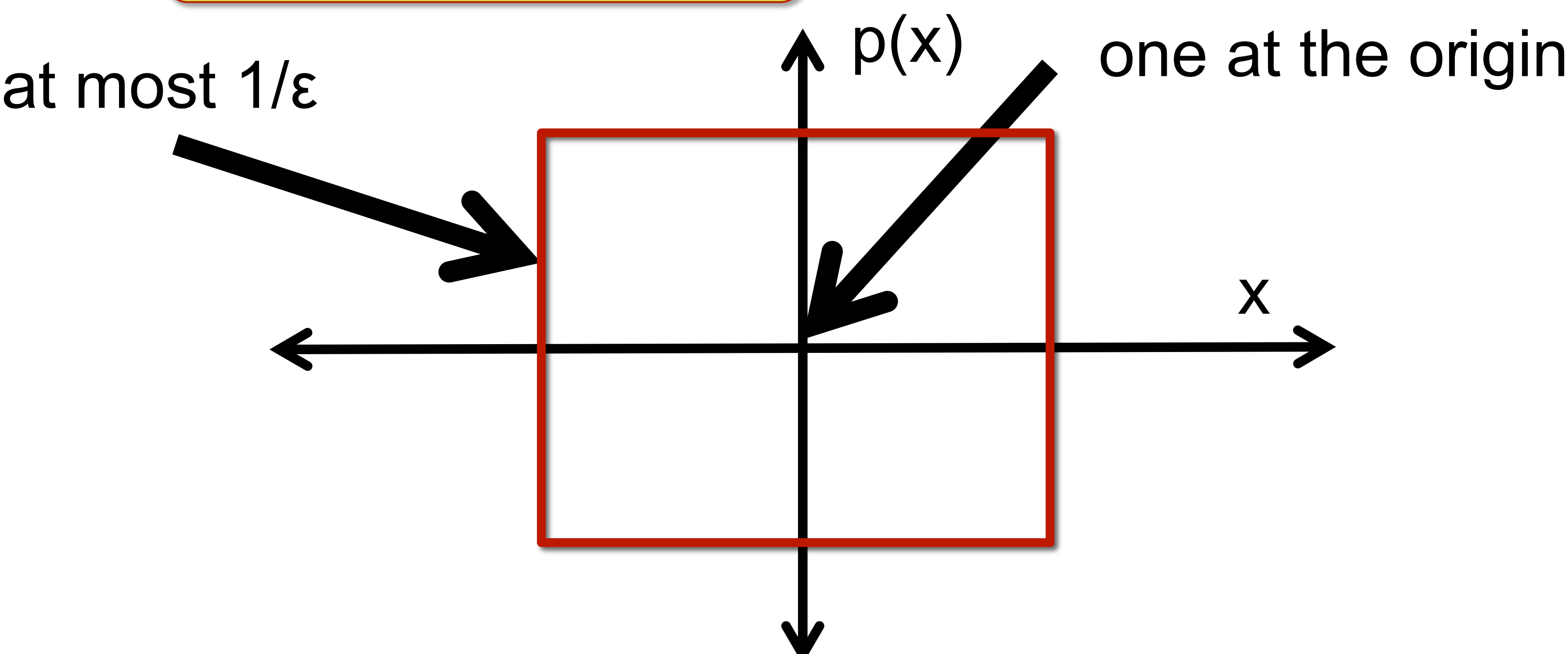
Try: $|p(x)| \leq 1/\varepsilon$ on $[-1,1]$



For all polynomials with p(0) =1 is it true that:

$$1 < \underbrace{\varepsilon \sup_{x \text{ in } [-1,1]}|p(x)|}+ C \sup_{x \text{ in } [-1,1]}|p(1- \mu+\mu x)| \text{ ?}$$

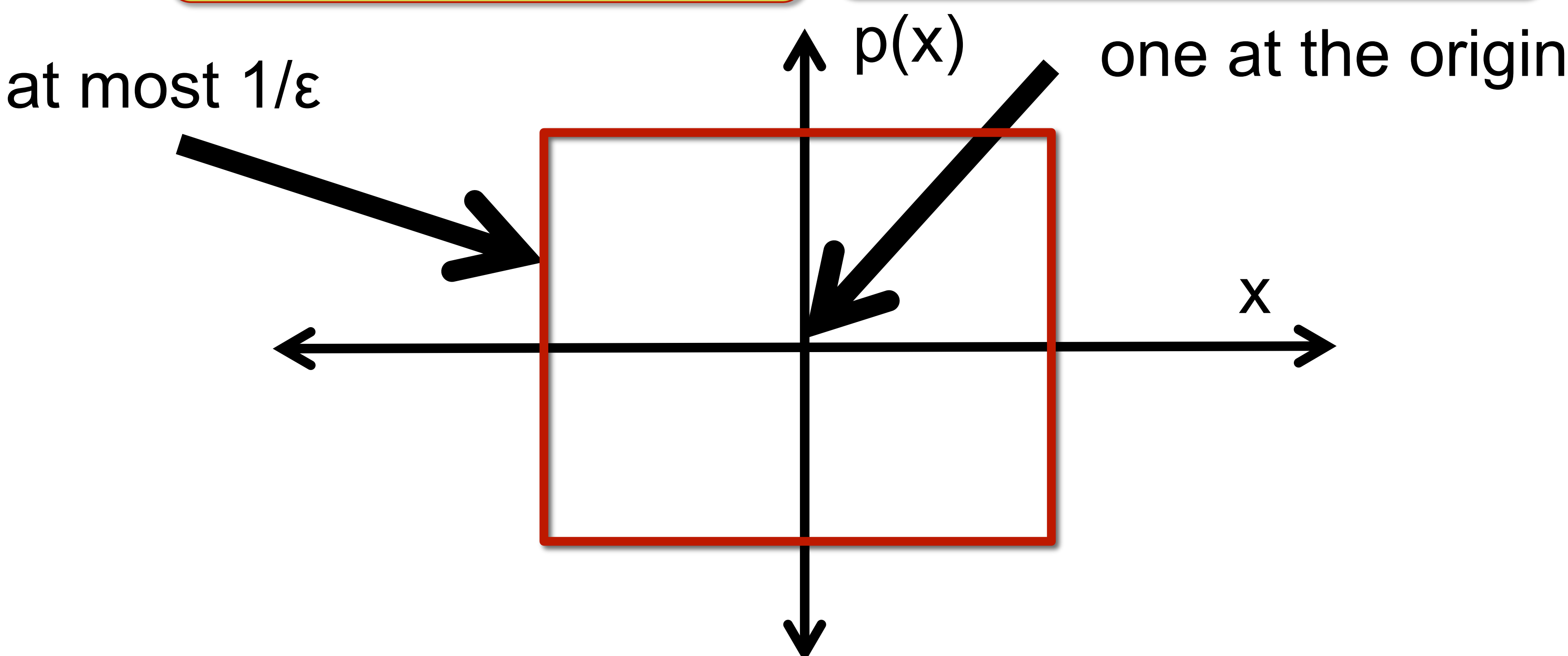
Try: $\boxed{|p(x)| \le 1/\varepsilon \text{ on } [-1,1]}$



For all polynomials with p(0) =1 is it true that:

$$1 < \varepsilon \sup_{x \text{ in } [-1,1]}|p(x)| + C \sup_{x \text{ in } [-1,1]}|p(1-\mu+\mu x)| \;?$$

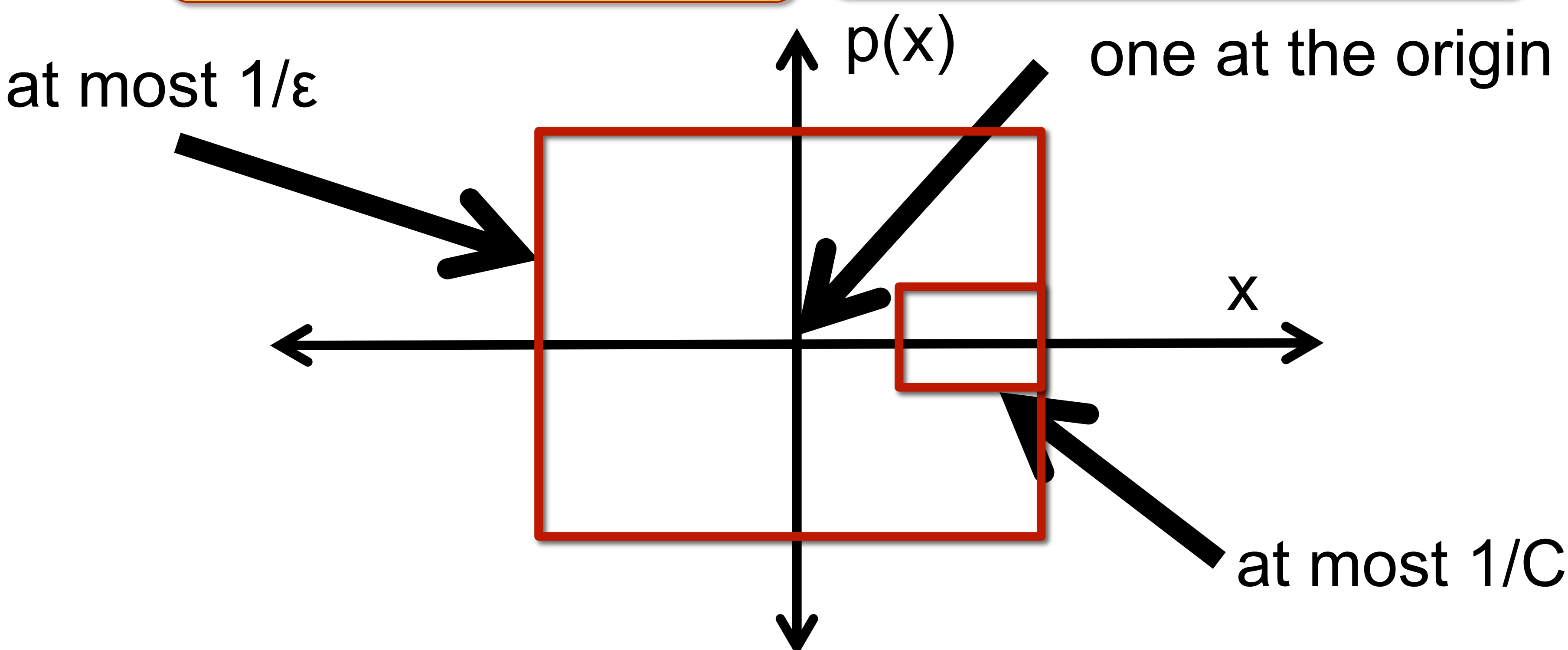
Try: $|p(x)| \leq 1/\varepsilon$ on $[-1,1]$ $\quad$ $|p(x)| \leq 1/C$ on $[1-2\mu,1]$



For all polynomials with p(0) =1 is it true that:

$$1 < \varepsilon \sup_{x \text{ in } [-1,1]} |p(x)| + C \sup_{x \text{ in } [-1,1]} |p(1-\mu+\mu x)| \ ?$$

Try: $|p(x)| \leq 1/\varepsilon$ on $[-1,1]$ $\quad$ $|p(x)| \leq 1/C$ on $[1-2\mu,1]$



For all polynomials with $p(0) = 1$ is it true that:

$$1 < \underbrace{\varepsilon \sup_{x \text{ in } [-1,1]} |p(x)|}_{} + \underbrace{C \sup_{x \text{ in } [-1,1]} |p(1-\mu+\mu x)|}_{} \ ?$$

Try: $|p(x)| \leq 1/\varepsilon$ on $[-1,1]$ $|p(x)| \leq 1/C$ on $[1-2\mu,1]$

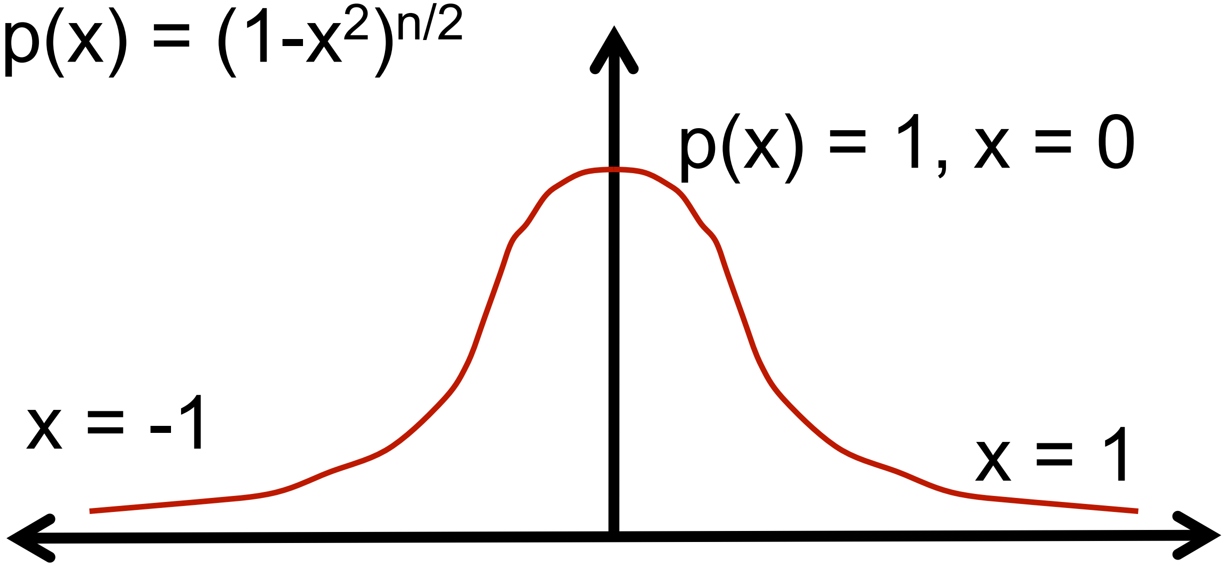
No, set $p(x) = (1-x^2)^{n/2}$

For all polynomials with $p(0) = 1$ is it true that:

$$1 < \varepsilon \sup_{x \text{ in } [-1,1]}|p(x)| + C \sup_{x \text{ in } [-1,1]}|p(1-\mu+\mu x)| \text{ ?}$$

Try: $|p(x)| \leq 1/\varepsilon$ on $[-1,1]$ $\qquad$ $|p(x)| \leq 1/C$ on $[1-2\mu,1]$

No, set $p(x) = (1-x^2)^{n/2}$



Does this p(x) refute our original conjecture too?

Does this p(x) refute our original conjecture too?

$$p(x) = (1-x^2)^{n/2}$$

Does this p(x) refute our original conjecture too?

$$p(x) = (1-x^2)^{n/2}$$

Is $||p||_{coeff}$ too small?

Does this p(x) refute our original conjecture too?

$$p(x) = (1-x^2)^{n/2}$$

Is $||p||_{coeff}$ too small?

No, it is **exponentially** large!

Does this p(x) refute our original conjecture too?

$$p(x) = (1-x^2)^{n/2}$$

Is $||p||_{coeff}$ too small?

No, it is **exponentially** large! Substitute x = i

$$p(i) = 2^{n/2}$$

Does this p(x) refute our original conjecture too?

$$p(x) = (1-x^2)^{n/2}$$

Is $||p||_{coeff}$ too small?

No, it is **exponentially** large! Substitute x = i

$$p(i) = 2^{n/2}$$

Claim: $||p||_{coeff} \geq \sup_{x \text{ in } D} |p(x)|$, where D is the unit complex disk

Relaxation #2

Relaxation #2

Claim: $||p||_{coeff} \geq \sup_{x \text{ in } D} |p(x)|$

Relaxation #2

Claim: $||p||_{coeff} \geq \sup_{x \text{ in } D} |p(x)|$

Proof: Consider x in D:

$$|p(x)| \leq \Sigma_i |p_i| \, |x^i| \leq \Sigma_i |p_i| = ||p||_{coeff}$$

Relaxation #2

Claim: $||p||_{coeff} \geq \sup_{x \text{ in } D} |p(x)|$

Proof: Consider x in D:

$$|p(x)| \leq \Sigma_i |p_i| \, |x^i| \leq \Sigma_i |p_i| = ||p||_{coeff}$$

New Question:

For all polynomials is it true that:

$$p(0) < \varepsilon \sup_{x \text{ in } D} |p(x)| + C \sup_{x \text{ in } D} |p(1- \mu+\mu x)| \ ?$$

For all polynomials with $p(0) = 1$ is it true that:

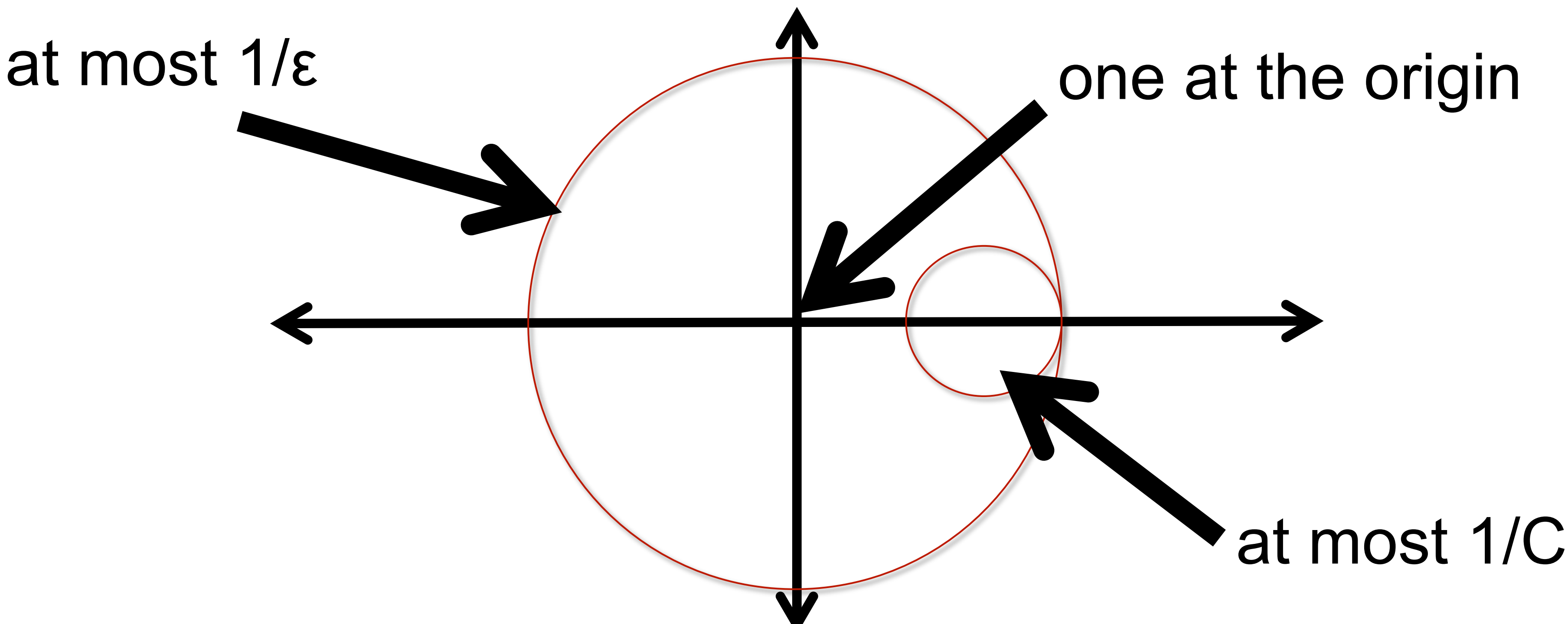
$$1 < \underbrace{\varepsilon \sup_{x \text{ in } D} |p(x)|}_{} + \underbrace{C \sup_{x \text{ in } D} |p(1-\mu+\mu x)|}_{} \text{ ?}$$

Try: $|p(x)| \leq 1/\varepsilon$ on D $|p(x)| \leq 1/C$ on $D(1-\mu, \mu)$

For all polynomials with $p(0) = 1$ is it true that:

$$1 < \varepsilon \sup_{x \text{ in } D} |p(x)| + C \sup_{x \text{ in } D} |p(1-\mu+\mu x)| \ ?$$

Try: $|p(x)| \leq 1/\varepsilon$ on D $\qquad$ $|p(x)| \leq 1/C$ on $D(1-\mu, \mu)$



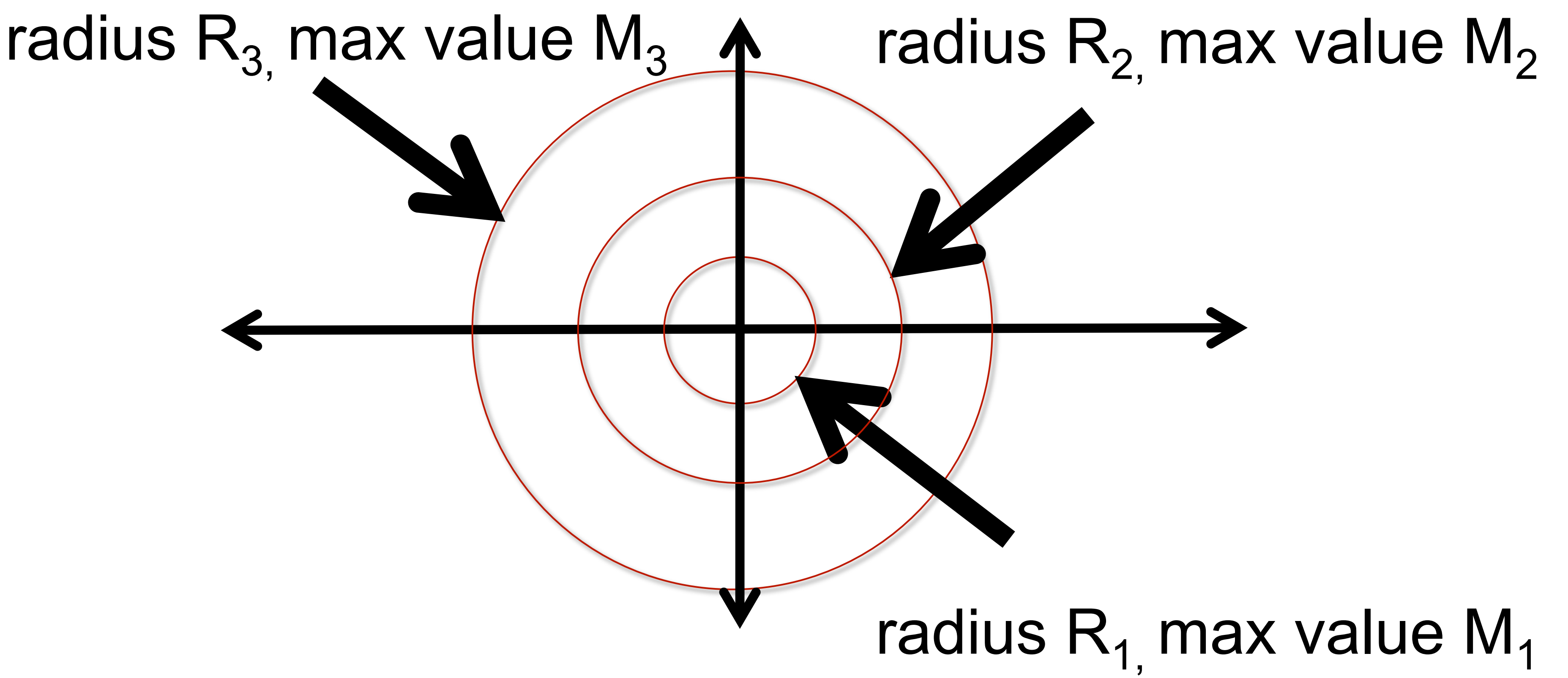
Hadamard Three Circle Theorem

Hadamard Three Circle Theorem

How can we bound the rate of growth of **holomorphic** functions in the complex plane?

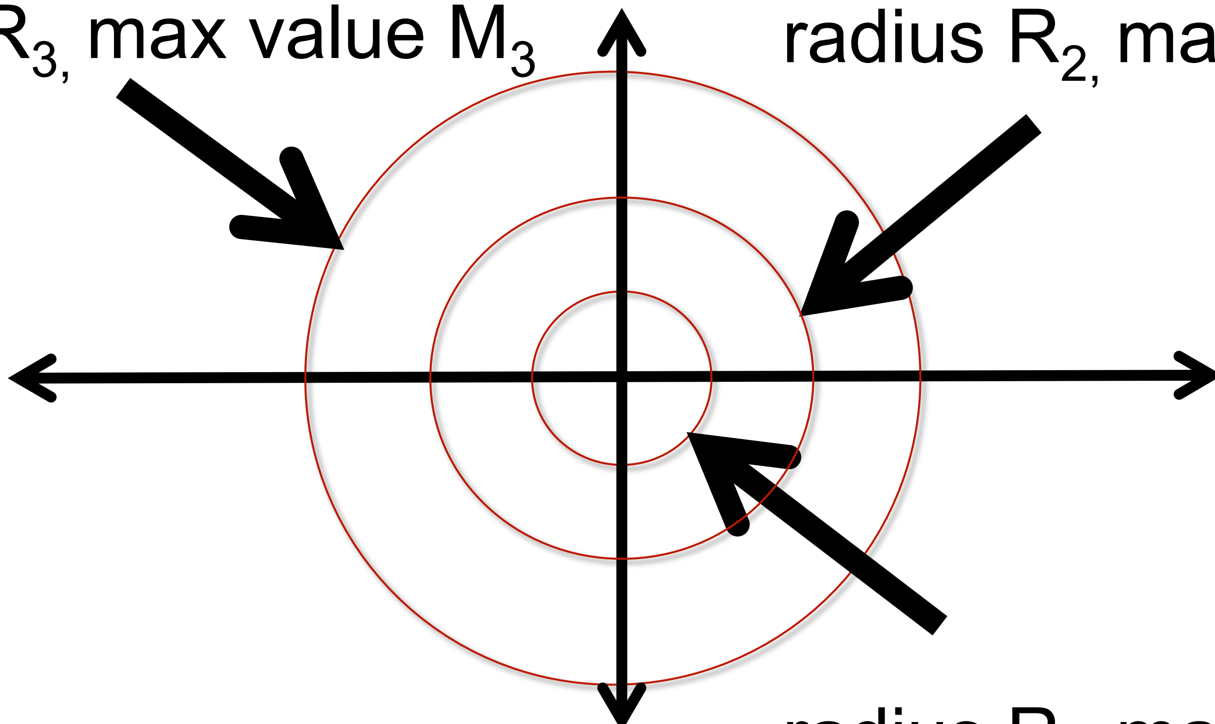
Hadamard Three Circle Theorem

How can we bound the rate of growth of **holomorphic** functions in the complex plane?



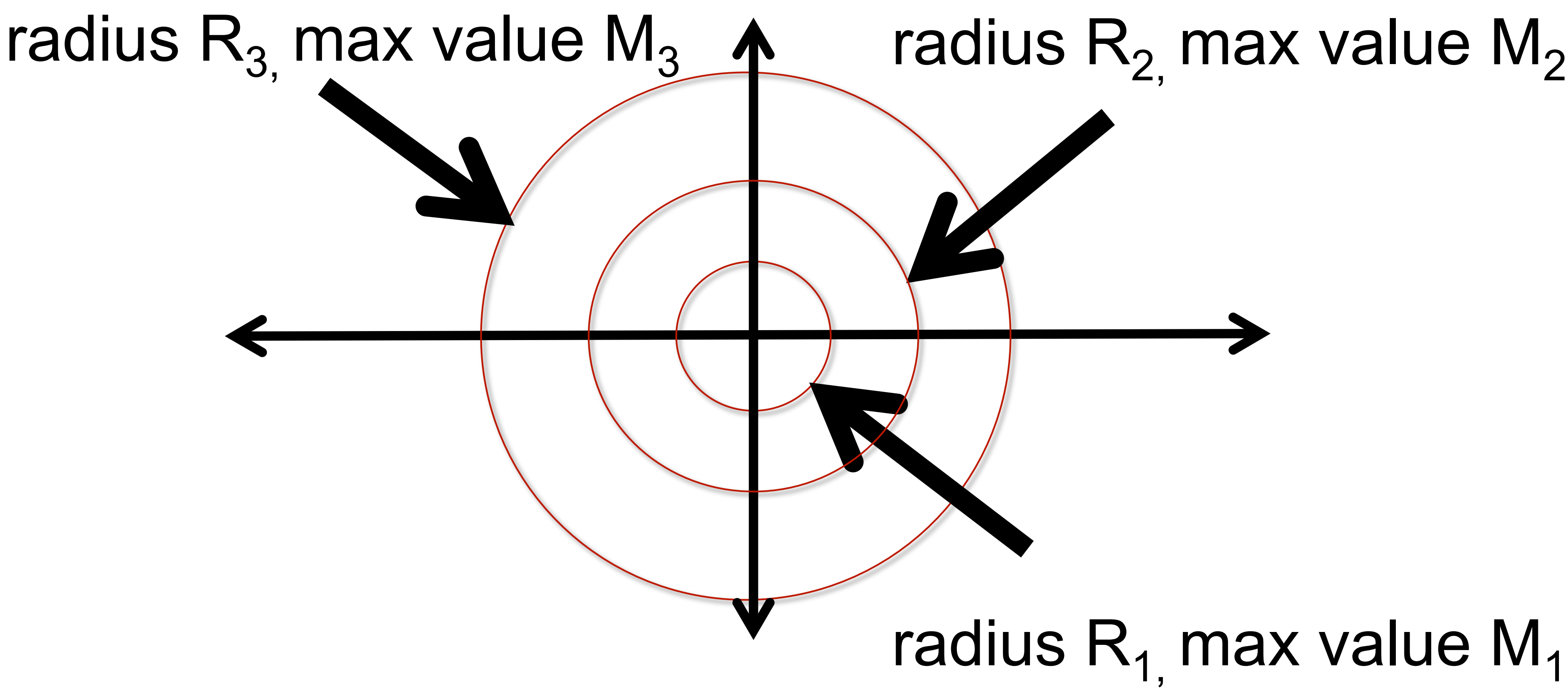
Hadamard Three Circle Theorem

$$\log\frac{R_3}{R_1}\log M_2 \le \log\frac{R_2}{R_1}\log M_3 + \log\frac{R_3}{R_2}\log M_1$$



Hadamard Three Circle Theorem

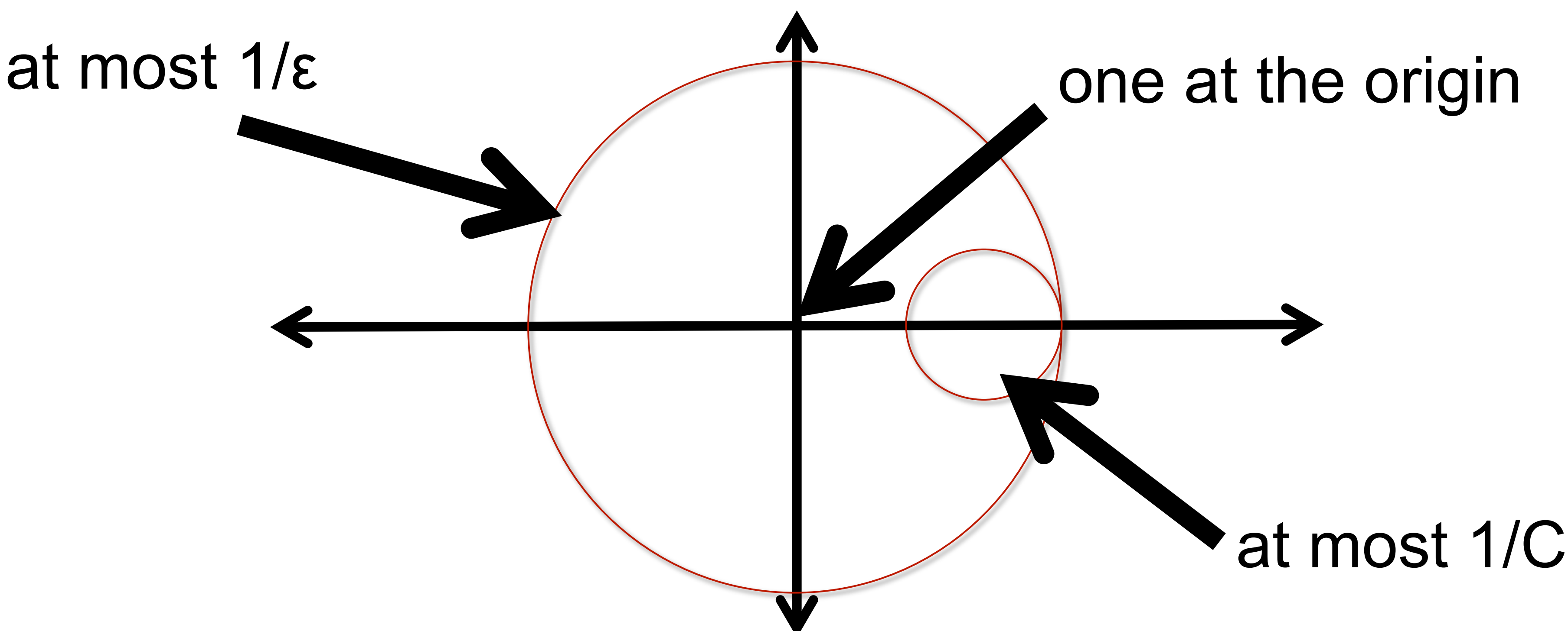
Hence M_2 is bounded by a geometric average of M_1 and M_3 (that depends on the radii)!



For all polynomials with $p(0) = 1$ is it true that:

$$1 < \varepsilon \sup_{x \text{ in } D} |p(x)| + C \sup_{x \text{ in } D} |p(1-\mu+\mu x)| \ ?$$

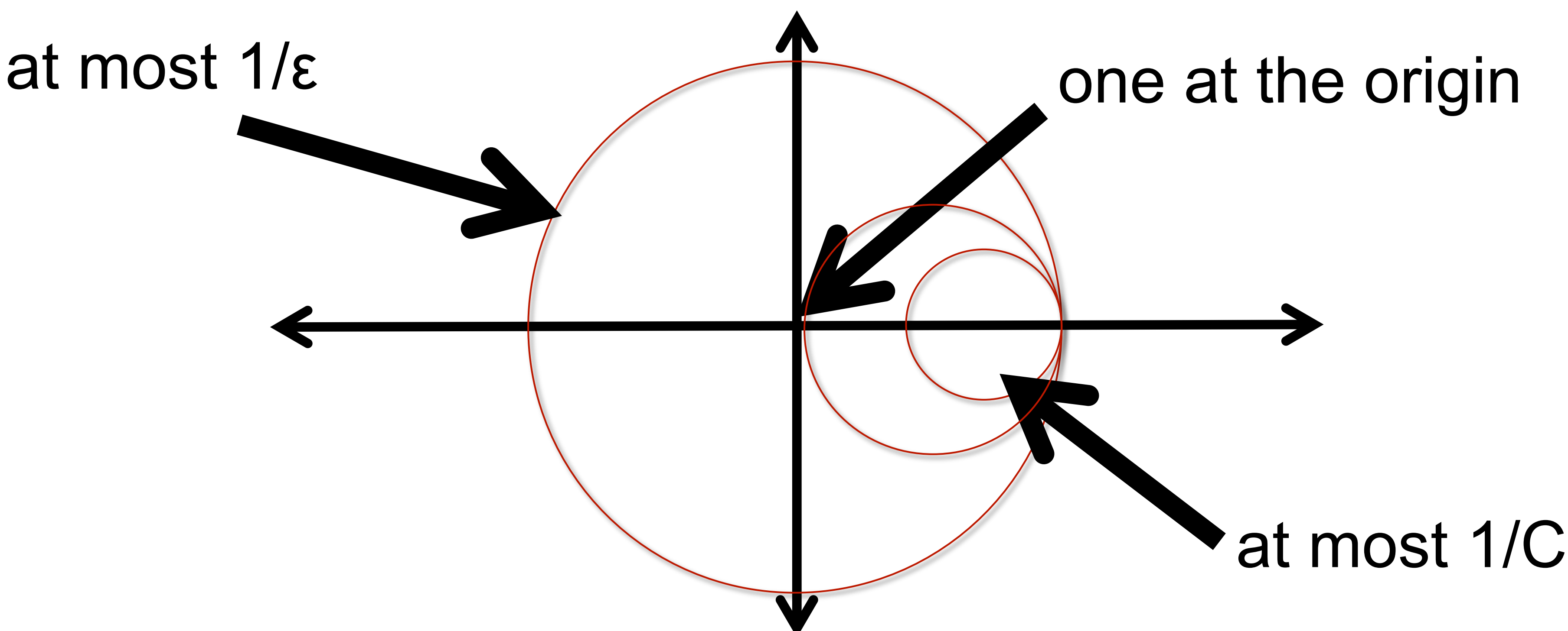
Try: $|p(x)| \leq 1/\varepsilon$ on D $\qquad$ $|p(x)| \leq 1/C$ on $D(1-\mu, \mu)$



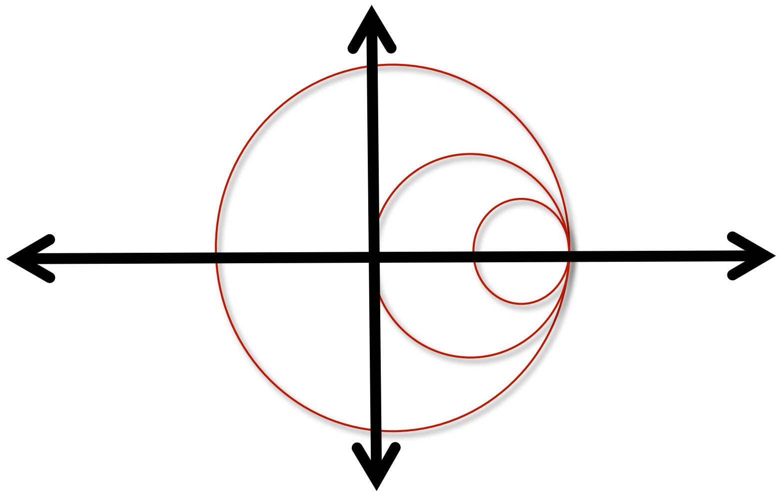
For all polynomials with $p(0) = 1$ is it true that:

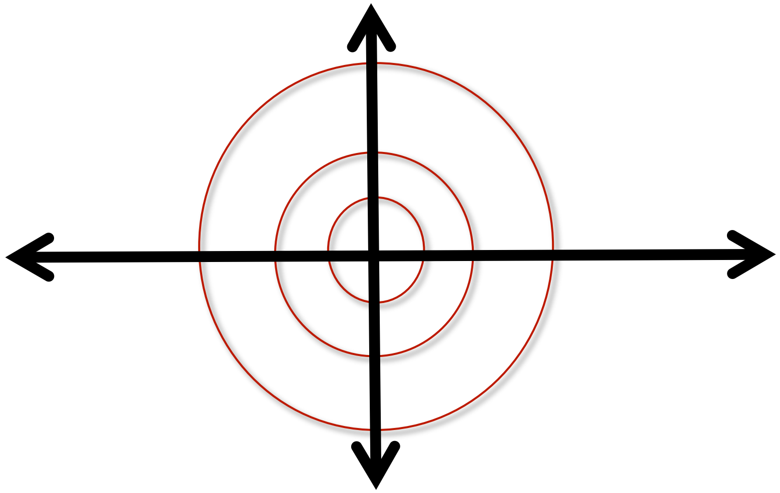
$$1 < \varepsilon \sup_{x \text{ in } D} |p(x)| + C \sup_{x \text{ in } D} |p(1-\mu+\mu x)| \ ?$$

Try: $|p(x)| \leq 1/\varepsilon$ on D $\quad$ $|p(x)| \leq 1/C$ on $D(1-\mu, \mu)$

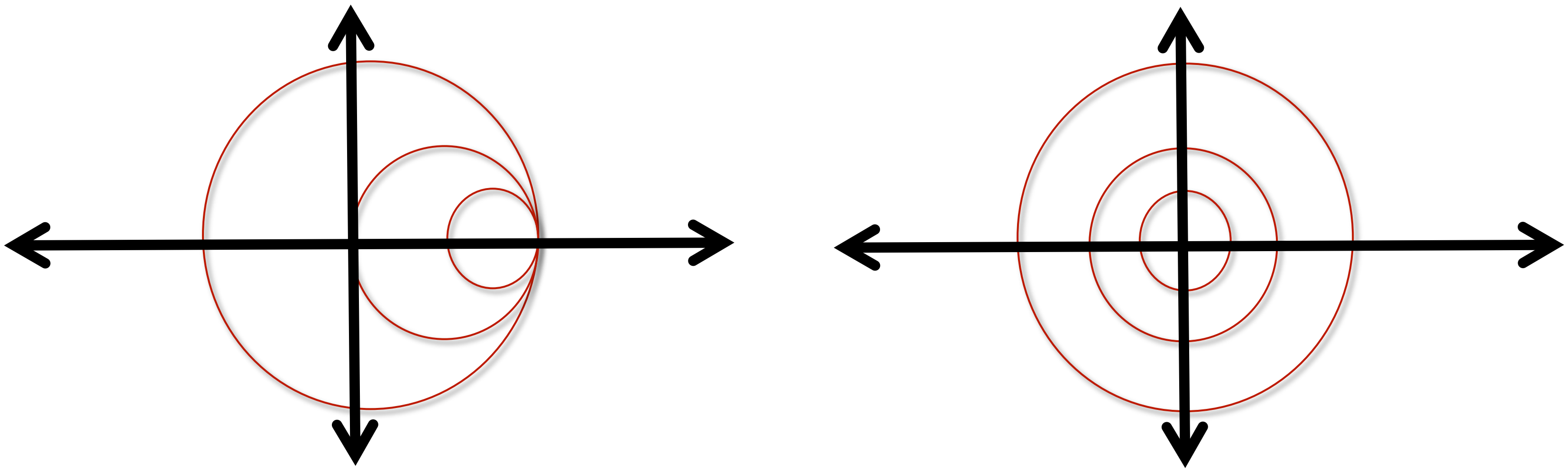


Is there a holomorphic map between these two pictures?



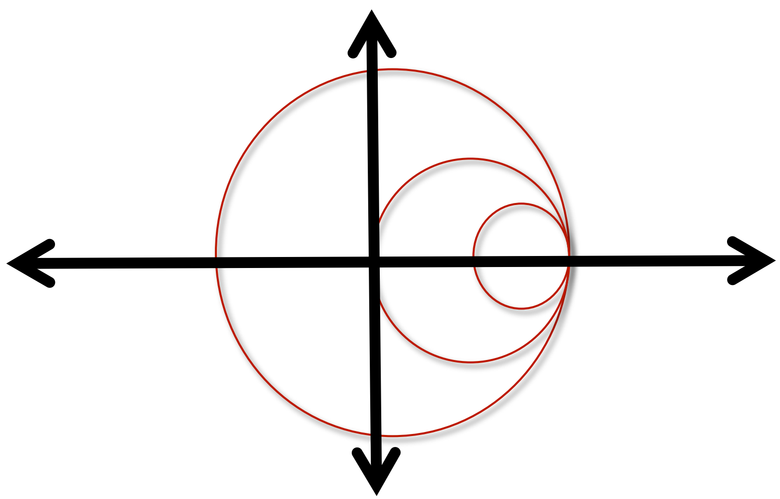


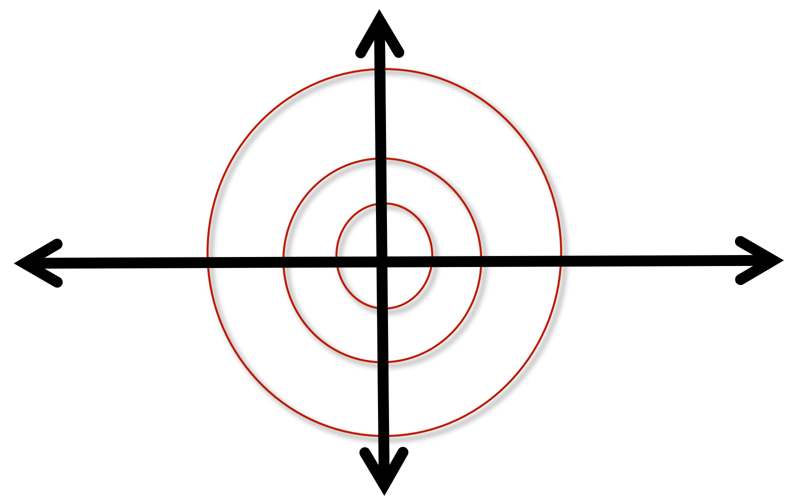
Is there a holomorphic map between these two pictures?



Three Circle Thm

Is there a holomorphic map between these two pictures?

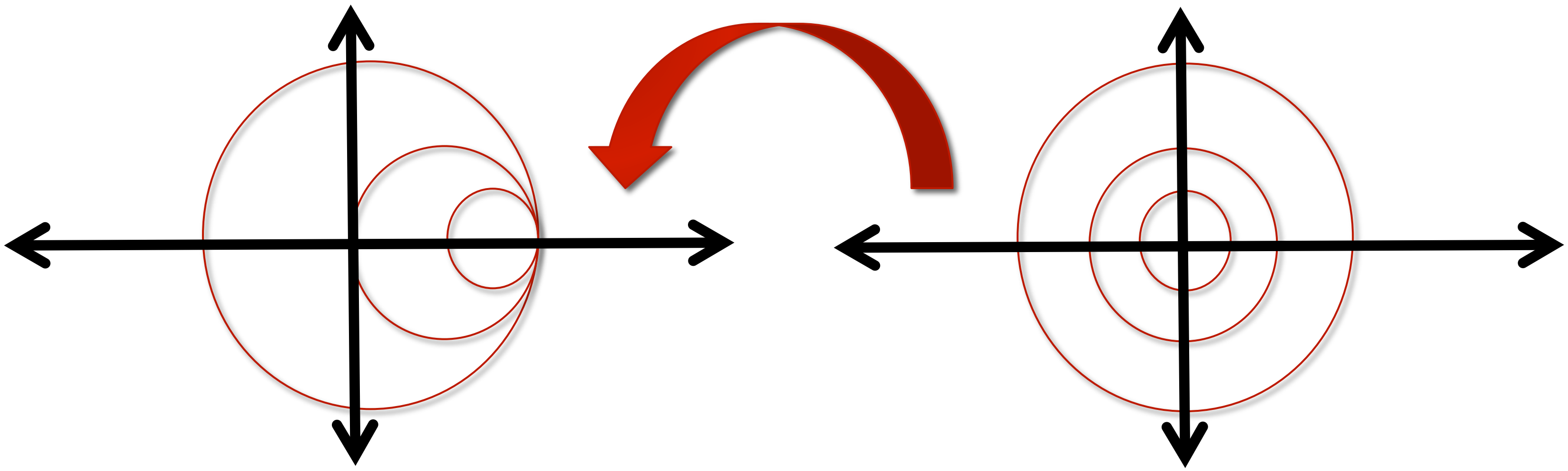




Can we analyze this?

Three Circle Thm

Is there a holomorphic map between these two pictures?



Can we analyze this? Three Circle Thm

Yes! And it is called the Mobius Transform

Outline

Outline

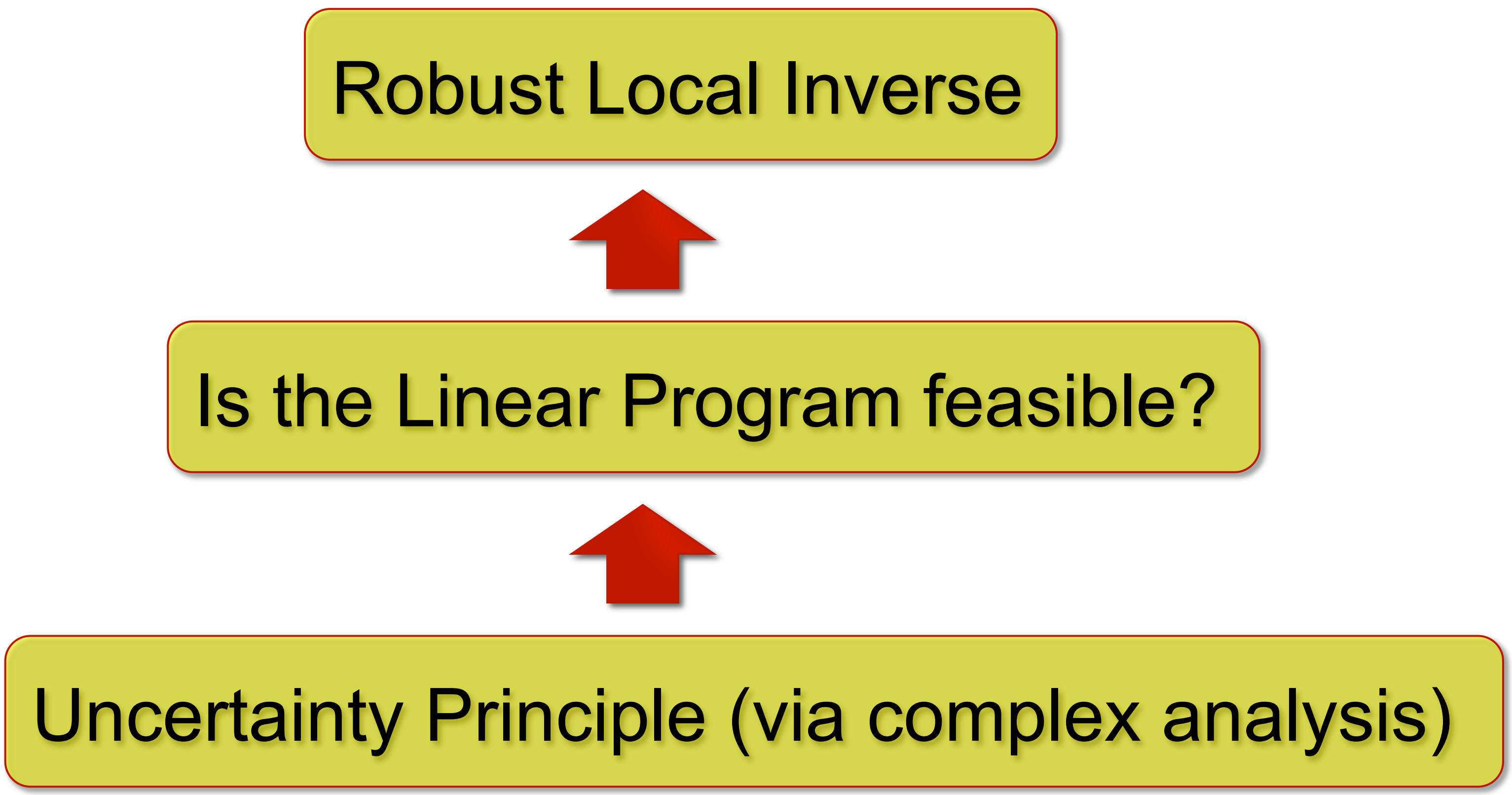
Uncertainty Principle (via complex analysis)

Outline

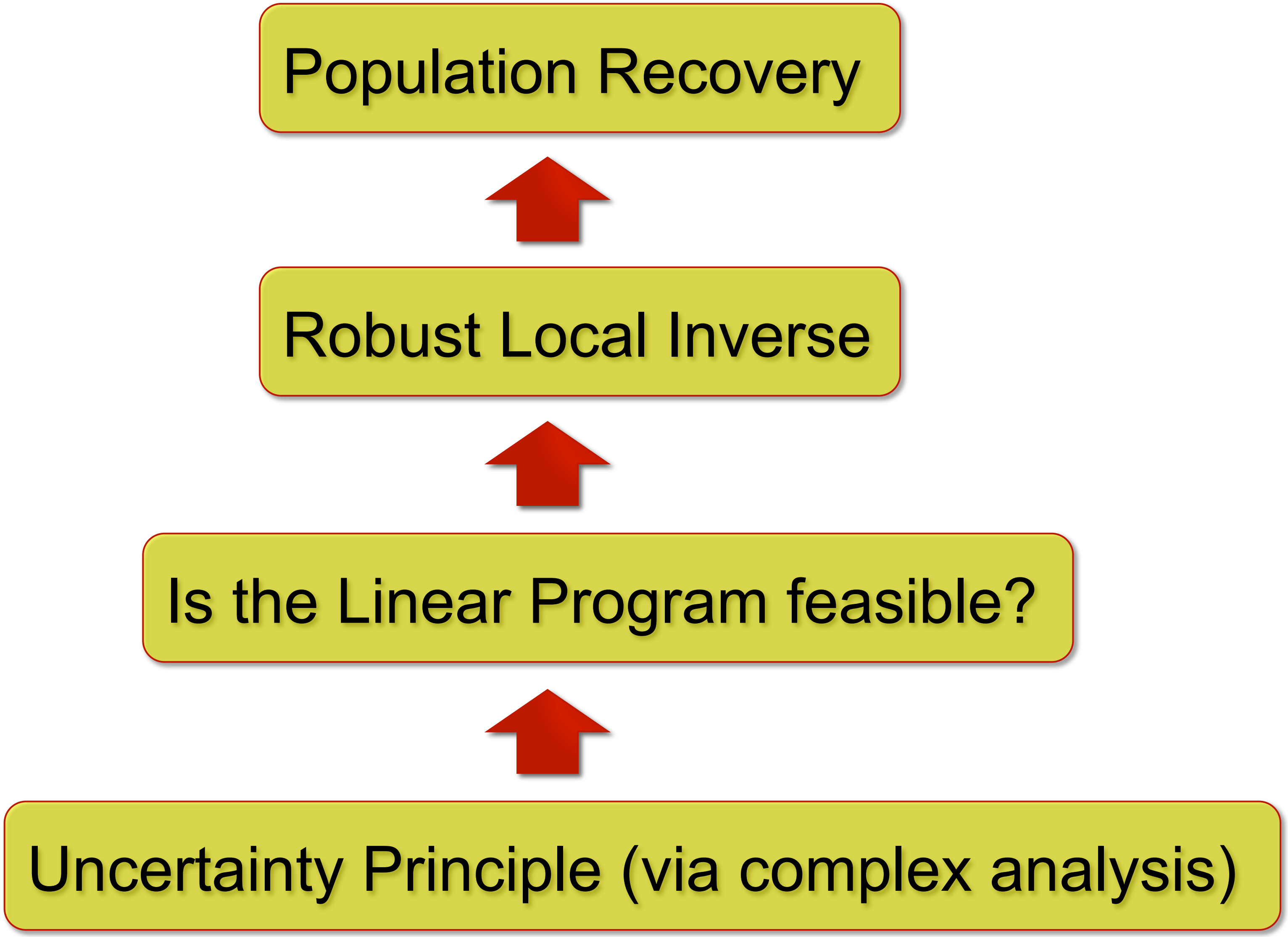
Is the Linear Program feasible?

Uncertainty Principle (via complex analysis)

Outline


Robust Local Inverse
Is the Linear Program feasible?
Uncertainty Principle (via complex analysis)

Outline


Population Recovery
Robust Local Inverse
Is the Linear Program feasible?
Uncertainty Principle (via complex analysis)

Summary and Open Questions

Summary and Open Questions

We solved an inverse problem, despite exponentially large condition number!

Summary and Open Questions

We solved an inverse problem, despite exponentially large condition number!

…using tools from complex analysis

Summary and Open Questions

We solved an inverse problem, despite exponentially large condition number!

…using tools from complex analysis

Can RLIs be useful for other problems in statistical inference?

Summary and Open Questions

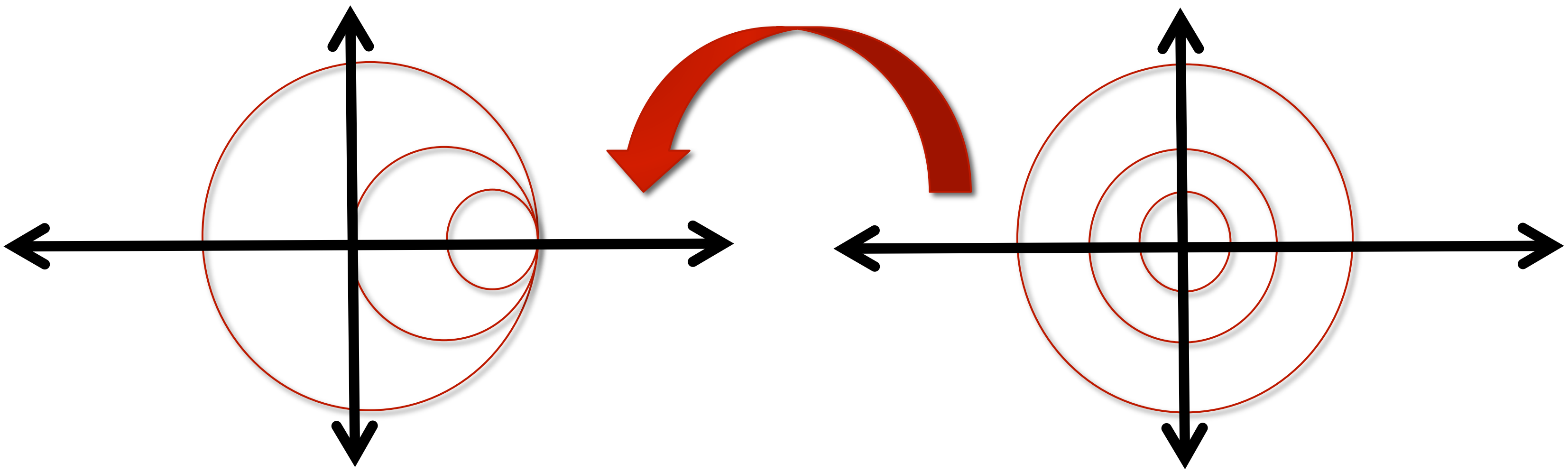
We solved an inverse problem, despite exponentially large condition number!

…using tools from complex analysis

Can RLIs be useful for other problems in statistical inference?

Is there a polynomial time algorithm for **noisy** population recovery?

Thanks!



Any Questions?