# Consumable Data via Quantum Communication

Siddhartha Jain (UT Austin)

Dar Gilboa, Jarrod McClean (Google Quantum AI)

Quantum AI

“The economics of data is a new but rapidly growing field.”

Jones & Tonetti, *Nonrivalry and the Economics of Data*

# Nonrivalry of classical data

# Nonrivalry of classical data

Can be copied at essentially zero cost

# Nonrivalry of classical data

## Can be copied at essentially zero cost

Concerns

# Nonrivalry of classical data

## Can be copied at essentially zero cost

Concerns

As an example, in ML

# Nonrivalry of classical data

## Can be copied at essentially zero cost

Concerns

As an example, in ML

- Privacy concerns during deployment of ML models

# Nonrivalry of classical data

## Can be copied at essentially zero cost

### Concerns

As an example, in ML

- Privacy concerns during deployment of ML models
- No economic incentive for generating curated datasets

# Nonrivalry of classical data

## Can be copied at essentially zero cost

### Concerns

As an example, in ML

- Privacy concerns during deployment of ML models
- No economic incentive for generating curated datasets

# Nonrivalry of classical data

## Can be copied at essentially zero cost

### Concerns

As an example, in ML

- Privacy concerns during deployment of ML models
- No economic incentive for generating curated datasets

### Hope

# Nonrivalry of classical data

## Can be copied at essentially zero cost

### Concerns

As an example, in ML

- Privacy concerns during deployment of ML models
- No economic incentive for generating curated datasets

### Hope

Destructive measurement in quantum mechanics to the rescue!

# Nonrivalry of classical data

## Can be copied at essentially zero cost

### Concerns

As an example, in ML

- Privacy concerns during deployment of ML models
- No economic incentive for generating curated datasets

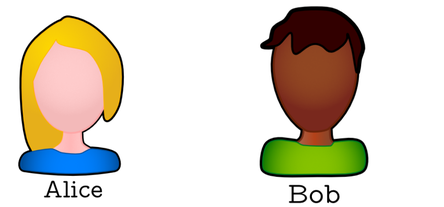
### Hope

Destructive measurement in quantum mechanics to the rescue!

# Our model: Communication Complexity

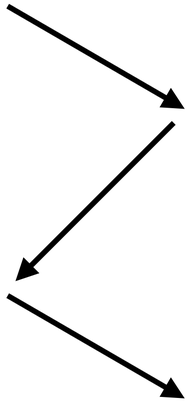
# Communication Complexity

# Communication Complexity



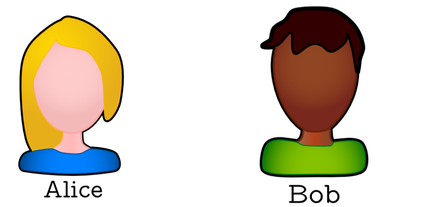




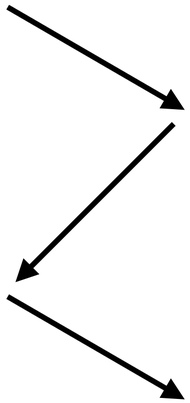



z: (x,y,z) is a solution

# Communication Complexity



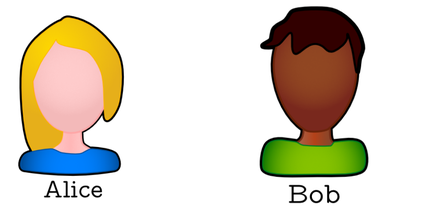




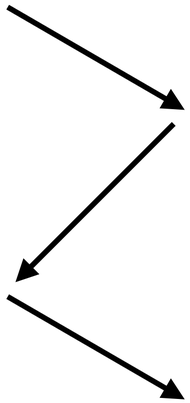
z: (x,y,z) is a solution

Expressive

# Communication Complexity





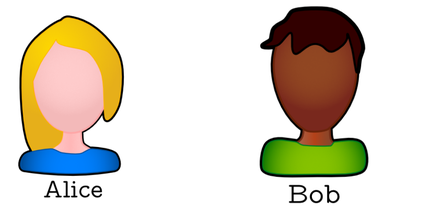


z: (x,y,z) is a solution

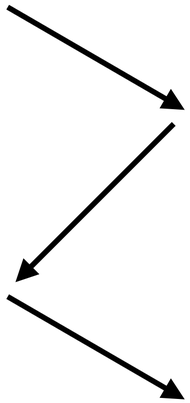
## Expressive

Can simulate models like query complexity, circuits, property testing, streaming and more

# Communication Complexity





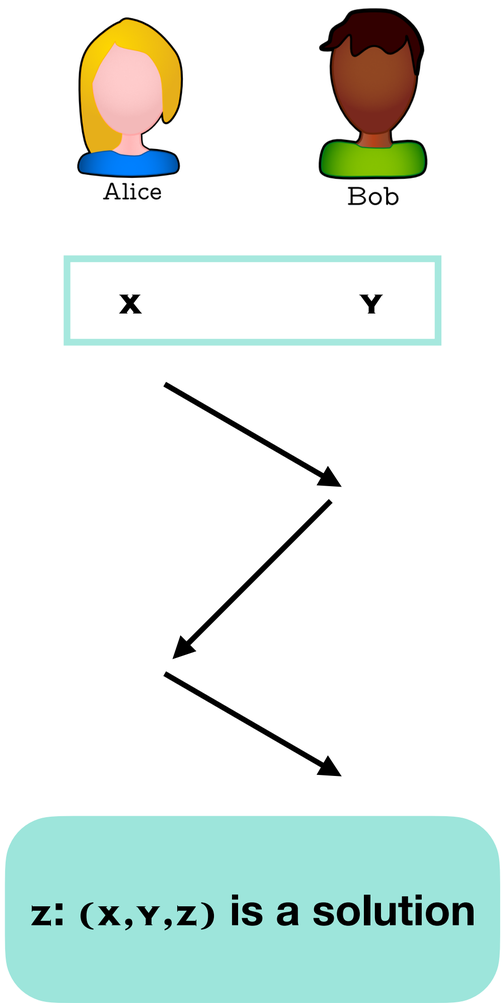


z: (x,y,z) is a solution

## Expressive

Can simulate models like query complexity, circuits, property testing, streaming and more

# Communication Complexity

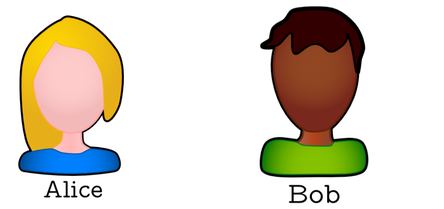



## Expressive

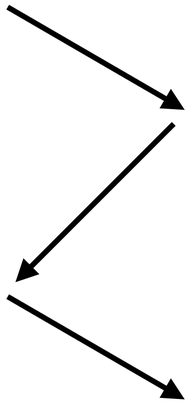
Can simulate models like query complexity, circuits, property testing, streaming and more

## Tractable

# Communication Complexity







z: (x,y,z) is a solution

## Expressive

Can simulate models like query complexity, circuits, property testing, streaming and more

## Tractable

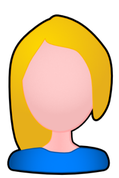
Can prove unconditional lower bounds for problems we care about

“Communication is everything,
everything is communication.”

Folklore

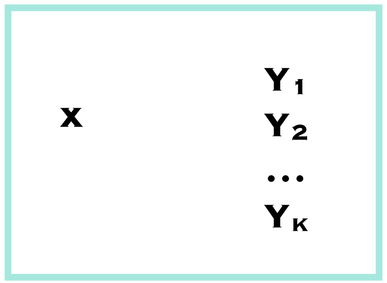
# An example market

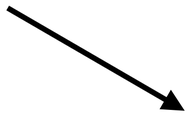
# An example market



Alice

Bob

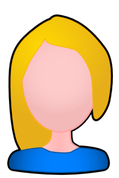


**Solutions for**
$(x,y_1)$, ... $(x,y_K)$

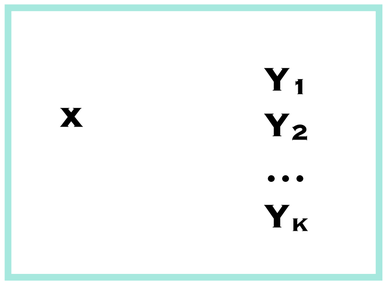
# An example market

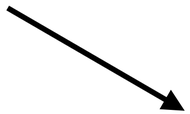
Data



Alice

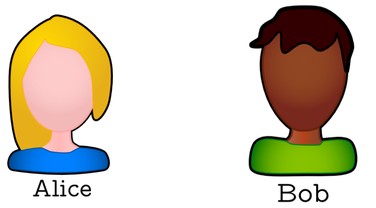
Bob



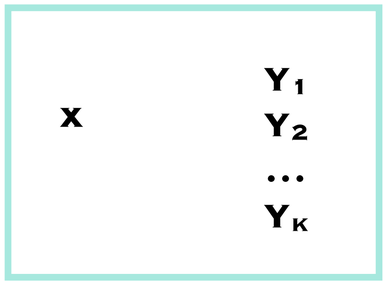
**Solutions for**
$(x,y_1)$, ... $(x,y_K)$

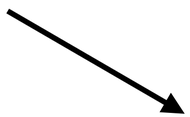
# An example market









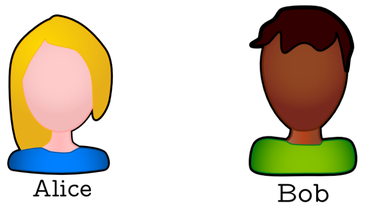



**Solutions for**
$(x,Y_1)$, ... $(x,Y_K)$

## Data

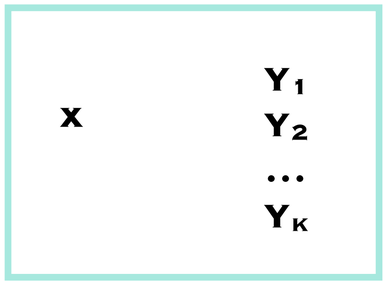
Alice holds some data $x$ which is useful for training models

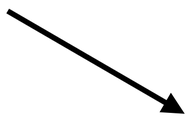
# An example market









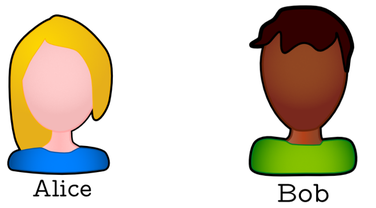



**Solutions for**
$(x,Y_1)$, ... $(x,Y_K)$

## Data

Alice holds some data $x$ which is useful for training models

# An example market




| x | $Y_1$<br>$Y_2$<br>…<br>$Y_K$ |
|---|---|

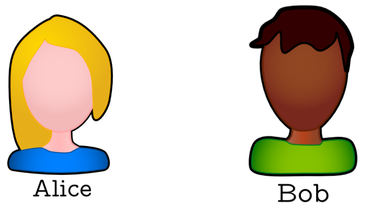
**Solutions for**
$(x,Y_1)$, … $(x,Y_K)$

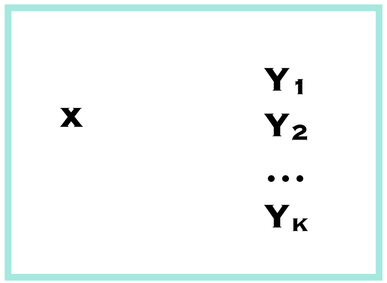
## Data

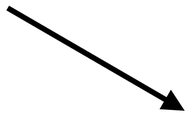
Alice holds some data $x$ which is useful for training models

## Buyer

# An example market



**Solutions for**
$(x,Y_1)$, ... $(x,Y_K)$

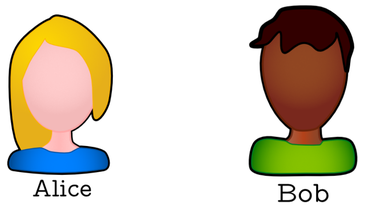
## Data

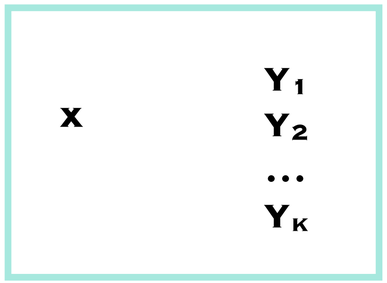
Alice holds some data $x$ which is useful for training models

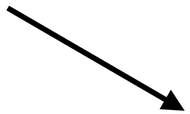
## Buyer

Bob has $K$ machine learning models which he wants to train using Alice's data

# An example market



**Solutions for**
$(x,Y_1)$, ... $(x,Y_K)$

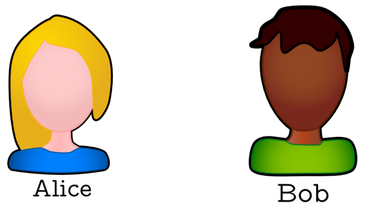
## Data

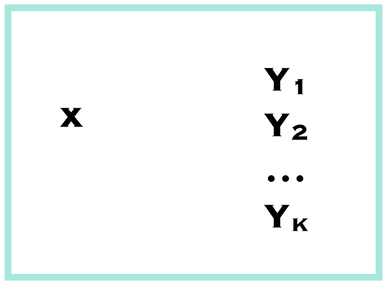
Alice holds some data $x$ which is useful for training models

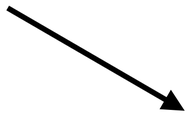
## Buyer

Bob has $K$ machine learning models which he wants to train using Alice's data

# An example market



**Solutions for**
$(x,y_1)$, ... $(x,y_\kappa)$

## Data

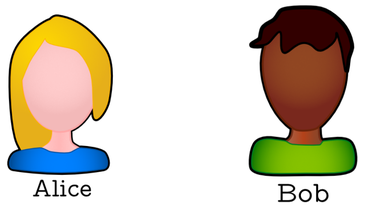
Alice holds some data $x$ which is useful for training models

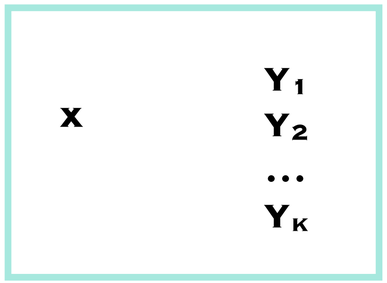
## Buyer

Bob has $\kappa$ machine learning models which he wants to train using Alice's data

## Communication

# An example market




**Solutions for**
$(x,Y_1)$, ... $(x,Y_K)$

## Data

Alice holds some data $x$ which is useful for training models

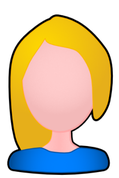
## Buyer

Bob has $K$ machine learning models which he wants to train using Alice's data

## Communication

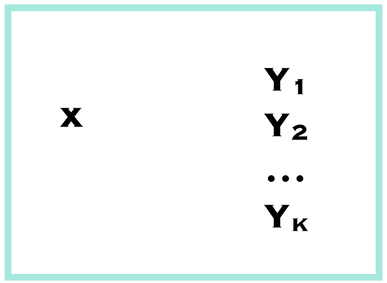
On payment, Alice sends a copy of her data to Bob

# Consumable Data



Alice

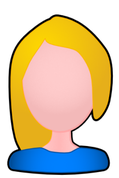
Bob




**Solutions for**
$(x,y_1)$, ... $(x,y_K)$

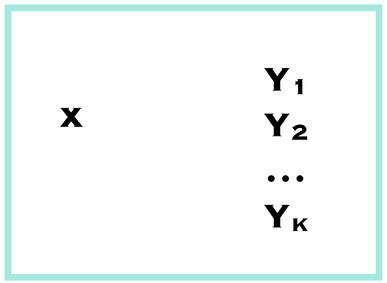
# Consumable Data

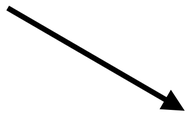
## Asymmetric Direct Sum for One-way Communication



Alice

Bob

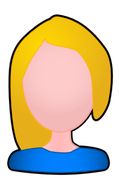


**Solutions for**
$(x,Y_1)$, ... $(x,Y_K)$

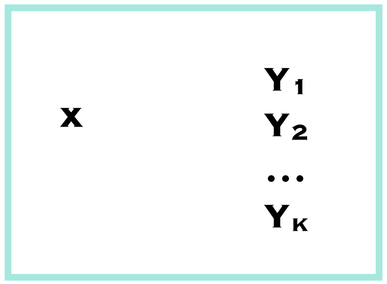
# Consumable Data

## Asymmetric Direct Sum for One-way Communication



Alice

Bob

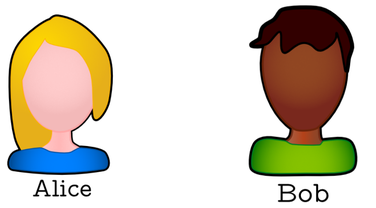



**Solutions for**
$(x,Y_1)$, ... $(x,Y_K)$

Defintion

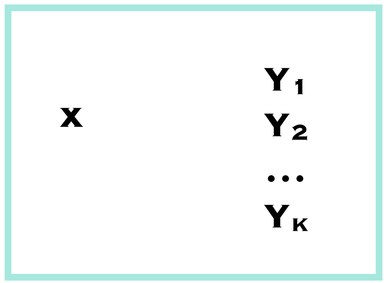
# Consumable Data

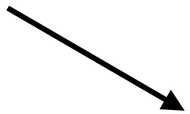
## Asymmetric Direct Sum for One-way Communication









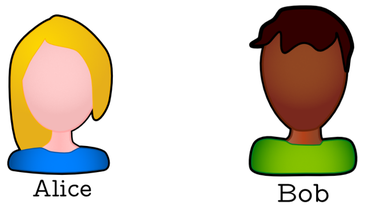
**Solutions for**
$(x,Y_1)$, ... $(x,Y_k)$

### Defintion

$R$ is a consumable data problem if $R^k$ requires $poly(k)\ CC(R)$ bits to be communicated

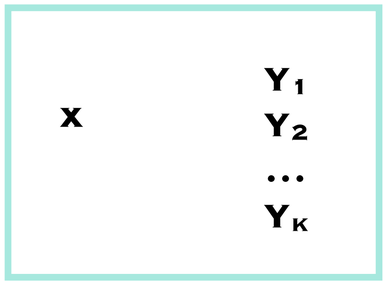
# Consumable Data

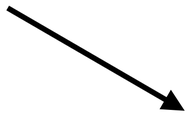
## Asymmetric Direct Sum for One-way Communication









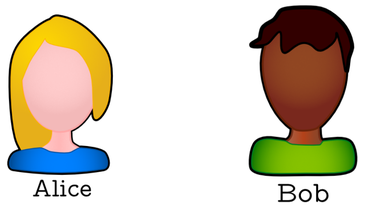
**Solutions for**
$(x,Y_1)$, ... $(x,Y_k)$

### Defintion

$R$ is a consumable data problem if $R^k$ requires $poly(k)\ CC(R)$ bits to be communicated

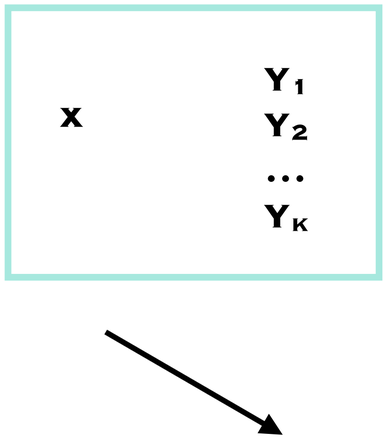
# Consumable Data

## Asymmetric Direct Sum for One-way Communication










**Solutions for**
$(x,Y_1)$, ... $(x,Y_k)$

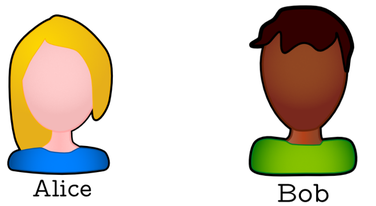
### Defintion

$R$ is a consumable data problem if $R^k$ requires $poly(k)$ $CC(R)$ bits to be communicated

### Strong version

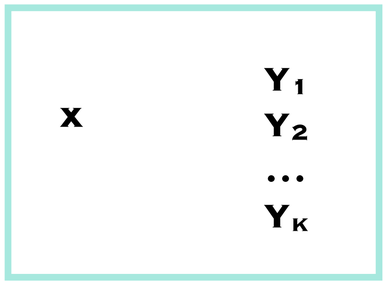
# Consumable Data

## Asymmetric Direct Sum for One-way Communication










**Solutions for**
$(x,Y_1)$, ... $(x,Y_k)$

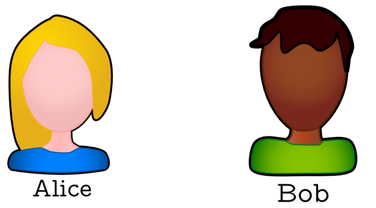
### Defintion

R is a consumable data problem if $R^k$ requires poly(k) CC(R) bits to be communicated

### Strong version

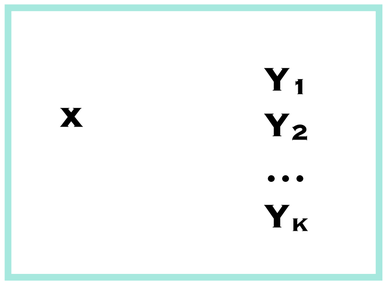
R is a strongly consumable data problem if the same lower bound holds when Bob wants to solve any 2/3 fraction of instances

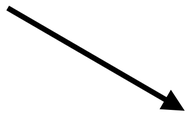
# Our results





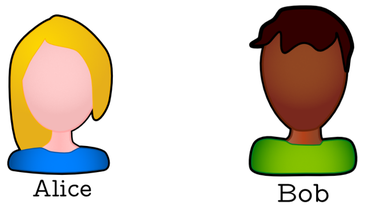




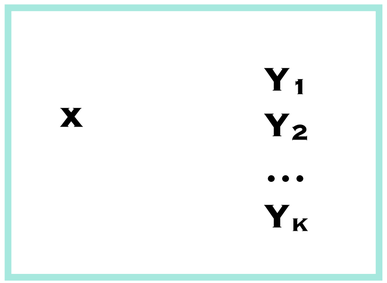
**Solutions for**
$(x,Y_1)$, ... $(x,Y_K)$

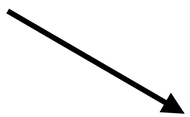
# Our results







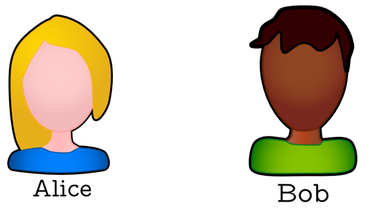


**Solutions for**
$(x,Y_1)$, ... $(x,Y_K)$

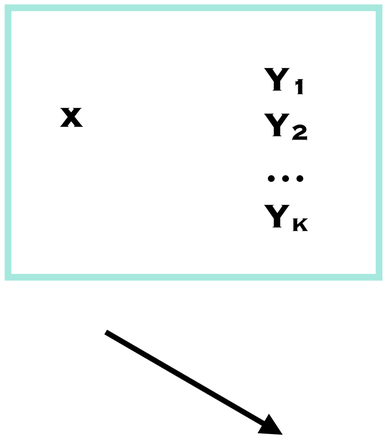
Proof-of-concept examples

# Our results







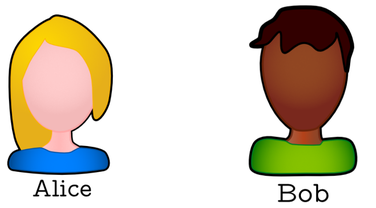



**Solutions for**
**$(x,y_1)$, ... $(x,y_K)$**

## Proof-of-concept examples

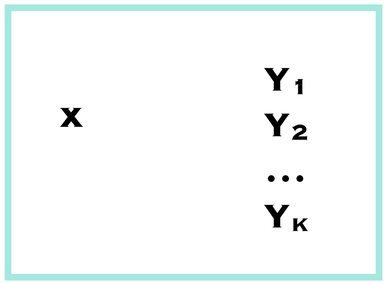
1. Linear Regression Sampling

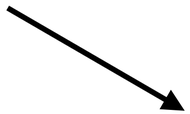
# Our results









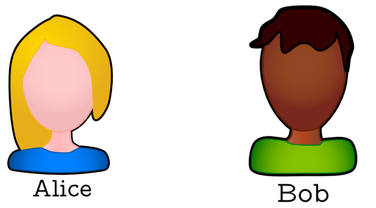



**Solutions for**
**$(x,y_1)$, ... $(x,y_K)$**

## Proof-of-concept examples

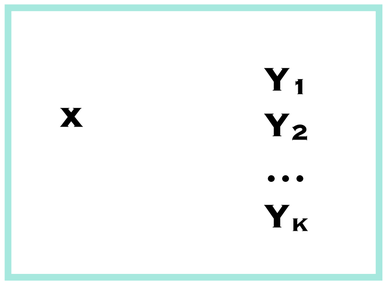
1. Linear Regression Sampling
2. Hidden Matching

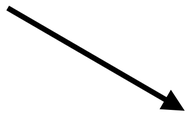
# Our results









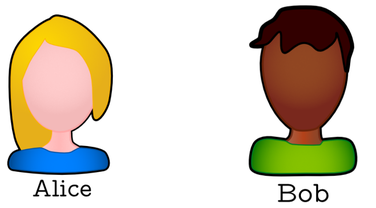



**Solutions for**
**$(x,y_1)$, ... $(x,y_K)$**

## Proof-of-concept examples

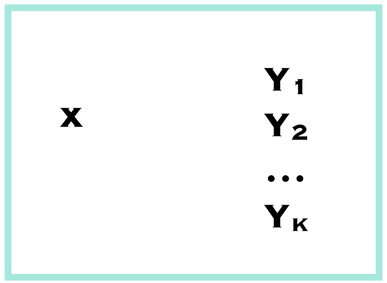
1. Linear Regression Sampling
2. Hidden Matching

# Our results









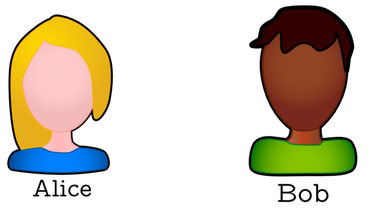

**Solutions for**
$(x,Y_1)$, ... $(x,Y_K)$

## Proof-of-concept examples

1. Linear Regression Sampling
2. Hidden Matching

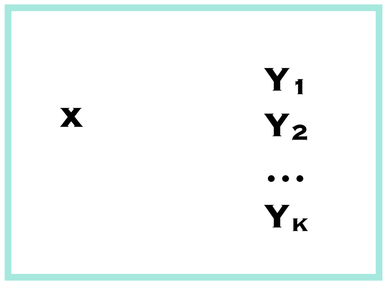
## Impossibility results

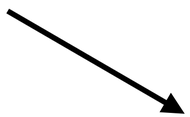
# Our results









**Solutions for**
$(x,Y_1)$, ... $(x,Y_K)$

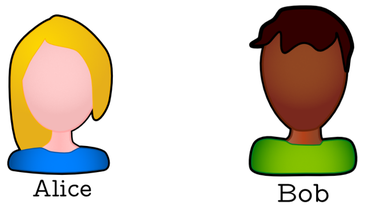
## Proof-of-concept examples

1. Linear Regression Sampling
2. Hidden Matching

## Impossibility results

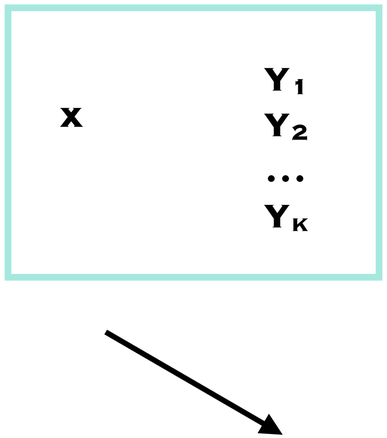
Decision problems

# Our results










**Solutions for**
**$(x,y_1)$, ... $(x,y_K)$**

## Proof-of-concept examples

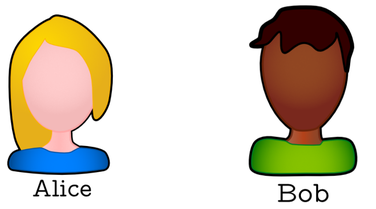
1. Linear Regression Sampling
2. Hidden Matching

Linear

## Impossibility results

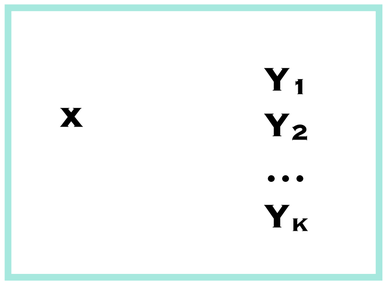
Decision problems

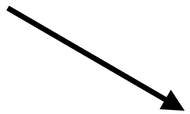
# Our results









**Solutions for**
**$(x,y_1)$, ... $(x,y_K)$**

## Proof-of-concept examples

1. Linear Regression Sampling
2. Hidden Matching

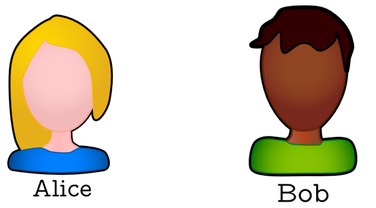
Linear

Square-root

## Impossibility results

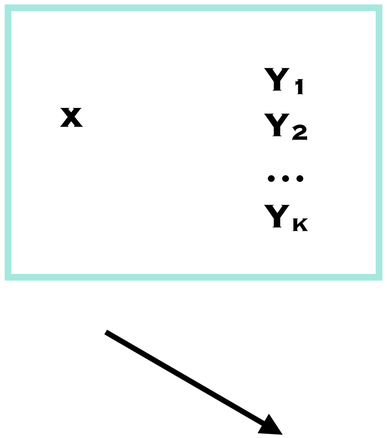
Decision problems

# Impossibility for decision problems



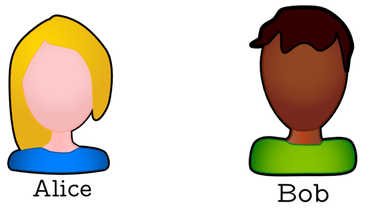






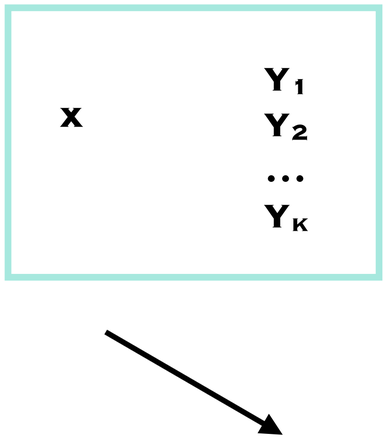

**Solutions for**
$(x,Y_1)$, ... $(x,Y_K)$

# Impossibility for decision problems





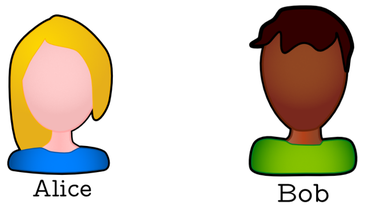





**Solutions for**
$(x,Y_1)$, ... $(x,Y_K)$

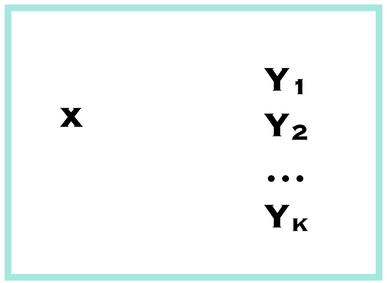
Key tool: Shadow Tomography

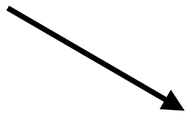
# Impossibility for decision problems









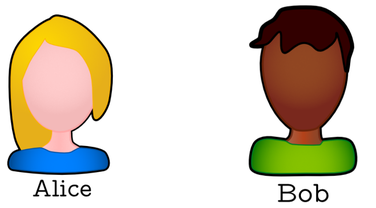



**Solutions for**
$(x,y_1)$, ... $(x,y_K)$

## Key tool: Shadow Tomography

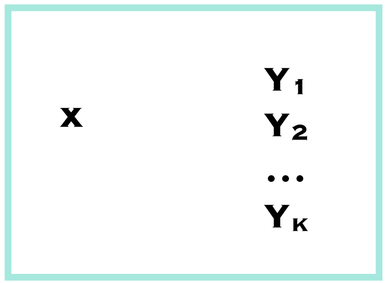
Introduced by Aaronson (2017), allows us to estimate the values of $K$ two-outcome observables applied to a $N$ qubit state using only $\text{POLYLOG}(N,K)$ samples

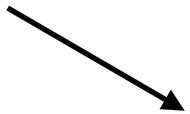
# Impossibility for decision problems









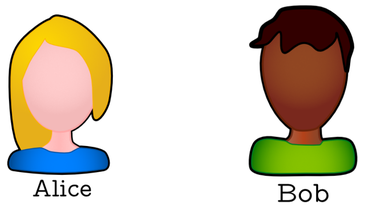



**Solutions for**
$(x,y_1)$, ... $(x,y_K)$

## Key tool: Shadow Tomography

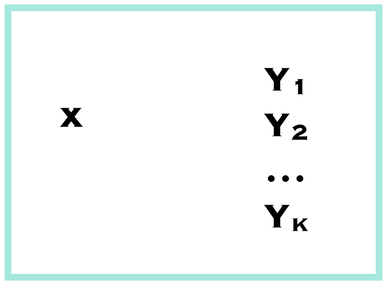
Introduced by Aaronson (2017), allows us to estimate the values of $K$ two-outcome observables applied to a $N$ qubit state using only $\text{POLYLOG}(N,K)$ samples

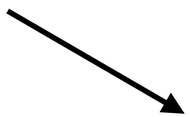
# Impossibility for decision problems









**Solutions for**
$(x,y_1)$, ... $(x,y_K)$

## Key tool: Shadow Tomography

Introduced by Aaronson (2017), allows us to estimate the values of $K$ two-outcome observables applied to a $N$ qubit state using only $\text{POLYLOG}(N,K)$ samples

This gives rise to an easy communication protocol

# Application: A fair data auction

# Posted price data auction

# Posted price data auction

$$V_A = PB$$

# Posted price data auction

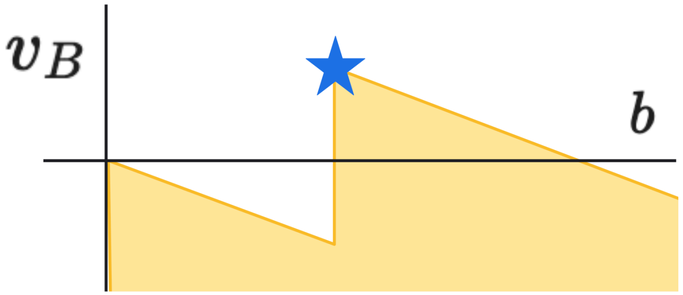
$$V_A = PB$$

$$V_B = S(K,B) - PB$$

# Posted price data auction

$$V_A = PB$$

$$V_B = S(K,B) - PB$$



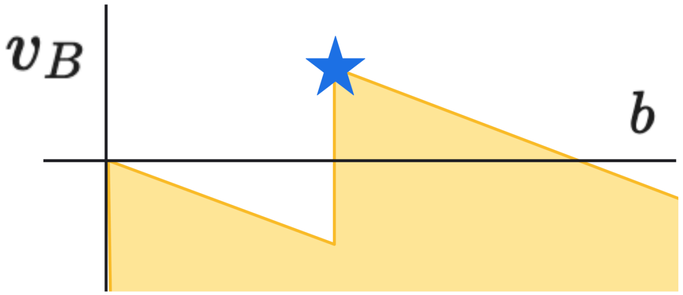

Classical

- $B^*$ is independent of $K$
- Alice doesn't know $K$
- Alice's payoff is $O(1)$

# Posted price data auction

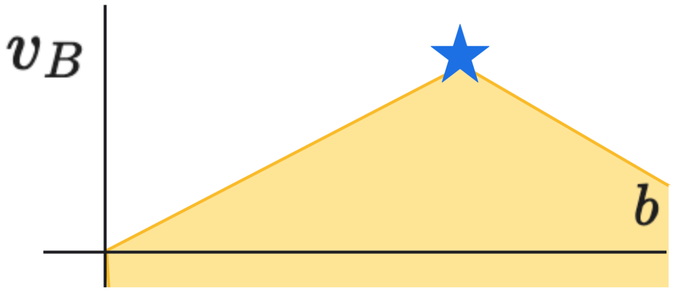
$$V_A = PB$$

$$V_B = S(K,B) - PB$$




## Classical

- $B^*$ is independent of $K$
- Alice doesn't know $K$
- Alice's payoff is $O(1)$



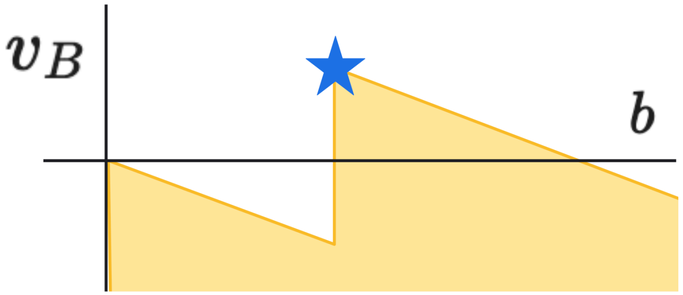

## Quantum

- $B^*$ scales linearly with $K$
- Alice's payoff is $O(K)$

# Posted price data auction

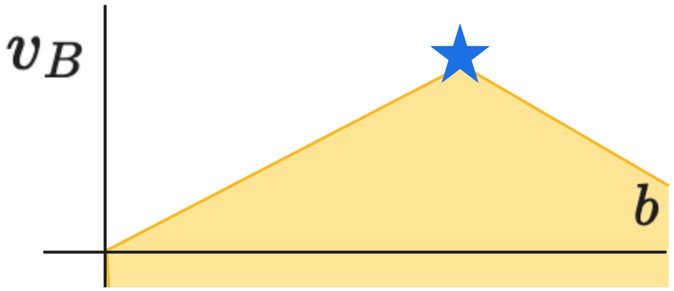
$$V_A = PB$$

$$V_B = S(K,B) - PB$$




## Classical

- $B^*$ is independent of $K$
- Alice doesn't know $K$
- Alice's payoff is $O(1)$




## Quantum

- $B^*$ scales linearly with $K$
- Alice's payoff is $O(K)$

## Takeaway

When using quantum communication, Alice's payoff is proportional to the number of times $x$ is used to generate solutions by Bob.

# Future work

# Open problems

# Open problems

- Can a non-cooperative communication model be used to get better consumable data properties information-theoretically?

# Open problems

- Can a non-cooperative communication model be used to get better consumable data properties information-theoretically?

We need new lower-bound techniques

# Open problems

- Can a non-cooperative communication model be used to get better consumable data properties information-theoretically?

We need new lower-bound techniques

- Can we use cryptographic primitives to create a more generic protocol?

# Open problems

- Can a non-cooperative communication model be used to get better consumable data properties information-theoretically?

We need new lower-bound techniques

- Can we use cryptographic primitives to create a more generic protocol?

Reminiscent of one-time programs

# Open problems

- Can a non-cooperative communication model be used to get better consumable data properties information-theoretically?

We need new lower-bound techniques

- Can we use cryptographic primitives to create a more generic protocol?

Reminiscent of one-time programs

- Can the lower bound for Hidden Matching be improved to linear in $\kappa$?

# Open problems

- Can a non-cooperative communication model be used to get better consumable data properties information-theoretically?

We need new lower-bound techniques

- Can we use cryptographic primitives to create a more generic protocol?

Reminiscent of one-time programs

- Can the lower bound for Hidden Matching be improved to linear in $\kappa$?

Proof needs to avoid classical upper bound when $\kappa > \sqrt{N}$

# Thanks for listening!

Au revoir