VC-Dimension and Rademacher Averages: From Statistical Learning Theory to Sampling Algorithms

Matteo Riondato[1,2] Eli Upfal[1]

[1]Department of Computer Science – Brown University

[2]Now at Two Sigma Investments

ECML PKDD'15 – Porto, September 11, 2015

Why this tutorial?

Over the years, we became convinced of the mathematical elegance of statistical learning theory. Experimental results then showed us its practical power.

We believe VC-dimension and Rademacher Averages are under-utilized in research and under-taught in classes, often relegated to small-font paragraphs in ML textbooks (this is slowly changing).

We feel that developing data mining algorithms using techniques considered to be only relevant for the theory of machine learning can help bringing the ML and the DM communities together...like ECML PKDD!

Logistics

You can find these slides at http://bit.ly/1JZEfF5

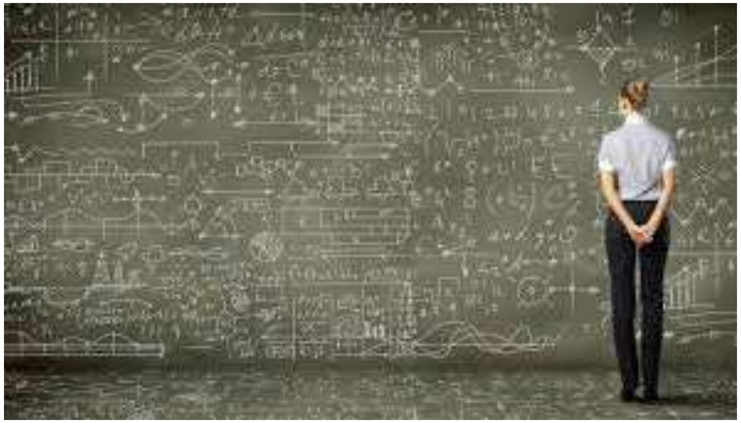
Tutorial mini-website: http://bigdata.cs.brown.edu/vctutorial/
Slides, bibliography, ...

Please interrupt me and ask questions at any time

Live tweet hashtag: #vctutorial

Two Views of Data

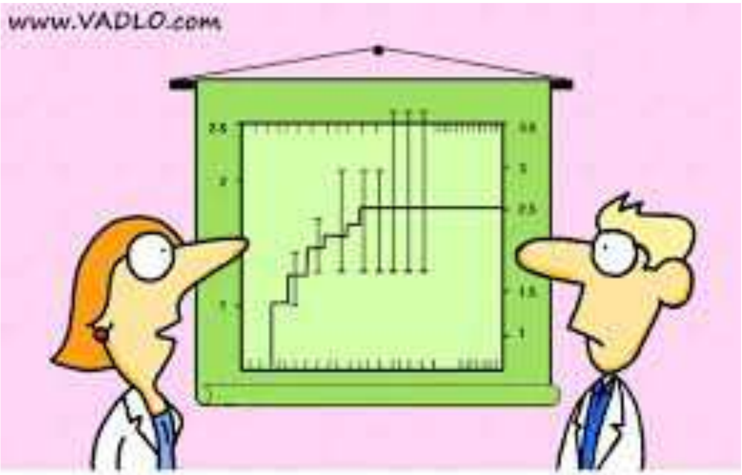
- The DB approach: The dataset is the whole truth (and nothing but the truth)
  - Example: Abnormality detection - Given a sequence of NASDAQ transactions, find suspicious transactions
- The scientific approach: The dataset is a sample of a larger process or system
  - Generalization: Valid observations on a sample must hold on other samples from the system



Data Analysis Through Sampling

- Sampling is a powerful technique –analyzing a sample, instead of the whole data set, saves computational time and space
- Analyzing a sample gives an approximation of the "true" result (but the whole data set may also be a sample)

The main question: How good is an approximation obtained from analyzing a sample (of a given size)?



"Did you really have to show the error bars?"

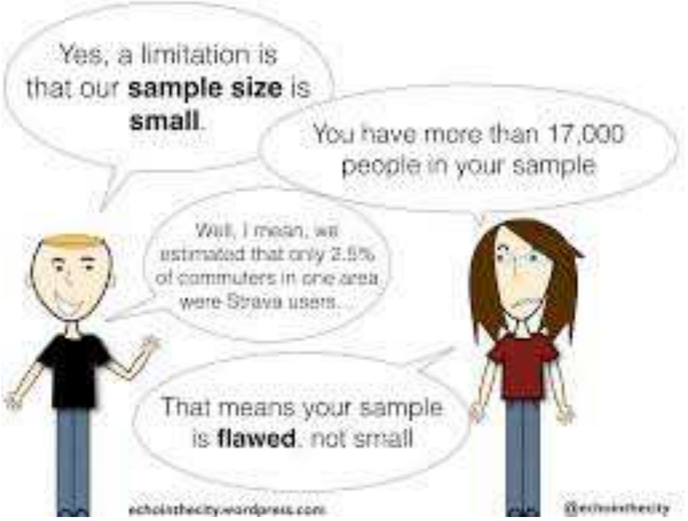
How Good is the Approximation?

- We “know” that increasing the sample size improves the approximation results
- Can we quantify this relation?
- We want a statement of the form:

“When analyzing a random sample of size n, with probability 1- δ, the results are within an ε factor (additive of multiplicative) of the true results.”

We need more information…

“I can see your error bars using google earth”


Yes, a limitation is that our **sample size** is **small**.
You have more than 17,000 people in your sample
Well, I mean, we estimated that only 2.5% of commuters in one area were Strava users.
That means your sample is **flawed**, not small
echoisthecity.wordpress.com
@echoisthecity

The Fundamental Tradeoff:

- Sample size
- Accuracy of the results
- Complexity of the analysis task

"I can prove it or disprove it! What do you want me to do?"

VC-Dimension and Rademacher Averages are two measures for the complexity of the analysis task

Using these measures we can obtained almost tight relations between the three quantities

A subset of the population.

Two Major Sampling Tasks

Detection:

- Detect internet flows sending more than 5% of total packet traffic
- Learn a classification rule from a training set

Estimation:

- Estimate the market share of all internet browsers with at least 5% market share
- Find the top frequent itemsets in a dataset

Basic Sampling Results

Consider uniform sampling S from a set U. Let $R \subseteq U$, such that $|R| \geq \epsilon |U|$.

Claim (Detection)

If S is a uniform random sample of U with size $\geq \frac{1}{\epsilon} \ln \frac{1}{\delta}$ then

$$\Pr(S \cap R = \emptyset) \leq (1-\epsilon)^{\frac{1}{\epsilon} \ln \frac{1}{\delta}} \leq \delta$$

Claim (Estimation – Chernoff bound)

If S is a uniform random sample of U with size $\geq \frac{3}{\epsilon^3} \ln \frac{2}{\delta}$ then

$$\Pr\left(\left| \frac{|S \cap R|}{|S|} - \frac{|R|}{|U|} \right| \leq \epsilon \frac{|R|}{|U|} \right) \geq 1 - \delta$$

Basic Sampling Results

Consider a distribution $\mathcal{D}$ with support U. Let $R \subseteq U$, such that $Pr(R) \geq \epsilon$.

Claim (Detection)

If S is a sample of $\mathcal{D}$ with size $\geq \frac{1}{\epsilon} \ln \frac{1}{\delta}$ then

$$\Pr(S \cap R = \emptyset) \leq (1-\epsilon)^{\frac{1}{\epsilon} \ln \frac{1}{\delta}} \leq \delta$$

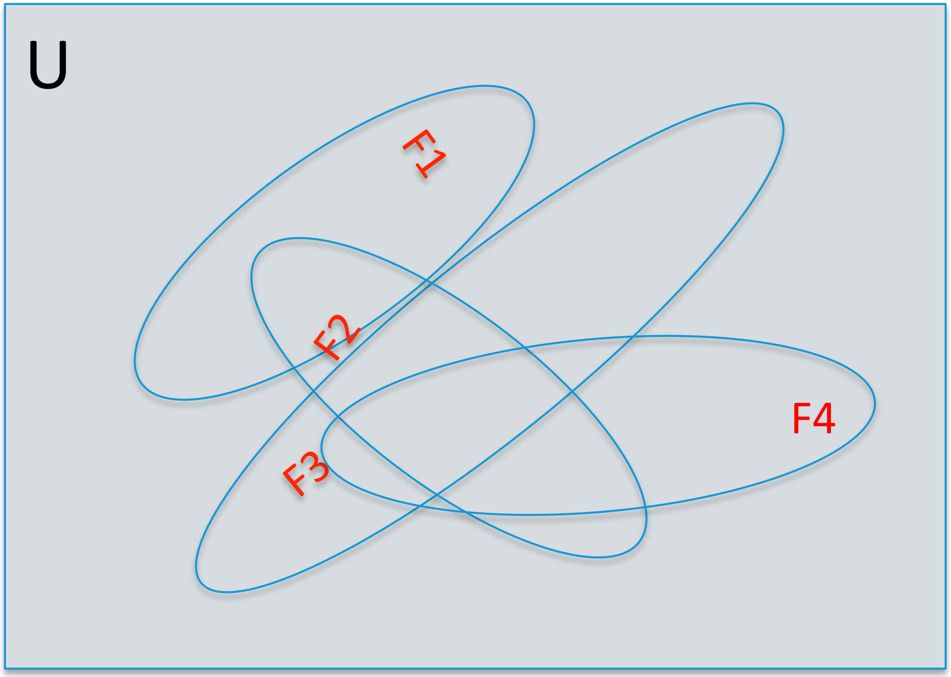
Claim (Estimation – Chernoff bound)

If S is a sample of $\mathcal{D}$ with size $\geq \frac{3}{\epsilon^3} \ln \frac{2}{\delta}$ then

$$\Pr\left(\left|\frac{|S \cap R|}{|S|} - \Pr(R)\right| \leq \epsilon \Pr(R)\right) \geq 1-\delta$$

The Multiple Events Problem

- Use a sample to detect internet flows sending more than 5% of total packet traffic
- The bound guarantee that we detect or estimate correctly one pre-defined event (set)



Need to detect or estimate simultaneously many events (sets).

Uniform convergence

Basic sampling results guarantee that *a sample intersects or approximate one given event (set), if the sample is not too small*

Instead, we want a sample that intersects or approximates simultaneously *all events (sets) that are not too small*

- More than one event
- Not fixed in advance

Classical solution to this problem: *Union bound*

The union bound

Consider uniform sampling S from a set U. Let $R_1, \ldots, R_n$ be subsets of U, such that $|R_i| \geq \epsilon |U|$, $1 \leq i \leq n$.

Claim (Estimation)

If S is a uniform random sample of U with size $\geq \frac{3}{\epsilon^3} \ln \frac{2n}{\delta}$ then

$$\Pr\left(\exists i \ s.t. \ \left| \frac{|S \cap R_i|}{|S|} - \frac{|R_i|}{|U|} \right| > \epsilon \frac{|R_i|}{|U|}\right) < \delta$$

The sample size now depends on the number n of sets we are interested in approximating!

Union bound consider events as if they were disjoint! This is far too loose!

Not practical for many applications
e.g., n is the number of itemsets, or of nodes in a graph!

PAC learning of a binary classifier

Consider:

- A probability distribution π on a domain $\mathcal{D}$
- A partition c of $\mathcal{D}$ into In and Out classes
- A concept class $\mathcal{C}$ – a collection of classification rules that includes the true classification c (realizable case)

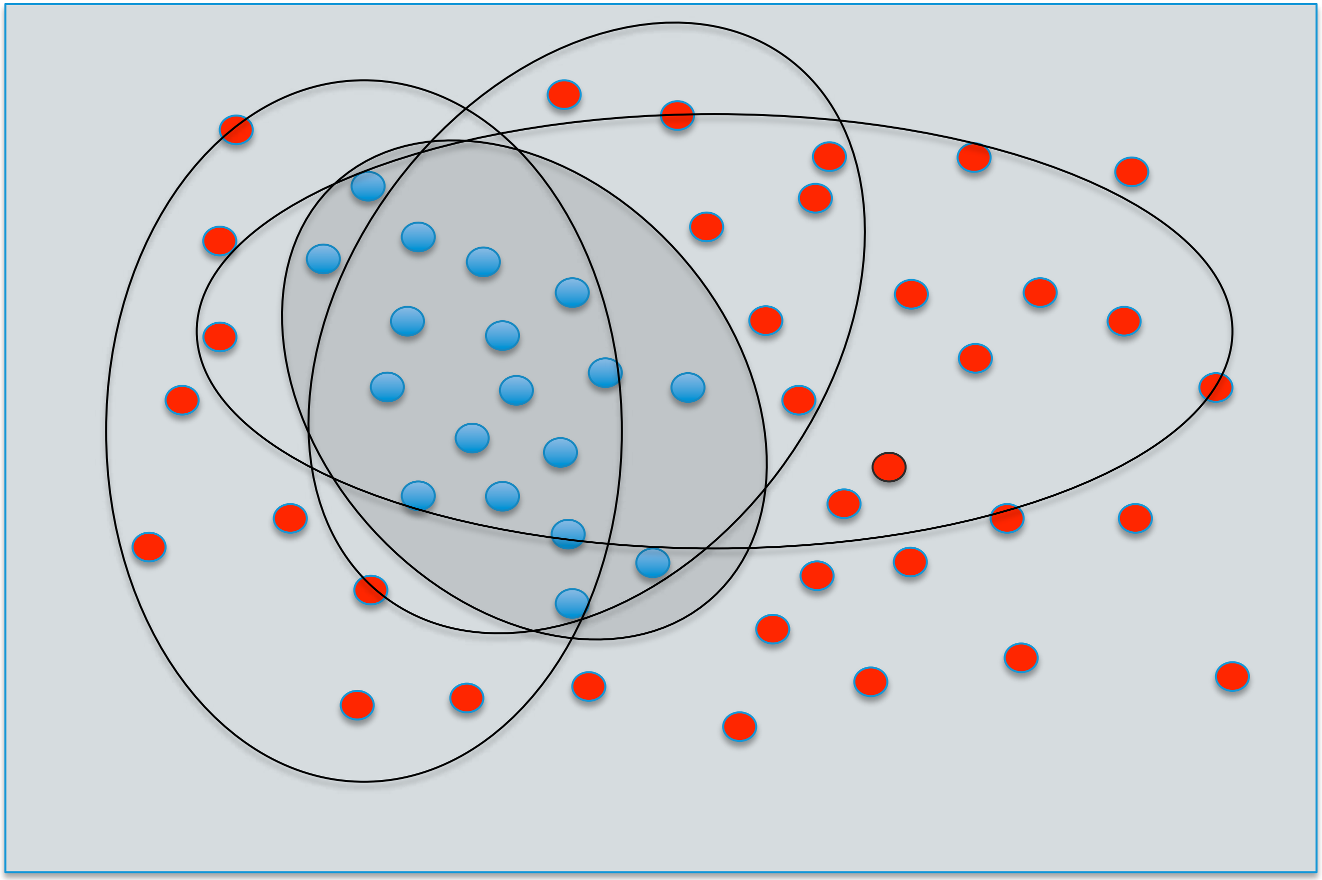
The learning algorithm gets m training examples $(x, \mathsf{c}(x))$, where x is sampled from π

Probably Approximately Correct (PAC) Learning:

With probability $1 - \delta$, the algorithm returns a classification rule from $\mathcal{C}$ that is correct (on elements sampled from π) with probability $1 - \varepsilon$

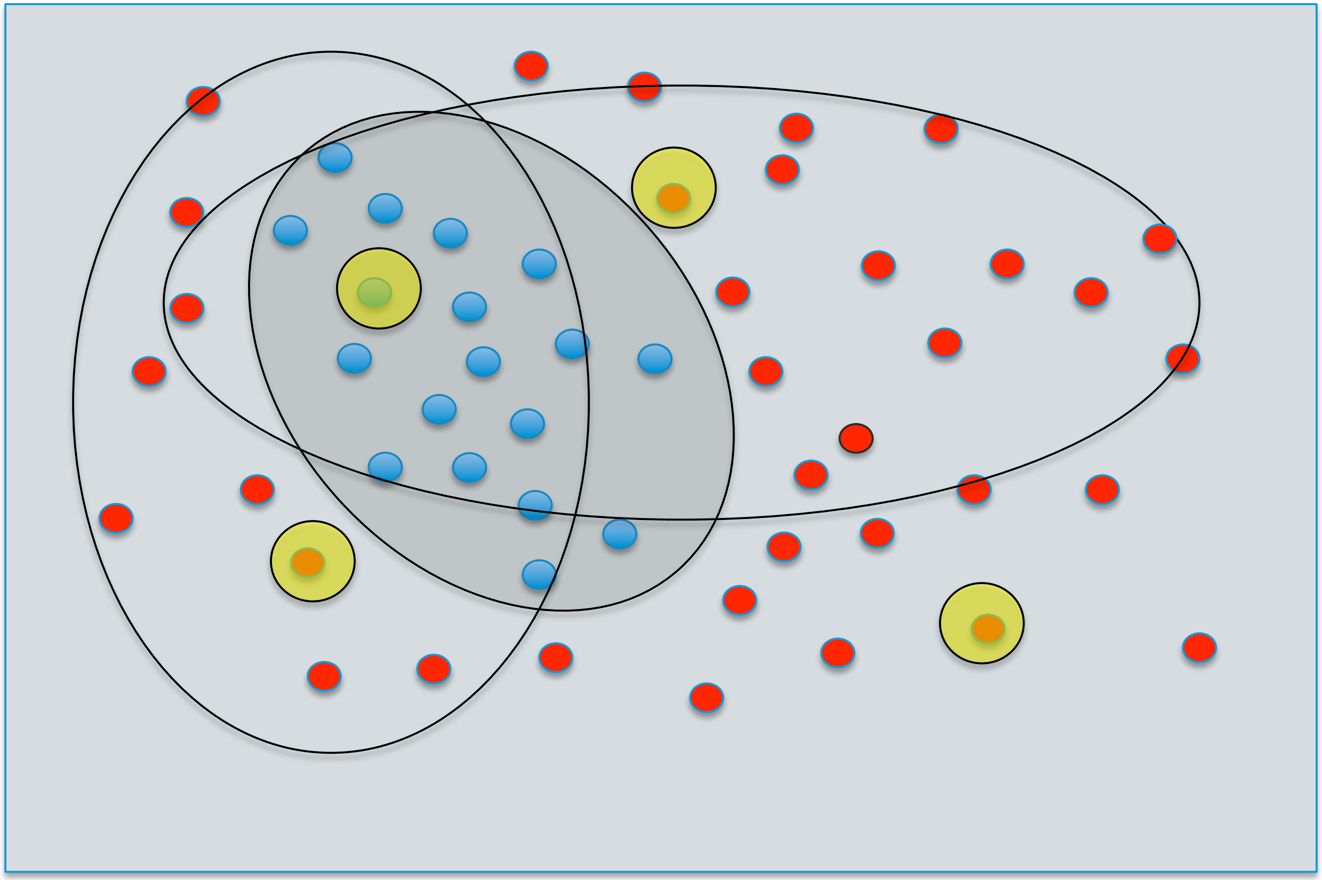
Learning a Binary Classifier

- Out and In items, and possible classification rules



Learning a Binary Classifier

- Red and blue items, possible classification rules, and the sample items ◉



When does the sample identify the correct rule?

- $\mathcal{C}$ - concept class - a collection of possible classification rules.
- $c \in \mathcal{C}$ - the correct rule.
- For any $h \in \mathcal{C}$ let $\Delta(c, h)$ be the set of items on which the two classifiers differ:

$$\Delta(c, h) = \{x \in U \mid h(x) \neq c(x)\}$$

- We need a sample that intersects every set in the family of sets

$$\{\Delta(c, h) \mid Pr(\Delta(c, h)) \geq \epsilon\}$$

Definition (ϵ-net)

An ϵ-net is a set $S \subseteq U$ such that for any $R \subseteq U$, if $\Pr(R) \geq \epsilon$ then $|R \cap S| \geq 1$.

Learnability - Uniform Convergence

Theorem

Any concept class $\mathcal{C}$ can be learned with $m = \frac{1}{\epsilon}(\ln|\mathcal{C}| + \ln\frac{1}{\delta})$ samples.

Proof.

We need a sample that intersects every set in the family of sets

$$\{\Delta(c, c') \mid \Pr(\Delta(c, c')) \geq \epsilon\}$$

. There are at most $|\mathcal{C}|$ such sets, and the probability that a sample is chosen inside a set is $\geq \epsilon$.

The probability that m random samples did not intersect with at least one of the sets is bounded by

$$|\mathcal{C}|(1-\epsilon)^m \leq |\mathcal{C}|e^{-\epsilon m} \leq |\mathcal{C}|e^{-(\ln|\mathcal{C}|+\ln\frac{1}{\delta})} \leq \delta.$$

□

How Good is this Bound?

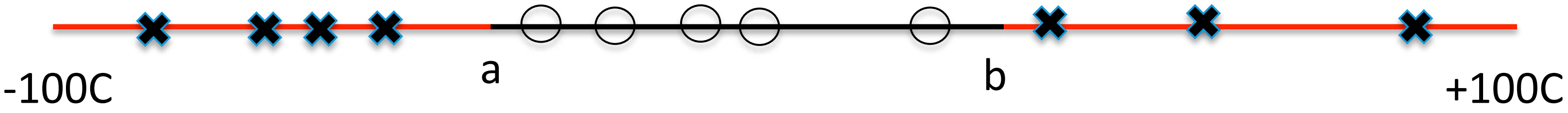
- Assume that we want to estimate the working temperature range of an iPhone.
- We sample temperatures in [-100C,+100C] and check if the iPhone works in each of these temperatures.

No Service

Temperature

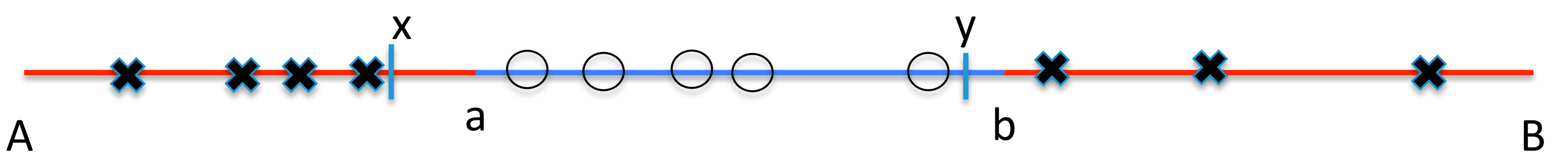
iPhone needs to cool down before you can use it.

slide for emergency



Learning an Interval

- Our universe U is an interval [A,B] on the line
- The “In” points are in the sub interval [a,b], the “out” points are outside [a,b]
- Our concept class is the collection of all the intervals [c,d], A ≤ c < d ≤ B
- If the learning algorithm returned the interval [x,y] then there were no samples in the sub-intervals [x,a] and [y,b]



Learning an Interval

- A distribution $\mathcal{D}$ is defined on universe that is an interval $[A, B]$.
- The true classification rule is defined by a sub-interval $[a, b] \subseteq [A, B]$.
- The concept class $\mathcal{C}$ is the collection of all intervals,

$$\mathcal{C} = \{[c, d] \mid [c, d] \subseteq [A, B]\}$$

Theorem

There is a learning algorithm that given a sample from $\mathcal{D}$ of size $m = \frac{2}{\epsilon} \ln \frac{2}{\delta}$, with probability $1 - \delta$, returns a classification rule (interval) $[x, y]$ that is correct with probability $1 - \epsilon$.

Note that the sample size is independent of the size of the concept class $|\mathcal{C}|$, which is $(B - A)^2$ if we assume that x and y must be integers, and infinite otherwise.

Proof.

Algorithm: Choose the smallest interval $[x, y]$ that includes all the "In" sample points.

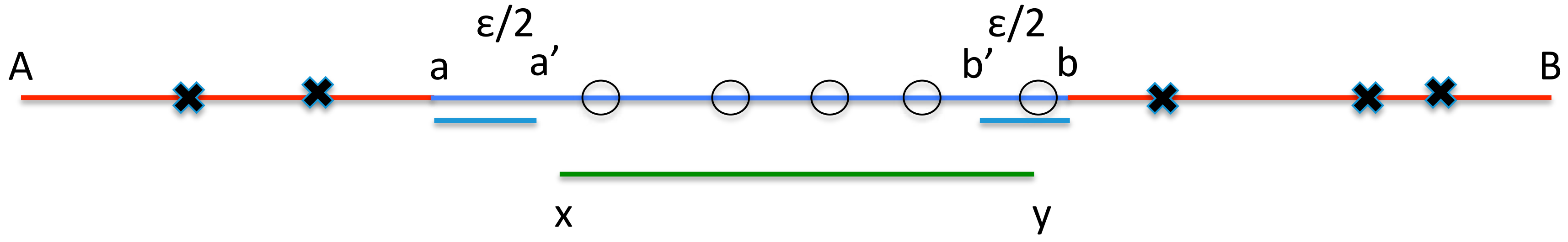
- Clearly $a \leq x < y \leq b$, and the algorithm can only err in classifying "In" points as "Out" points.
- Fix $a < a'$ and $b' < b$ such that $Pr([a, a']) = \epsilon/2$ and $Pr([b, b']) = \epsilon/2$.
- If the probability of error when using the classification $[x, y]$ is $\geq \epsilon$ then either $a' \leq x$ or $y \leq b'$ or both.
- The probability that the sample of size $m = \frac{2}{\epsilon} \ln \frac{2}{\delta}$ did not intersect with one of these intervals is bounded by

$$2(1 - \frac{\epsilon}{2})^m \leq e^{-\frac{\epsilon m}{2} + \ln 2} \leq \delta$$

□

Learning an Interval

- If the classification error is ≥ ε then the sample missed at least one of the the intervals [a,a’] or [b’,b] each of probability ≥ ε/2



Note that each sample excludes many possible intervals.

Questions?

Estimation: Frequent Itemsets Mining

Frequent Itemsets Mining: classic data mining problem with many applications

Settings:

Dataset $\mathcal{D}$
bread, milk
bread
milk, eggs
bread, milk, apple
bread, milk, eggs

Each line is a transaction, made of items from an alphabet $\mathcal{I}$

An itemset is a subset of $\mathcal{I}$. E.g., the itemset $\{\text{bread,milk}\}$

The frequency $f_{\mathcal{D}}(A)$ of $A \subseteq \mathcal{I}$ in $\mathcal{D}$ is the fraction of transactions of $\mathcal{D}$ that A is a subset of. E.g., $f_{\mathcal{D}}(\{\text{bread,milk}\}) = 3/5 = 0.6$

Frequent Itemsets Mining

Given a dataset $\mathcal{D}$ of transactions D find the k most frequent itemsets.

Exact algorithms are time and space expensive

Can we obtain a good approximation from a sample?

Problem: Rigorous approach that identifies the top frequent itemsets must have some estimate of all possible itemsets (exponential number)

Uniform Convergence

Data analysis through sampling requires simultaneous evaluations of many sets/events

Need sample that approximates/detects all relevant events (uniform convergence)

The union bound is too loose – events are not disjoint

VC-dimension and Rademacher averages allow us to obtain better bounds based on specific properties of the collection of events

Vapnik–Chervonenkis (VC) - Dimension

(X, R) is called a "range set":

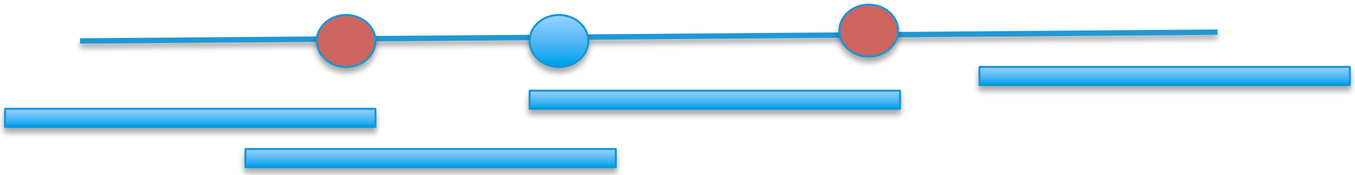
- $X =$ finite or infinite set (the set of objects to learn)
- R is a family of subsets of X, $R \subseteq 2^X$.
- In learning, $R = \mathcal{C}$, is a set of binary concepts, where $c \in \mathcal{C}$ is a subset $c = \{x \in X \mid c(x) = 1\} \subseteq X$
- For a finite set $S \subseteq X$, $s = |S|$, define the projection of $\mathcal{C}$ on S,

$$\Pi_{\mathcal{C}}(S) = \{c \cap S \mid c \in \mathcal{C}\}.$$

- If $|\Pi_{\mathcal{C}}(S)| = 2^s$ we say that $\mathcal{C}$ shatters S.
- The VC-dimension of $\mathcal{C}$ is the maximum size of S that is shattered by $\mathcal{C}$. If there is no maximum, the VC-dimension is ∞.

The VC-Dimension of a Collection of Intervals

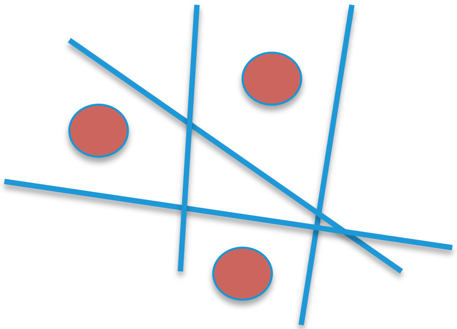
C = collections of intervals in [A,B] – can shatter 2 point but not 3. No interval includes only the two red points

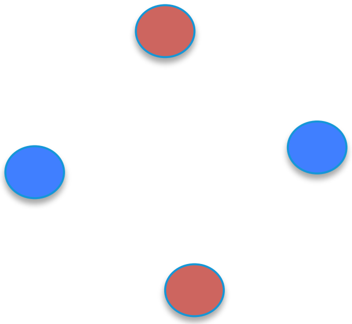


The VC-dimension of *C* is 2

Collection of Half Spaces in the Plane

C – all half space partitions in the plane. Any 3 points can be shattered:

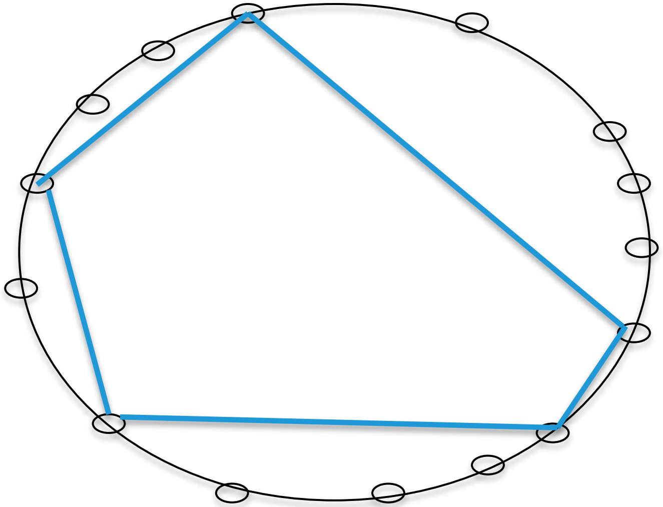




- Cannot partition the red from the blue points
- The VC-dimension of half spaces on the plane is 3
- The VC-dimension of half spaces in d-dimension space is d+1

Convex Bodies in the Plane

- *C* – all convex bodies on the plane



Any subset of the point can be included in a convex body.
The VC-dimension of *C* is ∞

Questions?

Learning a Classification

Theorem

Let $\mathcal{C}$ be a concept class with VC-dimension d then $\mathcal{C}$ is PAC learnable with

$$m = O(\frac{d}{\epsilon} \ln \frac{d}{\epsilon} + \frac{1}{\epsilon} \ln \frac{1}{\delta})$$

samples.

The sample size is not a function of the number of concepts, or the size of the domain!

Sauer's Lemma

VC-dinension is a measure of the complexity (or expressiveness) of a range space - how many different classification it defines on n elements.

For a finite set $S \subseteq X$, $s = |S|$, define the projection of R on S,

$$\Pi_R(S) = \{r \cap S \mid r \in R\}.$$

Theorem

Let (X, R) be a range space with VC-dimension d, for $S \subseteq X$, such that $|S| = n$,

$$|\Pi_R(S)| = \sum_{i=0}^{d} \binom{n}{i} \leq n^d.$$

The range space defines up to 2^d classifications for d elements, but no more than n^d for larger sets.

Proof

- By induction on d and (for each d) on n, obvious for $d = 0, 1$ with any n.
- Assume that the claim holds for all $|S'| \leq n - 1$ and $d' \leq d - 1$ and let $|S| = n$.
- Fix $x \in S$ and let $S' = S - \{x\}$.

$$\begin{aligned}
|\Pi_R(S)| &= |\{r \cap S \mid r \in R\}| \\
|\Pi_R(S')| &= |\{r \cap S' \mid r \in R\}| \\
|\Pi_{R(x)}(S')| &= |\{r \cap S' \mid r \in R \text{ and } x \notin r \text{ and } r \cup \{x\} \in R\}|
\end{aligned}$$

$$|\Pi_R(S)| = |\Pi_R(S')| + |\Pi_{R(x)}(S')|$$

- $(S', \Pi_{R(x)}(S'))$ has VC-dimension bounded by $d - 1$. If B is shattered by $(S', \Pi_{R(x)}(S'))$ then $B \cup \{x\}$ is shattered by (X, R)

$$\begin{aligned}|\Pi_R(S)| &\leq \sum_{i=0}^{d-1}\binom{n-1}{i} + \sum_{i=0}^{d}\binom{n-1}{i}\\ &= 1 + \sum_{i=1}^{d}(\binom{n-1}{i-1} + \binom{n-1}{i})\\ &= \sum_{i=0}^{d}\binom{n}{i} \leq (\frac{en}{d})^d \leq n^d\end{aligned}$$

[We use $\binom{n-1}{i-1} + \binom{n-1}{i} = \frac{(n-1)!}{(i-1)!(n-i-1)!}(\frac{1}{n-i} + \frac{1}{i}) = \binom{n}{i}$]

The number of distinct concepts on n elements grows polynomially in the VC-dimension!

ϵ-net

Let (X, R) be a range space and $\mathcal{D}$ a distribution on X.

Definition

An ϵ-net for a range space (X, R) is a subset $S \subseteq X$ such that for any $r \in R$, if $Pr(r) \geq \epsilon$ then $|S \cap r| \geq 1$.

Theorem

If (X, R) is a range space with VC-dimension d then a random sample of size

$$m = O(\frac{d}{\epsilon} \ln \frac{d}{\epsilon} + \frac{1}{\epsilon} \ln \frac{1}{\delta})$$

is with probability $1 - \delta$ an ϵ-net for (X, R).

ϵ-sample

Definition

An ε-sample for a range space (X, R) is a subset $N \subseteq X$ such that, for any $r \in R$,

$$\left|\Pr(r) - \frac{|N \cap r|}{|N|}\right| \leq \varepsilon \ .$$

Theorem

If (X, R) is a range space with VC-dimension d then a random sample of size

$$m = O(\frac{1}{\epsilon^2}(d + ln\frac{1}{\delta})$$

is, with probability $1 - \delta$, an ϵ-sample for (X, R).

The Double-Sampling Trick

Definition

An ϵ-net for a range space (X, R) is a subset $S \subseteq X$ such that for any $r \in R$, if $Pr(r) \geq \epsilon$ then $|S \cap r| \geq 1$.

- Let (X, R) be a range space with VC-dimension d. Let M be m independent samples from X.
- Let $E_1 = \{\exists r \in R \mid Pr(r) \geq \epsilon \text{ and } |r \cap M| = 0\}$. We want to show that $Pr(E_1) \leq \delta$.
- Choose a second sample T of m independent samples.
- Let $E_2 = \{\exists r \in R \mid Pr(r) \geq \epsilon \text{ and } |r \cap M| = 0 \text{ and } |r \cap T| \geq \epsilon m/2\}$

Lemma

$$Pr(E_2) \leq Pr(E_1) \leq 2Pr(E_2)$$

Lemma

$$Pr(E_2) \leq Pr(E_1) \leq 2Pr(E_2)$$

$E_1 = \{\exists r \in R \mid Pr(r) \geq \epsilon \text{ and } |r \cap M| = 0\}$

$E_2 = \{\exists r \in R \mid Pr(r) \geq \epsilon \text{ and } |r \cap M| = 0 \text{ and } |r \cap T| \geq \epsilon m/2\}$

$Pr(E_2) \leq Pr(E_1)$, but the additional condition holds with probability $\geq 1/2$:
Since $|T \cap r|$ has a Binomial distribution $B(m, \epsilon)$, for $m \geq 8/\epsilon$,

$$Pr(|T \cap r| < \epsilon m/2) \leq e^{-\epsilon m/8} < 1/2$$

Thus,

$$\frac{Pr(E_2)}{Pr(E_1)} = Pr(E_2 \mid E_1) \geq Pr(|T \cap r| \geq \epsilon m/2) \geq 1/2,$$

and it is sufficient to bound $Pr(E_2) \geq Pr(E_1)/2$.

$E_2 = \{\exists r \in R \mid Pr(r) \geq \epsilon \text{ and } |r \cap M| = 0 \text{ and } |r \cap T| \geq \epsilon m/2\}$

$E_2' = \{\exists r \in R \mid |r \cap M| = 0 \text{ and } |r \cap T| \geq \epsilon m/2\}$

Lemma

$$Pr(E_1) \leq 2Pr(E_2) \leq 2Pr(E_2') \leq 2(2m)^d 2^{-\epsilon m/2}.$$

- Instead of choosing M and T, we can choose a random sample Z of size $2m$ and divide it randomly to M and T.
- $Pr(E_2')$ is bounded by the probability that for an *arbitrary* set Z, there is $r \in R$ and $k = \epsilon m/2$, such that $|Z \cap r| \geq k$ but the random partition created M such that $|r \cap M| = 0$.
- For a fixed $r \in R$ let $E_r = \{|r \cap M| = 0 \text{ and } |r \cap T| \geq k\}$.

$$Pr(E_r) \leq Pr(|M \cap r| = 0 \mid |r \cap (M \cup T)| \geq k) = \frac{\binom{2m-k}{m}}{\binom{2m}{m}} \leq 2^{-\epsilon m/2}$$

- For a fixed $r \in R$ let $E_r = \{|r \cap M| = 0 \text{ and } |r \cap T| \geq k\}$.

$$Pr(E_r) \leq Pr(|M \cap r| = 0 \mid |r \cap (M \cup T)| \geq k) = \frac{\binom{2m-k}{m}}{\binom{2m}{m}} \leq 2^{-\epsilon m/2}$$

- For an arbitrary set Z the projection of R on Z gives $|\Pi_R(Z)| \leq (2m)^d$.
- Instead of a union bound on $|R|$ we union bound on $|\Pi_R(Z)| \leq (2m)^d$ sets.

$$Pr(E_1) \leq 2Pr(E_2') \leq 2(2m)^d 2^{-\epsilon m/2} \leq \delta$$

gives

$$m \geq \frac{8d}{\epsilon} \ln \frac{16d}{\epsilon} + \frac{4}{\epsilon} \ln \frac{4}{\delta}$$

- Independent of the size of R.

Lower Bound

The upper bound is almost tight:

Theorem

A random sample that gives an ϵ-net with probability $\geq 1 - \delta$ for a range space with VC-dimension d must have $\Omega(\frac{d}{\epsilon})$ samples.

Let $X = \{x_1, \ldots, x_d\}$ be a set that shattered $\mathcal{C}$.
W.l.o.g. $\mathcal{C} = \mathcal{C}(X)$, and $|\mathcal{C}| = 2^d$ - all possible classifications of d elements.

Define a probability distribution D:

$$\begin{aligned} Pr(x_1) &= 1 - 16\epsilon \\ Pr(x_2) &= Pr(x_3) = \cdots = Pr(x_d) = \frac{16\epsilon}{d-1} \end{aligned}$$

Let $X' = \{x_2, \ldots, x_d\}$.

Let S be a sample of $m = \frac{(d-1)}{64\epsilon}$ examples from the distribution D.

Let B be the event $|S \cap X'| \leq (d-1)/2$, then $Pr(B) \geq 1/2.$

Choose a random c - equivalent to choosing a random classification for each element.

$$Pr(\text{error on } c \mid B) \geq \frac{1}{2} 4\epsilon$$

Thus, with probability $\geq \delta \geq 1/2$ the error is $\geq \epsilon$.

ϵ-net and Learning a Classification

- Let X be a set of items, $\mathcal{D}$ a distribution on X, and $\mathcal{C}$ a set of concepts on X.
- $\Delta(c, c') = \{c \setminus c' \mid c' \in \mathcal{C}\} \cup \{c' \setminus c \mid c' \in \mathcal{C}\}$
- We take m samples and choose a concept c^*, while the correct concept is c.
- If $Pr_D(\{x \in X \mid c^*(x) \neq c(x)\}) > \epsilon$ then, $Pr(\Delta(c, c^*)) \geq \epsilon$, and no sample was chosen in $\Delta(c, c^*)$
- We need an ϵ-net for the range space $(X, \{\Delta(c, c') \mid c \in \mathcal{C}\})$.
- The VC-dimension of $(X, \{\Delta(c, c') \mid c \in \mathcal{C}\})$ is the same as the VC-dimension of $(X, \mathcal{C})$.
- $\{c' \cap S \mid c' \in \mathcal{C}\} \to \{(c' \setminus c) \cup (c \setminus c') \mid c' \in \mathcal{C}\}$ is a bijection.

Theorem

The number of samples needed to learn a (binary classification) concept class with VC-dimension d is

$$m = O(\frac{d}{\epsilon} \ln \frac{d}{\epsilon} + \frac{1}{\epsilon} \ln \frac{1}{\delta})$$

Questions?

Limitations of the VC-Dimension Approach:

- Hard to compute
- Combinatorial bound - ignores the distribution over the data.

Rademacher Averages:

- Incorporates the input distribution
- Applies to general functions not just classification
- Always at least as good bound as the VC-dimension
- Can be computed from a sample
- Still hard to compute

Rademacher Averages - Motivation

- Assume that S_1 and S_2 are two "uniform convergence" samples for estimating the expectations of any function in $\mathcal{F}$. Then, for any $f \in \mathcal{F}$, $\frac{1}{|S_1|}\sum_{x\in S_1} f(x) \approx \frac{1}{|S_2|}\sum_{y\in S_2} f(y) \approx E[f(x)]$, or

$$E[\sup_{f\in\mathcal{F}} |\frac{1}{|S_1|}\sum_{x\in S_1} f(x) - \frac{1}{|S_2|}\sum_{y\in S_2} f(y)|] \leq \epsilon$$

- *Rademacher Variables:* Instead of two samples we can take one sample $S = \{z_1, \ldots, z_m\}$ and split it randomly.

Definition

Let $\sigma = \sigma_1, \ldots, \sigma_m$ i.i.d r.v. $Pr(z_i = -1) = Pr(z_i = 1) = 1/2$. The *Empirical Rademacher Average* of $\mathcal{F}$ is defined as

$$\tilde{R}_m(\mathcal{F}) = E_\sigma\left[\sup_{f\in\mathcal{F}} \frac{1}{m}\sum_{i=1}^{m}\sigma_i f(z_i)\right]$$

Rademacher Averages - Motivation II

- Assume that $\mathcal{F}$ is a collection of $\{0, 1\}$ classifiers.
- A rich concept class $\mathcal{F}$ can approximate (correlate) any dichotomy, in particular a random one - represented by the random variables $\sigma = \sigma_1, \ldots, \sigma_m$.
- Thus, the Rademacher Average

$$\tilde{R}_m(\mathcal{F}) = E_\sigma \left[\sup_{f \in \mathcal{F}} \frac{1}{m} \sum_{i=1}^{m} \sigma_i f(z_i) \right]$$

represents the richness or expressiveness of the set $\mathcal{F}$.

Rademacher Averages (Complexity)

Given a fixed sample $S = \{z_1, \ldots, z_m\}$,

Definition

The *Empirical Rademacher Average* of $\mathcal{F}$ is defined as

$$\tilde{R}_m(\mathcal{F}) = E_\sigma \left[\sup_{f \in \mathcal{F}} \frac{1}{m} \sum_{i=1}^{m} \sigma_i f(z_i) \right]$$

Taking an expectation over the distribution $\mathcal{D}$ of the samples:

Definition

The *Rademacher Average* of $\mathcal{F}$ is defined as

$$R_m(\mathcal{F}) = E_{\mathcal{D}}[\tilde{R}_m(\mathcal{F})] = E_{\mathcal{D}} E_\sigma \left[\sup_{f \in \mathcal{F}} \frac{1}{m} \sum_{i=1}^{m} \sigma_i f(z_i) \right]$$

The Major Results

We first show that the Rademacher Average indeed captures the expected error in estimating the expectation of any function in a set of functions $\mathcal{F}$.

- Let $E_{\mathcal{D}}[f(z)]$ be the true expectation of a function f with distribution $\mathcal{D}$.
- For a sample $S = \{z_1, \ldots, z_m\}$ the estimate of $E_{\mathcal{D}}[f(z)]$ using the sample S is $\frac{1}{m}\sum_{i=1}^{m} f(z_i)$.

Theorem

$$E_{S\sim\mathcal{D}}\left[\sup_{f\in\mathcal{F}}\left(E_{\mathcal{D}}[f(z)] - \frac{1}{m}\sum_{i=1}^{m} f(z_i)\right)\right] \leq 2R_m(\mathcal{F}).$$

Proof Idea

Pick a second sample $S' = \{z'_1, \ldots, z'_m\}$.

$$
\begin{aligned}
& E_{S\sim\mathcal{D}}\left[\sup_{f\in\mathcal{F}}\left(E_{\mathcal{D}}[f(z)] - \frac{1}{m}\sum_{i=1}^{m} f(z_i)\right)\right] \\
= \;& E_{S\sim\mathcal{D}}\left[\sup_{f\in\mathcal{F}}\left(E_{S'\sim\mathcal{D}}\frac{1}{m}\sum_{i=1}^{m} f(z'_i) - \frac{1}{m}\sum_{i=1}^{m} f(z_i)\right)\right] \\
\leq \;& E_{S,S'\sim\mathcal{D}}\left[\sup_{f\in\mathcal{F}}\left(\frac{1}{m}\sum_{i=1}^{m} f(z'_i) - \frac{1}{m}\sum_{i=1}^{m} f(z_i)\right)\right] \qquad \text{Jensen's Inequlity} \\
\leq \;& E_{S,S',\sigma}\left[\sup_{f\in\mathcal{F}}\left(\frac{1}{m}\sum_{i=1}^{m} \sigma_i(f(z_i) - f(z'_i)\right)\right] \\
\leq \;& E_{S,\sigma}\left[\sup_{f\in\mathcal{F}}\frac{1}{m}\sum_{i=1}^{m} \sigma_i(f(z_i)\right] + E_{S',\sigma}\left[\sup_{f\in\mathcal{F}}\frac{1}{m}\sum_{i=1}^{m} \sigma_i(f(z'_i)\right] \\
= \;& 2R_m(\mathcal{F})
\end{aligned}
$$

Deviation Bound

Assume that that all $f \in \mathcal{F}$ satisfy $A_f \leq f(z) \leq A_f + c$.
Applying Azuma inequality to Doob's martingale of the function $\sup_{f\in\mathcal{F}}\left(E_{\mathcal{D}}[f(z)] - \frac{1}{m}\sum_{i=1}^{m} f(z_i)\right)$:

Theorem

Let $S = \{z_1, \ldots, z_n\}$ be a sample from $\mathcal{D}$ and let $\delta \in (0,1)$. For all $f \in \mathcal{F}$

1. $Pr(|E_{\mathcal{D}}[f(z)] - \frac{1}{m}\sum_{i=1}^{m} f(z_i)| \geq 2R_m(\mathcal{F}) + \epsilon) \leq e^{-2m\epsilon^2/c^2}$
2. $Pr(|E_{\mathcal{D}}[f(z)] - \frac{1}{m}\sum_{i=1}^{m} f(z_i)| \geq 2\tilde{R}_m(\mathcal{F}) + 2\epsilon) \leq 2e^{-2m\epsilon^2/c^2}$

Note that $|f(z)| \leq c$ is equivalent to $-c \leq f(z) \leq -c + 2c$.

McDiarmid's Inequality

Applying Azuma inequality to Doob's martingale:

Theorem

Let $X_1, \ldots, X_n$ be independent random variables and let $h(x_1, \ldots, x_n)$ be a function such that a change in variable x_i can change the value of the function by no more than c_i,

$$\sup_{x_1,\ldots,x_n,x_i'} |h(x_1, \ldots, x_i, \ldots, x_n) - h(x_1, \ldots, x_i', \ldots, x_n)| \leq c_i.$$

For any $\epsilon > 0$

$$Pr(h(X_1, \ldots, X_n) - E[h(X_1, \ldots, X_n)]| \geq \epsilon) \leq e^{-2\epsilon^2 / \sum_{i=1}^{n} c_i^2}.$$

Computing/Estimating Rademacher Averages

Can be estimated from a sample.

Theorem

Assume that $|\mathcal{F}|$ *is finite. Let* $S = \{z_1, \ldots, z_m\}$ *be a sample, and assume that*

$$\max_{f \in \mathcal{F}} \sqrt{\frac{1}{m} \sum_{i=1}^{m} f^2(z_i)} \leq C$$

then

$$\tilde{R}_m(\mathcal{F}) \leq \frac{C\sqrt{2 \log |\mathcal{F}|}}{m}.$$

VC-Dimension and Rademacher Averages: From Statistical Learning Theory to Sampling Algorithms

Matteo Riondato[1,2] Eli Upfal[1]

[1] Department of Computer Science – Brown University

[2] Now at Two Sigma Investments

ECML PKDD '15 – Porto, September 10, 2015

What time is it?

It's time to move from Statistical Learning Theory to Sampling Algorithms!

Why using sampling for Data Mining (DM) tasks?

The runtime of many DM algorithms has two components:

1. "problem runtime": due to the intrinsic complexity of the task
   (e.g., creating candidates, building a prefix tree, . . .)
2. "data runtime": due to the size of the input data (e.g., access to disks or network)

Many DM algorithms are impractical on huge inputs. How can we speed them up?

- Smarter algorithms cut the "problem runtime"(e.g., FP-growth vs Apriori) but the data runtime will always catch up and become dominant
- Analyzing only small subset(s) of the data cuts the "data runtime"
  but the output is an approximation of the exact results

Approximations are OK, when they have high-quality: many DM tasks are exploratory

Trade-off between accuracy and speed: the larger the samples, the better the approximation, the slower the execution

Why using VC-Dimension or Rademacher Averages in DM?

Many DM tasks require to compute many Quantities Of Interest (QOI)
Sometimes even an exponential number (Frequent Itemsets)

We want high-quality approximations for all the QUI
We need uniform convergence

Key question: How much to sample to get uniform convergence?
The sample size should depend on the DM task... but also on the data

Classical methods (Union bound) almost ignore the data and give too-large sample sizes

We say: No, let the data speak!
VC-Dimension and Rademacher Avgs use information about the data to derive better (smaller) sample sizes for uniform convergence

What are we going to show you now?

A recipe to formulate DM problems using VC-dimension and Rademacher averages

A VC-dimension-based sampling algorithm for betweenness centrality in graphs

A VC-dimension-based sampling algorithm for Frequent Itemsets Mining

A Rademacher-Averages-based progressive sampling algorithm for Frequent Itemsets Mining

A (empirical) VC-dimension-based algorithm to find statistically significant frequent itemsets when the dataset is a sample from an unknown distribution

General Recipe

1) Reformulate the DM task as an expectation estimation task:
Define the domain U, the family $\mathcal{F}$, and the probability distribution π on U, so that each QOI is the expectation $\mathbb{E}_\pi[f]$ of some $f \in \mathcal{F}$ w.r.t. π

2) Devise an efficient procedure to sample from U according to π
If the procedure is not efficient, the advantages of sampling are lost

3) Develop an efficient procedure to compute an upper bound to the VC-dimension of $\mathcal{F}$ or to the Rademacher Averages of $\mathcal{F}$ on the sample $\mathcal{S}$
If the procedure is not efficient, the advantages of sampling are lost

4) Determine the sample size using the bound and the ε-sample theorem, create the sample, and return the estimation on the sample (running an exact algorithm on the sample or a different procedure)

General Recipe

1) Reformulate the DM task as an expectation estimation task:
Define the domain U, the family $\mathcal{F}$, and the probability distribution π on U, so that each QOI is the expectation $\mathbb{E}_\pi[f]$ of some $f \in \mathcal{F}$ w.r.t. π

2) Devise an efficient procedure to sample from U according to π
If the procedure is not efficient, the advantages of sampling are lost

3) Develop an efficient procedure to compute an upper bound to the VC-dimension of $\mathcal{F}$ or to the Rademacher Averages of $\mathcal{F}$ on the sample $\mathcal{S}$
If the procedure is not efficient, the advantages of sampling are lost

4) Determine the sample size using the bound and the ε-sample theorem, create the sample, and return the estimation on the sample (running an exact algorithm on the sample or a different procedure)

5) Send the paper to KDD!

Application 1: Betweenness Centrality

VC-Dimension-based sampling algorithm for Node Betweenness Centrality
[R. and Kornaropoulos, WSDM 2014, DMKD 2015]

What vertices in a graph are important?

Betweenness centrality is one measure of vertex importance

Roughly, it is the fraction of Shortest Paths (SP) in a graph that go through a vertex

Let $G = (V, E)$, $|V| = n$, $|E| = m$. The betweenness centrality of $v \in V$ is:

$$\mathrm{b}(v) = \underbrace{\frac{1}{n(n-1)}}_{\text{normalization}} \sum_{p_{uw} \in \mathbb{S}_G} \underbrace{\frac{\mathbb{1}_{\mathcal{T}_v}(p_{uw})}{\sigma_{uw}}}_{\in[0,1]}$$

where:

- $\mathbb{S}_G$: set of all SPs in G
- $\mathcal{S}_{uw}$: set of all SPs from u to w ($\mathcal{S}_{uw} \subseteq \mathbb{S}_G$, $|\mathcal{S}_{uw}| = \sigma_{uw}$)
- $\mathcal{T}_v$: $\{p \in \mathbb{S}_G \ : \ v \in \mathrm{Int}(p)\}$

How to compute betweenness centrality?

Naïve algorithm: All Pairs SP computation, followed by aggregation
  Aggregation dominates runtime, $\Theta(n^3)$

[Brandes 2001]: Perform aggregation after each Single-Source SP (SSSP) computation
  Runtime: $O(nm)$ (unweighted G), $O(nm + n^2 \log n)$ (weighted G)
This is is still too much for graphs with $n = 10^9$, $m = 10^10$

Possible solution: perform fewer SPs computations by sampling
  We get approximate results, but that's OK!

What kind of approximation do we want ? What should we sample and how much?

What kind of approximation do we want?

We want uniform quality guarantees on the approximations of all vertices

Definition:

For $\varepsilon, \delta \in (0, 1)$, an (ε, δ)-approximation is a collection $\{\tilde{\mathsf{b}}(v), v \in V\}$ such that

$$\Pr(\exists v \in V \ : \ |\tilde{\mathsf{b}}(v) - \mathsf{b}(v)| > \varepsilon) < \delta$$

ε controls the accuracy, δ controls the confidence

Trade-off: smaller ε or $\delta \Rightarrow$ higher number of samples $\Rightarrow$ slower runtime

How can one get an (ε, δ)-approximation?

[Brandes and Pich, 2008]: only run SSSP and aggregation from a few sources

```
$r \leftarrow \frac{1}{\varepsilon^2}\left(\ln n + \ln 2 + \ln \frac{1}{\delta}\right)$ // sample size
$\tilde{b}(v) \leftarrow 0$, for all $v \in V$
for $i \leftarrow 1, \ldots, r$ do // the exact algorithm would iterate over $V$
    $v_i \leftarrow$ random vertex from $V$, chosen uniformly
    Perform single-source SP computation from $v_i$
    Perform partial aggregation, updating $\tilde{b}(u)$, $u \in V$, like in exact algorithm
end
Output $\{\tilde{b}v, v \in V\}$
```

Theorem: The output is an (ε, δ)-approximation

How do they prove it?

Start with bounding the deviation for a single vertex v (Hoeffding bound):

$$\Pr(|\tilde{\mathrm{b}}(v) - \mathrm{b}(v)| > \varepsilon) \leq 2e^{-2r\varepsilon^2}$$

Then take the union bound over n vertices to ensure uniform converge the sample size r must be such that

$$2e^{-2r\varepsilon^2} \leq \frac{\delta}{n}$$

That is, to get an (ε, δ)-approximation, we need

$$r \geq \frac{1}{2\varepsilon^2}\left(\ln n + \ln 2 + \ln \frac{1}{\delta}\right)$$

What is wrong with this approach?

1) We need

$$r \geq \frac{1}{2\varepsilon^2}\left(\ln n + \ln 2 + \ln\frac{1}{\delta}\right)$$

- This is loose, due to the union bound and does not scale well (experiments)
- The sample size depends on $\ln n$. This is not the right quantity: not all graphs of n nodes are equally "difficult": e.g., the n-star is "easier" than a random graph

The sample size r should depend on a more-specific characteristic of the graph

2) At each iteration, the algorithm performs a SSSP computation
Full exploration of the graph, no locality

How can we improve the sample size?

[R. and Kornaropoulos, 2014] present an algorithm that:

1) uses a sample size which depends on the vertex-diameter, a characteristic quantity of the graph. The derivation uses VC-dimension

2) samples SPs according to a specific, non-uniform distribution over $\mathbb{S}_G$. For each sample, it performs a single $s - t$ SP computation

- More locality: fewer edges touched than single-source SP
- Can use bidirectional search / A*, . . .

What is the algorithm?

```
$\mathrm{VD}(G) \leftarrow$ vertex-diameter of $G$ // stay tuned!
$r \leftarrow \frac{1}{2\varepsilon^2}\left(\lfloor \log_2(\mathrm{VD}(G) - 2\rfloor) + 1 + \ln(1/\delta)\right)$ // sample size
$\tilde{\mathrm{b}}(v) \leftarrow 0$, for all $v \in V$
for $i \leftarrow 1 \ldots, r$ do
    $(u, v) \leftarrow$ random pair of different vertices, chosen uniformly
    $\mathcal{S}_{uv} \leftarrow$ all SPs from $u$ to $v$ // Dijkstra, trunc. BFS, ...
    $p \leftarrow$ random element of $\mathcal{S}_{uv}$, chosen uniformly // not uniform over $\mathbb{S}_G$
    $\tilde{\mathrm{b}}(w) \leftarrow \tilde{\mathrm{b}}(w) + 1/r$, for all $w \in \mathrm{Int}(p)$ // update only nodes along $p$
end
Output $\{\tilde{\mathrm{b}}(v), v \in V\}$
```

Theorem: The output $\{\tilde{\mathrm{b}}(v), v \in V\}$ is an (ε, δ)-approximation

How can we prove the correctness?

We want to prove that the output $\{\tilde{\mathrm{b}}(v), v \in V\}$ is an (ε, δ)-approximation

Let's apply the recipe!

1. Define betweenness centrality computation as a expectation estimation problem (domain U, family $\mathcal{F}$, distribution π)
2. Show that the algorithm efficiently samples according to π
3. Show how to efficiently compute an upper bound to the VC-dimension
   Bonus: show tightness of bound
4. Apply the VC-dimension sampling theorem

How to define the expectation estimation task?

- The domain U is $\mathbb{S}_G$ (all SPs in G)
- The family is $\mathcal{F} = \{\mathbb{1}_{\mathcal{T}_v}, v \in V\}$, where $\mathcal{T}_v = \{p \in \mathbb{S}_G \;:\; : v \in \mathsf{Int}(p)\}$
- The probability distribution π on U is

$$\pi(p_{uw}) = \frac{1}{n(n-1)} \frac{1}{\sigma_{uw}}$$

The algorithm samples paths according to π

We have

$$\mathbb{E}_\pi[\mathbb{1}_{\mathcal{T}_v}] = \sum_{p_{uw} \in \mathbb{S}_G} \mathbb{1}_{\mathcal{T}_v} \pi(p_{uw}) = \sum_{p_{uw} \in \mathbb{S}_G} \mathbb{1}_{\mathcal{T}_v}(p_{uw}) \frac{1}{n(n-1)} \frac{1}{\sigma_{uw}} = \mathsf{b}(v)$$

How do we bound the VC-dimension?

Definition: The vertex-diameter $\mathsf{VD}(G)$ of G is the maximum number of vertices in a SP of G

$$\mathsf{VD}(G) = \max\{|p|, p \in \mathbb{S}_G\}$$

If G is unweighted, $\mathsf{VD}(G) = \Delta(G) + 1$. Otherwise no relationship
Very small in social networks, even huge ones (shrinking diameter effect)

Computing $\mathsf{VD}(G)$: $\left(2\frac{\text{max. edge weight}}{\text{min. edge weight}}\right)$-approximation via single-source SP

Theorem: The VC-dimension of $(\mathbb{S}_G, F)$ is at most $\lfloor \log_2 \mathsf{VD}(G) - 2 \rfloor + 1$

Let's prove it!

Theorem: The VC-dimension is at most $\lfloor \log_2 \text{VD}(G) - 2 \rfloor + 1$

Proof:

- For a set $A \subseteq \mathbb{S}_G$ of size $|A| = d$ to be shattered, any p in A must appear in at least 2^{d-1} different sets $\mathcal{T}_v$, one for each subset of A containing p.
- Any p appears only in the sets $\mathcal{T}_v$ such that $v \in \text{Int}(p)$
  There are $|\text{Int}(p)|$ such sets
- From the definition of the vertex-diameter $\text{VD}(G)$, we have $|\text{Int}(p)| \leq \text{VD}(G) - 2$
- To shatter A, d must be such that $2^{d-1} \leq \text{VD}(G) - 2$
- So d can be at most $\lfloor \log_2 \text{VD}(G) - 2 \rfloor + 1$, otherwise A can not be shattered

How to use the bound?

We have that:

- The estimation $\tilde{\mathrm{b}}(v)$ computed by the algorithm is the empirical average for $\mathrm{b}(v)$
- The algorithm samples SPs efficiently according to π
- We know an upper bound to the VC-dimension and how to compute it efficiently

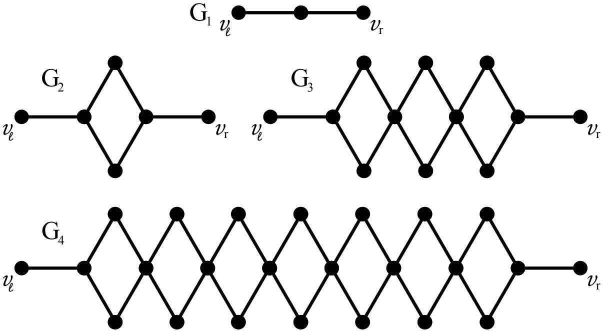
Thus we can apply the VC ε-sample theorem, and obtain that the algorithm outputs an (ε, δ)-approximation:

$$\Pr(\exists v \in V \ : \ |\tilde{\mathrm{b}}(v) - \mathrm{b}(v)| > \varepsilon) < \delta$$

Is the bound to the VC-dimension tight?

Yes! There is a class of graphs with VC-dimension exactly $\lfloor \log_2 \mathrm{VD}(G) - 2 \rfloor + 1$

The Concertina Graph Class $(G_i)_{i \in \mathbb{N}}$:



Theorem: The VC-dimension of $(\mathbb{S}_{G_i}, F)$ is $\lfloor \log_2 \mathrm{VD}(G) - 2 \rfloor + 1 = i$

Proof Intuition: The middle vertices are internal to a lot of SPs

Is the Vertex-Diameter the right quantity?

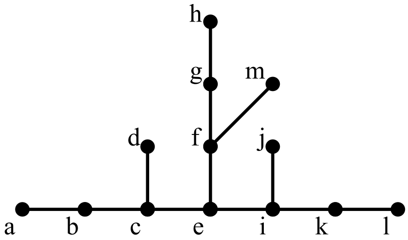
No! If G undirected and for every connected pair of nodes there is a unique SP, then the VC-dimension is at most 3

These graphs are not just trees!

Proof: in such a graph, two SPs that meet and separate can not meet again

(+ multiple case analysis)

The bound “3” is tight. In the following graph we can shatter 3 paths

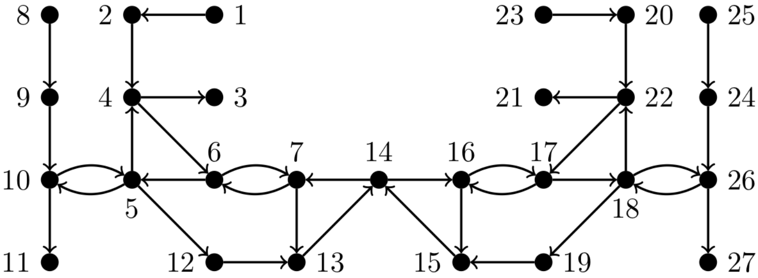


There is room for improvement using pseudodimension (we are working on that!)

What about directed graphs?

Does a similar result also hold for directed graphs with unique SP?

Not for the same constant 3. We built a graph with unique SPs between all connected nodes and we can shatter a set of 4 SPs



Yes, finding counterexamples is messy. . .

Does it hold for a different constant?

We do not know! Maybe you can work on that?

How well does the algorithm perform in practice?

It performs very well!

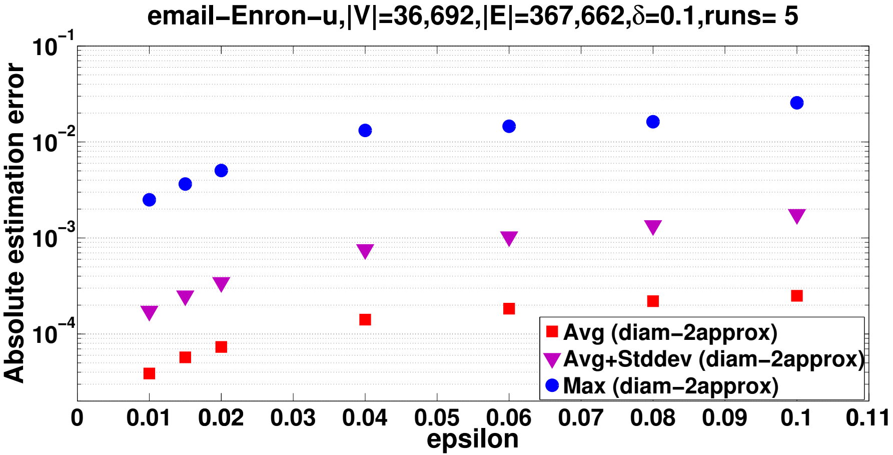
We tested the algorithm on real graphs (SNAP) and on artificial Barabasi-Albert graphs, to evalue its accuracy, speed, and scalability

Results: It blows away the exact algorithm and the union-bound-based sampling algorithm

How accurate is the algorithm?

In $O(10^3)$ runs of the algorithm on different graphs and with different parameters, we always had $|\tilde{\mathrm{b}}(v) - \mathrm{b}(v)| < \varepsilon$ for all nodes

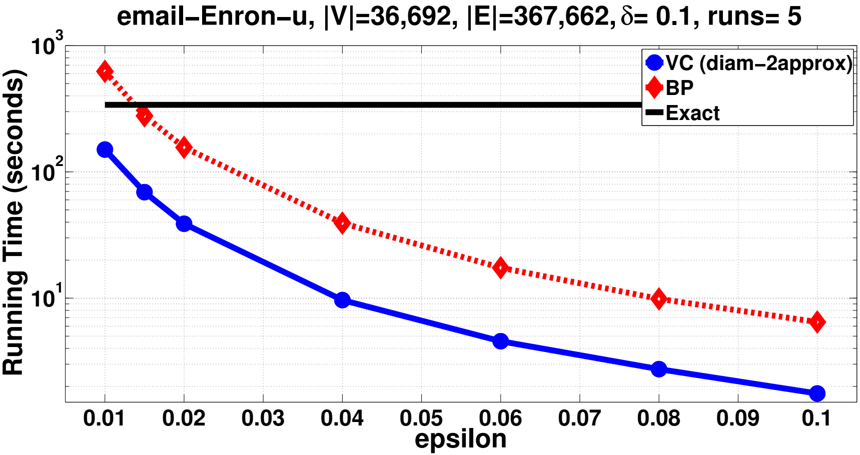
Actually, on average $|\tilde{\mathrm{b}}(v) - \mathrm{b}(v)| < \varepsilon/8$



How fast is the algorithm?

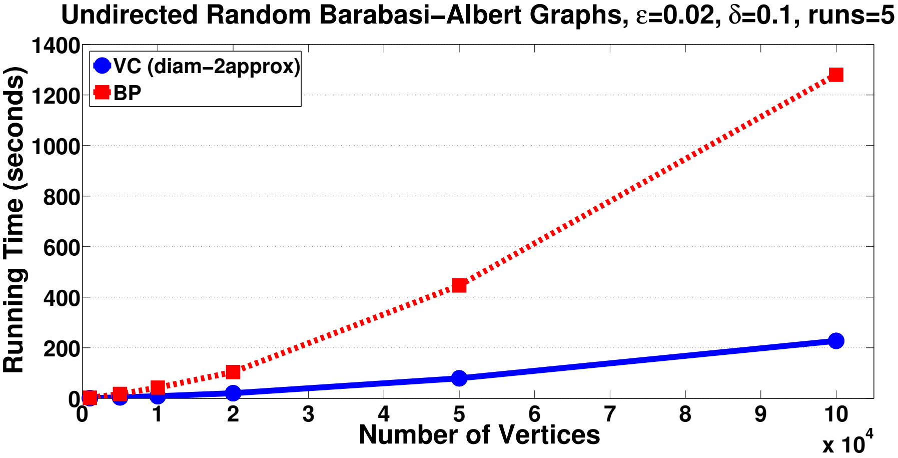
Approximately 8 times faster than the simple sampling algorithm

Variable speedup w.r.t. exact algorithm (200x – 4x), depending on ε



How scalable is the algorithm?

Much more scalable than the simple sampling algorithm, because the sample size does not depend on *n*



Conclusions (Betweenness Centrality)

We showed a sampling algorithm for betweenness centrality approximation that gives probabilistic guarantees on the quality of the approximation for all the vertices

The algorithm samples SPs according to a well-defined distribution, and the analysis relies on VC-dimension, which is bounded by the Vertex Diameter, a characteristic quantity of the graph that is small in real networks

The use of VC-dimension makes the algorithm much faster and more scalable than previous sampling approaches and than the exact algorithm

Questions?

Application 2: Frequent Itemsets Mining (FIM)

VC-Dimension-based sampling algorithm for FIM
[R. and U., ECML PKDD 2012, TKDD 2014]

Rademacher Averages-based sampling algorithm for FIM
[R. and U., KDD 2015]

Empirical-VC-dimension-based algorithm for finding statistically significant FIs
[R. and Vandin, SDM 2014]

What is Frequent Itemsets Mining (FIM)?

Frequent Itemsets Mining: classic data mining problem with many applications

Settings:

Dataset $\mathcal{D}$
bread, milk
bread
milk, eggs
bread, milk, apple
bread, milk, eggs

Each line is a transaction, made of items from an alphabet $\mathcal{I}$
An itemset is a subset of $\mathcal{I}$. E.g., the itemset $\{$bread,milk$\}$
The frequency $f_{\mathcal{D}}(A)$ of $A \subseteq \mathcal{I}$ in $\mathcal{D}$ is the fraction of transactions of $\mathcal{D}$ that A is a subset of. E.g., $f_{\mathcal{D}}(\{\text{bread,milk}\}) = 3/5 = 0.6$

Problem: Frequent Itemsets Mining (FIM)

Given $\theta \in [0, 1]$ find (i.e., mine) all itemsets $A \subseteq \mathcal{I}$ with $f_{\mathcal{D}}(A) \geq \theta$
I.e., compute the set $\mathrm{FI}(\mathcal{D}, \theta) = \{A \subseteq \mathcal{I} \ : \ f_{\mathcal{D}}(A) \geq \theta\}$

There exist exact algorithms for FI mining (Apriori, FP-Growth, . . .)

How to make FI mining faster?

Exact algorithms for FI mining do not scale with $|\mathcal{D}|$ (no. of transactions):
They scan $\mathcal{D}$ multiple times: painfully slow when accessing disk or network

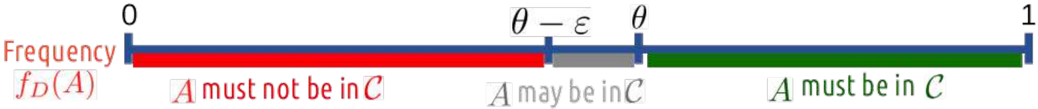
How to get faster? We could develop faster exact algorithms (difficult) or. . .
. . . only mine random samples of $\mathcal{D}$ that fit in main memory

Trading off accuracy for speed: we get an approximation of $\mathsf{FI}(\mathcal{D}, \theta)$ but we get it fast
Approximation is OK: FI mining is an exploratory task (the choice of θ is also often quite arbitrary)

Key question: How much to sample to get an approximation of given quality?

How to define an approximation of the FIs?

For $\varepsilon, \delta \in (0, 1)$, a (ε, δ)-approximation to FI$(\mathcal{D}, \theta)$ is a collection $\mathcal{C}$ of itemsets s.t., with prob. $\geq 1 - \delta$:


0
$\theta - \varepsilon$
θ
1
Frequency
$f_D(A)$
A must not be in $\mathcal{C}$
A may be in $\mathcal{C}$
A must be in $\mathcal{C}$

"Close" False Positives are allowed, but no False Negatives
This is the price to pay to get faster results: we lose accuracy

Still, $\mathcal{C}$ can act as set of candidate FIs to prune with fast scan of $\mathcal{D}$

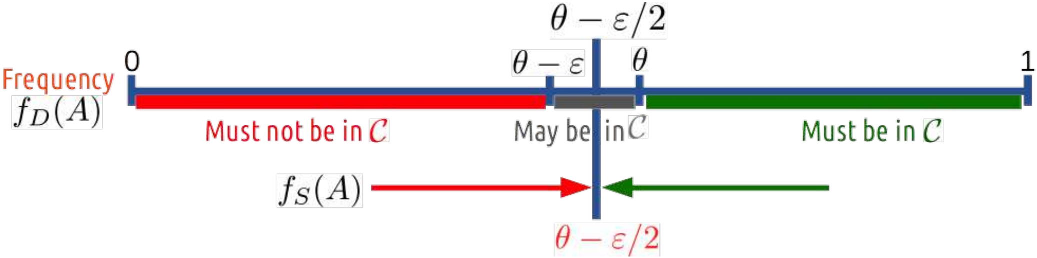
What do we really need?

We need a procedure that, given ε, δ, and $\mathcal{D}$, tells us how large should a sample $\mathcal{S}$ of $\mathcal{D}$ be so that

$$\Pr(\exists \text{ itemset } A \ : \ |f_{\mathcal{S}}(A) - f_{\mathcal{D}}(A)| > \varepsilon/2) < \delta$$

Theorem: When the above inequality holds, then $\mathsf{FI}(\mathcal{S}, \theta - \varepsilon/2)$ is an (ε, δ)-approximation

Proof (by picture):



Where does the union bound fall short?

For any itemset A, $|\mathcal{S}|f_{\mathcal{S}}(A)$ has a Binomial distribution with expectation $|\mathcal{S}|f_{\mathcal{D}}(A)$

We can use the Chernoff bound and have

$$\Pr(|f_{\mathcal{S}}(A) - f_{\mathcal{D}}(A)| > \varepsilon/2) \leq 2e^{-|\mathcal{S}|\varepsilon^2/12}$$

We then apply the union bound over all the itemsets to obtain uniform convergence

There are $2^{|\mathcal{I}|}$ itemsets, a priori. We need

$$2e^{-|\mathcal{S}|\varepsilon^2/12} \leq \delta/2^{|\mathcal{I}|}$$

Thus

$$|\mathcal{S}| \geq \frac{12}{\varepsilon^2}\left(|\mathcal{I}| + \ln 2 + \ln\frac{1}{\delta}\right)$$

The sample size depends on $|\mathcal{I}|$ but $\mathcal{I}$ can be very large

E.g., all the products sold by Amazon, all the pages on the Web, . . .

We need a smaller sample size that depends on some characteristic quantity of $\mathcal{D}$

How do we get a smaller sample size?

[R. and U. 2014, 2015]: Let's use VC-dimension! We apply the recipe

We define the task as an expectation estimation task:

- The domain is the dataset $\mathcal{D}$ (set of transactions)
- The family is $\mathcal{F} = \{\mathbb{1}_{\mathcal{T}_A}, A \subseteq 2^{\mathcal{I}}\}$, where $\mathcal{T}_A = \{\tau \in \mathcal{D} \ : \ A \subseteq \tau\}$ is the set of the transactions of $\mathcal{D}$ that contain A
- The distribution π is uniform over $\mathcal{D}$: $\pi(\tau) = 1/|\mathcal{D}|$, for each $\tau \in \mathcal{D}$

We sample transactions according to the uniform distribution, hence we have:

$$\mathbb{E}_\pi[\mathbb{1}_{\mathcal{T}_A}] = \sum_{\tau \in \mathcal{D}} \mathbb{1}_{\mathcal{T}_A}(\tau)\pi(\tau) = \sum_{\tau \in \mathcal{D}} \mathbb{1}_{\mathcal{T}_A}(\tau)\frac{1}{|\mathcal{D}|} = f_{\mathcal{D}}(A)$$

We then only need an efficient-to-compute upper bound to the VC-dimension

How do we bound the VC-dimension?

Enters the d-index of a dataset $\mathcal{D}$!

The d-index d of a dataset $\mathcal{D}$ is the maximum integer such that $\mathcal{D}$ contains at least d different transactions of length at least d

Example: The following dataset has d-index 3

bread	beer	milk	coffee
chips	coke	pasta	
bread	coke	chips	
milk	coffee		
pasta	milk		

It is similar but not equal to the h-index for published authors

It can be computed easily with a single scan of the dataset

Theorem: The VC-dimension is less or equal to the d-index d of $\mathcal{D}$

How do we prove the bound?

Theorem: The VC-dimension is less or equal to the d-index d of $\mathcal{D}$

Proof:

- Let $\ell > d$ and assume it is possible shatter a set $T \subseteq \mathcal{D}$ with $|T| = \ell$.
- Then any $\tau \in T$ appears in at least $2^{\ell-1}$ ranges $\mathcal{T}_A$ (there are $2^{\ell-1}$ subsets of T containing τ)
- But any τ only appears in the ranges $\mathcal{T}_A$ such that $A \subseteq \tau$. So it appears in $2^{|\tau|} - 1$ ranges
- From the definition of d, T must contain a transaction τ^* of length $|\tau^*| < \ell$
- This implies $2^{|\tau^*|} - 1 < 2^{\ell-1}$, so τ^* can not appear in $2^{\ell-1}$ ranges
- Then T can not be hattered. We reach a contradiction and the thesis is true

This theorem allows us to use the VC ε-sample theorem

What is the algorithm then?

```
$d \leftarrow$ d-index of $\mathcal{D}$
$r \leftarrow \frac{1}{\varepsilon^2}\left(d + \ln\frac{1}{\delta}\right)$
sample size
$\mathcal{S} \leftarrow \emptyset$
for $i \leftarrow 1, \ldots, r$ do
    $\tau_i \leftarrow$ random transaction from $\mathcal{D}$, chosen uniformly
    $\mathcal{S} \leftarrow \mathcal{S} \cup \{\tau_i\}$
end
Compute $\mathsf{FI}(\mathcal{S}, \theta - \varepsilon/2)$ using exact algorithm // Faster algos make our
approach faster!
Output $\mathsf{FI}(\mathcal{S}, \theta - \varepsilon/2)$
```

Theorem: The output of the algorithm is a (ε, δ)-approximation

We just proved it!

How does it perform in practice?

Very well!

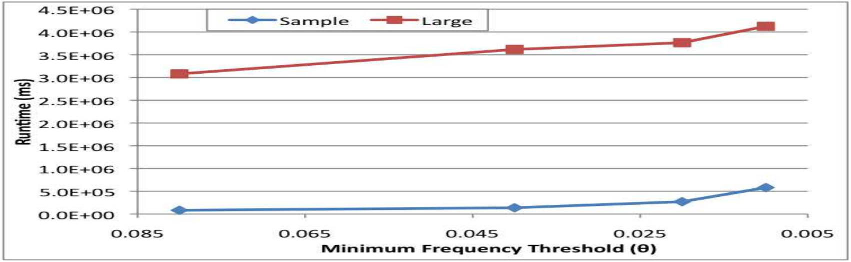
Great speedup w.r.t. an exact algorithm mining the whole dataset

Gets better as $\mathcal{D}$ grows, because the sample size does not depend on $|\mathcal{D}|$

Sample is small: 10^5 transactions for $\varepsilon = 0.01$, $\delta = 0.1$

The output always had the desired properties, not just with prob. $1 - \delta$

Maximum error $|f_{\mathcal{S}}(A) - f_{\mathcal{D}}(A)|$ much smaller than ε



Questions?

. . . so all is well, right?

There are some issues with the VC-dimension approach:

- Computing the d-index requires a full scan of the dataset

  This can still be expensive. Can we avoid it?

- The definition of the d-index depends on 'extreme' transactions:

  Maximum d such that $\mathcal{D}$ contains at least d transactions of length at least d

  This make the sample size too dependent on outliers. Can we do better?

- The VC-approach can not handle the following scenario:
  - We are given only a random sample $\mathcal{S}$ of $\mathcal{D}$ (no access to the dataset)
  - We are asked how good of an approximation we can get from this sample: given some δ, what is the minimum ε such that $\mathcal{S}$ is a (ε, δ)-approximation?

  Can we let the sample tells us how good it is?

[R. and U., 2015]: Let the sample speak! We use Progressive Random Sampling and Rademacher Averages to solve the above issues

What is Progressive Random Sampling?

Key question: How much to sample from $\mathcal{D}$ to obtain an (ε, δ)-approximation?

The VC-dimension algorithm a sufficient sample size, computed considering the worst-case dataset for the given d-index

Instead, let's start sampling, and have the data tell us when to stop we can get a better characterization of the data from the sample, and use it to sample less

Progressive Random Sampling is an iterative sampling scheme

Outline of PRS algorithm for approximating FI$(\mathcal{D}, \theta)$

At each iteration,

1. create sample $\mathcal{S}$ by drawing transactions from $\mathcal{D}$ uniformly and independently at random
2. Check a stopping condition on $\mathcal{S}$, to see if can get (ε, δ)-approximation from it
3. If stopping condition is satisfied, mine FI$(\mathcal{S}, \gamma)$ for some $\gamma < \theta$ and output it
4. Else, iterate with a larger sample

What are the challenges? What is our contribution?

The challenges are:

- Developing a stopping condition that
  - can be checked without expensive mining of each sample
  - guarantees that the output is a (ε, δ)-approximation
  - can be satisfied at small sample sizes
- Devising a method to choose the next sample size

Our contribution: We present the first algorithm that

- uses a stopping condition that does not mine each sample
- uses PRS to obtain an (ε, δ)-approximation of $\mathsf{FI}(\mathcal{D}, \theta)$
- computes the optimal next sample size on the fly

Previous contributions: heuristics (no guarantees) and/or required mining FIs from each sample (too expensive). They used predefined sample sizes (geometric schedule)

What do we really need?

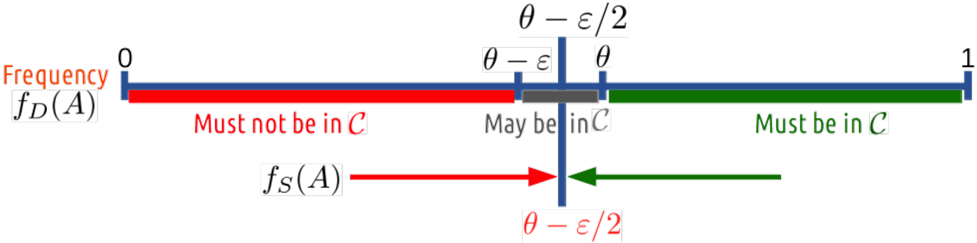
We need an efficient procedure that, *given a sample* $\mathcal{S}$ of $\mathcal{D}$, computes a value η s.t.

$$\Pr\left(\sup_{A\subseteq\mathcal{I}} |f_{\mathcal{D}}(A) - f_{\mathcal{S}}(A)| \le \eta\right) \ge 1-\delta$$

Then the stopping condition just tests if $\eta \le \varepsilon/2$

Theorem: If $\eta \le \varepsilon/2$, then $\mathsf{FI}(\mathcal{S}, \underbrace{\theta - \varepsilon/2}_{\gamma})$ is an (ε,δ)-approximation to $\mathsf{FI}(\mathcal{D},\mathcal{I},\theta)$

Proof (by picture) Like the one for the VC-dimension algorithm



How to compute η? Using Rademacher Averages!

What are Rademacher Averages? (Quick recall)

A measure of complexity of the task w.r.t. sampling (VC-dimension on steroids)
Definition is hairy: Let $\mathcal{S} = \{\tau_1, \ldots, \tau_{|\mathcal{S}|}\}$, the Rademacher Average on $\mathcal{S}$ is

$$\mathsf{R}(\mathcal{S}) = \mathbb{E}_\sigma \left[\sup_{A \subseteq \mathcal{I}} \frac{1}{\ell} \sum_{j=1}^{|\mathcal{S}|} \sigma_j \phi_A(\tau_j) \mid \mathcal{S} \right]$$

where the σ_i are Rademacher rv's and $\phi_A(\tau_i) = \mathbb{1}_{\tau_j}(A)$
The important part: $\mathsf{R}(\mathcal{S})$ is a sample-dependent quantity and we have:

$$\Pr\left(\sup_{A \subseteq \mathcal{I}} |f_{\mathcal{D}}(A) - f_{\mathcal{S}}(A)| \le \underbrace{2\mathsf{R}(\mathcal{S}) + \sqrt{\frac{2\ln(2/\delta)}{|\mathcal{S}|}}}_{\eta} \right) \ge 1 - \delta$$

We develop a method to efficiently compute an upper bound to $\mathsf{R}(\mathcal{S})$

So we can compute η and efficiently check the stopping condition "$\eta \le \varepsilon/2$?"

How can we bound the Rademacher average? (high level picture)

We compute an upper bound to the distribution of the frequencies in $\mathcal{S}$ of the Closed Itemsets (CIs) in $\mathcal{S}$ (An itemset is closed iff none of its supersets has the same frequency)

Connection with the CIs: $\sup_{A \subseteq \mathcal{I}} |f_{\mathcal{D}}(A) - f_{\mathcal{S}}(A)| = \sup_{A \in \text{CIs}} |f_{\mathcal{D}}(A) - f_{\mathcal{S}}(A)|$

Efficiency Constraint: use only information that can be obtained with a single scan of $\mathcal{S}$

How:

1. We use the frequency of the single items and the lengths of the transactions to define a (conceptual) partitioning of the CIs into classes, and to compute upper bounds to the size of each class and to the frequencies of the CIs in the class
2. We use these bounds to compute an upper bound to $\mathsf{R}(\mathcal{S})$ by minimizing a convex function in $\mathbb{R}^+$ (no constraints)

How can we bound the Rademacher average? (nitty-gritty details)

For any itemset $A \subseteq \mathcal{I}$, let $\mathbf{v}_{\mathcal{S}}(A)$ be the n-dimensional vector

$$\mathbf{v}_{\mathcal{S}}(A) = (\phi_A(\tau_1), \ldots, \phi_A(\tau_n)),$$

and let $V_{\mathcal{S}} = \{\mathbf{v}_{\mathcal{S}}(A), A \subseteq \mathcal{I}\}$ ($V_{\mathcal{S}}$ is a set)

Theorem (Variant of Massart's Lemma):

Let $w : \mathbb{R}^+ \to \mathbb{R}^+$ be the function

$$w(s) = \frac{1}{s} \ln \sum_{\mathbf{v} \in V_{\mathcal{S}}} \exp(s^2 \|\mathbf{v}\|^2 / (2n^2))$$

Then

$$\mathrm{R}(\mathcal{S}) \leq \min_{s \in \mathbb{R}^+} w(s)$$

Since $\tilde{w}$ is convex, its global minimum can be found efficiently

What does the set of vectors $V_{\mathcal{S}}$ look like?

Let $\mathsf{CI}(\mathcal{S})$ be the set of all Closed Itemsets in $\mathcal{S}$

Lemma: $V_{\mathcal{S}}$ contains all and only the vectors $\mathbf{v}_{\mathcal{S}}(A)$ for all $A \in \mathsf{CI}(\mathcal{S})$. Issue: Can not mine $\mathsf{CI}(\mathcal{S})$ to compute $w(s)$: it is too expensive!

Solution: Define a function $\tilde{w}(s)$ efficient to compute and minimize and s.t. $\tilde{w}(s) \geq w(s)$ for all s. Then use $\tilde{w}(s)$ to compute $\eta_{\mathcal{S}}$

How do we define the function $\tilde{w}$?

We define a partitioning $\mathcal{P}$ of $\mathsf{CI}(\mathcal{S})$

- Assume an ordering $<_{\mathcal{I}}$ of $\mathcal{I}$. For any $a \in \mathcal{I}$, assume an ordering $<_a$ of the transactions of $\mathcal{S}$ that contain a
- For any $A \in \mathsf{CI}(\mathcal{S})$, let $a \in A$ be the item in A that comes first wrt $<_{\mathcal{I}}$, and let τ be the transaction containing A that comes first wrt $<_a$. Assign A to class $\mathcal{P}_{a,\tau}$

For each class $\mathcal{P}_{a,\tau}$ we

- compute an upper bound to $|\mathcal{P}_{a,\tau}|$ using $|\tau|$ and $<_a$
- use $f_{\mathcal{S}}(a)$ as upper bound to $f_{\mathcal{S}}(A)$, for $A \in \mathcal{P}_{a,\tau}$

  Very efficient to compute $f_{\mathcal{S}}(a)$ while creating the sample

The new function $\tilde{w}$ used to compute $\mathsf{R}(\mathcal{S})$ is:

$$\tilde{w}(s) = \frac{1}{s} \ln \sum_{a \in \mathcal{I}_{\mathcal{S}}} \left(\left(1 + \sum_{r=1}^{\chi_a} \sum_{j=1}^{g_{a,r}} 2^{\min\{r, h_{a,r} - j\}} \right) e^{\frac{s^2 f_{\mathcal{S}}(a)}{2n}} \right)$$

Then

$$\eta_{\mathcal{S}} = \min_{s \in \mathbb{R}^+} \tilde{w}(s) + \sqrt{\frac{2 \ln(2/\delta)}{n}}$$

How to choose the next sample size ?

Previous works used a fixed sample schedule

Next sample size is current sample multiply by a user-specified parameter

We can compute the next sample size on the fly

Let the data speak: we use the quality of the current sample to compute the next sample size

First iteration: Use a sample of size at least $8\frac{\ln(2/\delta)}{\varepsilon^2}$

Why? It is impossible that $\eta \leq \varepsilon/2$ at smaller sample sizes

Successive iterations: multiply the sample size from the previous iteration by $\left(\frac{2\eta}{\varepsilon}\right)^2$

Intuition: If the frequencies of the items in the current iteration and the distribution of the transaction lengths are the same as in the previous iteration, then the stopping condition will be satisfied at this iteration

Experimental Evaluation

Greatly improved runtime over exact algorithm, one-shot sampling (vc), and fixed geometric schedules. Better and better than exact as $\mathcal{D}$ grows

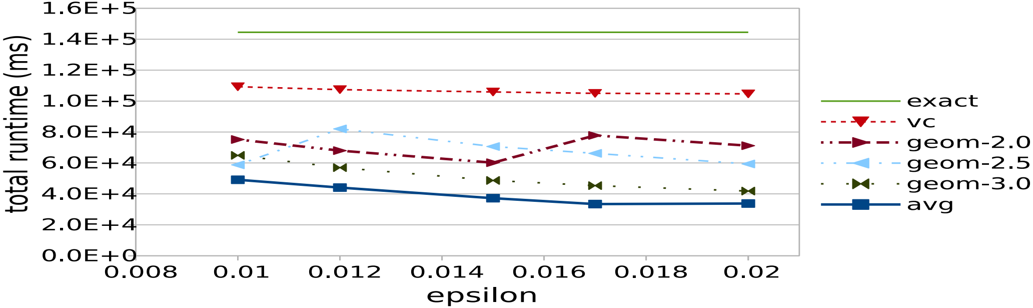


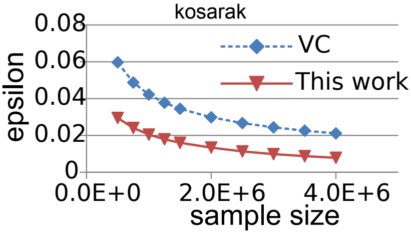
Figure: Running time for BMS-POS, $\theta = 0.015$.

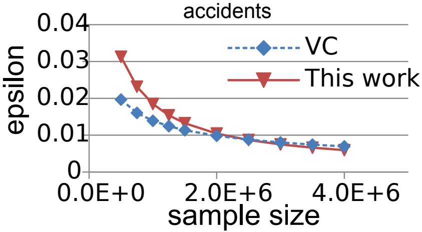
In 10K+ runs, the output was always an ε-approximation, not just with prob. $\geq 1 - \delta$

$\sup_{A\subseteq\mathcal{I}} |f_{\mathcal{D}}(A) - f_{\mathcal{S}}(A)|$ is 10x smaller than ε (50x smaller on average)

How does it compare to the VC-dimension algorithm?

Given a sample $\mathcal{S}$ and some $\delta \in (0,1)$, what is the smallest ε such that $\mathsf{FI}(\mathcal{S}, \theta - \varepsilon/2)$ is a (ε, δ)-approximation?





Note that this comparison is unfavorable to our algorithm: as we are allowing the VC-dimension approach to compute the d-index of $\mathcal{D}$ (but we don't have access to $\mathcal{D}$!)

We strongly believe that this is because we haven't optimized all the aspects of the bound to the Rademacher average. Once we do it, the Rademacher avg approach will most probably always be better

Recap

We show two algorithms for approximating the FIs using sampling

One uses VC-dimension and the d-index of the dataset to compute the sample size. This approach has some drawback

The second uses progressive sampling, with a stopping condition based on Rademacher averages, and solves most of the issues with the VC-approach

Questions?

Let's look at the data differently

The dataset $\mathcal{D}$ should (often) not be considered a perfect representation of reality
Rather, it is a sample from an unknown generative process

Reality is partially and noisily represented in the dataset
Itemsets may be frequent in $\mathcal{D}$ only due to random fluctuations

The real goal of mining is understanding the unknown generative process We should mine are the itemsets that have high probability of being generated: the True Frequent Itemsets [R. and Vandin, 2014]

What are the True Frequent Itemsets?

π: unknown probability distribution on $2^{\mathcal{I}}$ (samples are transactions)
No assumptions on π (e.g., no independence of items in transaction, or mixture model, ...)
$\mathcal{D}$: a collection of i.i.d. samples from π

The True Frequency of an itemset A is the probability that π generates a transaction containing A

$$\mathrm{t}(A) = \sum_{\substack{B \subseteq \mathcal{I} \\ A \subseteq B}} \pi(B)$$

Given $\theta \in [0,1]$, the True Frequent Itemsets w.r.t. θ are

$$\mathsf{TFI}(\pi, \theta) = \{A \subseteq \mathcal{I} \ : \ \mathrm{t}(A) \geq \theta\}$$

What can we really do?

We want to compute

$$\mathsf{TFI}(\pi,\theta) = \{A \subseteq \mathcal{I} \ : \ \mathsf{t}(A) \geq \theta\}$$

but we can not aim at getting the exact set: Any itemset A may have $f_{\mathcal{D}}(A) \leq \mathsf{t}(A)$ or $f_{\mathcal{D}}(A) \geq \mathsf{t}(A)$

Our Goal: Given $\delta \in (0,1)$, find $\gamma > \theta$ such that, with probability at least $1-\delta$, $\mathsf{FI}(\mathcal{D},\gamma) \subseteq \mathsf{TFI}(\pi,\theta)$, while minimizing $|\mathsf{TFI}(\pi\theta) \setminus \mathsf{FI}(\mathcal{D},\gamma)|$

In other words, we aim at controlling the Family-Wise Error Rate (classical goal in multiple hypothesis testing)

What are we actually looking for?

Let $\mathcal{B}^-(\mathsf{TFI}(\pi,\theta))$ be the negative border of $\mathsf{TFI}(\pi,\theta)$
the set of itemsets that are not in $\mathsf{TFI}(\pi,\theta)$ but whose supersets are all in $\mathsf{TFI}(\pi,\theta)$

Given $\delta \in (0,1)$, we want to compute the minimum ε such that,

$$\Pr(\exists A \in \mathcal{B}^-(\mathsf{TFI}(\pi,\theta)) \text{ s.t. } |f_{\mathcal{D}}(A) - \mathsf{t}(A)| > \varepsilon) < \delta$$

From the antimonotonicity of the frequency, this implies that, with probability at least $1-\delta$,

$$f_D(A) < \underbrace{\theta + \varepsilon}_{\gamma}, \text{ for all } A \notin \mathsf{TFI}(\pi,\theta)$$

so, with probability $\geq 1-\delta$, $\mathsf{FI}(\mathcal{D},\gamma) \subseteq \mathsf{TFI}(\pi,\theta)$

We can compute ε using the empirical VC-dimension of $\mathcal{B}^-(\mathsf{TFI}(\pi,\theta))$ on $\mathcal{D}$

What is the empirical VC-dimension?

The Empirical VC-dimension of a family $\mathcal{F}$ of functions on a sample is the VC-dimension of $\mathcal{F}$ on the sample

The size of the largest subset of the sample that can be shattered

Theorem (Variant of the VC ε-sample theorem using Empirical VC dimension): If the empirical VC-dimension is at most d, then, with probability at least $1 - \delta$,

$$|f_{\mathcal{D}}(A) - \mathrm{t}(A)| \leq 2\sqrt{\frac{2d \ln |\mathcal{S}|}{|\mathcal{S}|}} + \sqrt{\frac{2 \ln(2/\delta)}{|\mathcal{S}|}}, \text{ for all } A \subseteq \mathcal{I}$$

Key questions: what is the empirical VC-dimension in our case and how do we compute it?

What is the empirical VC-dimension of $\mathcal{B}^-(\mathsf{TFI}(\pi,\theta))$?

In our case, we are interested in the empirical VC-dimension of the family $\mathcal{F} = \{\mathbb{1}_{\mathcal{T}_A}, A \in \mathcal{B}^-(\mathsf{TFI}(\pi,\theta))$ on $\mathcal{D}$

This is different than the VC-dimension of all itemsets (i.e., $2^{\mathcal{I}}$), like in the "dataset is whole reality" case

Intuition: Bound like the d-index, but restricted to itemsets in $\mathcal{B}^-(\mathsf{TFI}(\pi,\theta))$ on $\mathcal{D}$

Involves solving a Set-Union Knapsack Problem: how many itemsets from $\mathcal{B}^-(\mathsf{TFI}(\pi,\theta))$ can we fit in a transactions of size ℓ?

We can exploit the fact that $\mathcal{B}^-(\mathsf{TFI}(\pi,\theta))$ is an *antichain*

Fewer itemsets can fit in a tranactions, hence tighter bound to empirical VC-dimension (but more convoluted computation)

What is the algorithm?

Roughly the following:

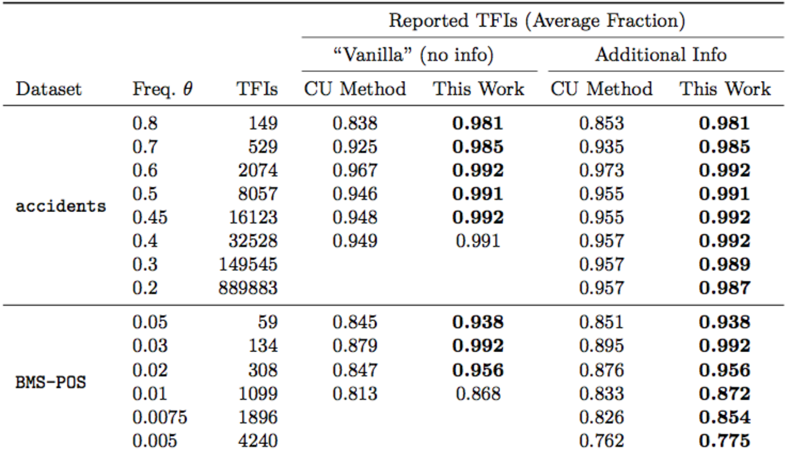
1. Compute the d-index d of $\mathcal{D}$ and $|\mathcal{I}|-1$
2. Compute the corresponding ε' and ε'' associated to these bounds. Let ε be the minimum.
3. Mine $A = \mathsf{FI}(\mathcal{D}, \theta - \varepsilon)$. Let $C = \mathcal{B}^{-}(A)$
4. Solve the SUKP associated to C to compute the empirical VC-dimension of C on $\mathcal{D}$
5. Compute the corresponding ε''
6. Mine and return $\mathsf{FI}(\mathcal{D}, \theta - \varepsilon'')$

Theorem: With probability at least $1-\delta$, $\mathsf{FI}(\mathcal{D}, \theta - \varepsilon'') \subseteq \mathsf{TFI}(\pi, \theta)$

How well does it perform in practice?

Always had $\mathsf{FI}(\mathcal{D}, \theta - \varepsilon'') \subseteq \mathsf{TFI}(\pi, \theta)$

Always reported more TFIs than previous approach with Chernoff+Union bound



			Reported TFIs (Average Fraction)			
			"Vanilla" (no info)		Additional Info	
Dataset	Freq. θ	TFIs	CU Method	This Work	CU Method	This Work
accidents	0.8	149	0.838	**0.981**	0.853	**0.981**
	0.7	529	0.925	**0.985**	0.935	**0.985**
	0.6	2074	0.967	**0.992**	0.973	**0.992**
	0.5	8057	0.946	**0.991**	0.955	**0.991**
	0.45	16123	0.948	**0.992**	0.955	**0.992**
	0.4	32528	0.949	0.991	0.957	**0.992**
	0.3	149545			0.957	**0.989**
	0.2	889883			0.957	**0.987**
BMS-POS	0.05	59	0.845	**0.938**	0.851	**0.938**
	0.03	134	0.879	**0.992**	0.895	**0.992**
	0.02	308	0.847	**0.956**	0.876	**0.956**
	0.01	1099	0.813	0.868	0.833	**0.872**
	0.0075	1896			0.826	**0.854**
	0.005	4240			0.762	**0.775**

Conclusions (Empirical VC-dimension)

The techniques and results we presented in the first part of the talk can be used also in the case where the dataset is a sample from an unknown distribution

Although the empirical VC-dimension is powerful, we believe that a Rademacher average approach would give better results also in this case

There is a lot to explore. . .

Questions?

General Conclusions

VC-dimension and Rademacher averages are a great addition to the DM algorithm designer toolkit They are

- Powerful
- Game-changing
- Intuitive (at least VC-dim. . .)
- Elegant
- Difficult to compute exactly but relatively easy to bound
- Extremely adaptible to different scenarios

We only scratched the surface and showed a few applications
There is much more (differential privacy, noisy datasets, . . .)

Embrace Statistical Learning Theory :-)

Contacts

Tutorial Website: http://bigdata.cs.brown.edu/vctutorial/

Come talk to us!

Matteo Riondato – matteo@twosigma.com
Two Sigma Investments
http://matteo.rionda.to – @riondabsd

Eli Upfal – eli@cs.brown.edu
Department of Computer Science – Brown University
http://cs.brown.edu/~eli