Sauer-Shelah Lemma

Ata Keskin

May 26, 2024

Abstract

The Sauer-Shelah Lemma is a fundamental result in extremal set theory and combinatorics, that guarantees the existence of a set T of size k which is shattered by a family of sets $\mathcal{F}$, if the cardinality of the family is greater than some bound dependent on k. A set T is said to be shattered by a family $\mathcal{F}$ if every subset of T can be obtained as an intersection of T with some set $S \in \mathcal{F}$. The Sauer-Shelah Lemma has found use in diverse fields such as computational geometry, approximation algorithms and machine learning. In this entry we formalize the notion of shattering and prove the generalized and standard versions of the Sauer-Shelah Lemma.

Contents

1 Introduction

The goal of this entry is to formalize the Sauer-Shelah Lemma. The result was first published by Sauer [2] and Shelah [3] independently from one another. The proof presented in this entry is based on an article by Kalai [1].

The lemma has a wide range of applications. Vapnik and Červonenkis [4] reproved and used the lemma in the context of statistical learning theory. For instance, the VC-dimension of a family of sets is defined as the size of the largest set the family shatters. In this context, the Sauer-Shelah Lemma is a result tying the VC-dimension of a family to the number of sets in the family.

2 Definitions and lemmas about shattering

In this section, we introduce the predicate *shatters* and the term for the family of sets that a family shatters *shattered-by*.

```
theory Shattering
  imports Main
begin
```

2.1 Intersection of a family of sets with a set

```
abbreviation IntF :: 'a set set ⇒ 'a set ⇒ 'a set set (infixl ∩* 60)
  where F ∩* S ≡ ((∩) S) ` F

lemma idem-IntF:
  assumes ⋃A ⊆ Y
  shows A ∩* Y = A
proof −
  from assms have A ⊆ A ∩* Y by blast
  thus ?thesis by fastforce
qed

lemma subset-IntF:
  assumes A ⊆ B
  shows A ∩* X ⊆ B ∩* X
  using assms by (rule image-mono)

lemma Int-IntF: (A ∩* Y) ∩* X = A ∩* (Y ∩ X)
proof
  show A ∩* Y ∩* X ⊆ A ∩* (Y ∩ X)
  proof
    fix S
    assume S ∈ A ∩* Y ∩* X
     then obtain a-y where A-Y0: a-y ∈ A ∩* Y and A-Y1: a-y ∩ X = S by
blast
    from A-Y0 obtain a where A0: a ∈ A and A1: a ∩ Y = a-y by blast
    from A-Y1 A1 have a ∩ (Y ∩ X) = S by fast
    with A0 show S ∈ A ∩* (Y ∩ X) by blast
  qed
next
  show A ∩* (Y ∩ X) ⊆ A ∩* Y ∩* X
  proof
```

fix S
assume $S \in A \cap* (Y \cap X)$
then obtain a **where** $A0$: $a \in A$ **and** $A1$: $a \cap (Y \cap X) = S$ **by** *blast*
from $A0$ **have** $a \cap Y \in A \cap* Y$ **by** *blast*
with $A1$ **show** $S \in (A \cap* Y) \cap* X$ **by** *blast*
qed
qed

insert distributes over $(\cap*)$

lemma *insert-IntF*:
shows $insert\ x\ `\ (H \cap* S) = (insert\ x\ `\ H) \cap* (insert\ x\ S)$
proof
show $insert\ x\ `\ (H \cap* S) \subseteq (insert\ x\ `\ H) \cap* (insert\ x\ S)$
proof
fix $y\text{-}x$
assume $y\text{-}x \in insert\ x\ `\ (H \cap* S)$
then obtain y **where** 0: $y \in (H \cap* S)$ **and** 1: $y\text{-}x = y \cup \{x\}$ **by** *blast*
from 0 **obtain** yh **where** 2: $yh \in H$ **and** 3: $y = yh \cap S$ **by** *blast*
from $1\ 3$ **have** $y\text{-}x = (yh \cup \{x\}) \cap (S \cup \{x\})$ **by** *simp*
with 2 **show** $y\text{-}x \in (insert\ x\ `\ H) \cap* (insert\ x\ S)$ **by** *blast*
qed
next
show $insert\ x\ `\ H \cap* (insert\ x\ S) \subseteq insert\ x\ `\ (H \cap* S)$
proof
fix $y\text{-}x$
assume $y\text{-}x \in insert\ x\ `\ H \cap* (insert\ x\ S)$
then obtain $yh\text{-}x$ **where** 0: $yh\text{-}x \in (\lambda Y.\ Y \cup \{x\})\ `\ H$ **and** 1: $y\text{-}x = yh\text{-}x \cap (S \cup \{x\})$ **by** *blast*
from 0 **obtain** yh **where** 2: $yh \in H$ **and** 3: $yh\text{-}x = yh \cup \{x\}$ **by** *blast*
from $1\ 3$ **have** $y\text{-}x = (yh \cap S) \cup \{x\}$ **by** *simp*
with 2 **show** $y\text{-}x \in insert\ x\ `\ (H \cap* S)$ **by** *blast*
qed
qed

2.2 Definition of *shatters*, *VC-dim* and *shattered-by*

abbreviation *shatters* :: $'a\ set\ set \Rightarrow 'a\ set \Rightarrow bool$ (**infixl** *shatters 70*)
where $H\ shatters\ A \equiv H \cap* A = Pow\ A$

definition *VC-dim* :: $'a\ set\ set \Rightarrow nat$
where $VC\text{-}dim\ F = Sup\ \{card\ S \mid S.\ F\ shatters\ S\}$

definition *shattered-by* :: $'a\ set\ set \Rightarrow 'a\ set\ set$
where $shattered\text{-}by\ F \equiv \{A.\ F\ shatters\ A\}$

lemma *shattered-by-in-Pow*:
shows $shattered\text{-}by\ F \subseteq Pow\ (\bigcup F)$
unfolding *shattered-by-def* **by** *blast*

lemma *subset-shatters*:

```
  assumes A ⊆ B and A shatters X
  shows B shatters X
proof -
  from assms(1) have A ∩* X ⊆ B ∩* X by blast
  with assms(2) have Pow X ⊆ B ∩* X  by presburger
  thus ?thesis by blast
qed

lemma supset-shatters:
  assumes Y ⊆ X and A shatters X
  shows A shatters Y
proof -
  have h: ⋃(Pow Y) ⊆ Y by simp
  from assms have 0: Pow Y ⊆ A ∩* X by auto
  from subset-IntF[OF 0, of Y] Int-IntF[of Y X A] idem-IntF[OF h] have Pow
Y ⊆ A ∩* (X ∩ Y) by argo
  with Int-absorb2[OF assms(1)] Int-commute[of X Y] have Pow Y ⊆ A ∩* Y
by presburger
  then show ?thesis by fast
qed

lemma shatters-empty:
  assumes F ≠ {}
  shows F shatters {}
using assms by fastforce

lemma subset-shattered-by:
  assumes A ⊆ B
  shows shattered-by A ⊆ shattered-by B
unfolding shattered-by-def using subset-shatters[OF assms] by force

lemma finite-shattered-by:
  assumes finite (⋃ F)
  shows finite (shattered-by F)
  using assms rev-finite-subset[OF - shattered-by-in-Pow, of F] by fast
```

The following example shows that requiring finiteness of a family of sets is not enough, to ensure that *shattered-by* also stays finite.

```
lemma ∃F::nat set set. finite F ∧ infinite (shattered-by F)
proof -
  let ?F = {odd -` {True}, odd -` {False}}
  have 0: finite ?F by simp

  let ?f = λn::nat. {n}
  let ?N = range ?f
  have inj (λn. {n}) by simp
  with infinite-iff-countable-subset[of ?N] have infinite-N: infinite ?N by blast
  have F-shatters-any-singleton: ?F shatters {n::nat} for n
  proof -
```

```
    have Pow-n: Pow {n} = {{n}, {}} by blast
    have 1: Pow {n} ⊆ ?F ∩* {n}
    proof (cases odd n)
      case True
      from True have (odd −‘ {False}) ∩ {n} = {} by blast
      hence 0: {} ∈ ?F ∩* {n}  by blast
      from True have (odd −‘ {True}) ∩ {n} = {n} by blast
      hence 1: {n} ∈ ?F ∩* {n}  by blast
      from 0 1 Pow-n show ?thesis by simp
    next
      case False
      from False have (odd −‘ {True}) ∩ {n} = {} by blast
      hence 0: {} ∈ ?F ∩* {n} by blast
      from False have (odd −‘ {False}) ∩ {n} = {n} by blast
      hence 1: {n} ∈ ?F ∩* {n} by blast
      from 0 1 Pow-n show ?thesis by simp
    qed
    thus ?thesis by fastforce
  qed
  then have ?N ⊆ shattered-by ?F unfolding shattered-by-def by force
  from 0 infinite-super[OF this infinite-N] show ?thesis by blast
qed

end
```

3 Lemmas involving the cardinality of sets

In this section, we prove some lemmas that make use of the term *card* or provide bounds for it.

```
theory Card-Lemmas
  imports Main
begin


lemma card-Int-copy:
  assumes finite X and A ∪ B ⊆ X and ∃f. inj-on f (A ∩ B) ∧ (A ∪ B) ∩ (f ‘ (A ∩ B)) = {} ∧ f ‘ (A ∩ B) ⊆ X
  shows card A + card B ≤ card X
proof −
  from rev-finite-subset[OF assms(1), of A] rev-finite-subset[OF assms(1), of B] assms(2)
  have finite-A: finite A and finite-B: finite B by blast+
  then have finite-A-Un-B: finite (A ∪ B) and finite-A-Int-B: finite (A ∩ B) by blast+
  from assms(3) obtain f where f-inj-on: inj-on f (A ∩ B)
                         and f-disjnt: (A ∪ B) ∩ (f ‘ (A ∩ B)) = {}
                         and f-imj-in: f ‘ (A ∩ B) ⊆ X by blast
  from finite-A-Int-B have finite-f-img: finite (f ‘ (A ∩ B)) by blast
  from assms(2) f-imj-in have union-in: (A ∪ B) ∪ f ‘ (A ∩ B) ⊆ X by blast
```

from *card-Un-Int*[*OF finite-A finite-B*] **have** *card A* + *card B* = *card* (*A* ∪ *B*) + *card* (*A* ∩ *B*) **.**
also from *card-image*[*OF f-inj-on*] **have** ... = *card* (*A* ∪ *B*) + *card* (*f* ' (*A* ∩ *B*)) **by** *presburger*
also from *card-Un-disjoint*[*OF finite-A-Un-B finite-f-img f-disjnt*] **have** ... = *card* ((*A* ∪ *B*) ∪ *f* ' (*A* ∩ *B*)) **by** *argo*
also from *card-mono*[*OF assms*(*1*) *union-in*] **have** ... ≤ *card X* **by** *blast*
finally show *?thesis* **.**
qed

lemma *finite-diff-not-empty*:
assumes *finite Y* **and** *card Y* < *card X*
shows *X* − *Y* ≠ {}
proof
assume *X* − *Y* = {}
hence *X* ⊆ *Y* **by** *simp*
from *card-mono*[*OF assms*(*1*) *this*] *assms*(*2*) **show** *False* **by** *linarith*
qed

lemma *obtain-difference-element*:
fixes *F* :: *'a set set*
assumes *2* ≤ *card F*
obtains *x* **where** *x*∈ ⋃ *F x* ∉ ⋂ *F*
proof −
from *assms card-le-Suc-iff*[*of 1 F*] **obtain** *A F'* **where** *0*: *F* = *insert A F'* **and** *1*: *A* ∉ *F'* **and** *2*: *1* ≤ *card F'* **by** *auto*
from *2 card-le-Suc-iff*[*of 0 F'*] **obtain** *B F''* **where** *3*: *F'* = *insert B F''* **by** *auto*
from *1 3* **have** *A-noteq-B*: *A* ≠ *B* **by** *blast*
from *0 3* **have** *A-in-F*: *A* ∈ *F* **and** *B-in-F*: *B* ∈ *F* **by** *blast+*
from *A-noteq-B* **have** (*A* − *B*) ∪ (*B* − *A*) ≠ {} **by** *simp*
with *A-in-F B-in-F that* **show** *thesis* **by** *blast*
qed

end

4 Lemmas involving the binomial coefficient

In this section, we prove lemmas that use the term for the binomial coefficient *choose*.

theory *Binomial-Lemmas*
imports *Main*
begin

lemma *choose-mono*:
assumes *x* ≤ *y*
shows *x choose n* ≤ *y choose n*

```
proof −
  have finite {0..<y} by blast
  with finite-Pow-iff[of {0..<y}] have finiteness: finite {K ∈ Pow {0..<y}. card
K = n} by simp
  from assms have Pow {0..<x} ⊆ Pow {0..<y} by force
  then have {K ∈ Pow {0..<x}. card K = n} ⊆ {K ∈ Pow {0..<y}. card K =
n} by blast
  from card-mono[OF finiteness this] show ?thesis unfolding binomial-def .
qed

lemma choose-row-sum-set:
  assumes finite (⋃F)
  shows card {S. S ⊆ ⋃F ∧ card S ≤ k} = (∑i≤k. card (⋃ F) choose i)
proof (induction k)
  case 0
  from rev-finite-subset[OF assms] have S ⊆ ⋃F ∧ card S ≤ 0 ⟷ S = {} for
S by fastforce
  then show ?case by simp
next
  case (Suc k)
  let ?FS = {S. S ⊆ ⋃ F ∧ card S ≤ Suc k}
  and ?F-Asm = {S. S ⊆ ⋃ F ∧ card S ≤ k}
  and ?F-Step = {S. S ⊆ ⋃ F ∧ card S = Suc k}

  from finite-Pow-iff[of ⋃F] assms have finite-Pow-Un-F: finite (Pow (⋃ F)) ..
  have ?F-Asm ⊆ Pow (⋃ F) and ?F-Step ⊆ Pow (⋃ F) by fast+
   with rev-finite-subset[OF finite-Pow-Un-F] have finite-F-Asm: finite ?F-Asm
and finite-F-Step: finite ?F-Step by presburger+

  have F-Un: ?FS = ?F-Asm ∪ ?F-Step and F-disjoint: ?F-Asm ∩ ?F-Step = {}
by fastforce+
  from card-Un-disjoint[OF finite-F-Asm finite-F-Step F-disjoint] F-Un have card
?FS = card ?F-Asm + card ?F-Step by argo
  also from Suc have ... = (∑i≤k. card (⋃ F) choose i) + card ?F-Step by argo
   also from n-subsets[OF assms, of Suc k] have ... = (∑i≤Suc k. card (⋃ F)
choose i) by force
  finally show ?case by blast
qed

end
```

5 Sauer-Shelah Lemma

```
theory Sauer-Shelah-Lemma
  imports Shattering Card-Lemmas Binomial-Lemmas
begin
```

5.1 Generalized Sauer-Shelah Lemma

To prove the Sauer-Shelah Lemma, we will first prove a slightly stronger fact that every family F shatters at least as many sets as *card F*. We first fix an element $x \in \bigcup F$ and consider the subfamily *F0* of sets in the family not containing it. By induction, *F0* shatters at least as many elements of F as *card F0*. Next, we consider the subfamily *F1* of sets in the family that contain x. Again, by induction, *F1* shatters as many elements of F as its cardinality. The number of elements of F shattered by *F0* and *F1* sum up to at least *card F0* + *card F1* = *card F*. When a set $S \in F$ is shattered by only one of the two subfamilies, say *F0*, it contributes one unit to the set *shattered-by F0* and to *shattered-by F*. However, when the set is shattered by both subfamilies, both S and $S \cup \{x\}$ are in *shattered-by F*, so S contributes two units to *shattered-by F0* ∪ *shattered-by F1*. Therefore, the cardinality of *shattered-by F* is at least equal to the cardinality of *shattered-by F0* ∪ *shattered-by F1*, which is at least *card F*.

lemma *sauer-shelah-0*:
  fixes *F* :: *'a set set*
  shows *finite* (⋃ *F*) ⟹ *card F* ≤ *card* (*shattered-by F*)
proof (*induction F rule*: *measure-induct-rule*[*of card*])
  case (*less F*)
  note *finite-F* = *finite-UnionD*[*OF less*(*2*)]
  note *finite-shF* = *finite-shattered-by*[*OF less*(*2*)]
  show *?case*
  proof (*cases 2* ≤ *card F*)
    case *True*
    from *obtain-difference-element*[*OF True*]
    obtain *x* :: *'a* **where** *x-in-Union-F*: *x* ∈ ⋃*F*
        and *x-not-in-Int-F*: *x* ∉ ⋂*F* **by** *blast*

Define F0 as the subfamily of F containing sets that don't contain x.

    let *?F0* = {*S* ∈ *F*. *x* ∉ *S*}
    from *x-in-Union-F* **have** *F0-psubset-F*: *?F0* ⊂ *F* **by** *blast*
    from *F0-psubset-F* **have** *F0-in-F*: *?F0* ⊆ *F* **by** *blast*
    from *subset-shattered-by*[*OF F0-in-F*] **have** *shF0-subset-shF*: *shattered-by ?F0* ⊆ *shattered-by F* **.**
    from *F0-in-F* **have** *Un-F0-in-Un-F*:⋃ *?F0* ⊆ ⋃ *F* **by** *blast*

F0 shatters at least as many sets as *card F0* by the induction hypothesis.

  note *IH-F0* = *less*(*1*)[*OF psubset-card-mono*[*OF finite-F F0-psubset-F*] *rev-finite-subset*[*OF less*(*2*) *Un-F0-in-Un-F*]]

Define F1 as the subfamily of F containing sets that contain x.

    let *?F1* = {*S* ∈ *F*. *x* ∈ *S*}
    from *x-not-in-Int-F* **have** *F1-psubset-F*: *?F1* ⊂ *F* **by** *blast*
    from *F1-psubset-F* **have** *F1-in-F*: *?F1* ⊆ *F* **by** *blast*
    from *subset-shattered-by*[*OF F1-in-F*] **have** *shF1-subset-shF*: *shattered-by ?F1* ⊆ *shattered-by F* **.**

from *F1-in-F* **have** *Un-F1-in-Un-F*:$\bigcup$ *?F1* $\subseteq \bigcup$ *F* **by** *blast*

F1 shatters at least as many sets as *card F1* by the induction hypothesis.

note *IH-F1* = *less*(*1*)[*OF psubset-card-mono*[*OF finite-F F1-psubset-F*] *rev-finite-subset*[*OF less*(*2*) *Un-F1-in-Un-F*]]

from *shF0-subset-shF shF1-subset-shF*
have *shattered-subset*: (*shattered-by ?F0*) ∪ (*shattered-by ?F1*) ⊆ *shattered-by F* **by** *simp*

There is a set with the same cardinality as the intersection of *shattered-by F0* and *shattered-by F1* which is disjoint from their union and is also contained in *shattered-by F*.

have *f-copies-the-intersection*:
∃*f*. *inj-on f* (*shattered-by ?F0* ∩ *shattered-by ?F1*) ∧
(*shattered-by ?F0* ∪ *shattered-by ?F1*) ∩ (*f* ‘ (*shattered-by ?F0* ∩ *shattered-by ?F1*)) = {} ∧
f ‘ (*shattered-by ?F0* ∩ *shattered-by ?F1*) ⊆ *shattered-by F*
proof
have *x-not-in-shattered*: ∀ *S*∈(*shattered-by ?F0*) ∪ (*shattered-by ?F1*). *x* ∉ *S* **unfolding** *shattered-by-def* **by** *blast*

This set is precisely the image of the intersection under *insert x*.

let *?f* = *insert x*
have *0*: *inj-on ?f* (*shattered-by ?F0* ∩ *shattered-by ?F1*)
proof
fix *X Y*
assume *x0*: *X* ∈ (*shattered-by ?F0* ∩ *shattered-by ?F1*) **and** *y0*: *Y* ∈ (*shattered-by ?F0* ∩ *shattered-by ?F1*)
and *0*: *?f X* = *?f Y*
from *x-not-in-shattered x0* **have** *X* = *?f X* − {*x*} **by** *blast*
also from *0* **have** ... = *?f Y* − {*x*} **by** *argo*
also from *x-not-in-shattered y0* **have** ... = *Y* **by** *blast*
finally show *X* = *Y* **.**
qed

The set is disjoint from the union.

have *1*: (*shattered-by ?F0* ∪ *shattered-by ?F1*) ∩ *?f* ‘ (*shattered-by ?F0* ∩ *shattered-by ?F1*) = {}
proof (*rule ccontr*)
assume (*shattered-by ?F0* ∪ *shattered-by ?F1*) ∩ *?f* ‘ (*shattered-by ?F0* ∩ *shattered-by ?F1*) ≠ {}
then obtain *S* **where** *10*: *S* ∈ (*shattered-by ?F0* ∪ *shattered-by ?F1*)
and *11*: *S* ∈ *?f* ‘ (*shattered-by ?F0* ∩ *shattered-by ?F1*) **by** *auto*
from *10 x-not-in-shattered* **have** *x* ∉ *S* **by** *blast*
with *11* **show** *False* **by** *blast*
qed

This set is also in *shattered-by F*.

have *2*: *?f* ‘ (*shattered-by ?F0* ∩ *shattered-by ?F1*) ⊆ *shattered-by F*
proof
fix *S-x*
assume *S-x* ∈ *?f* ‘ (*shattered-by ?F0* ∩ *shattered-by ?F1*)
then obtain *S* **where** *20*: *S* ∈ *shattered-by ?F0*
and *21*: *S* ∈ *shattered-by ?F1*
and *22*: *S-x* = *?f S* **by** *blast*
from *x-not-in-shattered 20* **have** *x-not-in-S*: *x* ∉ *S* **by** *blast*

from *22 Pow-insert*[*of x S*] **have** *Pow S-x* = *Pow S* ∪ *?f* ‘ *Pow S* **by** *fast*
also from *20* **have** ... = (*?F0* ∩∗ *S*) ∪ (*?f* ‘ *Pow S*) **unfolding** *shattered-by-def* **by** *blast*
also from *21* **have** ... = (*?F0* ∩∗ *S*) ∪ (*?f* ‘ (*?F1* ∩∗ *S*)) **unfolding** *shattered-by-def* **by** *force*
also from *insert-IntF*[*of x S ?F1*] **have** ... = (*?F0* ∩∗ *S*) ∪ (*?f* ‘ *?F1* ∩∗ (*?f S*)) **by** *argo*
also from *22* **have** ... = (*?F0* ∩∗ *S*) ∪ (*?F1* ∩∗ *S-x*) **by** *blast*
also from *22* **have** ... = (*?F0* ∩∗ *S-x*) ∪ (*?F1* ∩∗ *S-x*) **by** *blast*
also from *subset-IntF*[*OF F0-in-F*, *of S-x*] *subset-IntF*[*OF F1-in-F*, *of S-x*] **have** ... ⊆ (*F* ∩∗ *S-x*) **by** *blast*
finally have *Pow S-x* ⊆ (*F* ∩∗ *S-x*) **.**
thus *S-x* ∈ *shattered-by F* **unfolding** *shattered-by-def* **by** *blast*
qed

from *0 1 2* **show** *inj-on ?f* (*shattered-by ?F0* ∩ *shattered-by ?F1*) ∧
(*shattered-by ?F0* ∪ *shattered-by ?F1*) ∩ (*?f* ‘ (*shattered-by ?F0* ∩ *shattered-by ?F1*)) = {} ∧
?f ‘ (*shattered-by ?F0* ∩ *shattered-by ?F1*) ⊆ *shattered-by F* **by** *blast*
qed

have *F0-union-F1-is-F*: *?F0* ∪ *?F1* = *F* **by** *fastforce*
from *finite-F* **have** *finite-F0*: *finite ?F0* **and** *finite-F1*: *finite ?F1* **by** *fastforce+*
have *disjoint-F0-F1*: *?F0* ∩ *?F1* = {} **by** *fastforce*

We have the following lower bound on the cardinality of *shattered-by F*:

from *F0-union-F1-is-F card-Un-disjoint*[*OF finite-F0 finite-F1 disjoint-F0-F1*]

have *card F* = *card ?F0* + *card ?F1* **by** *argo*
also from *IH-F0*
have ... ≤ *card* (*shattered-by ?F0*) + *card ?F1* **by** *linarith*
also from *IH-F1*
have ... ≤ *card* (*shattered-by ?F0*) + *card* (*shattered-by ?F1*) **by** *linarith*
also from *card-Int-copy*[*OF finite-shF shattered-subset f-copies-the-intersection*]
have ... ≤ *card* (*shattered-by F*) **by** *argo*
finally show *?thesis* **.**
next

If F contains less than 2 sets, the statement follows trivially.

case *False*

hence *card F* = *0* ∨ *card F* = *1* **by** *force*
thus *?thesis*
proof
assume *card F* = *0*
thus *?thesis* **by** *auto*
next
assume *asm*: *card F* = *1*
hence *F-not-empty*: *F* ≠ {} **by** *fastforce*
from *shatters-empty*[*OF F-not-empty*] **have** {{}} ⊆ *shattered-by F* **unfolding** *shattered-by-def* **by** *fastforce*
from *card-mono*[*OF finite-shF this*] *asm* **show** *?thesis* **by** *fastforce*
qed
qed
qed

5.2 Sauer-Shelah Lemma

The generalized version immediately implies the Sauer-Shelah Lemma, because only ($\sum i{\le}k.$ *n choose i*) of the subsets of an *n*-item universe have cardinality less than *k* + *1*. Thus, when ($\sum i{\le}k.$ *n choose i*) < *card F*, there are not enough sets to be shattered, so one of the shattered sets must have cardinality at least *k* + *1*.

corollary *sauer-shelah*:
fixes *F* :: *'a set set*
assumes *finite* ($\bigcup$ *F*) **and** ($\sum i{\le}k.$ *card* ($\bigcup$ *F*) *choose i*) < *card F*
shows ∃ *S*. (*F shatters S* ∧ *card S* = *k* + *1*)
proof −
let *?K* = {*S*. *S* ⊆ $\bigcup$ *F* ∧ *card S* ≤ *k*}
from *finite-Pow-iff*[*of F*] *assms*(*1*) **have** *finite-Pow-Un*: *finite* (*Pow* ($\bigcup$ *F*)) **by** *fast*

from *sauer-shelah-0*[*OF assms*(*1*)] *assms*(*2*) **have** ($\sum i{\le}k.$ *card* ($\bigcup$ *F*) *choose i*) < *card* (*shattered-by F*) **by** *linarith*
with *choose-row-sum-set*[*OF assms*(*1*), *of k*] **have** *card ?K* < *card* (*shattered-by F*) **by** *presburger*

from *finite-diff-not-empty*[*OF finite-subset*[*OF - finite-Pow-Un*] *this*]
obtain *S* **where** *S* ∈ *shattered-by F* − *?K* **by** *blast*
then have *F-shatters-S*: *F shatters S* **and** *S* ⊆ $\bigcup$ *F* **and** ¬(*S* ⊆ $\bigcup$ *F* ∧ *card S* ≤ *k*) **unfolding** *shattered-by-def* **by** *blast*+
then have *card-S-ge-Suc-k*: *k* + *1* ≤ *card S* **by** *simp*
from *obtain-subset-with-card-n*[*OF card-S-ge-Suc-k*] **obtain** *S′* **where** *card S′* = *k* + *1* **and** *S′* ⊆ *S* **by** *blast*
from *this*(*1*) *supset-shatters*[*OF this*(*2*) *F-shatters-S*] **show** *?thesis* **by** *blast*
qed

5.3 Sauer-Shelah Lemma for hypergraphs

If we designate X to be the set of hyperedges and S the set of vertices, we can also formulate the Sauer-Shelah Lemma in terms of hypergraphs. In this form, the statement provides a sufficient condition for the existence of an hyperedge of a given cardinality which is shattered by the set of edges.

corollary *sauer-shelah-2*:
  fixes X :: *'a set set* **and** S :: *'a set*
  assumes *finite S* **and** $X \subseteq$ *Pow S* **and** $(\sum i \leq k.$ *card S choose i*$) <$ *card X*
  shows $\exists Y.$ (*X shatters Y* $\wedge$ *card Y* $= k + 1$)
proof −
  from *assms*(*2*) **have** *0*: $\bigcup X \subseteq S$ **by** *blast*
  then have $(\sum i \leq k.$ *card* $(\bigcup X)$ *choose i*$) \leq (\sum i \leq k.$ *card S choose i*$)$
    by (*simp add*: *assms*(*1*) *card-mono choose-mono sum-mono*)
  then show *?thesis*
    using *0 assms finite-subset sauer-shelah* **by** *fastforce*
qed

5.4 Alternative statement of the Sauer-Shelah Lemma

We can also state the Sauer-Shelah Lemma in terms of the *VC-dim*. If the VC-dimension of F is k then F can consist at most of $(\sum i \leq k.$ *card* $(\bigcup F)$ *choose i*$)$ sets which is in $\mathcal{O}($*card* $(\bigcup F)$ ^$k)$.

corollary *sauer-shelah-alt*:
  assumes *finite* $(\bigcup F)$ **and** *VC-dim F* $= k$
  shows *card F* $\leq (\sum i \leq k.$ *card* $(\bigcup F)$ *choose i*$)$
proof (*rule ccontr*)
  assume ¬ *card F* $\leq (\sum i \leq k.$ *card* $(\bigcup F)$ *choose i*$)$ **hence** $(\sum i \leq k.$ *card* $(\bigcup F)$ *choose i*$) <$ *card F* **by** *linarith*
  then obtain *S* **where** *F shatters S* **and** *card S* $= k + 1$
    by (*meson assms*(*1*) *sauer-shelah*)
  then have §: $k + 1 \in \{$*card S* $\mid$ *S. F shatters S*$\}$
    by *simp metis*
  have *finite* {*A. F shatters A*}
    by (*metis* ‹*finite* $(\bigcup F)$› *finite-shattered-by shattered-by-def*)
  then have *bdd-above* {*card A* |*A. F shatters A*}
    by *simp*
  then have $k + 1 \leq$ *Sup* {*card A* |*A. F shatters A*}
    by (*smt* (*verit*, *best*) § *cSup-upper*)
  then have $k + 1 \leq$ *VC-dim F*
    by (*simp add*: *VC-dim-def*)
  then show *False*
    using *assms*(*2*) **by** *auto*
qed

end

References

[1] G. Kalai. Extremal combinatorics iii: Some basic theorems, Sep 2008.

[2] N. Sauer. On the density of families of sets. *Journal of Combinatorial Theory, Series A*, 13(1):145–147, 1972.

[3] S. Shelah. A combinatorial problem; stability and order for models and theories in infinitary languages. *Pacific Journal of Mathematics*, 41(1):247 – 261, 1972.

[4] V. N. Vapnik and A. J. Červonenkis. The uniform convergence of frequencies of the appearance of events to their probabilities. *Teor. Verojatnost. i Primenen.*, 16:264–279, 1971.