ACM SIGMOD/PODS

Chicago, IL, USA

May 15, 2017

A Relational Framework for Classifier Engineering

Benny Kimelfeld

Technion, Israel

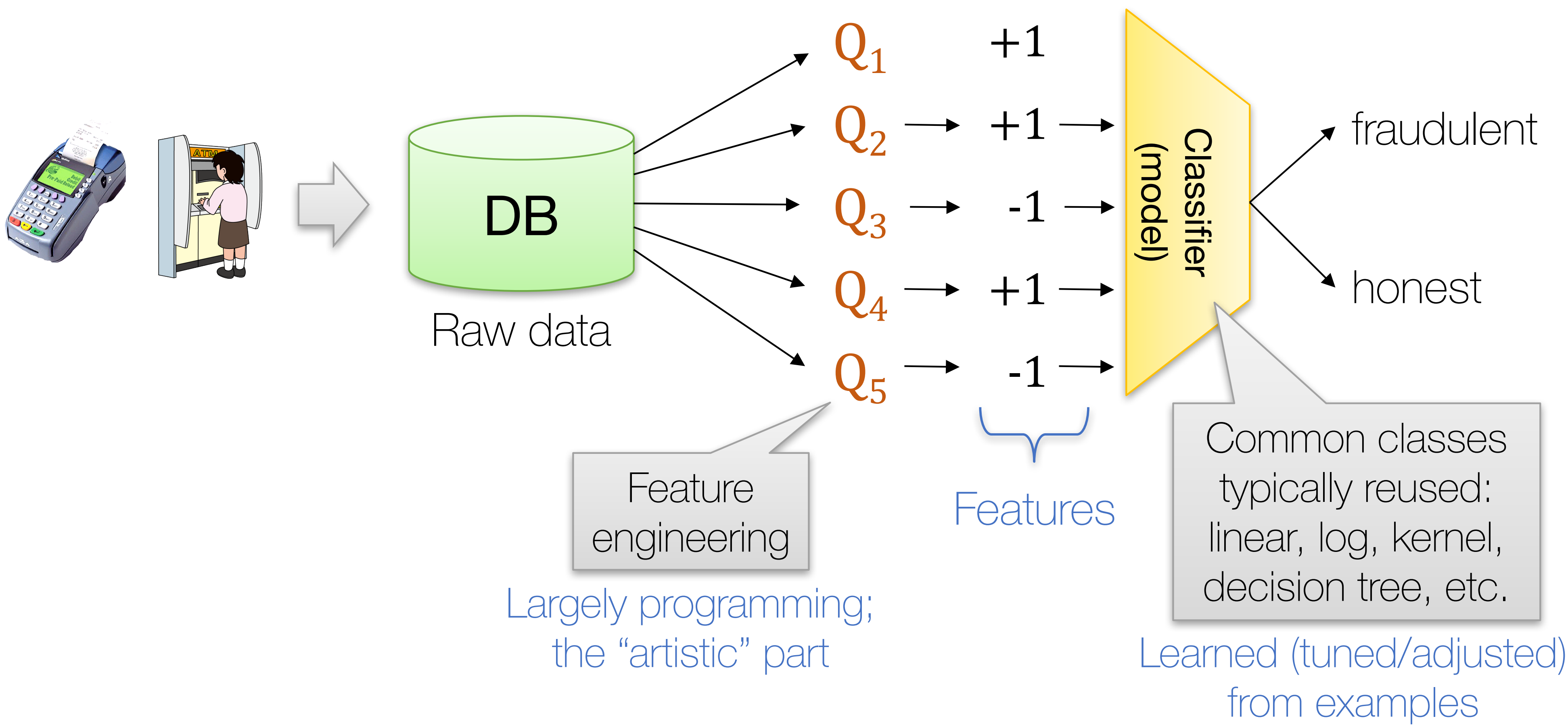
Christopher Ré

Stanford University

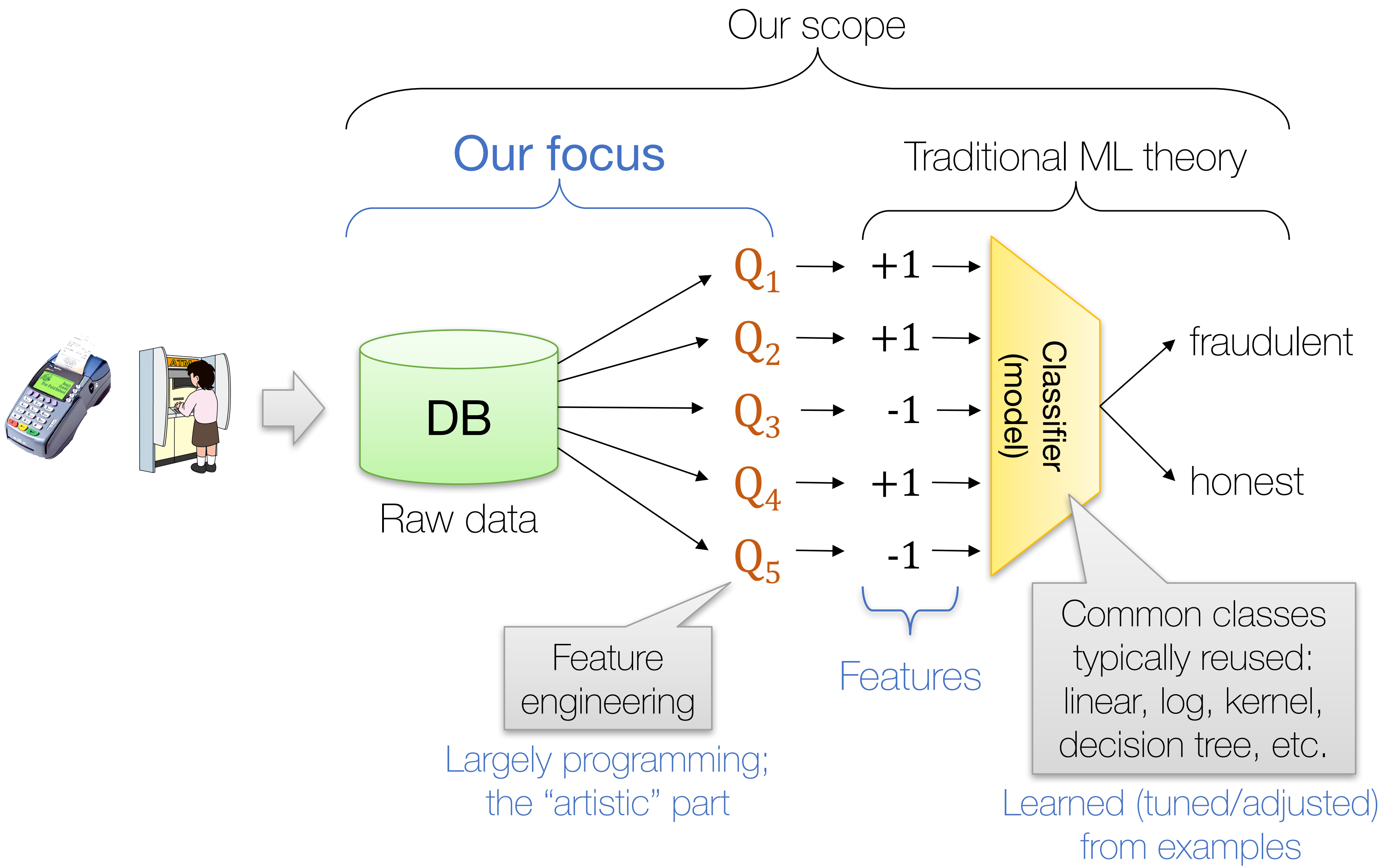
Background

- ML application constantly increasing
  - e.g., by 2020 >50% Intel servers will run ML (D. Bryant, Intel SVP)
- Rising interest in DB research for ML
  - e.g., query optimization for feature selection / evaluation [Zhang+14, Kumar+15,16], ML on factorized DB [Schleich+16]
  - DEEM workshop on *Data Management for End-to-End ML*
  - Dagstuhl Presp. Workshop 16151: *Research Directions for PDM*
- Feature Engineering (FE) critical for quality
  - Yet heavy resource consumer in ML development
  - Tooling and principles [Guyon+06 book]
  - Standard practice; here to stay!
    - *Deep Learning* avoids FE; applicable in certain areas / domains w/ massive training data available

Classic ML Classification Flow



Classic ML Classification Flow



Framework Goal

- DB “understands” how *entities* become *features*
  - Relational structure, constraints, queries
- Can be used for assisting FE?
  - Estimate feature quality?
  - Suggest new features?
  - Test for suitability of a feature language?
  - Detect engineering faults?
  - Implication of underlying languages on computational complexity?
  - Benefit from decades of DB theory?
- Setup for attacking questions
- Step towards DB theory for ML engineering

Outline

- Formal Setup
- Computational Problems
- Complexity Results
- Directions

Scope

- ML task: **binary classification**
  - Learn a mapping entity ⟶ +1/-1
- Boolean features
  - Simplifies the framework
  - Common in practice
    - e.g., *binning / bucketing*
- Hence, a *classifier* has the form

$$\mathbf{C} : \{+1,-1\}^n \longrightarrow \{+1,-1\}$$

Txn

id
1
2
3
4

TxnInfo

TXN	card	country	state
1	100	US	GA
2	100	US	NY
3	101	BR	RJ
4	102	US	CA

Card

id	ssn	country	state
100	200	US	GA
101	201	US	NY
102	202	BR	RJ

Txn

id	λ(e)
1	-1
2	+1
3	-1
4	-1

TxnInfo

TXN	card	country	state
1	100	US	GA
2	100	US	NY
3	101	BR	RJ
4	102	US	CA

Card

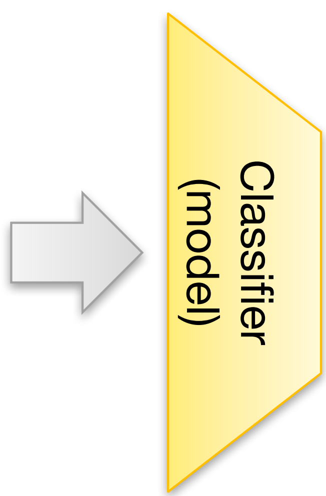
id	ssn	country	state
100	200	US	GA
101	201	US	NY
102	202	BR	RJ

(txn in owner's state) $Q_1(x) \leftarrow \mathrm{TxnInfo}(x, n, c, s), \mathrm{Card}(n, c, s)$

(txn in owner's country) $Q_2(x) \leftarrow \mathrm{TxnInfo}(x, n, c, s), \mathrm{Card}(n, d, s)$

(txn in NY) $Q_3(x) \leftarrow \mathrm{TxnInfo}(x, n, c, \text{'NY'})$

+1	+1	-1	-1
-1	+1	+1	+1
-1	-1	-1	-1
-1	-1	-1	-1



Formal Setup

- Entity schema: $(\mathbf{S},\eta)$
  - $\mathbf{S}$ is a relational schema (signature, constraints)
  - η is a unary relation in $\mathbf{S}$, representing *entities*
- An instance $\mathbf{I}$ of $\mathbf{S}$ defines:
  - An entity set η^{I} (the η relation of $\mathbf{I}$)
  - Information on the entities (all other relations)
- Feature query: unary query $\mathbf{Q}$ over $\mathbf{S}$
- Statistic: series $\mathbf{\Pi}=(\mathrm{Q}_1,\ldots,\mathrm{Q}_n)$ of feature queries
- Each $\mathrm{e} \in \eta^{\mathrm{I}}$ has a feature vector $\mathbf{\Pi}(\mathrm{e}) = (\mathrm{f}_1,\ldots,\mathrm{f}_n)$

$$\mathrm{f_i} = \begin{cases} +1 & \text{if } \mathrm{e} \in \mathrm{Q_i}(\mathrm{I}) \\ -1 & \text{if } \mathrm{e} \notin \mathrm{Q_i}(\mathrm{I}) \end{cases}$$

S

Txn

id
1
2
3
4

η

TxnInfo

TXN	card	country	state
1	100	US	GA
2	100	US	NY
3	101	BR	RJ
4	102	US	CA

Card

id	ssn	country	state
100	200	US	GA
101	201	US	NY
102	202	BR	RJ

(txn in owner's state) $Q_1(x) \leftarrow \mathrm{TxnInfo}(x, n, c, s), \mathrm{Card}(n, c, s)$

(txn in owner's country) $Q_2(x) \leftarrow \mathrm{TxnInfo}(x, n, c, s), \mathrm{Card}(n, d, s)$

(txn in NY) $Q_3(x) \leftarrow \mathrm{TxnInfo}(x, n, c, \text{'NY'})$

Feature queries

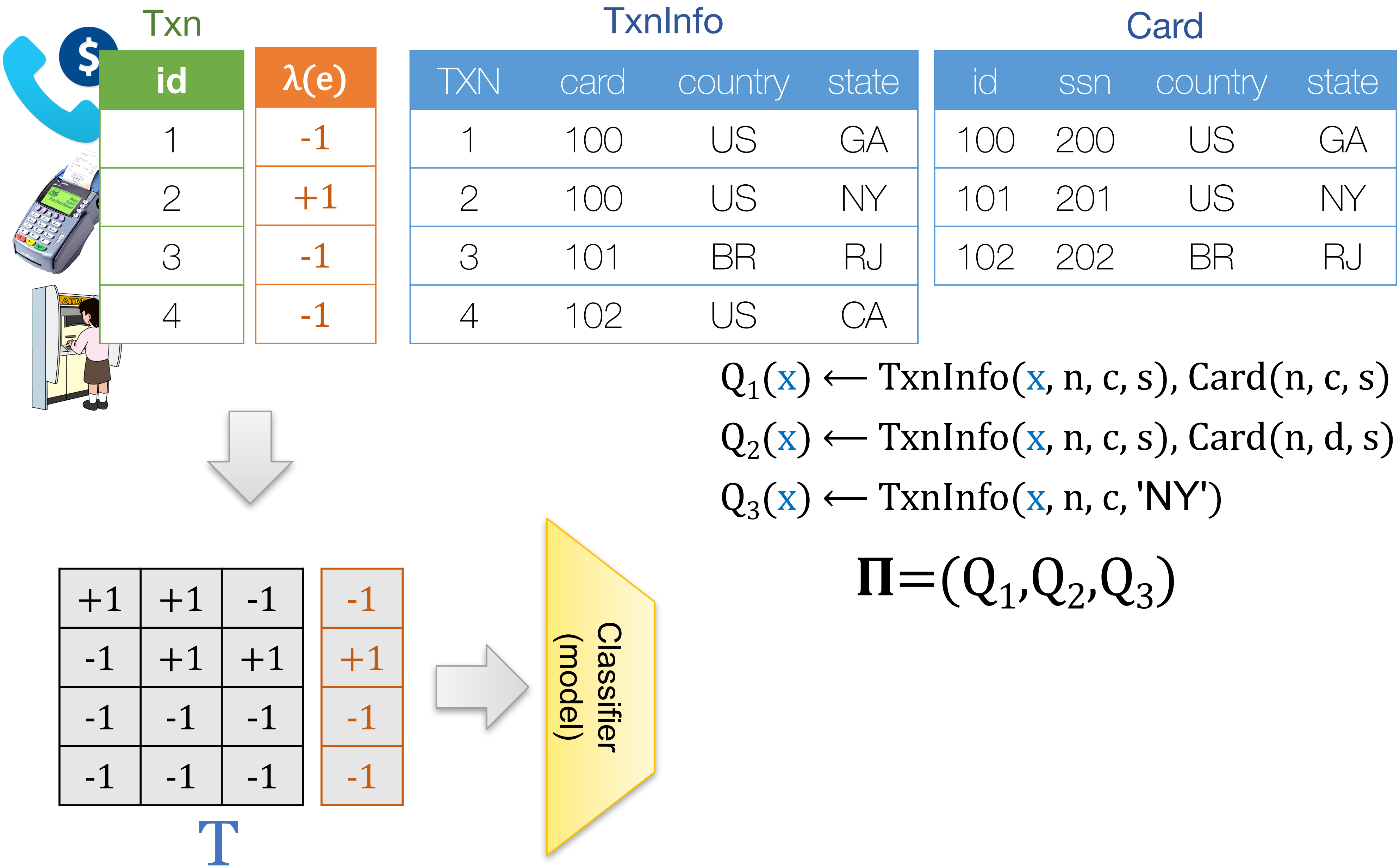
Statistic: $\mathbf{\Pi} = (Q_1, Q_2, Q_3)$

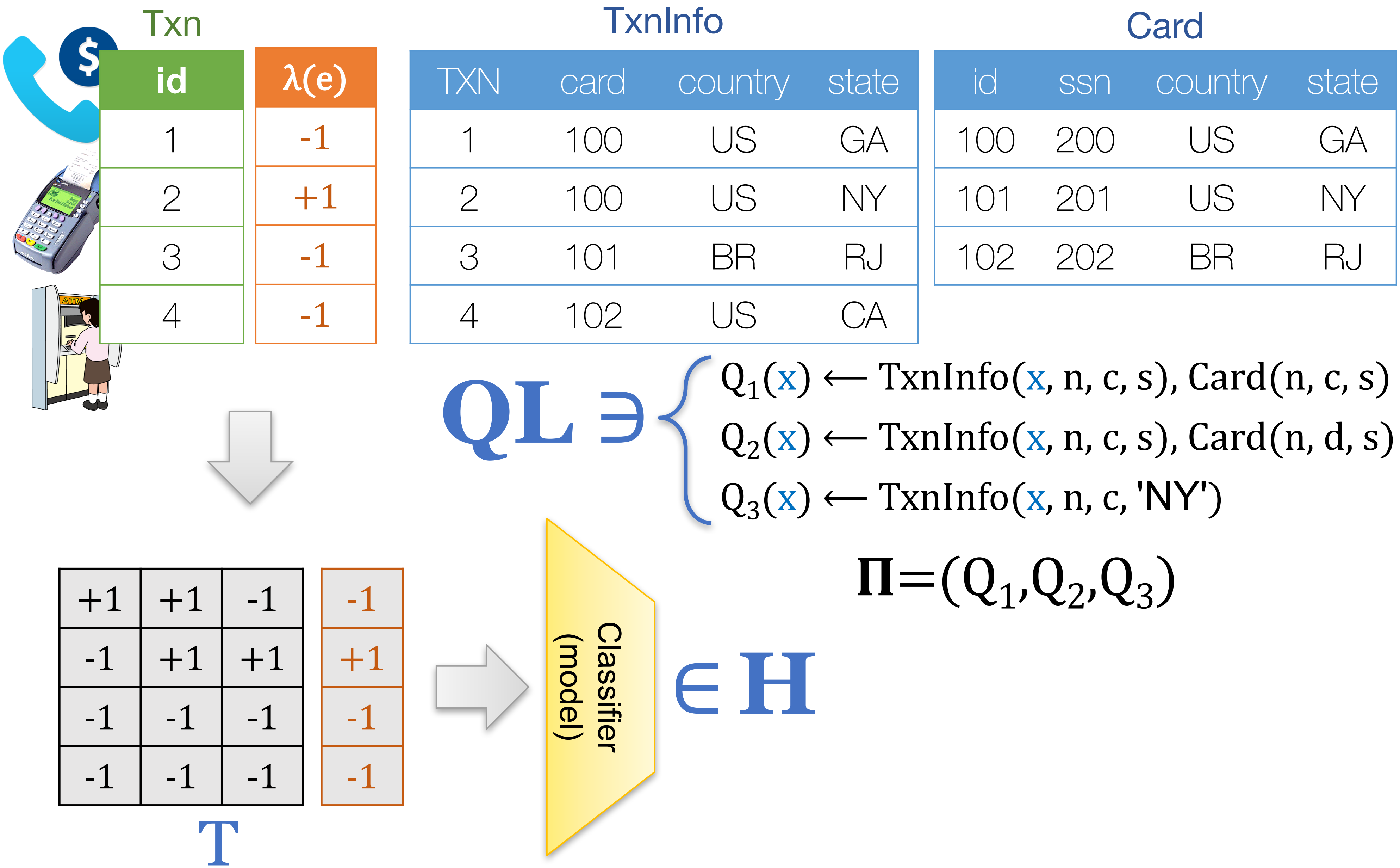
Training

- Let $(\mathbf{S},\eta)$ be an entity schema
- A *training instance* is a pair (I,λ) where
  - I is an instance over $\mathbf{S}$
  - $\lambda\colon \eta^{\mathrm{I}} \longrightarrow \{+1,-1\}$ is a labeling function
- (I,λ) + statistic $\mathbf{\Pi}$ define the training collection

$$\mathrm{T}=\{\,\langle \mathbf{\Pi}(e),\lambda(e)\rangle \mid e \in \eta^{\mathrm{I}}\,\}$$

- Training finds a classifier from a *hypothesis class* $\mathbf{H}$ by minimizing a *risk function* over T


Txn
id
1
2
3
4
λ(e)
-1
+1
-1
-1
TxnInfo
TXN card country state
1 100 US GA
2 100 US NY
3 101 BR RJ
4 102 US CA
Card
id ssn country state
100 200 US GA
101 201 US NY
102 202 BR RJ
$Q_1(x) \longleftarrow \mathrm{TxnInfo}(x, n, c, s), \mathrm{Card}(n, c, s)$
$Q_2(x) \longleftarrow \mathrm{TxnInfo}(x, n, c, s), \mathrm{Card}(n, d, s)$
$Q_3(x) \longleftarrow \mathrm{TxnInfo}(x, n, c, \text{'NY'})$
$\Pi = (Q_1, Q_2, Q_3)$
+1 +1 -1
-1 +1 +1
-1 -1 -1
-1 -1 -1
-1
+1
-1
-1
T
Classifier
(model)


Txn
id
1
2
3
4
λ(e)
-1
+1
-1
-1
TxnInfo
TXN card country state
1 100 US GA
2 100 US NY
3 101 BR RJ
4 102 US CA
Card
id ssn country state
100 200 US GA
101 201 US NY
102 202 BR RJ
$\mathbf{QL} \ni \{ Q_1(x) \leftarrow \mathrm{TxnInfo}(x, n, c, s), \mathrm{Card}(n, c, s)$
$Q_2(x) \leftarrow \mathrm{TxnInfo}(x, n, c, s), \mathrm{Card}(n, d, s)$
$Q_3(x) \leftarrow \mathrm{TxnInfo}(x, n, c, \text{'NY'})$
$\mathbf{\Pi}=(Q_1,Q_2,Q_3)$
+1 +1 -1
-1 +1 +1
-1 -1 -1
-1 -1 -1
-1
+1
-1
-1
T
Classifier
(model)
$\in \mathbf{H}$

Outline

- Formal Setup
- Computational Problems
- Complexity Results
- Directions

Txn

id	$\lambda(e)$
1	-1
2	+1
3	-1
4	-1

TxnInfo

TXN	card	country	state
1	100	US	GA
2	100	US	NY
3	101	BR	RJ
4	102	US	CA

Card

id	ssn	country	state
100	200	US	GA
101	201	US	NY
102	202	BR	RJ

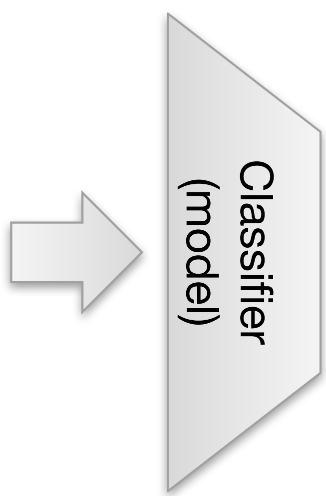
(txn in owner's state) $Q_1(x) \leftarrow \mathrm{TxnInfo}(x, n, c, s), \mathrm{Card}(n, c, s)$

(txn in owner's country) $Q_2(x) \leftarrow \mathrm{TxnInfo}(x, n, c, s), \mathrm{Card}(n, d, s)$

(txn in NY) $Q_3(x) \leftarrow \mathrm{TxnInfo}(x, n, c, \text{'NY'})$

+1	+1	-1
-1	+1	+1
-1	-1	-1
-1	-1	-1

-1
+1
-1
-1



Can we fit feature queries & classifier of chosen families?

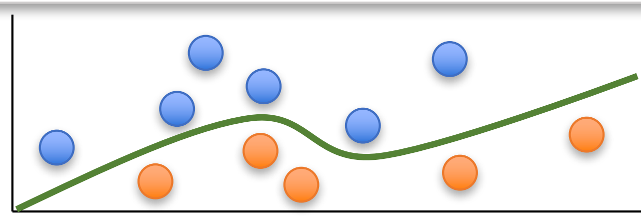
Problem 1: Separability

The naïve "noise-free" training from ML textbooks:

Is full separation possible?

($\mathbf{H}$,$\mathbf{QL}$)-separablity

Given a training instance $(\mathbf{I},\lambda)$ over a schema $(\mathbf{S},\eta)$, is there any statistic $\mathbf{\Pi}$ in $\mathbf{QL}$ such that $(\mathbf{I},\lambda)$ can be perfectly realized by a classifier in $\mathbf{H}$?



Txn

id	$\lambda(e)$
1	-1
2	+1
3	-1
4	-1

TxnInfo

TXN	card	country	state
1	100	US	GA
2	100	US	NY
3	101	BR	RJ
4	102	US	CA

Card

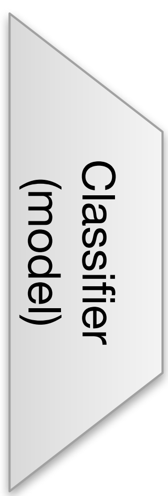
id	ssn	country	state
100	200	US	GA
101	201	US	NY
102	202	BR	RJ

$$Q_1(x) \longleftarrow \mathrm{TxnInfo}(x, n, c, s), \mathrm{Card}(n, c, s)$$
$$Q_2(x) \longleftarrow \mathrm{TxnInfo}(x, n, c, s), \mathrm{Card}(n, d, s)$$
$$Q_3(x) \longleftarrow \mathrm{TxnInfo}(x, n, c, \text{'NY'})$$

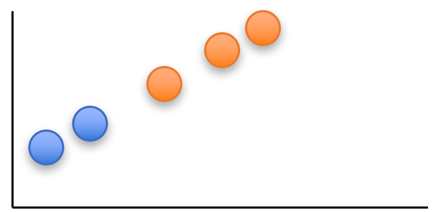
Redundancy in features?

+1	+1	-1
-1	+1	+1
-1	-1	-1
-1	-1	-1

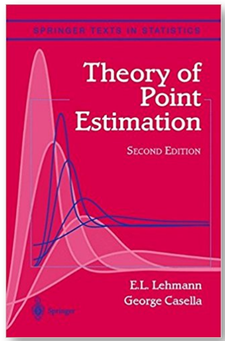
-1
+1
-1
-1



Redundancy / Identifiablity



- Linear column dependence in the feature matrix often means redundant features
  - e.g., linear/logistic classification/regression
- ML libraries often require full column rank
  - For their optimization solution to be "identifiable"
  - c.f. "Theory of Point Estimation" [LehmannCasella83]



TXN in owner's US state	TXN in different US state	TXN in East Coast	TXN US but not East Coast
+1	-1	+1	-1
-1	+1	-1	+1
-1	-1	-1	-1

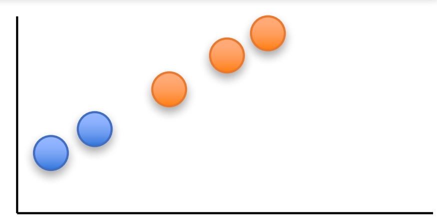
TXN in owner's US state	TXN in different US state	TXN in East Coast	TXN US but not East Coast
+1	-1	+1	-1
-1	+1	-1	+1
-1	-1	-1	-1

sum = sum

Problem 2: Identifiability

QL-identifiability

Given a statistic $\mathbf{\Pi}$ in **QL** over entity schema $(\mathbf{S},\eta)$, is there any instance $\mathbf{I}$ with a column-independent feature matrix?



Two variants:

- Linear independence (arises in, e.g., least-square minimization)
- Affine independence (arises in, e.g., entropy minimization)

Txn

id	$\lambda(e)$
1	-1
2	+1
3	-1
4	-1

TxnInfo

TXN	card	country	state
1	100	US	GA
2	100	US	NY
3	101	BR	RJ
4	102	US	CA

Card

id	ssn	country	state
100	200	US	GA
101	201	US	NY
102	202	BR	RJ

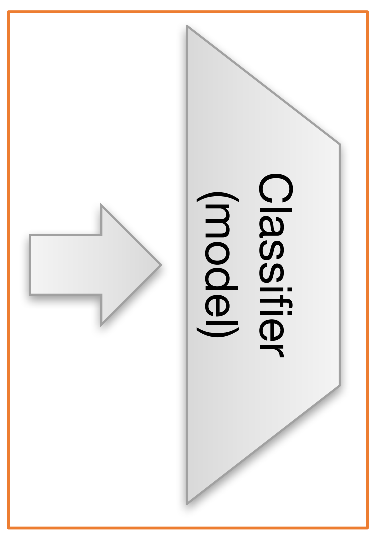
(txn in owner's state) $Q_1(x) \leftarrow \mathrm{TxnInfo}(x, n, c, s), \mathrm{Card}(n, c, s)$

(txn in owner's country) $Q_2(x) \leftarrow \mathrm{TxnInfo}(x, n, c, s), \mathrm{Card}(n, d, s)$

(txn in NY) $Q_3(x) \leftarrow \mathrm{TxnInfo}(x, n, c, \text{'NY'})$

+1	+1	-1
-1	+1	+1
-1	-1	-1
-1	-1	-1

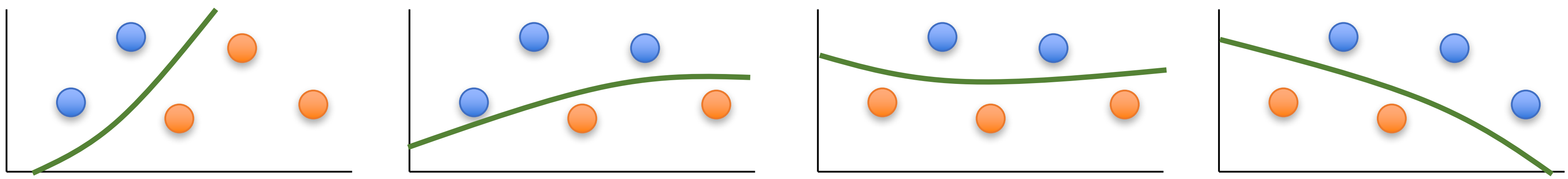
-1
+1
-1
-1



How much training to learn over our features?

Vapnik-Chervonenkis (VC) Dimension

- *What is the max #entities that can be shattered*
  - That is, perfectly classified on every possible labeling?

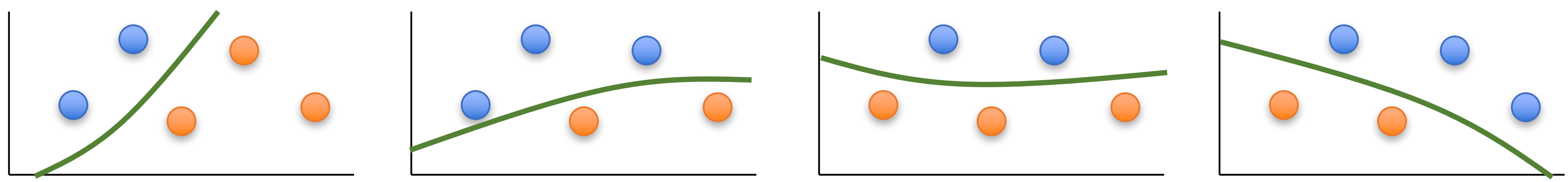


- Complexity measure for learnability
  - (not the only one)
- Estimate training amount to avoid overfitting

Problem 3: Dimensionality

($\mathbf{H}$,$\mathbf{QL}$)-dimensionality

Given a statistic $\mathbf{\Pi}$ in $\mathbf{QL}$ over an entity schema $(\mathbf{S},\eta)$, what is the max $\mathbf{m}$ such that some instance with $\mathbf{m}$ entities can be shattered by $\mathbf{H}$?



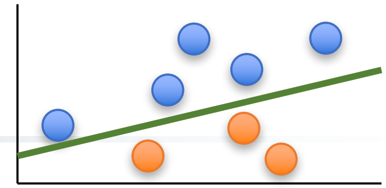
Outline

- Formal Setup
- Computational Problems
- Complexity Results
- Directions

Scope of Results

- Complexity analysis in a specific setting:
  - Hypothesis class $\mathbf{H} = \textbf{Lin}$: linear classifiers
  - Query language $\mathbf{QL} = \textbf{CQ}$: conjunctive queries
    - Without constants
  - No schema constraints
- Mostly intractable complexity classes (expected)
- Baseline & justification for future assumptions
- Next, a few highlights

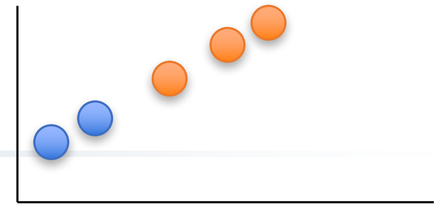
(Lin,CQ)-Separability



Given a training instance $(\mathbf{I},\lambda)$ over a schema $(\mathbf{S},\eta)$, is there any statistic $\mathbf{\Pi}$ in **CQ** such that $(\mathbf{I},\lambda)$ can be perfectly realized by a classifier in **Lin**?

- Every training instance is separable, unless entities with different labels are indistinguishable by CQs
  - That is, there are $\mathbf{e}$ and $\mathbf{e'}$ with $\lambda(\mathbf{e})\neq\lambda(\mathbf{e'})$ and endomorphism that maps $\mathbf{e}$ and $\mathbf{e'}$ and vice versa
  - Relationship to *CQ-query-by-example*
    - [Willard10,tenCateDalmau15,BarcelóRomero16]
  - coNP-complete
- Avoiding self joins ⟶ **harder**: Σ_2^P-complete!

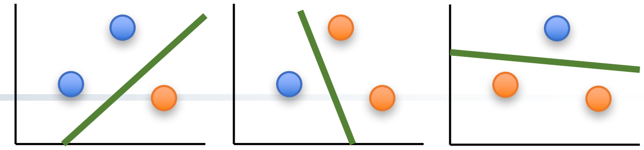
CQ-Identifiability



Given a statistic $\mathbf{\Pi}$ in **CQ** over entity schema $(\mathbf{S},\eta)$, is there any instance $\mathbf{I}$ with a column-independent feature matrix?

- The following are equivalent if CQs are connected:
  - $\mathbf{\Pi}$ is linearly identifiable
  - $\mathbf{\Pi}$ is affinely identifiable
  - $\mathbf{\Pi}$ is non-redundant (no equivalent feature queries)
- Pairwise equivalences break if:
  - CQs can be disconnected
  - CQs can have negation
- Generalized characterization for disconnected CQs
- coNP-complete

(Lin,CQ)-Dimensionality



Given a statistic $\mathbf{\Pi}$ in **CQ** over entity schema $(\mathbf{S},\eta)$, what is the max $\mathbf{m}$ such that some instance with $\mathbf{m}$ entities can be shattered by **Lin**?

- For connected CQs VC dim w.r.t. $\mathbf{\Pi}$ is $\mathbf{d+1}$

  $\mathbf{d}$ = #(equivalence classes among CQs in $\mathbf{\Pi}$)

  – In particular, containment among CQs does not reduce the VC dimension compared to vanilla linear classification

- Can go down if we allow:
  – Disconnected CQs
  – Negation

Outline

- Formal Setup
- Computational Problems
- Complexity Results
- Directions

Directions for Future Research

- Schema constraints
- Generalized features / tasks
  - Numeric, aggregate, multi-label, regression
- Realistic variants of separability
  - Approximate/noisy, incremental
- Restrict model complexity
  - Small/shallow feature queries, low statistic dimension
- Connection to prob. DBs (statistical guarantees?)
- Context of text analysis
  - Doc. spanners [Fagin+2014], DeepDive [Shin+2015]
- …

Summary

- Framework for classifier engineering over DBs
  - Entity schema, feature query, statistic, training instance
- Goal: DB smartness (schema, constraints, queries) to aid feature engineering
- Illustrated on several computational problems
  - Separability, dimensionality, identifiability
  - Preliminary results for linear classifiers and CQs
- Plethora of problems / directions to pursue

Thank you! Questions?