

# Concentration Inequalities

Emin Karayel and Yong Kiam Tan\*

February 6, 2026

## Abstract

Concentration inequalities provide bounds on how a random variable (or a sum/composition of random variables) deviate from their expectation, usually based on moments/independence of the variables.

The most important concentration inequalities (the Markov, Chebyshev, and Hoelder inequalities and the Chernoff bounds) are already part of HOL-Probability. This entry collects more advanced results, such as Bennett's/Bernstein's Inequality, Bienaymé's Identity, Cantelli's Inequality, the Efron-Stein Inequality, McDiarmid's Inequality, and the Paley-Zygmund Inequality.

## Contents

|          |                                 |           |
|----------|---------------------------------|-----------|
| <b>1</b> | <b>Preliminary results</b>      | <b>1</b>  |
| <b>2</b> | <b>Bennett's Inequality</b>     | <b>8</b>  |
| <b>3</b> | <b>Bienaymé's identity</b>      | <b>20</b> |
| <b>4</b> | <b>Cantelli's Inequality</b>    | <b>24</b> |
| <b>5</b> | <b>Efron-Stein Inequality</b>   | <b>27</b> |
| <b>6</b> | <b>McDiarmid's inequality</b>   | <b>34</b> |
| <b>7</b> | <b>Paley-Zygmund Inequality</b> | <b>54</b> |

## 1 Preliminary results

```
theory Concentration-Inequalities-Preliminary
  imports Lp.Lp
begin
```

Version of Cauchy-Schwartz for the Lebesgue integral:

---

\*The authors contributed equally to this work.

**lemma** *cauchy-schwartz*:  
**fixes**  $f\ g :: - \Rightarrow \text{real}$   
**assumes**  $f \in \text{borel-measurable } M\ g \in \text{borel-measurable } M$   
**assumes**  $\text{integrable } M (\lambda x. (f\ x) \wedge 2)\ \text{integrable } M (\lambda x. (g\ x) \wedge 2)$   
**shows**  $\text{integrable } M (\lambda x. f\ x * g\ x)$  (**is** ?A)  
 $(\int x. f\ x * g\ x\ \partial M) \leq (\int x. (f\ x) \wedge 2\ \partial M)\ \text{powr } (1/2) * (\int x. (g\ x) \wedge 2\ \partial M)\ \text{powr } (1/2)$   
**(is** ?L  $\leq$  ?R)  
**proof** –  
**show** 0: ?A  
**using** *assms* **by** (*intro Holder-inequality(1)*[**where**  $p=2$  **and**  $q=2$ ])  
*auto*  
  
**have** ?L  $\leq (\int x. |f\ x * g\ x|\ \partial M)$   
**using** 0 **by** (*intro integral-mono*) *auto*  
**also have** ...  $\leq (\int x. |f\ x|\ \text{powr } 2\ \partial M)\ \text{powr } (1/2) * (\int x. |g\ x|\ \text{powr } 2\ \partial M)\ \text{powr } (1/2)$   
**using** *assms* **by** (*intro Holder-inequality(2)*) *auto*  
**also have** ... = ?R **by** *simp*  
**finally show** ?L  $\leq$  ?R **by** *simp*  
**qed**

Generalization of *prob-space.indep-vars-iff-distr-eq-PiM'*:

**lemma** (**in** *prob-space*) *indep-vars-iff-distr-eq-PiM''*:  
**fixes**  $I :: 'i\ \text{set}$  **and**  $X :: 'i \Rightarrow 'a \Rightarrow 'b$   
**assumes**  $rv: \bigwedge i. i \in I \implies \text{random-variable } (M'\ i)\ (X\ i)$   
**shows**  $\text{indep-vars } M'\ X\ I \longleftrightarrow$   
 $\text{distr } M\ (\Pi_M\ i \in I. M'\ i)\ (\lambda x. \lambda i \in I. X\ i\ x) = (\Pi_M\ i \in I. \text{distr } M\ (M'\ i)\ (X\ i))$   
**proof** (*cases*  $I = \{\}$ )  
**case** *True*  
**have** 0:  $\text{indicator } A\ (\lambda -. \text{undefined}) = \text{emeasure } (\text{count-space } \{\lambda -. \text{undefined}\})\ A$  (**is** ?L = ?R)  
**if**  $A \subseteq \{\lambda -. \text{undefined}\}$  **for**  $A :: ('i \Rightarrow 'b)\ \text{set}$   
**proof** –  
**have** 1:  $A \neq \{\} \implies A = \{\lambda -. \text{undefined}\}$   
**using** *that* **by** *auto*  
  
**have** ?R = *of-nat* (*card*  $A$ )  
**using** *finite-subset that* **by** (*intro emeasure-count-space-finite that*)  
*auto*  
**also have** ... = ?L  
**using** 1 **by** (*cases*  $A = \{\}$ ) *auto*  
**finally show** ?thesis **by** *simp*  
**qed**  
  
**have**  $\text{distr } M\ (\Pi_M\ i \in I. M'\ i)\ (\lambda x. \lambda i \in I. X\ i\ x) =$   
 $\text{distr } M\ (\text{count-space } \{\lambda -. \text{undefined}\})\ (\lambda -. (\lambda -. \text{undefined}))$   
**unfolding** *True PiM-empty* **by** (*intro distr-cong*) (*auto simp: restrict-def*)

**also have** ... = return (count-space {λ-. undefined}) (λ-. undefined)  
**by** (intro distr-const) auto  
**also have** ... = count-space ({λ-. undefined} :: ('i ⇒ 'b) set)  
**by** (intro measure-eqI) (auto simp:0)  
**also have** ... = (Π<sub>M</sub> i∈I. distr M (M' i) (X i))  
**unfolding** True PiM-empty **by** simp  
**finally have** distr M (Π<sub>M</sub> i∈I. M' i) (λx. λi∈I. X i x) = (Π<sub>M</sub> i∈I.  
distr M (M' i) (X i)) ↔ True  
**by** simp  
**also have** ... ↔ indep-vars M' X I  
**unfolding** indep-vars-def **by** (auto simp add: space-PiM indep-sets-def)  
(auto simp add: True)  
**finally show** ?thesis **by** simp  
**next**  
**case** False  
**thus** ?thesis  
**by** (intro indep-vars-iff-distr-eq-PiM' assms) auto  
**qed**

**lemma** proj-indep:

**assumes**  $\bigwedge i. i \in I \implies \text{prob-space } (M i)$   
**shows** prob-space.indep-vars (PiM I M) M (λi ω. ω i) I

**proof** –

**interpret** prob-space (PiM I M)  
**by** (intro prob-space-PiM assms)

**have** distr (Pi<sub>M</sub> I M) (Pi<sub>M</sub> I M) (λx. restrict x I) = PiM I M  
**by** (intro distr-PiM-reindex assms) auto

**also have** ... = Pi<sub>M</sub> I (λi. distr (Pi<sub>M</sub> I M) (M i) (λω. ω i))  
**by** (intro PiM-cong refl distr-PiM-component[symmetric] assms)

**finally have**  
distr (Pi<sub>M</sub> I M) (Pi<sub>M</sub> I M) (λx. restrict x I) = Pi<sub>M</sub> I (λi. distr  
(Pi<sub>M</sub> I M) (M i) (λω. ω i))

**by** simp  
**thus** indep-vars M (λi ω. ω i) I  
**by** (intro iffD2[OF indep-vars-iff-distr-eq-PiM'']) simp-all

**qed**

**lemma** forall-Pi-to-PiE:

**assumes**  $\bigwedge x. P x = P (\text{restrict } x I)$   
**shows**  $(\forall x \in \text{Pi } I A. P x) = (\forall x \in \text{PiE } I A. P x)$   
**using** assms **by** (simp add: PiE-def Pi-def set-eq-iff, force)

**lemma** PiE-reindex:

**assumes** inj-on f I  
**shows** PiE I (A ∘ f) = (λa. restrict (a ∘ f) I) ' PiE (f ' I) A (is  
?lhs = ?g ' ?rhs)

**proof** –

**have** ?lhs ⊆ ?g ' ?rhs

**proof** (*rule subsetI*)  
**fix**  $x$   
**assume**  $a: x \in \text{PiE } I (A \circ f)$   
**define**  $y$  **where**  $y\text{-def}: y = (\lambda k. \text{if } k \in f^{-1} I \text{ then } x \text{ (the-inv-into } I$   
 $f k) \text{ else undefined})$   
**have**  $b: y \in \text{PiE } (f^{-1} I) A$   
**using**  $a$  *assms the-inv-into-f-eq [OF assms]*  
**by** (*simp add: y-def PiE-iff extensional-def*)  
**have**  $c: x = (\lambda a. \text{restrict } (a \circ f) I) y$   
**using**  $a$  *assms the-inv-into-f-eq extensional-arb*  
**by** (*intro ext, simp add: y-def PiE-iff, fastforce*)  
**show**  $x \in ?g^{-1} ?rhs$  **using**  $b c$  **by** *blast*  
**qed**  
**moreover** **have**  $?g^{-1} ?rhs \subseteq ?lhs$   
**by** (*rule image-subsetI, simp add: Pi-def PiE-def*)  
**ultimately** **show** *?thesis* **by** *blast*  
**qed**

**context** *prob-space*  
**begin**

**lemma** *indep-sets-reindex:*

**assumes** *inj-on f I*  
**shows** *indep-sets A (f^{-1} I) = indep-sets (\lambda i. A (f i)) I*  
**proof** –  
**have**  $a: \bigwedge J. J \subseteq I \implies (\prod j \in f^{-1} J. g j) = (\prod j \in J. g (f j))$   
**by** (*metis assms prod.reindex-cong inj-on-subset*)  
  
**have**  $b: J \subseteq I \implies (\prod_E i \in J. A (f i)) = (\lambda a. \text{restrict } (a \circ f) J) A$   
 $\text{PiE } (f^{-1} J) A$  **for**  $J$   
**using** *assms inj-on-subset*  
**by** (*subst PiE-reindex[symmetric] auto*)  
  
**have**  $c: \bigwedge J. J \subseteq I \implies \text{finite } (f^{-1} J) = \text{finite } J$   
**by** (*meson assms finite-image-iff inj-on-subset*)  
  
**show** *?thesis*  
**by** (*simp add: indep-sets-def all-subset-image a c*) (*simp-all add: forall-Pi-to-PiE*  
 $b$ )  
**qed**

**lemma** *indep-vars-reindex:*

**assumes** *inj-on f I*  
**assumes** *indep-vars M' X' (f^{-1} I)*  
**shows** *indep-vars (M' \circ f) (\lambda k \omega. X' (f k) \omega) I*  
**using** *assms* **by** (*simp add: indep-vars-def2 indep-sets-reindex*)

**lemma** *indep-vars-cong-AE:*

**assumes** *AE x in M. (\forall i \in I. X' i x = Y' i x)*

**assumes** *indep-vars*  $M' X' I$   
**assumes**  $\bigwedge i. i \in I \implies \text{random-variable } (M' i) (Y' i)$   
**shows** *indep-vars*  $M' Y' I$   
**proof** –  
**have**  $a: AE\ x\ \text{in}\ M. (\lambda i \in I. Y' i\ x) = (\lambda i \in I. X' i\ x)$   
**by** (*rule*  $AE\text{-mp}[OF\ \text{assms}(1)]$ , *rule*  $AE\text{-I2}$ , *simp cong:restrict-cong*)  
**have**  $b: \bigwedge i. i \in I \implies \text{random-variable } (M' i) (X' i)$   
**using**  $\text{assms}(2)$  **by** (*simp add:indep-vars-def2*)  
**have**  $c: \bigwedge x. x \in I \implies AE\ xa\ \text{in}\ M. X' x\ xa = Y' x\ xa$   
**by** (*rule*  $AE\text{-mp}[OF\ \text{assms}(1)]$ , *rule*  $AE\text{-I2}$ , *simp*)  
  
**have**  $\text{distr } M (Pi_M\ I\ M') (\lambda x. \lambda i \in I. Y' i\ x) = \text{distr } M (Pi_M\ I\ M')$   
 $(\lambda x. \lambda i \in I. X' i\ x)$   
**by** (*intro distr-cong-AE measurable-restrict a b assms(3)*) *auto*  
**also have**  $\dots = Pi_M\ I (\lambda i. \text{distr } M (M' i) (X' i))$   
**using**  $\text{assms } b$  **by** (*subst indep-vars-iff-distr-eq-PiM''[symmetric]*)  
*auto*  
**also have**  $\dots = Pi_M\ I (\lambda i. \text{distr } M (M' i) (Y' i))$   
**by** (*intro PiM-cong distr-cong-AE c assms(3) b*) *auto*  
**finally have**  $\text{distr } M (Pi_M\ I\ M') (\lambda x. \lambda i \in I. Y' i\ x) = Pi_M\ I (\lambda i.$   
 $\text{distr } M (M' i) (Y' i))$   
**by** *simp*  
  
**thus** *?thesis*  
**using**  $\text{assms}(3)$   
**by** (*subst indep-vars-iff-distr-eq-PiM''*) *auto*  
**qed**

**end**

Integrability of bounded functions on finite measure spaces:

**lemma** *bounded-const*:  $\text{bounded } ((\lambda x. (c::\text{real})) ' T)$   
**by** (*intro boundedI[where B=norm c]*) *auto*

**lemma** *bounded-exp*:  
**fixes**  $f :: 'a \Rightarrow \text{real}$   
**assumes**  $\text{bounded } ((\lambda x. f\ x) ' T)$   
**shows**  $\text{bounded } ((\lambda x. \exp (f\ x)) ' T)$

**proof** –  
**obtain**  $m$  **where**  $\text{norm } (f\ x) \leq m$  **if**  $x \in T$  **for**  $x$   
**using**  $\text{assms}$  **unfolding** *bounded-iff* **by** *auto*

**thus** *?thesis*  
**by** (*intro boundedI[where B=exp m]*) *fastforce*  
**qed**

**lemma** *bounded-mult-comp*:  
**fixes**  $f :: 'a \Rightarrow \text{real}$   
**assumes**  $\text{bounded } (f ' T)$   $\text{bounded } (g ' T)$

**shows** *bounded*  $((\lambda x. (f x) * (g x)) \text{ ' } T)$   
**proof** –  
**obtain**  $m1$  **where**  $norm (f x) \leq m1$   $m1 \geq 0$  **if**  $x \in T$  **for**  $x$   
**using** *assms unfolding bounded-iff* **by** *fastforce*  
**moreover obtain**  $m2$  **where**  $norm (g x) \leq m2$   $m2 \geq 0$  **if**  $x \in T$   
**for**  $x$   
**using** *assms unfolding bounded-iff* **by** *fastforce*  
  
**ultimately show** *?thesis*  
**by** (*intro boundedI*[**where**  $B=m1 * m2$ ]) (*auto intro!*: *mult-mono simp:abs-mult*)  
**qed**

**lemma** *bounded-sum*:  
**fixes**  $f :: 'i \Rightarrow 'a \Rightarrow real$   
**assumes** *finite I*  
**assumes**  $\bigwedge i. i \in I \implies bounded (f i \text{ ' } T)$   
**shows** *bounded*  $((\lambda x. (\sum i \in I. f i x)) \text{ ' } T)$   
**using** *assms* **by** (*induction I*) (*auto intro:bounded-plus-comp bounded-const*)

**lemma** *bounded-pow*:  
**fixes**  $f :: 'a \Rightarrow real$   
**assumes** *bounded*  $((\lambda x. f x) \text{ ' } T)$   
**shows** *bounded*  $((\lambda x. (f x)^{\hat{n}}) \text{ ' } T)$   
**proof** –  
**obtain**  $m$  **where**  $norm (f x) \leq m$  **if**  $x \in T$  **for**  $x$   
**using** *assms unfolding bounded-iff* **by** *auto*  
**hence**  $norm ((f x)^{\hat{n}}) \leq m^{\hat{n}}$  **if**  $x \in T$  **for**  $x$   
**using** *that unfolding norm-power* **by** (*intro power-mono*) *auto*  
**thus** *?thesis* **by** (*intro boundedI*[**where**  $B=m^{\hat{n}}$ ]) *auto*  
**qed**

**lemma** *bounded-sum-list*:  
**fixes**  $f :: 'i \Rightarrow 'a \Rightarrow real$   
**assumes**  $\bigwedge y. y \in set ys \implies bounded (f y \text{ ' } T)$   
**shows** *bounded*  $((\lambda x. (\sum y \leftarrow ys. f y x)) \text{ ' } T)$   
**using** *assms* **by** (*induction ys*) (*auto intro:bounded-plus-comp bounded-const*)

**lemma** (*in finite-measure*) *bounded-int*:  
**fixes**  $f :: 'i \Rightarrow 'a \Rightarrow real$   
**assumes** *bounded*  $((\lambda x. f (fst x) (snd x)) \text{ ' } (T \times space M))$   
**shows** *bounded*  $((\lambda x. (\int \omega. (f x \omega) \partial M)) \text{ ' } T)$   
**proof** –  
**obtain**  $m$  **where**  $\bigwedge x y. x \in T \implies y \in space M \implies norm (f x y) \leq m$   
**using** *assms unfolding bounded-iff* **by** *auto*  
**hence**  $m: \bigwedge x y. x \in T \implies y \in space M \implies norm (f x y) \leq max m 0$   
**by** *fastforce*

```

have norm (∫ ω. (f x ω) ∂M) ≤ max m 0 * measure M (space M)
(is ?L ≤ ?R) if x ∈ T for x
proof –
  have ?L ≤ (∫ ω. norm (f x ω) ∂M) by simp
  also have ... ≤ (∫ ω. max m 0 ∂M)
    using that m by (intro integral-mono') auto
  also have ... = ?R
    by simp
  finally show ?thesis by simp
qed
thus ?thesis
  by (intro boundedI[where B=max m 0 * measure M (space M)])
auto
qed

```

```

lemmas bounded-intros =
  bounded-minus-comp bounded-plus-comp bounded-mult-comp bounded-sum
  finite-measure.bounded-int
  bounded-const bounded-exp bounded-pow bounded-sum-list

```

```

lemma (in prob-space) integrable-bounded:
  fixes f :: - ⇒ ('b :: {banach,second-countable-topology})
  assumes bounded (f ' space M)
  assumes f ∈ M →M borel
  shows integrable M f
proof –
  obtain m where norm (f x) ≤ m if x ∈ space M for x
    using assms(1) unfolding bounded-iff by auto
  thus ?thesis
    by (intro integrable-const-bound[where B=m] AE-I2 assms(2))
qed

```

```

lemma integrable-bounded-pmf:
  fixes f :: - ⇒ ('b :: {banach,second-countable-topology})
  assumes bounded (f ' set-pmf M)
  shows integrable (measure-pmf M) f
proof –
  obtain m where norm (f x) ≤ m if x ∈ set-pmf M for x
    using assms(1) unfolding bounded-iff by auto
  thus ?thesis by (intro measure-pmf.integrable-const-bound[where
  B=m] AE-pmfI) auto
qed

```

**end**

## 2 Bennett's Inequality

In this section we verify Bennett's inequality [1] and a (weak) version of Bernstein's inequality as a corollary. Both inequalities give concentration bounds for sums of independent random variables. The statement and proofs follow a summary paper by Boucheron et al. [2].

**theory** *Bennett-Inequality*

**imports** *Concentration-Inequalities-Preliminary*

**begin**

**context** *prob-space*

**begin**

**lemma** *indep-vars-Chernoff-ineq-ge:*

**assumes** *I: finite I*

**assumes** *ind: indep-vars ( $\lambda$  -. borel) X I*

**assumes** *sge:  $s \geq 0$*

**assumes** *int:  $\bigwedge i. i \in I \implies$  integrable M ( $\lambda x. \exp(s * X i x)$ )*

**shows** *prob { $x \in$  space M. ( $\sum i \in I. X i x - \text{expectation } (X i)$ )  $\geq$  t}  $\leq$*

*$\exp(-s*t) *$*

*$(\prod i \in I. \text{expectation } (\lambda x. \exp(s * (X i x - \text{expectation } (X i))))$ )*

**proof** (*cases s = 0*)

**case** [*simp*]: *True*

**thus** *?thesis*

**by** (*simp add: prob-space*)

**next**

**case** *False*

**then have** *s:  $s > 0$  using sge by auto*

**have** [*measurable*]:  *$\bigwedge i. i \in I \implies$  random-variable borel (X i)*

**using** *ind unfolding indep-vars-def by blast*

**have** *indep1: indep-vars ( $\lambda$ -. borel)*

*( $\lambda i \omega. \exp(s * (X i \omega - \text{expectation } (X i)))$ ) I*

**apply** (*intro indep-vars-compose[OF ind, unfolded o-def]*)

**by** *auto*

**define** *S where S = ( $\lambda x. (\sum i \in I. X i x - \text{expectation } (X i))$ )*

**have** *int1:  $\bigwedge i. i \in I \implies$*

*integrable M ( $\lambda \omega. \exp(s * (X i \omega - \text{expectation } (X i)))$ )*

**by** (*auto simp add: algebra-simps exp-diff int*)

**have** *eprod:  $\bigwedge x. \exp(s * S x) = (\prod i \in I. \exp(s * (X i x - \text{expectation } (X i))))$*

**unfolding** *S-def*  
**by** (*simp add: assms(1) exp-sum vector-space-over-itself.scale-sum-right*)

**from** *indep-vars-integrable[OF I indep1 int1]*  
**have** *intS: integrable M (λx. exp (s \* S x))*  
**unfolding** *eprod by auto*

**then have** *si: set-integrable M (space M) (λx. exp (s \* S x))*  
**unfolding** *set-integrable-def*  
**apply** (*intro integrable-mult-indicator*)  
**by** *auto*

**from** *Chernoff-ineq-ge[OF s si]*  
**have** *prob {x ∈ space M. S x ≥ t} ≤ exp (- s \* t) \* (∫ x∈space M. exp (s \* S x) ∂M)*  
**by** *auto*

**also have** ( $\int x \in \text{space } M. \exp (s * S x) \partial M = \text{expectation } (\lambda x. \exp (s * S x))$ )  
**unfolding** *set-integral-space[OF intS] by auto*

**also have**  $\dots = \text{expectation } (\lambda x. \prod_{i \in I}. \exp (s * (X i x - \text{expectation } (X i))))$   
**unfolding** *S-def*  
**by** (*simp add: assms(1) exp-sum vector-space-over-itself.scale-sum-right*)  
**also have**  $\dots = (\prod_{i \in I}. \text{expectation } (\lambda x. \exp (s * (X i x - \text{expectation } (X i))))$   
**apply** (*intro indep-vars-lebesgue-integral[OF I indep1 int1]*) .  
**finally show** *?thesis*  
**unfolding** *S-def*  
**by** *auto*

**qed**

**definition** *bennett-h::real ⇒ real*  
**where** *bennett-h u = (1 + u) \* ln (1 + u) - u*

**lemma** *exp-sub-two-terms-eq:*  
**fixes** *x :: real*  
**shows**  $\exp x - x - 1 = (\sum n. x^{\wedge}(n+2) / \text{fact } (n+2))$   
 $\text{summable } (\lambda n. x^{\wedge}(n+2) / \text{fact } (n+2))$   
**proof** –  
**have** ( $\sum_{i < 2}. \text{inverse } (\text{fact } i) * x^{\wedge} i = 1 + x$ )  
**by** (*simp add:numeral-eq-Suc*)  
**thus**  $\exp x - x - 1 = (\sum n. x^{\wedge}(n+2) / \text{fact } (n+2))$   
**unfolding** *exp-def*  
**apply** (*subst suminf-split-initial-segment[where k = 2]*)  
**by** (*auto simp add: summable-exp divide-inverse-commute*)  
**have** *summable (λn. x^ $\wedge$ n / fact n)*  
**by** (*simp add: divide-inverse-commute summable-exp*)

**then have**  $\text{summable } (\lambda n. x^{\wedge}(\text{Suc } (\text{Suc } n)) / \text{fact } (\text{Suc } (\text{Suc } n)))$   
**apply** (*subst summable-Suc-iff*)  
**apply** (*subst summable-Suc-iff*)  
**by** *auto*  
**thus**  $\text{summable } (\lambda n. x^{\wedge}(n+2) / \text{fact } (n+2))$  **by** *auto*  
**qed**

**lemma** *psi-mono*:

**defines**  $f \equiv (\lambda x. (\text{exp } x - x - 1) - x^{\wedge}2 / 2)$

**assumes**  $xy: a \leq (b::\text{real})$

**shows**  $f a \leq f b$

**proof** –

**have** 1: (*f has-real-derivative* ( $\text{exp } x - x - 1$ )) (*at x*) **for**  $x$

**unfolding** *f-def*

**by** (*auto intro!*; *derivative-eq-intros*)

**have** 2:  $\bigwedge x. x \in \{a..b\} \implies 0 \leq \text{exp } x - x - 1$

**by** (*smt* (*verit*) *exp-ge-add-one-self*)

**from** *deriv-nonneg-imp-mono*[*OF* 1 2  $xy$ ]

**show** *?thesis* **by** *auto*

**qed**

**lemma** *psi-inequality*:

**assumes**  $le: x \leq (y::\text{real}) \ y \geq 0$

**shows**  $y^{\wedge}2 * (\text{exp } x - x - 1) \leq x^{\wedge}2 * (\text{exp } y - y - 1)$

**proof** –

**have**  $x: \text{exp } x - x - 1 = (\sum n. (x^{\wedge}(n+2) / \text{fact } (n+2)))$

*summable* ( $\lambda n. x^{\wedge}(n+2) / \text{fact } (n+2)$ )

**using** *exp-sub-two-terms-eq* .

**have**  $y: \text{exp } y - y - 1 = (\sum n. (y^{\wedge}(n+2) / \text{fact } (n+2)))$

*summable* ( $\lambda n. y^{\wedge}(n+2) / \text{fact } (n+2)$ )

**using** *exp-sub-two-terms-eq* .

**have**  $l: y^{\wedge}2 * (\text{exp } x - x - 1) = (\sum n. y^{\wedge}2 * (x^{\wedge}(n+2) / \text{fact } (n+2)))$

**using**  $x$

**apply** (*subst suminf-mult*)

**by** *auto*

**have**  $ls: \text{summable } (\lambda n. y^{\wedge}2 * (x^{\wedge}(n+2) / \text{fact } (n+2)))$

**by** (*intro summable-mult*[*OF*  $x(2)$ ])

**have**  $r: x^{\wedge}2 * (\text{exp } y - y - 1) = (\sum n. x^{\wedge}2 * (y^{\wedge}(n+2) / \text{fact } (n+2)))$

**using**  $y$

```

apply (subst suminf-mult)
by auto
have rs: summable ( $\lambda n. x^2 * (y^{(n+2)} / \text{fact } (n+2))$ )
by (intro summable-mult[OF y(2)])

have  $|x| \leq |y| \vee |y| < |x|$  by auto
moreover {
  assume  $|x| \leq |y|$ 
  then have  $x^n \leq y^n$  for  $n$ 
  by (smt (verit, ccfv-threshold) bot-nat-0.not-eq-extremum le power-0
real-root-less-mono real-root-power-cancel root-abs-power)
  then have  $(x^2 * y^2) * x^n \leq (x^2 * y^2) * y^n$  for  $n$ 
  by (simp add: mult-left-mono)
  then have  $y^2 * (x^{(n+2)}) \leq x^2 * (y^{(n+2)})$  for  $n$ 
  by (metis (full-types) ab-semigroup-mult-class.mult-ac(1) mult.commute
power-add)
  then have  $y^2 * (x^{(n+2)} / \text{fact } (n+2)) \leq x^2 * (y^{(n+2)} / \text{fact } (n+2))$ 
/ fact (n+2) for  $n$ 
  by (meson divide-right-mono fact-ge-zero)
  then have  $(\sum n. y^2 * (x^{(n+2)} / \text{fact } (n+2))) \leq (\sum n. x^2 * (y^{(n+2)} / \text{fact } (n+2)))$ 
apply (intro suminf-le[OF - ls rs])
by auto
  then have  $y^2 * (\exp x - x - 1) \leq x^2 * (\exp y - y - 1)$ 
using l r by presburger
}
moreover {
  assume ineq:  $|y| < |x|$ 

  from psi-mono[OF assms(1)]
  have  $(\exp x - x - 1) - x^2 / 2 \leq (\exp y - y - 1) - y^2 / 2$  .

  then have  $y^2 * ((\exp x - x - 1) - x^2 / 2) \leq x^2 * ((\exp y - y - 1) - y^2 / 2)$ 
  by (smt (verit, best) ineq diff-divide-distrib exp-lower-Taylor-quadratic
le(1) le(2) mult-nonneg-nonneg one-less-exp-iff power-zero-numeral prob-space.psi-mono
prob-space-completion right-diff-distrib zero-le-power2))

  then have  $y^2 * (\exp x - x - 1) \leq x^2 * (\exp y - y - 1)$ 
  by (simp add: mult.commute right-diff-distrib)
}
ultimately show ?thesis by auto
qed

```

```

lemma bennett-inequality-1:
assumes I: finite I
assumes ind: indep-vars ( $\lambda . \text{borel}$ ) X I
assumes intsq:  $\bigwedge i. i \in I \implies \text{integrable } M (\lambda x. (X i x)^2)$ 

```

```

assumes bnd:  $\bigwedge i. i \in I \implies AE\ x\ in\ M. X\ i\ x \leq 1$ 
assumes t:  $t \geq 0$ 
defines V  $\equiv (\sum i \in I. expectation(\lambda x. X\ i\ x^2))$ 
shows prob { $x \in space\ M. (\sum i \in I. X\ i\ x - expectation\ (X\ i)) \geq$ 
t}  $\leq$ 
  exp (-V * bennett-h (t / V))
proof (cases V = 0)
  case True
  then show ?thesis
  by auto
next
  case f: False
  have V  $\geq 0$ 
  unfolding V-def
  apply (intro sum-nonneg integral-nonneg-AE)
  by auto
  then have Vpos: V > 0 using f by auto

define l :: real where l = ln(1 + t / V)
then have l: l  $\geq 0$ 
  using t Vpos by auto
have rv[measurable]:  $\bigwedge i. i \in I \implies random\ variable\ borel\ (X\ i)$ 
  using ind unfolding indep-vars-def by blast

define  $\psi$  where  $\psi = (\lambda x::real. exp(x) - x - 1)$ 

have rw: exp y = 1 + y +  $\psi$  y for y
  unfolding  $\psi$ -def by auto

have ebnd:  $\bigwedge i. i \in I \implies$ 
  AE x in M. exp (l * X i x)  $\leq exp\ l$ 
  apply (drule bnd)
  using l by (auto simp add: mult-left-le)

have int:  $\bigwedge i. i \in I \implies integrable\ M\ (\lambda x. (X\ i\ x))$ 
using rv intsq square-integrable-imp-integrable by blast

have intl:  $\bigwedge i. i \in I \implies integrable\ M\ (\lambda x. (l * X\ i\ x))$ 
using int by blast

have interpl:  $\bigwedge i. i \in I \implies integrable\ M\ (\lambda x. exp\ (l * X\ i\ x))$ 
apply (intro integrable-const-bound[where B = exp l])
using ebnd by auto

have intpsi:  $\bigwedge i. i \in I \implies integrable\ M\ (\lambda x. \psi\ (l * X\ i\ x))$ 
unfolding  $\psi$ -def
using intl interpl by auto

```

**have** \*\*:  $\bigwedge i. i \in I \implies$   
 $expectation (\lambda x. \psi (l * X i x)) \leq \psi l * expectation (\lambda x. (X i x)^{\wedge 2})$   
**proof** –  
**fix**  $i$  **assume**  $i: i \in I$   
**then have**  $AE\ x\ in\ M. l * X\ i\ x \leq l$   
**using**  $ebnd$  **by**  $auto$   
**then have**  $AE\ x\ in\ M. l^{\wedge 2} * \psi (l * X i x) \leq (l * X i x)^{\wedge 2} * \psi l$   
**using**  $psi-inequality[OF\ -\ l]$  **unfolding**  $\psi-def$   
**by**  $auto$   
**then have**  $AE\ x\ in\ M. l^{\wedge 2} * \psi (l * X i x) \leq l^{\wedge 2} * (\psi l * (X\ i\ x)^{\wedge 2})$   
**by**  $(auto\ simp\ add: field-simps)$   
**then have**  $AE\ x\ in\ M. \psi (l * X i x) \leq \psi l * (X i x)^{\wedge 2}$   
**by**  $(smt\ (verit,\ best)\ AE-cong\ \psi-def\ exp-eq-one-iff\ mult-cancel-left\ mult-eq-0-iff\ mult-left-mono\ zero-eq-power2\ zero-le-power2)$   
**then have**  $AE\ x\ in\ M. 0 \leq \psi l * (X i x)^{\wedge 2} - \psi (l * X i x)$   
**by**  $auto$   
**then have**  $expectation (\lambda x. \psi l * (X i x)^{\wedge 2} + (- \psi (l * X i x)))$   
 $\geq 0$   
**by**  $(simp\ add: integral-nonneg-AE)$   
**also have**  $expectation (\lambda x. \psi l * (X i x)^{\wedge 2} + (- \psi (l * X i x))) =$   
 $\psi l * expectation (\lambda x. (X i x)^{\wedge 2}) - expectation (\lambda x. \psi (l * X i$   
 $x))$   
**apply**  $(subst\ Bochner-Integration.integral-add)$   
**using**  $intpsi[OF\ i]$   $intsq[OF\ i]$  **by**  $auto$   
**finally show**  $expectation (\lambda x. \psi (l * X i x)) \leq \psi l * expectation$   
 $(\lambda x. (X i x)^{\wedge 2})$   
**by**  $auto$   
**qed**

**then have** \*:  $\bigwedge i. i \in I \implies$   
 $expectation (\lambda x. exp (l * X i x)) \leq$   
 $exp (l * expectation (X i)) * exp (\psi l * expectation (\lambda x. X i x^{\wedge 2}))$   
**proof** –  
**fix**  $i$   
**assume**  $iI: i \in I$   
**have**  $expectation (\lambda x. exp (l * X i x)) =$   
 $1 + l * expectation (\lambda x. X i x) +$   
 $expectation (\lambda x. \psi (l * X i x))$   
**unfolding**  $rw$   
**apply**  $(subst\ Bochner-Integration.integral-add)$   
**using**  $iI\ intl\ intpsi$  **apply**  $auto[2]$   
**apply**  $(subst\ Bochner-Integration.integral-add)$   
**using**  $intl\ iI\ prob-space$  **by**  $auto$   
**also have**  $\dots = l * expectation (X i) + 1 + expectation (\lambda x. \psi (l$   
 $* X i x))$   
**by**  $auto$   
**also have**  $\dots \leq 1 + l * expectation (X i) + \psi l * expectation (\lambda x.$

$X i x \hat{2}$ )  
**using**  $**[OF iI]$  **by** *auto*  
**also have**  $\dots \leq \exp (l * \text{expectation } (X i)) * \exp (\psi l * \text{expectation } (\lambda x. X i x \hat{2}))$   
**by** (*simp add: is-num-normalize(1) mult-exp-exp*)  
**finally show**  $\text{expectation } (\lambda x. \exp (l * X i x)) \leq \exp (l * \text{expectation } (X i)) * \exp (\psi l * \text{expectation } (\lambda x. X i x \hat{2}))$

•

**qed**

**have**  $(\prod_{i \in I. \text{expectation } (\lambda x. \exp (l * (X i x)))}) \leq (\prod_{i \in I. \exp (l * \text{expectation } (X i)) * \exp (\psi l * \text{expectation } (\lambda x. X i x \hat{2})))$   
**by** (*auto intro!: prod-mono simp add: \**)  
**also have**  $\dots = (\prod_{i \in I. \exp (l * \text{expectation } (X i))}) * (\prod_{i \in I. \exp (\psi l * \text{expectation } (\lambda x. X i x \hat{2})))$   
**by** (*auto simp add: prod.distrib*)  
**finally have**  $**:$   
 $(\prod_{i \in I. \text{expectation } (\lambda x. \exp (l * (X i x)))}) \leq (\prod_{i \in I. \exp (l * \text{expectation } (X i))}) * \exp (\psi l * V)$   
**by** (*simp add: V-def I exp-sum sum-distrib-left*)

**from** *indep-vars-Chernoff-ineq-ge[OF I ind l interpl]*  
**have** *prob*  $\{x \in \text{space } M. (\sum i \in I. X i x - \text{expectation } (X i)) \geq t\}$   
 $\leq \exp (-l * t) * (\prod_{i \in I. \text{expectation } (\lambda x. \exp (l * (X i x - \text{expectation } (X i)))))$   
**by** *auto*  
**also have**  $(\prod_{i \in I. \text{expectation } (\lambda x. \exp (l * (X i x - \text{expectation } (X i))))}) = (\prod_{i \in I. \text{expectation } (\lambda x. \exp (l * (X i x)))}) * \exp (-l * \text{expectation } (X i))$   
**by** (*auto intro!: prod.cong simp add: field-simps exp-diff exp-minus-inverse*)  
**also have**  $\dots = (\prod_{i \in I. \exp (-l * \text{expectation } (X i))}) * (\prod_{i \in I. \text{expectation } (\lambda x. \exp (l * (X i x)))})$   
**by** (*auto simp add: prod.distrib*)  
**also have**  $\dots \leq (\prod_{i \in I. \exp (-l * \text{expectation } (X i))}) * ((\prod_{i \in I. \exp (l * \text{expectation } (X i))}) * \exp (\psi l * V))$   
**apply** (*intro mult-left-mono[OF \*\*]*)  
**by** (*meson exp-ge-zero prod-nonneg*)  
**also have**  $\dots = \exp (\psi l * V)$   
**apply** (*simp add: prod.distrib [symmetric]*)  
**by** (*smt (verit, ccfv-threshold) exp-minus-inverse prod.not-neutral-contains-not-neutral*)  
**finally have** *prob*  $\{x \in \text{space } M. (\sum i \in I. X i x - \text{expectation } (X i)) \geq t\} \leq \exp (\psi l * V - l * t)$

by (simp add: mult-exp-exp)  
 also have  $\psi \ l * V - l * t = -V * \text{bennett-h } (t / V)$   
 unfolding  $\psi\text{-def } l\text{-def } \text{bennett-h-def}$   
 apply (subst exp-ln)  
 subgoal by (smt (verit) Vpos divide-nonneg-nonneg t)  
 by (auto simp add: algebra-simps)  
 finally show ?thesis .  
 qed

**lemma** *real-AE-le-sum*:  
 assumes  $\bigwedge i. i \in I \implies AE\ x\ \text{in } M. f\ i\ x \leq (g\ i\ x::\text{real})$   
 shows  $AE\ x\ \text{in } M. (\sum i \in I. f\ i\ x) \leq (\sum i \in I. g\ i\ x)$   
**proof** (cases)  
 assume finite I  
 with *AE-finite-allI*[OF this assms] have  $0:AE\ x\ \text{in } M. (\forall i \in I. f\ i\ x \leq g\ i\ x)$  by auto  
 show ?thesis by (intro eventually-mono[OF 0] sum-mono) auto  
 qed simp

**lemma** *real-AE-eq-sum*:  
 assumes  $\bigwedge i. i \in I \implies AE\ x\ \text{in } M. f\ i\ x = (g\ i\ x::\text{real})$   
 shows  $AE\ x\ \text{in } M. (\sum i \in I. f\ i\ x) = (\sum i \in I. g\ i\ x)$   
**proof** -  
 have 1:  $AE\ x\ \text{in } M. (\sum i \in I. f\ i\ x) \leq (\sum i \in I. g\ i\ x)$   
 apply (intro real-AE-le-sum)  
 apply (drule assms)  
 by auto  
 have 2:  $AE\ x\ \text{in } M. (\sum i \in I. g\ i\ x) \leq (\sum i \in I. f\ i\ x)$   
 apply (intro real-AE-le-sum)  
 apply (drule assms)  
 by auto  
 show ?thesis  
 using 1 2  
 by auto  
 qed

**theorem** *bennett-inequality*:  
 assumes *I*: finite I  
 assumes *ind*: indep-vars ( $\lambda \cdot$ , borel) X I  
 assumes *intsq*:  $\bigwedge i. i \in I \implies \text{integrable } M\ (\lambda x. (X\ i\ x)^{\wedge} 2)$   
 assumes *bnd*:  $\bigwedge i. i \in I \implies AE\ x\ \text{in } M. X\ i\ x \leq B$   
 assumes *t*:  $t \geq 0$   
 assumes *B*:  $B > 0$   
 defines  $V \equiv (\sum i \in I. \text{expectation } (\lambda x. X\ i\ x^{\wedge} 2))$   
 shows  $\text{prob } \{x \in \text{space } M. (\sum i \in I. X\ i\ x - \text{expectation } (X\ i)) \geq t\} \leq \exp (-V / B^{\wedge} 2 * \text{bennett-h } (t * B / V))$   
**proof** -

```

define  $Y$  where  $Y = (\lambda i x. X i x / B)$ 

from indep-vars-compose[OF ind, where  $Y = \lambda i x. x / B$ ]
have  $1$ : indep-vars ( $\lambda \cdot$ . borel)  $Y I$ 
  unfolding  $Y$ -def by (auto simp add: o-def)
have  $2$ :  $\bigwedge i. i \in I \implies \text{integrable } M (\lambda x. (Y i x)^2)$ 
  unfolding  $Y$ -def apply (drule intsq)
  by (auto simp add: field-simps)
have  $3$ :  $\bigwedge i. i \in I \implies AE x \text{ in } M. Y i x \leq 1$ 
  unfolding  $Y$ -def apply (drule bnd)
  using  $B$  by auto
have  $4$ :  $0 \leq t / B$  using  $t B$  by auto

have  $rw1$ :  $(\sum i \in I. Y i x - \text{expectation } (Y i)) =$ 
   $(\sum i \in I. X i x - \text{expectation } (X i)) / B$  for  $x$ 
  unfolding  $Y$ -def
  by (auto simp: diff-divide-distrib sum-divide-distrib)

have  $rw2$ :  $\text{expectation } (\lambda x. (Y i x)^2) =$ 
   $\text{expectation } (\lambda x. (X i x)^2) / B^2$  for  $i$ 
  unfolding  $Y$ -def
  by (simp add: power-divide)

have  $rw3$ :  $-(\sum i \in I. \text{expectation } (\lambda x. (X i x)^2) / B^2) = - V /$ 
 $B^2$ 
  unfolding  $V$ -def
  by (auto simp add: sum-divide-distrib)

have  $t / B / (\sum i \in I. \text{expectation } (\lambda x. (X i x)^2) / B^2) =$ 
 $t / B / (V / B^2)$ 
  unfolding  $V$ -def
  by (auto simp add: sum-divide-distrib)
then have  $rw4$ :  $t / B / (\sum i \in I. \text{expectation } (\lambda x. (X i x)^2) / B^2)$ 
 $=$ 
   $t * B / V$ 
  by (simp add: power2-eq-square)
have  $\text{prob } \{x \in \text{space } M. t \leq (\sum i \in I. X i x - \text{expectation } (X i))\}$ 
 $=$ 
 $\text{prob}\{x \in \text{space } M. t / B \leq (\sum i \in I. X i x - \text{expectation } (X i)) /$ 
 $B\}$ 
  by (smt (verit, best) B Collect-cong divide-cancel-right divide-right-mono)
also have  $\dots \leq$ 
 $\text{exp } (- V / B^2 * \text{bennett-h } (t * B / V))$ 
  using bennett-inequality-1[OF I 1 2 3 4]
  unfolding  $rw1$   $rw2$   $rw3$   $rw4$  .
finally show ?thesis .
qed

```

```

lemma bennett-h-bernstein-bound:
  assumes  $x \geq 0$ 
  shows  $\text{bennett-h } x \geq x^2 / (2 * (1 + x / 3))$ 
proof -
  have  $\text{eq}: x^2 / (2 * (1 + x / 3)) = 3/2 * x - 9/2 * (x / (x+3))$ 
    using assms
    by (sos (()) & ()))

  define g where  $g = (\lambda x. \text{bennett-h } x - (3/2 * x - 9/2 * (x / (x+3))))$ 

  define g' where  $g' = (\lambda x::\text{real}. \ln(1 + x) + 27 / (2 * (x+3)^2) - 3 / 2)$ 
  define g'' where  $g'' = (\lambda x::\text{real}. 1 / (1 + x) - 27 / (x+3)^3)$ 

  have  $54 / ((2 * x + 6)^2) = 27 / (2 * (x + 3)^2)$  (is ?L = ?R)
for  $x :: \text{real}$ 
proof -
  have  $?L = 54 / (2^2 * (x + 3)^2)$ 
  unfolding power-mult-distrib[symmetric] by (simp add: algebra-simps)
  also have  $\dots = ?R$  by simp
  finally show ?thesis by simp
qed

  hence  $1: x \geq 0 \implies (g \text{ has-real-derivative } (g' x)) \text{ (at } x \text{) for } x$ 
  unfolding g-def g'-def bennett-h-def by (auto intro!: derivative-eq-intros simp: power2-eq-square)
  have  $2: x \geq 0 \implies (g' \text{ has-real-derivative } (g'' x)) \text{ (at } x \text{) for } x$ 
  unfolding g'-def g''-def
  apply (auto intro!: derivative-eq-intros)[1]
  by (sos (()) & ()))

  have gz:  $g 0 = 0$ 
  unfolding g-def bennett-h-def by auto
  have g1z:  $g' 0 = 0$ 
  unfolding g'-def by auto

  have p2:  $g'' x \geq 0$  if  $x \geq 0$  for  $x$ 
proof -
  have  $27 * (1+x) \leq (x+3)^3$ 
  using that unfolding power3-eq-cube by (auto simp: algebra-simps)
  hence  $27 / (x + 3)^3 \leq 1 / (1+x)$ 
  using that by (subst frac-le-eq) (auto intro!: divide-nonpos-pos)
  thus ?thesis unfolding g''-def by simp
qed

  from deriv-nonneg-imp-mono[OF 2 p2 -]

```

**have**  $x \geq 0 \implies g' x \geq 0$  **for**  $x$  **using**  $g1z$   
**by** (*metis atLeastAtMost-iff*)

**from** *deriv-nonneg-imp-mono*[*OF 1 this -*]  
**have**  $x \geq 0 \implies g x \geq 0$  **for**  $x$  **using**  $gz$   
**by** (*metis atLeastAtMost-iff*)

**thus** *?thesis*  
**using** *assms eq g-def* **by** *force*  
**qed**

**lemma** *sum-sq-exp-eq-zero-imp-zero*:  
**assumes** *finite I i ∈ I*  
**assumes** *intsq: integrable M (λx. (X i x)^2)*  
**assumes**  $(\sum i \in I. \text{expectation } (\lambda x. X i x^2)) = 0$   
**shows** *AE x in M. X i x = (0::real)*  
**proof** –  
**have**  $(\forall i \in I. \text{expectation } (\lambda x. X i x^2) = 0)$   
**using** *assms*  
**apply** (*subst sum-nonneg-eq-0-iff[symmetric]*)  
**by** *auto*  
**then have**  $\text{expectation } (\lambda x. X i x^2) = 0$   
**using** *assms(2)* **by** *blast*  
**thus** *?thesis*  
**using** *integral-nonneg-eq-0-iff-AE*[*OF intsq*]  
**by** *auto*  
**qed**

**corollary** *bernstein-inequality*:  
**assumes** *I: finite I*  
**assumes** *ind: indep-vars (λ -, borel) X I*  
**assumes** *intsq:  $\bigwedge i. i \in I \implies \text{integrable } M (\lambda x. (X i x)^2)$*   
**assumes** *bnf:  $\bigwedge i. i \in I \implies \text{AE } x \text{ in } M. X i x \leq B$*   
**assumes** *t:  $t \geq 0$*   
**assumes** *B:  $B > 0$*   
**defines**  $V \equiv (\sum i \in I. \text{expectation } (\lambda x. X i x^2))$   
**shows**  $\text{prob } \{x \in \text{space } M. (\sum i \in I. X i x - \text{expectation } (X i)) \geq t\} \leq$   
 $\text{exp } (- (t^2 / (2 * (V + t * B / 3))))$   
**proof** (*cases V = 0*)  
**case** *True*  
**then have**  $1: \bigwedge i. i \in I \implies \text{AE } x \text{ in } M. X i x = 0$   
**unfolding** *V-def*  
**using** *sum-sq-exp-eq-zero-imp-zero*  
**by** (*metis I intsq*)  
**then have**  $2: \bigwedge i. i \in I \implies \text{expectation } (X i) = 0$   
**using** *integral-eq-zero-AE* **by** *blast*

**have** *AE x in M.  $(\sum i \in I. X i x - \text{expectation } (X i)) = (\sum i \in I.$*

0)

```

apply (intro real-AE-eq-sum)
using 1 2
by auto
then have *: AE x in M. ( $\sum i \in I. X i x - \text{expectation } (X i) = 0$ )
by force

moreover {
  assume t > 0
  then have prob {x ∈ space M. ( $\sum i \in I. X i x - \text{expectation } (X i) \geq t$ ) = 0}
  apply (intro prob-eq-0-AE)
  using * by auto
  then have ?thesis by auto
}
ultimately show ?thesis
apply (cases t = 0) using t by auto
next
case f: False
have V ≥ 0
  unfolding V-def
  apply (intro sum-nonneg integral-nonneg-AE)
  by auto
then have V: V > 0 using f by auto

have t * B / V ≥ 0 using t B V by auto
from bennett-h-bernstein-bound[OF this]
have (t * B / V)2 / (2 * (1 + t * B / V / 3))
  ≤ bennett-h (t * B / V) .

then have (- V / B2) * bennett-h (t * B / V) ≤
  (- V / B2) * ((t * B / V)2 / (2 * (1 + t * B / V / 3)))
  apply (subst mult-left-mono-neg)
  using B V by auto
also have ... =
  ((- V / B2) * (t * B / V)2) / (2 * (1 + t * B / V / 3))
  by auto
also have ((- V / B2) * (t * B / V)2) = -(t2) / V
  using V B by (auto simp add: field-simps power2-eq-square)
finally have *: (- V / B2) * bennett-h (t * B / V) ≤
  -(t2) / (2 * (V + t * B / 3))
  using V by (auto simp add: field-simps)

from bennett-inequality[OF assms(1-6)]
have prob {x ∈ space M. ( $\sum i \in I. X i x - \text{expectation } (X i) \geq t$ )}
  ≤
  exp (- V / B2 * bennett-h (t * B / V))
  using V-def by auto
also have ... ≤ exp (- (t2 / (2 * (V + t * B / 3))))

```

```

    using *
    by auto
    finally show ?thesis .
qed

end

end

```

### 3 Bienaymé's identity

Bienaymé's identity [5, §17] can be used to deduce the variance of a sum of random variables, if their co-variance is known. A common use-case of the identity is the computation of the variance of the mean of pair-wise independent variables.

**theory** *Bienaymes-Identity*

```

imports Concentration-Inequalities-Preliminary
begin

```

```

context prob-space
begin

```

**lemma** *variance-divide:*

```

fixes  $f :: 'a \Rightarrow \text{real}$ 
assumes integrable M f
shows  $\text{variance } (\lambda\omega. f \ \omega / r) = \text{variance } f / r^2$ 
using assms
by (subst Bochner-Integration.integral-divide[OF assms(1)])
    (simp add:diff-divide-distrib[symmetric] power2-eq-square algebra-simps)

```

**definition** *covariance where*

```

covariance f g = expectation  $(\lambda\omega. (f \ \omega - \text{expectation } f) * (g \ \omega - \text{expectation } g))$ 

```

**lemma** *covariance-eq:*

```

fixes  $f :: 'a \Rightarrow \text{real}$ 
assumes  $f \in \text{borel-measurable } M$   $g \in \text{borel-measurable } M$ 
assumes integrable M ( $\lambda\omega. f \ \omega^2$ ) integrable M ( $\lambda\omega. g \ \omega^2$ )
shows  $\text{covariance } f g = \text{expectation } (\lambda\omega. f \ \omega * g \ \omega) - \text{expectation } f$ 
     $* \text{expectation } g$ 
proof –
have integrable M f using square-integrable-imp-integrable assms by
auto
moreover have integrable M g using square-integrable-imp-integrable
assms by auto
ultimately show ?thesis
    using assms cauchy-schwartz(1)[where M=M]
    by (simp add:covariance-def algebra-simps prob-space)

```

qed

**lemma** *covar-integrable*:

**fixes**  $f\ g :: 'a \Rightarrow \text{real}$

**assumes**  $f \in \text{borel-measurable } M\ g \in \text{borel-measurable } M$

**assumes**  $\text{integrable } M\ (\lambda\omega. f\ \omega^{\wedge}2)\ \text{integrable } M\ (\lambda\omega. g\ \omega^{\wedge}2)$

**shows**  $\text{integrable } M\ (\lambda\omega. (f\ \omega - \text{expectation } f) * (g\ \omega - \text{expectation } g))$

**proof** –

**have**  $\text{integrable } M\ f$  **using** *square-integrable-imp-integrable assms by auto*

**moreover have**  $\text{integrable } M\ g$  **using** *square-integrable-imp-integrable assms by auto*

**ultimately show** *?thesis using assms cauchy-schwartz(1)[where  $M=M$ ]* **by** (*simp add: algebra-simps*)

qed

**lemma** *sum-square-int*:

**fixes**  $f :: 'b \Rightarrow 'a \Rightarrow \text{real}$

**assumes** *finite I*

**assumes**  $\bigwedge i. i \in I \implies f\ i \in \text{borel-measurable } M$

**assumes**  $\bigwedge i. i \in I \implies \text{integrable } M\ (\lambda\omega. f\ i\ \omega^{\wedge}2)$

**shows**  $\text{integrable } M\ (\lambda\omega. (\sum i \in I. f\ i\ \omega)^2)$

**proof** –

**have**  $\text{integrable } M\ (\lambda\omega. \sum i \in I. \sum j \in I. f\ j\ \omega * f\ i\ \omega)$

**using** *assms*

**by** (*intro Bochner-Integration.integrable-sum cauchy-schwartz(1)[where  $M=M$ ], auto*)

**thus** *?thesis*

**by** (*simp add: power2-eq-square sum-distrib-left sum-distrib-right*)

qed

**theorem** *bienaymes-identity*:

**fixes**  $f :: 'b \Rightarrow 'a \Rightarrow \text{real}$

**assumes** *finite I*

**assumes**  $\bigwedge i. i \in I \implies f\ i \in \text{borel-measurable } M$

**assumes**  $\bigwedge i. i \in I \implies \text{integrable } M\ (\lambda\omega. f\ i\ \omega^{\wedge}2)$

**shows**

$\text{variance } (\lambda\omega. (\sum i \in I. f\ i\ \omega)) = (\sum i \in I. (\sum j \in I. \text{covariance } (f\ i)\ (f\ j)))$

**proof** –

**have**  $a: \bigwedge i\ j. i \in I \implies j \in I \implies$

$\text{integrable } M\ (\lambda\omega. (f\ i\ \omega - \text{expectation } (f\ i)) * (f\ j\ \omega - \text{expectation } (f\ j)))$

**using** *assms covar-integrable by simp*

**have**  $\text{variance } (\lambda\omega. (\sum i \in I. f\ i\ \omega)) = \text{expectation } (\lambda\omega. (\sum i \in I. f\ i\ \omega - \text{expectation } (f\ i))^2)$

**using** *square-integrable-imp-integrable[OF assms(2,3)]*

**by** (*simp add: Bochner-Integration.integral-sum sum-subtractf*)

**also have** ... =  $\text{expectation } (\lambda\omega. (\sum i \in I. (\sum j \in I. (f i \omega - \text{expectation } (f i)) * (f j \omega - \text{expectation } (f j))))))$   
**by** (*simp add: power2-eq-square sum-distrib-right sum-distrib-left mult.commute*)  
**also have** ... =  $(\sum i \in I. (\sum j \in I. \text{covariance } (f i) (f j)))$   
**using** *a* **by** (*simp add: Bochner-Integration.integral-sum covariance-def*)  
**finally show** *?thesis* **by** *simp*  
**qed**

**lemma** *covar-self-eq*:  
**fixes** *f* :: 'a  $\Rightarrow$  real  
**shows**  $\text{covariance } f f = \text{variance } f$   
**by** (*simp add: covariance-def power2-eq-square*)

**lemma** *covar-indep-eq-zero*:  
**fixes** *f g* :: 'a  $\Rightarrow$  real  
**assumes** *integrable M f*  
**assumes** *integrable M g*  
**assumes** *indep-var borel f borel g*  
**shows**  $\text{covariance } f g = 0$

**proof** –  
**have** *a: indep-var borel (( $\lambda t. t - \text{expectation } f$ )  $\circ$  f) borel (( $\lambda t. t - \text{expectation } g$ )  $\circ$  g)*  
**by** (*rule indep-var-compose[OF assms(3)], auto*)

**have** *b: expectation ( $\lambda\omega. (f \omega - \text{expectation } f) * (g \omega - \text{expectation } g)) = 0$*   
**using** *a* **assms** **by** (*subst indep-var-lebesgue-integral, auto simp add: comp-def prob-space*)

**thus** *?thesis* **by** (*simp add: covariance-def*)  
**qed**

**lemma** *bienaymes-identity-2*:  
**fixes** *f* :: 'b  $\Rightarrow$  'a  $\Rightarrow$  real  
**assumes** *finite I*  
**assumes**  $\bigwedge i. i \in I \implies f i \in \text{borel-measurable } M$   
**assumes**  $\bigwedge i. i \in I \implies \text{integrable } M (\lambda\omega. f i \omega^2)$   
**shows**  $\text{variance } (\lambda\omega. (\sum i \in I. f i \omega)) =$   
 $(\sum i \in I. \text{variance } (f i)) + (\sum i \in I. \sum j \in I - \{i\}. \text{covariance } (f i) (f j))$

**proof** –  
**have**  $\text{variance } (\lambda\omega. (\sum i \in I. f i \omega)) = (\sum i \in I. \sum j \in I. \text{covariance } (f i) (f j))$   
**by** (*simp add: bienaymes-identity[OF assms(1,2,3)]*)  
**also have** ... =  $(\sum i \in I. \text{covariance } (f i) (f i) + (\sum j \in I - \{i\}. \text{covariance } (f i) (f j)))$   
**using** *assms* **by** (*subst sum.insert[symmetric], auto simp add: insert-absorb*)

**also have** ... =  $(\sum_{i \in I}. \text{variance } (f\ i)) + (\sum_{i \in I}. (\sum_{j \in I - \{i\}}. \text{covariance } (f\ i) (f\ j)))$   
**by** (*simp add: covar-self-eq[symmetric] sum.distrib*)  
**finally show** ?thesis **by simp**  
**qed**

**theorem** *bienaymes-identity-pairwise-indep:*

**fixes**  $f :: 'b \Rightarrow 'a \Rightarrow \text{real}$   
**assumes** *finite I*  
**assumes**  $\bigwedge i. i \in I \Rightarrow f\ i \in \text{borel-measurable } M$   
**assumes**  $\bigwedge i. i \in I \Rightarrow \text{integrable } M (\lambda \omega. f\ i\ \omega^{\wedge} 2)$   
**assumes**  $\bigwedge i\ j. i \in I \Rightarrow j \in I \Rightarrow i \neq j \Rightarrow \text{indep-var borel } (f\ i)$   
*borel } (f\ j)*  
**shows**  $\text{variance } (\lambda \omega. (\sum_{i \in I}. f\ i\ \omega)) = (\sum_{i \in I}. \text{variance } (f\ i))$   
**proof** –  
**have**  $\bigwedge i\ j. i \in I \Rightarrow j \in I - \{i\} \Rightarrow \text{covariance } (f\ i) (f\ j) = 0$   
**using** *covar-indep-eq-zero assms(4) square-integrable-imp-integrable[OF assms(2,3)] by auto*  
**hence**  $a: (\sum_{i \in I}. \sum_{j \in I - \{i\}}. \text{covariance } (f\ i) (f\ j)) = 0$   
**by simp**  
**thus** ?thesis **by** (*simp add: bienaymes-identity-2[OF assms(1,2,3)]*)  
**qed**

**lemma** *indep-var-from-indep-vars:*

**assumes**  $i \neq j$   
**assumes** *indep-vars*  $(\lambda-. M') f \{i, j\}$   
**shows** *indep-var*  $M' (f\ i) M' (f\ j)$   
**proof** –  
**have**  $a: \text{inj } (\text{case-bool } i\ j)$  **using** *assms(1)*  
**by** (*simp add: bool.case-eq-if inj-def*)  
**have**  $b: \text{range } (\text{case-bool } i\ j) = \{i, j\}$   
**by** (*simp add: UNIV-bool insert-commute*)  
**have**  $c: \text{indep-vars } (\lambda-. M') f (\text{range } (\text{case-bool } i\ j))$  **using** *assms(2)*  
*b by simp*

**have**  $\text{True} = \text{indep-vars } (\lambda x. M') (\lambda x. f (\text{case-bool } i\ j\ x)) \text{ UNIV}$   
**using** *indep-vars-reindex[OF a c]*  
**by** (*simp add: comp-def*)  
**also have** ... = *indep-vars*  $(\lambda x. \text{case-bool } M' M' x) (\lambda x. \text{case-bool } (f\ i) (f\ j) x) \text{ UNIV}$   
**by** (*rule indep-vars-cong, auto simp: bool.case-distrib bool.case-eq-if*)  
**also have** ... = ?thesis  
**by** (*simp add: indep-var-def*)  
**finally show** ?thesis **by simp**  
**qed**

**lemma** *bienaymes-identity-pairwise-indep-2:*

**fixes**  $f :: 'b \Rightarrow 'a \Rightarrow \text{real}$   
**assumes** *finite I*

```

assumes  $\bigwedge i. i \in I \implies f i \in \text{borel-measurable } M$ 
assumes  $\bigwedge i. i \in I \implies \text{integrable } M (\lambda \omega. f i \omega^2)$ 
assumes  $\bigwedge J. J \subseteq I \implies \text{card } J = 2 \implies \text{indep-vars } (\lambda -. \text{borel}) f J$ 
shows  $\text{variance } (\lambda \omega. (\sum i \in I. f i \omega)) = (\sum i \in I. \text{variance } (f i))$ 
using assms(4)
by (intro bienaymes-identity-pairwise-indep[OF assms(1,2,3)] indep-var-from-indep-vars, auto)

```

**lemma** *bienaymes-identity-full-indep:*

```

fixes  $f :: 'b \Rightarrow 'a \Rightarrow \text{real}$ 
assumes finite I
assumes  $\bigwedge i. i \in I \implies f i \in \text{borel-measurable } M$ 
assumes  $\bigwedge i. i \in I \implies \text{integrable } M (\lambda \omega. f i \omega^2)$ 
assumes  $\text{indep-vars } (\lambda -. \text{borel}) f I$ 
shows  $\text{variance } (\lambda \omega. (\sum i \in I. f i \omega)) = (\sum i \in I. \text{variance } (f i))$ 
by (intro bienaymes-identity-pairwise-indep-2[OF assms(1,2,3)] indep-vars-subset[OF assms(4)] auto)

```

**end**

**end**

## 4 Cantelli's Inequality

Cantelli's inequality [3] is an improvement of Chebyshev's inequality for one-sided tail bounds.

**theory** *Cantelli-Inequality*

**imports** *HOL-Probability.Probability*

**begin**

**context** *prob-space*

**begin**

**lemma** *cantelli-arith:*

```

assumes  $a > (0::\text{real})$ 
shows  $(V + (V / a)^2) / (a + (V / a)^2) = V / (a^2 + V)$  (is ?L = ?R)

```

**proof** –

```

have  $?L = ((V * a^2 + V^2) / a^2) / ((a^2 + V)^2 / a^2)$ 
using assms by (intro arg-cong2[where f=(/)] (simp-all add:field-simps power2-eq-square))

```

```

also have  $\dots = (V * a^2 + V^2) / (a^2 + V)^2$ 
using assms unfolding divide-divide-times-eq by simp

```

```

also have  $\dots = V * (a^2 + V) / (a^2 + V)^2$ 
by (intro arg-cong2[where f=(/)] (simp-all add: algebra-simps power2-eq-square))

```

```

also have  $\dots = ?R$  by (simp add:power2-eq-square)

```

finally show *?thesis* by simp  
qed

**theorem** *cantelli-inequality*:

**assumes** *[measurable]*: random-variable borel  $Z$

**assumes** *intZsq*: integrable  $M$  ( $\lambda z. Z z^2$ )

**assumes**  $a: a > 0$

**shows**  $\text{prob} \{z \in \text{space } M. Z z - \text{expectation } Z \geq a\} \leq$   
 $\text{variance } Z / (a^2 + \text{variance } Z)$

**proof** –

**define**  $u$  where  $u = \text{variance } Z / a$

**have**  $u: u \geq 0$

**unfolding**  $u\text{-def}$

**by** (*simp add: a divide-nonneg-pos*)

**define**  $Y$  where  $Y = (\lambda z. Z z + (-\text{expectation } Z))$

**have** random-variable borel ( $\lambda z. |Y z + u|$ )

**unfolding**  $Y\text{-def}$

**by** *auto*

**then have**  $ev: \{z \in \text{space } M. a + u \leq |Y z + u|\} \in \text{events}$

**by** *auto*

**have**  $\text{int}Z:\text{integrable } M Z$

**apply** (*subst square-integrable-imp-integrable[OF - intZsq]*)

**by** *auto*

**then have**  $i1: \text{integrable } M (\lambda z. (Z z - \text{expectation } Z + u)^2)$

**unfolding** *power2-sum power2-diff* **using**  $\text{intZsq}$

**by** *auto*

**have**  $\text{int}Y:\text{integrable } M Y$

**unfolding**  $Y\text{-def}$  **using**  $\text{int}Z$  **by** *auto*

**have**  $\text{int}Ysq:\text{integrable } M (\lambda z. Y z^2)$

**unfolding**  $Y\text{-def}$  *power2-sum* **using**  $\text{intZsq}$   $\text{int}Z$  **by** *auto*

**have**  $\text{expectation } Y = 0$

**unfolding**  $Y\text{-def}$

**apply** (*subst Bochner-Integration.integral-add[OF intZ]*)

**using** *prob-space* **by** *auto*

**then have**  $\text{expectation } (\lambda z. (Y z + u)^2) =$

$\text{expectation } (\lambda z. (Y z)^2) + u^2$

**unfolding** *power2-sum*

**apply** (*subst Bochner-Integration.integral-add[OF - -]*)

**using**  $\text{int}Y$   $\text{int}Ysq$  **apply** *auto*[2]

**apply** (*subst Bochner-Integration.integral-add[OF - -]*)

**using**  $\text{int}Y$   $\text{int}Ysq$  **apply** *auto*[2]

**using** *prob-space* **by** *auto*

**then have**  $*$ :  $\text{expectation } (\lambda z. (Y z + u)^2) = \text{variance } Z + u^2$

**unfolding**  $Y\text{-def}$  **by** *auto*

**have**  
 $\text{prob } \{z \in \text{space } M. Z z - \text{expectation } Z \geq a\} =$   
 $\text{prob } \{z \in \text{space } M. Y z + u \geq a + u\}$   
**apply** (*intro arg-cong*[**where**  $f = \text{prob}$ ])  
**using** *Y-def* **by** *auto*  
**also have**  $\dots \leq \text{prob } \{z \in \text{space } M. a + u \leq |Y z + u|\}$   
**apply** (*intro finite-measure-mono*[*OF - ev*])  
**by** *auto*  
  
**also have**  $\dots \leq \text{expectation } (\lambda z. (Y z + u)^2) / (a + u)^2$   
**apply** (*intro second-moment-method*)  
**unfolding** *Y-def* **using** *a u i1* **by** *auto*  
**also have**  $\dots = ((\text{variance } Z) + u^2) / (a + u)^2$   
**using** *\** **by** *auto*  
**also have**  $\dots = \text{variance } Z / (a^2 + \text{variance } Z)$   
**unfolding** *u-def* **using** *a* **by** (*auto intro!*: *cantelli-arith*)  
**finally show** *?thesis* .  
**qed**

**corollary** *cantelli-inequality-neg*:

**assumes** [*measurable*]: *random-variable borel*  $Z$   
**assumes** *intZsq*: *integrable*  $M (\lambda z. Z z^2)$   
**assumes**  $a: a > 0$   
**shows**  $\text{prob } \{z \in \text{space } M. Z z - \text{expectation } Z \leq -a\} \leq$   
 $\text{variance } Z / (a^2 + \text{variance } Z)$   
**proof** –  
**define**  $nZ$  **where** [*simp*]:  $nZ = (\lambda z. -Z z)$   
**have**  $vnZ$ : *variance*  $nZ = \text{variance } Z$   
**unfolding** *nZ-def*  
**by** (*auto simp add*: *power2-commute*)  
  
**have** *1*: *random-variable borel*  $nZ$  **by** *auto*  
**have** *2*: *integrable*  $M (\lambda z. (nZ z)^2)$   
**using** *intZsq* **by** *auto*  
**from** *cantelli-inequality*[*OF 1 2 a*]  
**have**  $\text{prob } \{z \in \text{space } M. a \leq nZ z - \text{expectation } nZ\} \leq$   
 $\text{variance } nZ / (a^2 + \text{variance } nZ)$   
**by** *auto*  
**thus** *?thesis* **unfolding**  $vnZ$  **apply** *auto*[*1*]  
**by** (*smt* (*verit*, *del-insts*) *Collect-cong*)  
**qed**

**end**

**end**

## 5 Efron-Stein Inequality

In this section we verify the Efron-Stein inequality. The verified theorem is stated as Efron-Stein inequality for non-symmetric functions by Steele [8]. However most textbook refer to this version as “the Efron-Stein inequality”. The original result that was shown by Efron and Stein is a tail bound for the variance of a symmetric functions of i.i.d. random variables [4].

**theory** *Efron-Stein-Inequality*

**imports** *Concentration-Inequalities-Preliminary*

**begin**

**theorem** *efron-stein-inequality-distr*:

**fixes**  $f :: - \Rightarrow \text{real}$

**assumes** *finite I*

**assumes**  $\bigwedge i. i \in I \implies \text{prob-space } (M\ i)$

**assumes** *integrable (PiM I M) (λx. f x^2) and f-meas: f ∈ borel-measurable (PiM I M)*

**shows** *prob-space.variance (PiM I M) f ≤*

$(\sum i \in I. (\int x. (f (\lambda j. x (j, \text{False})) - f (\lambda j. x (j, j=i)))^2 \partial \text{PiM } (I \times \text{UNIV}) (M \circ \text{fst}))) / 2$

**(is ?L ≤ ?R)**

**proof** –

**let**  $?M = \text{PiM } (I \times (\text{UNIV} :: \text{bool set})) (M \circ \text{fst})$

**have** *prob: prob-space (PiM I M)*

**using** *assms(2) by (intro prob-space-PiM) auto*

**interpret** *prob-space ?M*

**using** *assms(2) by (intro prob-space-PiM) auto*

**define**  $n$  **where**  $n = \text{card } I$

**obtain**  $q :: - \Rightarrow \text{nat}$  **where**  $q: \text{bij-betw } q\ I\ \{..<n\}$

**unfolding** *n-def using ex-bij-betw-finite-nat[OF assms(1)] atLeast0LessThan*  
**by** *auto*

**let**  $?φ = (\lambda n\ x. f (\lambda j. x (j, q\ j < n)))$

**let**  $?τ = (\lambda n\ x. f (\lambda j. x (j, q\ j = n)))$

**let**  $?σ = (\lambda x. f (\lambda j. x (j, \text{False})))$

**let**  $?χ = (\lambda x. f (\lambda j. x (j, \text{True})))$

**have** *meas-1: (λω. f (g ω)) ∈ borel-measurable ?M*

**if**  $g \in \text{PiM } (I \times \text{UNIV}) (M \circ \text{fst}) \rightarrow_M \text{PiM } I\ M$  **for**  $g$

**using** *that by (intro measurable-compose[OF f-meas])*

**have** *meas-2: (λx j. x (j, h j)) ∈ ?M →M PiM I M for h*

**proof** –

**have**  $?thesis \longleftrightarrow (\lambda x. (\lambda j \in I. x (j, h j))) \in ?M \rightarrow_M Pi_M I M$   
**by** (*intro measurable-cong*) (*auto simp:space-PiM PiE-def extensional-def*)  
**also have**  $\dots \longleftrightarrow True$   
**unfolding** *eq-True*  
**by** (*intro measurable-restrict measurable-PiM-component-rev*) *auto*  
**finally show**  $?thesis$  **by** *simp*  
**qed**

**have** *int-1: integrable ?M*  $(\lambda x. (g x - h x) \hat{=} 2)$   
**if** *integrable ?M*  $(\lambda x. (g x) \hat{=} 2)$  *integrable ?M*  $(\lambda x. (h x) \hat{=} 2)$   
**and**  $g \in \text{borel-measurable } ?M$   $h \in \text{borel-measurable } ?M$   
**for**  $g h :: - \Rightarrow \text{real}$   
**proof** –  
**have** *integrable ?M*  $(\lambda x. (g x) \hat{=} 2 + (h x) \hat{=} 2 - 2 * (g x * h x))$   
**using** *that* **by** (*intro Bochner-Integration.integrable-add Bochner-Integration.integrable-diff integrable-mult-right cauchy-schwartz(1)*)  
**thus**  $?thesis$  **by** (*simp add:algebra-simps power2-eq-square*)  
**qed**

**note** *meas-rules = borel-measurable-add borel-measurable-times borel-measurable-diff borel-measurable-power meas-1 meas-2*

**have** *f-int: integrable (Pi\_M I M) f*  
**by** (*intro finite-measure.square-integrable-imp-integrable[OF - f-meas assms(3)] prob-space.finite-measure prob*)  
**moreover have** *integrable (Pi\_M I M)  $(\lambda x. f (restrict x I)) = integrable (Pi_M I M) f$*   
**by** (*intro Bochner-Integration.integrable-cong*) (*auto simp:space-PiM*)  
**ultimately have** *f-int-2: integrable (Pi\_M I M)  $(\lambda x. f (restrict x I))$*   
**by** *simp*

**have** *cong:  $(\int x. g (\lambda j \in I. x (j, h j)) \partial ?M) = (\int x. g (\lambda j. x (j, h j)) \partial ?M)$*  (*is ?L1 = ?R1*)  
**for**  $g :: - \Rightarrow \text{real}$  **and**  $h$   
**by** (*intro Bochner-Integration.integral-cong arg-cong[where f=g] refl*)  
*(auto simp add:space-PiM PiE-def extensional-def restrict-def)*

**have** *lift:  $(\int x. g x \partial Pi_M I M) = (\int x. g (\lambda j. x (j, h j)) \partial ?M)$*  (*is ?L1 = ?R1*)  
**if**  $g \in \text{borel-measurable } (Pi_M I M)$   
**for**  $g :: - \Rightarrow \text{real}$  **and**  $h$   
**proof** –  
**let**  $?J = (\lambda i. (i, h i)) ' I$   
**have**  $?R1 = (\int x. g (\lambda j \in I. x (j, h j)) \partial ?M)$   
**by** (*intro cong[symmetric]*)  
**also have**  $\dots = (\int x. g x \partial \text{distr } ?M (Pi_M I (\lambda i. (M \circ \text{fst}) (i, h i)))$

$(\lambda x. (\lambda j \in I. x (j, h j)))$   
**using that**  
**by** (*intro integral-distr[symmetric] measurable-restrict measurable-component-singleton*) *auto*  
**also have** ... =  $(\int x. g x \partial PiM I (\lambda i. (M \circ fst) (i, h i)))$   
**using** *assms(2)*  
**by** (*intro arg-cong2[where f=integral<sup>L</sup>] refl distr-PiM-reindex inj-onI*) *auto*  
**also have** ... = ?L1  
**by** *auto*  
**finally show** ?thesis  
**by** *simp*  
**qed**

**have** *lift-int: integrable ?M*  $(\lambda x. g (\lambda j. x (j, h j)))$  **if** *integrable (PiM I M) g*  
**for**  $g :: - \Rightarrow \text{real}$  **and**  $h$   
**proof** –  
**have** *0:integrable (distr ?M (PiM I (\lambda i. (M \circ fst) (i, h i)))*  $(\lambda x. (\lambda j \in I. x (j, h j)))$   $g$   
**using that** *assms(2)* **by** (*subst distr-PiM-reindex*) (*auto intro:inj-onI*)  
**have** *integrable ?M*  $(\lambda x. g (\lambda j \in I. x (j, h j)))$   
**by** (*intro integrable-distr[OF - 0] measurable-restrict measurable-component-singleton*) *auto*  
**moreover have** *integrable ?M*  $(\lambda x. g (\lambda j \in I. x (j, h j))) \longleftrightarrow ?thesis$   
**by** (*intro Bochner-Integration.integrable-cong refl arg-cong[where f=g] ext*)  
*(auto simp:PiE-def space-PiM extensional-def)*  
**ultimately show** ?thesis  
**by** *simp*  
**qed**

**note** *int-rules = cauchy-schwartz(1) int-1 lift-int assms(3) f-int f-int-2*

**have**  $(\int x. g x \partial ?M) = (\int x. g (\lambda(j,v). x (j, v \neq h j)) \partial ?M)$  (**is** ?L1 = ?R1)  
**if**  $g$  **in** *borel-measurable ?M* **for**  $g :: - \Rightarrow \text{real}$  **and**  $h$   
**proof** –  
**have** ?L1 =  $(\int x. g x \partial \text{distr } ?M (PiM (I \times UNIV) (\lambda i. (M \circ fst) (fst i, snd i \neq h (fst i))))$   
 $(\lambda x. (\lambda i \in I \times UNIV. x (fst i, snd i \neq h (fst i))))$   
**by** (*subst distr-PiM-reindex*) (*auto intro:inj-onI assms(2) simp:comp-def*)  
**also have** ... =  $(\int x. g (\lambda i \in I \times UNIV. x (fst i, snd i \neq h (fst i))) \partial ?M)$   
**using that by** (*intro integral-distr measurable-restrict measurable-component-singleton*)  
*(auto simp:comp-def)*

**also have** ... = ?R1  
**by** (intro Bochner-Integration.integral-cong refl arg-cong[where f=g] ext)  
(auto simp add:space-PiM PiE-def extensional-def restrict-def)  
**finally show** ?thesis  
**by** simp  
**qed**

**hence** switch:  $(\int x. g x \partial^?M) = (\int x. h x \partial^?M)$   
**if**  $\bigwedge x. h x = g (\lambda(j,v). x (j, v \neq u j))$   $g \in \text{borel-measurable } ?M$   
**for**  $g h :: - \Rightarrow \text{real}$  **and**  $u$   
**using** that **by** simp

**have** 1:  $(\int x. (?\sigma x) * (?\varphi i x - ?\varphi (i+1) x) \partial^?M) \leq (\int x. (?\sigma x - ?\tau i x)^2 \partial^?M) / 2$   
**(is** ?L1  $\leq$  ?R1)  
**if**  $i < n$  **for**  $i$   
**proof** -  
**have** ?L1 =  $(\int x. (?\tau i x) * (?\varphi (i+1) x - ?\varphi i x) \partial^?M)$   
**by** (intro switch[of - - ( $\lambda j. q j = i$ )] arg-cong2[where f=(\*)] arg-cong2[where f=(-)] arg-cong[where f=f] ext meas-rules)  
(auto intro:arg-cong)  
**hence** ?L1 =  $(?L1 + (\int x. (?\tau i x) * (?\varphi (i+1) x - ?\varphi i x) \partial^?M)) / 2$   
**by** simp  
**also have** ... =  $(\int x. (?\sigma x) * (?\varphi i x - ?\varphi(i+1) x) + (?\tau i x) * (?\varphi(i+1) x - ?\varphi i x) \partial^?M) / 2$   
**by** (intro Bochner-Integration.integral-add[symmetric] arg-cong2[where f=(/)] refl  
int-rules meas-rules)  
**also have** ... =  $(\int x. (?\sigma x - ?\tau i x) * (?\varphi i x - ?\varphi(i+1) x) \partial^?M) / 2$   
**by** (intro arg-cong2[where f=(/)] Bochner-Integration.integral-cong)  
(auto simp:algebra-simps)  
**also have** ...  $\leq ((\int x. (?\sigma x - ?\tau i x)^2 \partial^?M)^{\text{powr}(1/2)} * (\int x. (?\varphi i x - ?\varphi(i+1) x)^2 \partial^?M)^{\text{powr}(1/2)}) / 2$   
**by** (intro divide-right-mono cauchy-schwartz meas-rules int-rules)  
auto  
**also have** ... =  $((\int x. (?\sigma x - ?\tau i x)^2 \partial^?M)^{\text{powr}(1/2)} * (\int x. (?\sigma x - ?\tau i x)^2 \partial^?M)^{\text{powr}(1/2)}) / 2$   
**by** (intro arg-cong2[where f=(/)] arg-cong2[where f=(\*)] arg-cong2[where f=(powr)] refl  
switch[of - - ( $\lambda j. q j < i$ )] arg-cong2[where f=powr] arg-cong2[where f=(-)]  
arg-cong[where f=f] ext meas-rules) (auto intro:arg-cong)  
**also have** ... =  $(\int x. (?\sigma x - ?\tau i x)^2 \partial^?M) / 2$   
**by** (simp add:powr-add[symmetric])  
**finally show** ?thesis **by** simp  
**qed**

**have** *indep-vars* ( $M \circ \text{fst}$ ) ( $\lambda i \omega. \omega i$ ) ( $I \times \text{UNIV}$ )  
**using** *assms(2)* **by** (*intro proj-indep*) *auto*  
**hence**  $2:\text{indep-var}$  ( $Pi_M (I \times \{False\}) (M \circ \text{fst})$ ) ( $\lambda x. \lambda j \in I \times \{False\}. x j$ )  
 $(Pi_M (I \times \{True\}) (M \circ \text{fst})) (\lambda x. \lambda j \in I \times \{True\}. x j)$   
**by** (*intro indep-var-restrict*[**where**  $I = I \times \text{UNIV}$ ]) *auto*  
**have** *indep-var*  
 $(Pi_M I M) ((\lambda x. (\lambda i \in I. x (i, False))) \circ (\lambda x. (\lambda j \in I \times \{False\}. x j)))$   
 $(Pi_M I M) ((\lambda x. (\lambda i \in I. x (i, True))) \circ (\lambda x. (\lambda j \in I \times \{True\}. x j)))$   
**by** (*intro indep-var-compose*[*OF 2*] *measurable-restrict measurable-PiM-component-rev*) *auto*  
**hence** *indep-var* ( $Pi_M I M$ ) ( $\lambda x. (\lambda j \in I. x (j, False))$ ) ( $Pi_M I M$ )  
 $(\lambda x. (\lambda j \in I. x (j, True)))$   
**unfolding** *comp-def* **by** (*simp add:restrict-def cong:if-cong*)  
  
**hence** *indep-var borel* ( $f \circ (\lambda x. (\lambda j \in I. x (j, False)))$ ) *borel* ( $f \circ (\lambda x. (\lambda j \in I. x (j, True)))$ )  
**by** (*intro indep-var-compose*[*OF - assms(4,4)*]) *auto*  
**hence** *indep:indep-var borel* ( $\lambda x. f (\lambda j \in I. x (j, False))$ ) *borel* ( $\lambda x. f (\lambda j \in I. x (j, True))$ )  
**by** (*simp add:comp-def*)  
  
**have**  $3: \omega (j, q j = q i) = \omega (j, j = i)$  **if**  
 $\omega \in PiE (I \times \text{UNIV}) (\lambda i. \text{space } (M (\text{fst } i))) i \in I$  **for**  $i j \omega$   
**proof** (*cases j \in I*)  
**case** *True*  
**hence** ( $q j = q i$ ) = ( $j = i$ )  
**using** *that inj-onD*[*OF bij-betw-imp-inj-on*[*OF q*]] **by** *blast*  
**thus** *?thesis* **by** *simp*  
**next**  
**case** *False*  
**hence**  $\omega (j, a) = \text{undefined}$  **for**  $a$   
**using** *that unfolding PiE-def extensional-def* **by** *simp*  
**thus** *?thesis* **by** *simp*  
**qed**  
  
**have**  $?L = (\int x. (f x)^2 \partial PiM I M) - (\int x. (f x) \partial PiM I M)^2$   
**by** (*intro prob-space.variance-eq f-int assms(3) prob*)  
**also have**  $\dots = (\int x. (f x)^2 \partial PiM I M) - (\int x. f x \partial PiM I M) * (\int x. f x \partial PiM I M)$   
**by** (*simp add:power2-eq-square*)  
**also have**  $\dots = (\int x. (?\sigma x)^2 \partial ?M) - (\int x. ?\sigma x \partial ?M) * (\int x. ?\chi x \partial ?M)$   
**by** (*intro arg-cong2*[**where**  $f = (-)$ ] *lift arg-cong2*[**where**  $f = (*)$ ]) *meas-rules f-meas*)  
**also have**  $\dots = (\int x. (?\sigma x)^2 \partial ?M) - (\int x. f (\lambda j \in I. x (j, False))$

$\partial^?M) * (\int x. f(\lambda j \in I. x(j, True)) \partial^?M)$   
**by** (intro arg-cong2[**where**  $f=(-)$ ] arg-cong2[**where**  $f=(*)$ ] cong[symmetric] refl)  
**also have** ... =  $(\int x. (?\sigma x) \wedge^2 \partial^?M) - (\int x. f(\lambda j \in I. x(j, False)) * f(\lambda j \in I. x(j, True)) \partial^?M)$   
**by** (intro arg-cong2[**where**  $f=(-)$ ] indep-var-lebesgue-integral[symmetric] refl int-rules indep)  
**also have** ... =  $(\int x. (?\sigma x) * (?\varphi 0 x) \partial^?M) - (\int x. (?\sigma x) * (?\varphi n x) \partial^?M)$   
**using** bij-betw-apply[OF q] **by** (intro arg-cong2[**where**  $f=(-)$ ] arg-cong2[**where**  $f=(*)$ ] ext  
arg-cong[**where**  $f=f$ ] Bochner-Integration.integral-cong)  
(auto simp:space-PiM power2-eq-square PiE-def extensional-def)  
**also have** ... =  $(\sum i < n. (\int x. (?\sigma x) * (?\varphi i x) \partial^?M) - (\int x. (?\sigma x) * (?\varphi (Suc i) x) \partial^?M))$   
**unfolding** power2-eq-square **by** (intro sum-lessThan-telescope'[symmetric])  
**also have** ... =  $(\sum i < n. (\int x. (?\sigma x) * (?\varphi i x) - (?\sigma x) * (?\varphi (Suc i) x) \partial^?M))$   
**by** (intro sum.cong Bochner-Integration.integral-diff[symmetric] int-rules meas-rules) auto  
**also have** ... =  $(\sum i < n. (\int x. (?\sigma x) * (?\varphi i x - ?\varphi (i+1) x) \partial^?M))$   
**by** (simp-all add:power2-eq-square algebra-simps)  
**also have** ...  $\leq (\sum i < n. ((\int x. (?\sigma x - ?\tau i x) \wedge^2 \partial^?M)) / 2)$   
**by** (intro sum-mono 1) auto  
**also have** ... =  $(\sum i \in I. ((\int x. (f(\lambda j. x(j, False)) - f(\lambda j. x(j, q j=q i))) \wedge^2 \partial^?M)) / 2)$   
**by** (intro sum.reindex-bij-betw[OF q, symmetric])  
**also have** ... =  $(\sum i \in I. ((\int x. (f(\lambda j. x(j, False)) - f(\lambda j. x(j, q j=q i))) \wedge^2 \partial^?M))) / 2$   
**unfolding** sum-divide-distrib[symmetric] **by** simp  
**also have** ... = ?R  
**using** inj-onD[OF bij-betw-imp-inj-on[OF q]]  
**by** (intro arg-cong2[**where**  $f=(/)$ ] arg-cong2[**where**  $f=(-)$ ] arg-cong2[**where**  $f=power$ ] arg-cong[**where**  $f=f$ ] Bochner-Integration.integral-cong sum.cong refl ext 3)  
(auto simp add:space-PiM )  
**finally show** ?thesis  
**by** simp  
qed

**theorem (in prob-space) efron-stein-inequality-classic:**

**fixes**  $f :: - \Rightarrow real$

**assumes** finite I

**assumes** indep-vars ( $M' \circ fst$ ) X (I  $\times$  (UNIV :: bool set))

**assumes**  $f \in borel-measurable (PiM I M')$

**assumes** integrable M ( $\lambda \omega. f(\lambda i \in I. X(i, False) \omega) \wedge^2$ )

**assumes**  $\bigwedge i. i \in I \implies distr M (M' i) (X(i, True)) = distr M (M'$

*i*) ( $X (i, False)$ )  
**shows** *variance* ( $\lambda\omega. f (\lambda i \in I. X (i, False) \omega) \leq$   
 $(\sum j \in I. expectation (\lambda\omega. (f (\lambda i \in I. X (i, False) \omega) - f (\lambda i \in I. X$   
 $(i, i=j) \omega))^2) / 2$   
 $(is ?L \leq ?R)$ )

**proof** –  
**let**  $?D = distr M (PiM I M') (\lambda\omega. \lambda i \in I. X (i, False) \omega)$

**let**  $?M = PiM I (\lambda i. distr M (M' i) (X (i, False)))$   
**let**  $?N = PiM (I \times (UNIV::bool set)) ((\lambda i. distr M (M' i) (X$   
 $(i, False))) \circ fst)$

**have** *rv: random-variable* ( $M' i$ ) ( $X (i, j)$ ) **if**  $i \in I$  **for**  $i j$   
**using** *assms(2)* **that unfolding indep-vars-def by auto**

**have** *proj-meas:* ( $\lambda x j. x (j, h j) \in PiM (I \times UNIV) (M' \circ fst)$   
 $\rightarrow_M PiM I M'$   
**for**  $h :: - \Rightarrow bool$

**proof** –  
**have**  $?thesis \longleftrightarrow (\lambda x. (\lambda j \in I. x (j, h j))) \in PiM (I \times UNIV)$   
 $(M' \circ fst) \rightarrow_M PiM I M'$   
**by** (*intro measurable-cong*) (*auto simp:space-PiM PiE-def exten-*  
*sional-def*)  
**also have**  $\dots \longleftrightarrow True$   
**unfolding** *eq-True*  
**by** (*intro measurable-restrict measurable-PiM-component-rew*) *auto*  
**finally show**  $?thesis$  **by** *simp*

**qed**

**note** *meas-rules = borel-measurable-add borel-measurable-times borel-measurable-diff*  
*proj-meas*  
*borel-measurable-power assms(3) measurable-restrict measurable-compose[OF*  
*- assms(3)]*

**have** *indep-vars* ( $(M' \circ fst) \circ (\lambda i. (i, False))$ ) ( $\lambda i. X (i, False)$ )  $I$   
**by** (*intro indep-vars-reindex indep-vars-subset[OF assms(2)] inj-onI*)  
*auto*  
**hence** *indep-vars*  $M' (\lambda i. X (i, False)) I$  **by** (*simp add: comp-def*)  
**hence**  $0: ?D = PiM I (\lambda i. distr M (M' i) (X (i, False)))$   
**by** (*intro iffD1[OF indep-vars-iff-distr-eq-PiM'] rv*)

**have**  $distr M (M' (fst x)) (X (fst x, False)) = distr M (M' (fst x))$   
 $(X x)$   
**if**  $x \in I \times UNIV$  **for**  $x$   
**using** *that assms(5)* **by** (*cases x, cases snd x*) *auto*

**hence**  $1: ?N = PiM (I \times UNIV) (\lambda i. distr M ((M' \circ fst) i) (X i))$   
**using** *assms(3)* **by** (*intro PiM-cong refl*) (*simp add: comp-def*)  
**also have**  $\dots = distr M (PiM (I \times UNIV) (M' \circ fst)) (\lambda x. \lambda i \in I \times$

```

UNIV. X i x)
  using rv by (intro iffD1[OF indep-vars-iff-distr-eq-PiM'', symmetric]
  assms(2)) auto
  finally have 2: ?N = distr M (PiM (I × UNIV) (M' ∘ fst)) (λx.
  λi∈I × UNIV. X i x)
    by simp

  have 3: integrable (PiM I (λi. distr M (M' i) (X (i, False)))) (λx.
  (f x)2)
    unfolding 0[symmetric] by (intro iffD2[OF integrable-distr-eq]
  meas-rules assms rv)

  have ?L = (∫ x. (f x - expectation (λω. f (λi∈I. X (i, False) ω)))2
  ∂?D)
    using rv by (intro integral-distr[symmetric] meas-rules measur-
  able-restrict) auto
    also have ... = prob-space.variance ?D f
      by (intro arg-cong[where f=integralL ?D] arg-cong2[where f=(-)]
  arg-cong2[where f=power]
      refl ext integral-distr[symmetric] measurable-restrict rv assms(3))
    also have ... = prob-space.variance ?M f
      unfolding 0 by simp
    also have ... ≤ (∑ i∈I. (∫ x. (f (λj. x (j, False)) - f (λj. x (j, j =
  i)))2 ∂?N)) / 2
      using assms(3) by (intro efron-stein-inequality-distr prob-space-distr
  rv assms(1) 3) auto
    also have ... = (∑ i∈I. expectation (λω. (f (λj. (λi∈I × UNIV. X i
  ω) (j, False)) -
      f (λj. (λi∈I × UNIV. X i ω) (j, j=i)))2)) / 2
      using rv unfolding 2
      by (intro sum.cong arg-cong2[where f=(/)] integral-distr refl
  meas-rules) auto
    also have ... = ?R
      by (simp add: restrict-def)
    finally show ?thesis
      by simp
qed

end

```

## 6 McDiarmid's inequality

In this section we verify McDiarmid's inequality [6, Lemma 1.2]. In the source and also further sources sometimes refer to the result as the “independent bounded differences” inequality.

```

theory McDiarmid-Inequality
  imports Concentration-Inequalities-Preliminary
begin

```

**lemma** *Collect-restr-cong*:  
**assumes**  $A = B$   
**assumes**  $\bigwedge x. x \in A \implies P x = Q x$   
**shows**  $\{x \in A. P x\} = \{x \in B. Q x\}$   
**using** *assms* **by** *auto*

**lemma** *ineq-chain*:  
**fixes**  $h :: nat \Rightarrow real$   
**assumes**  $\bigwedge i. i < n \implies h (i+1) \leq h i$   
**shows**  $h n \leq h 0$   
**using** *assms* **by** (*induction n*) *force+*

**lemma** *restrict-subset-eq*:  
**assumes**  $A \subseteq B$   
**assumes**  $restrict f B = restrict g B$   
**shows**  $restrict f A = restrict g A$   
**using** *assms* **unfolding** *restrict-def* **by** (*meson subsetD*)

Bochner Integral version of Hoeffding's Lemma using *interval-bounded-random-variable.Hoeffding*

**lemma** (*in prob-space*) *Hoeffdings-lemma-bochner*:  
**assumes**  $l > 0$  **and** *E0*: *expectation f = 0*  
**assumes** *random-variable borel f*  
**assumes** *AE x in M. f x ∈ {a..b::real}*  
**shows**  $expectation (\lambda x. exp (l * f x)) \leq exp (l^2 * (b - a)^2 / 8)$  (*is ?L ≤ ?R*)

**proof** –

**interpret** *interval-bounded-random-variable M f a b*  
**using** *assms* **by** (*unfold-locales*) *auto*

**have** *integrable M (λx. exp (l \* f x))*  
**using** *assms(1,3,4)* **by** (*intro integrable-const-bound[where B=exp (l \* b)] simp-all*)

**hence** *ennreal (?L) = (∫<sup>+</sup> x. exp (l \* f x) ∂M)*  
**by** (*intro nn-integral-eq-integral[symmetric]*) *auto*

**also have**  $... \leq ennreal (?R)$

**by** (*intro Hoeffdings-lemma-nn-integral-0 assms*)

**finally have**  $0:ennreal (?L) \leq ennreal ?R$

**by** *simp*

**show** *?thesis*

**proof** (*cases ?L ≥ 0*)

**case** *True*

**thus** *?thesis* **using** *0* **by** *simp*

**next**

**case** *False*

**hence**  $?L \leq 0$  **by** *simp*

**also have**  $... \leq ?R$  **by** *simp*

**finally show** *?thesis* **by** *simp*

qed  
qed

lemma (in prob-space) Hoeffdings-lemma-bochner-2:

assumes  $l > 0$  and  $E0$ : expectation  $f = 0$

assumes random-variable borel  $f$

assumes  $\bigwedge x y. \{x,y\} \subseteq \text{space } M \implies |f x - f y| \leq (c::\text{real})$

shows expectation  $(\lambda x. \exp (l * f x)) \leq \exp (l^2 * c^2 / 8)$  (is ?L  
 $\leq ?R$ )

proof -

define  $a :: \text{real}$  where  $a = (\text{INF } x \in \text{space } M. f x)$

define  $b :: \text{real}$  where  $b = a + c$

obtain  $\omega$  where  $\omega : \omega \in \text{space } M$  using not-empty by auto

hence  $0 : f ' \text{space } M \neq \{\}$  by auto

have  $1 : c = b - a$  unfolding b-def by simp

have bdd-below  $(f ' \text{space } M)$

using  $\omega$  assms(4) unfolding abs-le-iff

by (intro bdd-belowI[where  $m=f \omega - c$ ]) (auto simp add:algebra-simps)

hence  $f x \geq a$  if  $x \in \text{space } M$  for  $x$  unfolding a-def by (intro  
 $c\text{INF-lower that}$ )

moreover have  $f x \leq b$  if  $x\text{-space}: x \in \text{space } M$  for  $x$

proof (rule ccontr)

assume  $\neg(f x \leq b)$

hence  $a : f x > a + c$  unfolding b-def by simp

have  $f y \geq f x - c$  if  $y \in \text{space } M$  for  $y$

using that  $x\text{-space assms}(4)$  unfolding abs-le-iff by (simp  
 $\text{add:algebra-simps}$ )

hence  $f x - c \leq a$  unfolding a-def using  $c\text{Inf-greatest}[OF 0]$  by  
auto

thus False using a by simp

qed

ultimately have  $f x \in \{a..b\}$  if  $x \in \text{space } M$  for  $x$  using that by  
auto

hence  $AE x$  in  $M. f x \in \{a..b\}$  by simp

thus ?thesis unfolding 1 by (intro Hoeffdings-lemma-bochner assms(1,2,3))

qed

lemma (in prob-space) Hoeffdings-lemma-bochner-3:

assumes expectation  $f = 0$

assumes random-variable borel  $f$

assumes  $\bigwedge x y. \{x,y\} \subseteq \text{space } M \implies |f x - f y| \leq (c::\text{real})$

shows expectation  $(\lambda x. \exp (l * f x)) \leq \exp (l^2 * c^2 / 8)$  (is ?L  
 $\leq ?R$ )

proof -

consider (a)  $l > 0$  | (b)  $l = 0$  | (c)  $l < 0$

by argo

then show ?thesis

```

proof (cases)
  case a thus ?thesis by (intro Hoeffdings-lemma-bochner-2 assms)
auto
next
  case b thus ?thesis by simp
next
  case c
  have ?L = expectation ( $\lambda x. \exp((-l) * (-f x))$ ) by simp
  also have ...  $\leq \exp((-l)^2 * c^2/8)$  using c assms by (intro
Hoeffdings-lemma-bochner-2) auto
  also have ... = ?R by simp
  finally show ?thesis by simp
qed
qed

```

Version of *product-sigma-finite.product-integral-singleton* without the condition that  $M i$  has to be sigma finite for all  $i$ :

```

lemma product-integral-singleton:
  fixes f :: -  $\Rightarrow$  -::{banach, second-countable-topology}
  assumes sigma-finite-measure (M i)
  assumes f  $\in$  borel-measurable (M i)
  shows ( $\int x. f (x i) \partial(\text{PiM } \{i\} M)$ ) = ( $\int x. f x \partial(M i)$ ) (is ?L =
?R)
proof -
  define M' where M' j = (if j=i then M i else count-space {undefined})
for j

  interpret product-sigma-finite M'
  using assms(1) unfolding product-sigma-finite-def M'-def
  by (auto intro!:sigma-finite-measure-count-space-finite)

  have ?L =  $\int x. f (x i) \partial(\text{PiM } \{i\} M')$ 
  by (intro Bochner-Integration.integral-cong PiM-cong) (simp-all
add:M'-def)
  also have ... = ( $\int x. f x \partial(M' i)$ )
  using assms(2) by (intro product-integral-singleton) (simp add:M'-def)
  also have ... = ?R
  by (intro Bochner-Integration.integral-cong PiM-cong) (simp-all
add:M'-def)
  finally show ?thesis by simp
qed

```

Version of *product-sigma-finite.product-integral-fold* without the condition that  $M i$  has to be sigma finite for all  $i$ :

```

lemma product-integral-fold:
  fixes f :: -  $\Rightarrow$  -::{banach, second-countable-topology}
  assumes  $\bigwedge i. i \in I \cup J \implies$  sigma-finite-measure (M i)
  assumes  $I \cap J = \{\}$ 
  assumes finite I

```

**assumes** *finite J*  
**assumes** *integrable (PiM (I ∪ J) M) f*  
**shows**  $(\int x. f x \partial \text{PiM } (I \cup J) M) = (\int x. (\int y. f (\text{merge } I J(x,y)) \partial \text{PiM } J M) \partial \text{PiM } I M)$  (**is** ?L = ?R)  
**and** *integrable (PiM I M) ( $\lambda x. (\int y. f (\text{merge } I J(x,y)) \partial \text{PiM } J M$ )* (**is** ?I)  
**and** *AE x in PiM I M. integrable (PiM J M) ( $\lambda y. f (\text{merge } I J(x,y))$ )* (**is** ?T)  
**proof** –  
**define** *M' where M' i = (if i ∈ I ∪ J then M i else count-space {undefined})* **for** *i*  
  
**interpret** *product-sigma-finite M'*  
**using** *assms(1) unfolding product-sigma-finite-def M'-def*  
**by** (*auto intro!:sigma-finite-measure-count-space-finite*)  
  
**interpret** *pair-sigma-finite Pi<sub>M</sub> I M' Pi<sub>M</sub> J M'*  
**using** *assms(3,4) sigma-finite unfolding pair-sigma-finite-def* **by** *blast*  
  
**have** *0: integrable (Pi<sub>M</sub> (I ∪ J) M') f = integrable (Pi<sub>M</sub> (I ∪ J) M) f*  
**by** (*intro Bochner-Integration.integrable-cong PiM-cong*) (*simp-all add:M'-def*)  
  
**have** *?L = ( $\int x. f x \partial \text{PiM } (I \cup J) M'$ )*  
**by** (*intro Bochner-Integration.integral-cong PiM-cong*) (*simp-all add:M'-def*)  
**also have** *... = ( $\int x. (\int y. f (\text{merge } I J (x,y)) \partial \text{PiM } J M') \partial \text{PiM } I M'$ )*  
**using** *assms(5) by (intro product-integral-fold assms(2,3,4))* (*simp add:0*)  
**also have** *... = ?R*  
**by** (*intro Bochner-Integration.integral-cong PiM-cong*) (*simp-all add:M'-def*)  
**finally show** *?L = ?R by simp*  
  
**have** *integrable (Pi<sub>M</sub> (I ∪ J) M') f = integrable (PiM I M' ⊗<sub>M</sub> PiM J M') ( $\lambda x. f (\text{merge } I J x)$ )*  
**using** *assms(5) apply (subst distr-merge[OF assms(2,3,4),symmetric])*  
**by** (*intro integrable-distr-eq*) (*simp-all add:0[symmetric]*)  
**hence** *1: integrable (PiM I M' ⊗<sub>M</sub> PiM J M') ( $\lambda x. f (\text{merge } I J x)$ )*  
**using** *assms(5) 0 by simp*  
  
**hence** *integrable (PiM I M') ( $\lambda x. (\int y. f (\text{merge } I J(x,y)) \partial \text{PiM } J M')$ )* (**is** ?I')  
**by** (*intro integrable-fst'*) *auto*  
**moreover have** *?I' = ?I*

**by** (intro Bochner-Integration.integrable-cong PiM-cong ext Bochner-Integration.integral-cong)  
 (simp-all add:M'-def)  
**ultimately show** ?I  
**by** simp  
  
**have** AE x in PiM I M'. integrable (PiM J M') (λy. f (merge I J  
 (x, y))) (is ?T')  
**by** (intro AE-integrable-fst'[OF 1])  
**moreover have** ?T' = ?T  
**by** (intro arg-cong2[where f=almost-everywhere] PiM-cong ext  
 Bochner-Integration.integrable-cong)  
 (simp-all add:M'-def)  
**ultimately show** ?T  
**by** simp  
**qed**

**lemma** product-integral-insert:

**fixes** f :: - ⇒ -::{banach, second-countable-topology}  
**assumes**  $\bigwedge k. k \in \{i\} \cup J \implies \text{sigma-finite-measure } (M k)$   
**assumes**  $i \notin J$   
**assumes** finite J  
**assumes** integrable (PiM (insert i J) M) f  
**shows**  $(\int x. f x \partial \text{PiM } (\text{insert } i \text{ J}) M) = (\int x. (\int y. f (y(i := x)) \partial \text{PiM } J M) \partial M i)$  (is ?L = ?R)  
**proof** –  
**note** meas-cong = iffD1[OF measurable-cong]

**have** integrable (PiM {i} M) (λx. ( $\int y. f (\text{merge } \{i\} J (x,y)) \partial \text{PiM } J M$ ))  
**using** assms **by** (intro product-integral-fold) auto  
**hence** 0:(λx. ( $\int y. f (\text{merge } \{i\} J (x,y)) \partial \text{PiM } J M$ )) ∈ borel-measurable  
 (PiM {i} M)  
**using** borel-measurable-integrable **by** simp  
**have** 1:(λx. ( $\int y. f (y(i := (x i))) \partial \text{PiM } J M$ )) ∈ borel-measurable  
 (PiM {i} M)  
**by** (intro meas-cong[OF - 0] Bochner-Integration.integral-cong  
 arg-cong[where f=f])  
 (auto simp add:space-PiM merge-def fun-upd-def PiE-def exten-  
 sional-def)  
**have** (λx. ( $\int y. f (y(i := (\lambda i \in \{i\}. x) i)) \partial \text{PiM } J M$ )) ∈ borel-measurable  
 (M i)  
**by** (intro measurable-compose[OF - 1, where f=(λx. (λi ∈ {i}. x))] measurable-restrict) auto  
**hence** 2:(λx. ( $\int y. f (y(i := x)) \partial \text{PiM } J M$ )) ∈ borel-measurable  
 (M i) **by** simp

**have** ?L = ( $\int x. f x \partial \text{PiM } (\{i\} \cup J) M$ ) **by** simp  
**also have** ... = ( $\int x. (\int y. f (\text{merge } \{i\} J (x,y)) \partial \text{PiM } J M) \partial \text{PiM } \{i\} M$ )

**using** *assms(2,4)* **by** (*intro product-integral-fold assms(1,3)*) *auto*  
**also have** ... =  $(\int x. (\int y. f (y(i := (x i)))) \partial PiM J M) \partial PiM \{i\} M)$   
**by** (*intro Bochner-Integration.integral-cong refl arg-cong[where f=f]*)  
*(auto simp add:space-PiM merge-def fun-upd-def PiE-def extensional-def)*  
**also have** ... = ?R  
**using** *assms(1,4)* **by** (*intro product-integral-singleton assms(1) 2*)  
*auto*  
**finally show** ?thesis **by** *simp*  
**qed**

**lemma** *product-integral-insert-rev*:  
**fixes** *f :: -  $\Rightarrow$  -::\{banach, second-countable-topology\}*  
**assumes**  $\bigwedge k. k \in \{i\} \cup J \implies \text{sigma-finite-measure } (M k)$   
**assumes**  $i \notin J$   
**assumes** *finite J*  
**assumes** *integrable (PiM (insert i J) M) f*  
**shows**  $(\int x. f x \partial PiM (insert i J) M) = (\int y. (\int x. f (y(i := x))) \partial M i) \partial PiM J M)$  (**is** ?L = ?R)  
**proof** –  
**have** ?L =  $(\int x. f x \partial PiM (J \cup \{i\}) M)$  **by** *simp*  
**also have** ... =  $(\int x. (\int y. f (merge J \{i\} (x,y))) \partial PiM \{i\} M) \partial PiM J M)$   
**using** *assms(2,4)* **by** (*intro product-integral-fold assms(1,3)*) *auto*  
**also have** ... =  $(\int x. (\int y. f (x(i := (y i)))) \partial PiM \{i\} M) \partial PiM J M)$   
**unfolding** *merge-singleton[OF assms(2)]*  
**by** (*intro Bochner-Integration.integral-cong refl arg-cong[where f=f]*)  
*(metis PiE-restrict assms(2) restrict-upd space-PiM)*  
**also have** ... = ?R  
**using** *assms(1,4)* **by** (*intro Bochner-Integration.integral-cong product-integral-singleton*) *auto*  
**finally show** ?thesis **by** *simp*  
**qed**

**lemma** *merge-empty[simp]*:  
 $merge \{ \} I (y,x) = restrict x I$   
 $merge I \{ \} (y,x) = restrict y I$   
**unfolding** *merge-def restrict-def* **by** *auto*

**lemma** *merge-cong*:  
**assumes**  $restrict x1 I = restrict x2 I$   
**assumes**  $restrict y1 J = restrict y2 J$   
**shows**  $merge I J (x1,y1) = merge I J (x2,y2)$   
**using** *assms* **unfolding** *merge-def restrict-def*  
**by** (*intro ext*) (*smt (verit, best) case-prod-conv*)

**lemma** *restrict-merge*:

*restrict (merge I J x) K = merge (I ∩ K) (J ∩ K) x*

**unfolding** *restrict-def merge-def* **by** (*intro ext*) (*auto simp: case-prod-beta*)

**lemma** *map-prod-measurable*:

**assumes**  $f \in M \rightarrow_M M'$

**assumes**  $g \in N \rightarrow_M N'$

**shows**  $\text{map-prod } f g \in M \otimes_M N \rightarrow_M M' \otimes_M N'$

**using** *assms* **by** (*subst measurable-pair-iff*) *simp*

**lemma** *mc-diarmid-inequality-aux*:

**fixes**  $f :: (\text{nat} \Rightarrow 'a) \Rightarrow \text{real}$

**fixes**  $n :: \text{nat}$

**assumes**  $\bigwedge i. i < n \implies \text{prob-space } (M i)$

**assumes**  $\bigwedge i x y. i < n \implies \{x, y\} \subseteq \text{space } (PiM \{..<n\} M) \implies$

$(\forall j \in \{..<n\} - \{i\}. x j = y j) \implies |f x - f y| \leq c i$

**assumes** *f-meas*:  $f \in \text{borel-measurable } (PiM \{..<n\} M)$  **and**  $\varepsilon\text{-gt-0}$ :

$\varepsilon > 0$

**shows**  $\mathcal{P}(\omega \text{ in } PiM \{..<n\} M. f \omega - (\int \xi. f \xi \partial PiM \{..<n\} M)) \geq$

$\varepsilon) \leq \exp(-2 * \varepsilon^2) / (\sum i < n. (c i)^2)$

(**is** ?L ≤ ?R)

**proof** –

**define** *h* **where**  $h k = (\lambda \xi. (\int \omega. f (\text{merge } \{..<k\} \{k..<n\} (\xi, \omega))$

$\partial PiM \{k..<n\} M))$  **for** *k*

**define** *t* **where**  $t = 4 * \varepsilon / (\sum i < n. (c i)^2)$

**define** *V* **where**  $V i \xi = h (\text{Suc } i) \xi - h i \xi$  **for** *i*  $\xi$

**obtain** *x0* **where**  $x0 : x0 \in \text{space } (PiM \{..<n\} M)$

**using** *prob-space.not-empty*[*OF prob-space-PiM*] *assms(1)* **by** *fast-force*

**have** *delta*:  $|f x - f y| \leq c i$  **if**  $i < n$

$x \in PiE \{..<n\} (\lambda i. \text{space } (M i))$   $y \in PiE \{..<n\} (\lambda i. \text{space } (M i))$

$\text{restrict } x (\{..<n\} - \{i\}) = \text{restrict } y (\{..<n\} - \{i\})$

**for**  $x y i$

**proof** (*rule assms(2)*[*OF that(1)*], *goal-cases*)

**case** 1

**then show** ?*case* **using** *that(2,3)* **unfolding** *space-PiM* **by** *auto*

**next**

**case** 2

**then show** ?*case* **using** *that(4)* **by** (*intro ballI*) (*metis restrict-apply'*)

**qed**

**have** *c-ge-0*:  $c j \geq 0$  **if**  $j < n$  **for** *j*

**proof** –

**have**  $0 \leq |f x0 - f x0|$  **by** *simp*  
**also have**  $\dots \leq c j$  **using** *x0 unfolding space-PiM* **by** (*intro delta that*) *auto*  
**finally show** *?thesis* **by** *simp*  
**qed**  
**hence** *sum-c-ge-0*:  $(\sum_{i < n}. (c i)^2) \geq 0$  **by** (*meson sum-nonneg zero-le-power2*)

**hence** *t-ge-0*:  $t \geq 0$  **using** *ε-gt-0 unfolding t-def* **by** *simp*

**note** *borel-rules* =  
*borel-measurable-sum measurable-compose[OF - borel-measurable-exp]*  
*borel-measurable-times*

**note** *int-rules* =  
*prob-space-PiM assms(1) borel-rules*  
*prob-space.integrable-bounded bounded-intros*  
**have** *h-n*:  $h n \xi = f \xi$  **if**  $\xi \in \text{space } (PiM \{..<n\} M)$  **for**  $\xi$   
**proof** –  
**have**  $h n \xi = (\int \omega. f (\lambda i \in \{..<n\}. \xi i) \partial PiM \{ \} M)$   
**unfolding** *h-def* **using** *leD*  
**by** (*intro Bochner-Integration.integral-cong PiM-cong arg-cong*[**where** *f=f*] *restrict-cong*)  
*auto*  
**also have**  $\dots = f (\text{restrict } \xi \{..<n\})$   
**unfolding** *PiM-empty* **by** *simp*  
**also have**  $\dots = f \xi$   
**using** *that* **unfolding** *space-PiM PiE-def*  
**by** (*simp add: extensional-restrict*)  
**finally show** *?thesis*  
**by** *simp*  
**qed**

**have** *h-0*:  $h 0 \xi = (\int \omega. f \omega \partial PiM \{..<n\} M)$  **for**  $\xi$   
**unfolding** *h-def* **by** (*intro Bochner-Integration.integral-cong PiM-cong refl*)  
*(simp-all add:space-PiM atLeast0LessThan)*

**have** *h-cong*:  $h j \omega = h j \xi$  **if**  $\text{restrict } \omega \{..<j\} = \text{restrict } \xi \{..<j\}$   
**for**  $j \omega \xi$   
**using** *that* **unfolding** *h-def*  
**by** (*intro Bochner-Integration.integral-cong refl arg-cong*[**where** *f=f*] *merge-cong*) *auto*

**have** *h-meas*:  $h i \in \text{borel-measurable } (PiM I M)$  **if**  $i \leq n$   $\{..<i\} \subseteq I$   
**for**  $i I$   
**proof** –  
**have**  $0: \{..<n\} = \{..<i\} \cup \{i..<n\}$   
**using** *that(1)* **by** *auto*

**have** 1:  $\text{merge } \{..<i\} \{i..<n\} = \text{merge } \{..<i\} \{i..<n\} \circ \text{map-prod}$   
 $(\lambda x. \text{restrict } x \{..<i\}) \text{id}$   
**unfolding**  $\text{merge-def map-prod-def restrict-def comp-def}$   
**by**  $(\text{intro ext}) (\text{auto simp:case-prod-beta}')$

**have**  $\text{merge } \{..<i\} \{i..<n\} \in \text{Pi}_M I M \otimes_M \text{Pi}_M \{i..<n\} M \rightarrow_M$   
 $\text{Pi}_M \{..<n\} M$   
**unfolding** 0 **by**  $(\text{subst } 1) (\text{intro measurable-comp}[OF - \text{measurable-merge}] \text{map-prod-measurable}$   
 $\text{measurable-ident measurable-restrict-subset that}(2))$   
**hence**  $(\lambda x. f (\text{merge } \{..<i\} \{i..<n\} x)) \in \text{borel-measurable } (\text{Pi}_M$   
 $I M \otimes_M \text{Pi}_M \{i..<n\} M)$   
**by**  $(\text{intro measurable-compose}[OF - f-meas])$   
**thus** ?thesis  
**unfolding** h-def **by**  $(\text{intro sigma-finite-measure.borel-measurable-lebesgue-integral}$   
 $\text{prob-space-imp-sigma-finite prob-space-PiM assms}(1)) (\text{auto}$   
 $\text{simp:case-prod-beta}')$   
**qed**

**have**  $\text{merge-space-aux:merge } \{..<j\} \{j..<n\} u \in (\Pi_E i \in \{..<n\}. \text{space}$   
 $(M i))$   
**if**  $j \leq n$  **fst**  $u \in \text{Pi } \{..<j\} (\lambda i. \text{space } (M i))$  **snd**  $u \in \text{Pi } \{j..<n\}$   
 $(\lambda i. \text{space } (M i))$   
**for**  $u j$   
**proof** –  
**have**  $\text{merge } \{..<j\} \{j..<n\} (\text{fst } u, \text{snd } u) \in (\text{PiE } (\{..<j\} \cup \{j..<n\}))$   
 $(\lambda i. \text{space } (M i))$   
**using** that **by**  $(\text{intro iffD2}[OF \text{PiE-cancel-merge}] \text{auto})$   
**also have**  $\dots = (\Pi_E i \in \{..<n\}. \text{space } (M i))$   
**using** that **by**  $(\text{intro arg-cong2}[\text{where } f=\text{PiE}] \text{refl}) \text{auto}$   
**finally show** ?thesis **by**  $\text{simp}$   
**qed**

**have**  $\text{merge-space:merge } \{..<j\} \{j..<n\} (u, v) \in (\Pi_E i \in \{..<n\}. \text{space}$   
 $(M i))$   
**if**  $j \leq n$   $u \in \text{PiE } \{..<j\} (\lambda i. \text{space } (M i))$   $v \in \text{PiE } \{j..<n\} (\lambda i.$   
 $\text{space } (M i))$   
**for**  $u v j$   
**using** that **by**  $(\text{intro merge-space-aux}) (\text{simp-all add:PiE-def})$

**have**  $\text{delta!}: |f x - f y| \leq (\sum i < n. c i)$   
**if**  $x \in \text{PiE } \{..<n\} (\lambda i. \text{space } (M i))$   $y \in \text{PiE } \{..<n\} (\lambda i. \text{space } (M$   
 $i))$  **for**  $x y$   
**proof** –  
**define**  $m$  **where**  $m i = \text{merge } \{..<i\} \{i..<n\} (x, y)$  **for**  $i$

**have** 0:  $z \in \text{Pi } I (\lambda i. \text{space } (M i))$  **if**  $z \in \text{PiE } \{..<n\} (\lambda i. \text{space}$   
 $(M i))$

$I \subseteq \{..<n\}$  for  $z I$   
**using that unfolding PiE-def by auto**

**have**  $\exists$ :  $\{..<Suc\ i\} \cap (\{..<n\} - \{i\}) = \{..<i\}$   
 $\{Suc\ i..<n\} \cap (\{..<n\} - \{i\}) = \{Suc\ i..<n\}$   
 $\{..<i\} \cap (\{..<n\} - \{i\}) = \{..<i\}$   
 $\{i..<n\} \cap (\{..<n\} - \{i\}) = \{Suc\ i..<n\}$   
**if**  $i < n$  **for**  $i$   
**using that by auto**

**have**  $|f\ x - f\ y| = |f\ (m\ n) - f\ (m\ 0)|$   
**using that unfolding m-def by (simp add:atLeast0LessThan)**  
**also have**  $... = |\sum\ i < n. f\ (m\ (Suc\ i)) - f\ (m\ i)|$   
**by (subst sum-lessThan-telescope) simp**  
**also have**  $... \leq (\sum\ i < n. |f\ (m\ (Suc\ i)) - f\ (m\ i)|)$   
**by simp**  
**also have**  $... \leq (\sum\ i < n. c\ i)$   
**using that unfolding m-def by (intro delta sum-mono merge-space-aux**  
 $0\ subsetI)$   
 $(simp-all\ add:restrict-merge\ \exists)$   
**finally show ?thesis**  
**by simp**  
**qed**

**have**  $norm\ (f\ x) \leq norm\ (f\ x0) + sum\ c\ \{..<n\}$  **if**  $x \in space\ (Pi_M$   
 $\{..<n\}\ M)$  **for**  $x$   
**proof -**  
**have**  $|f\ x - f\ x0| \leq sum\ c\ \{..<n\}$   
**using**  $x0$  **that unfolding space-PiM by (intro delta') auto**  
**thus ?thesis**  
**by simp**  
**qed**  
**hence**  $f$ -bounded: bounded  $(f\ 'space\ (Pi_M\ \{..<n\}\ M))$   
**by (intro boundedI[where B=norm (f x0) + (\sum i<n. c i)]) auto**

**have**  $f$ -merge-bounded:  
 $bounded\ ((\lambda\ \omega. (f\ (merge\ \{..<j\}\ \{j..<n\}\ (u, \omega))))\ 'space\ (Pi_M$   
 $\{j..<n\}\ M))$   
**if**  $j \leq n$   $u \in PiE\ \{..<j\}$   $(\lambda i. space\ (M\ i))$  **for**  $u\ j$   
**proof -**  
**have**  $(\lambda\ \omega. merge\ \{..<j\}\ \{j..<n\}\ (u, \omega))\ 'space\ (Pi_M\ \{j..<n\}\ M)$   
 $\subseteq space\ (Pi_M\ \{..<n\}\ M)$   
**using that unfolding space-PiM**  
**by (intro image-subsetI merge-space) auto**  
**thus ?thesis**  
**by (subst image-image[of f,symmetric]) (intro bounded-subset[OF**  
 $f$ -bounded] image-mono)  
**qed**

**have** *f-merge-meas-aux*:  
 $(\lambda\omega. f (\text{merge } \{..\<j\} \{j..\<n\} (u, \omega))) \in \text{borel-measurable } (Pi_M \{j..\<n\} M)$   
**if**  $j \leq n$   $u \in Pi \{..\<j\} (\lambda i. \text{space } (M i))$  **for**  $j$   $u$   
**proof** –

**have**  $0: \{..\<n\} = \{..\<j\} \cup \{j..\<n\}$   
**using** *that(1)* **by** *auto*

**have**  $1: \text{merge } \{..\<j\} \{j..\<n\} (u, \omega) = \text{merge } \{..\<j\} \{j..\<n\}$   
*(restrict u {..\<j}, \omega)* **for**  $\omega$   
**by** *(intro merge-cong) auto*

**have**  $(\lambda\omega. \text{merge } \{..\<j\} \{j..\<n\} (u, \omega)) \in Pi_M \{j..\<n\} M \rightarrow_M Pi_M \{..\<n\} M$   
**using** *that unfolding 0 1*  
**by** *(intro measurable-compose[OF - measurable-merge] measurable-Pair1')*  
*(simp add:space-PiM)*  
**thus** *?thesis*  
**by** *(intro measurable-compose[OF - f-meas])*  
**qed**

**have** *f-merge-meas*:  $(\lambda\omega. f (\text{merge } \{..\<j\} \{j..\<n\} (u, \omega))) \in \text{borel-measurable } (Pi_M \{j..\<n\} M)$   
**if**  $j \leq n$   $u \in PiE \{..\<j\} (\lambda i. \text{space } (M i))$  **for**  $j$   $u$   
**using** *that unfolding PiE-def* **by** *(intro f-merge-meas-aux) auto*

**have** *h-bounded*: *bounded*  $(h i \text{ 'space } (PiM I M))$   
**if** *h-bounded-assms*:  $i \leq n$   $\{..\<i\} \subseteq I$  **for**  $i$   $I$   
**proof** –

**have**  $\text{merge } \{..\<i\} \{i..\<n\} x \in \text{space } (Pi_M \{..\<n\} M)$   
**if**  $x \in (\Pi_E i \in I. \text{space } (M i)) \times (\Pi_E i \in \{i..\<n\}. \text{space } (M i))$  **for**  $x$   
**using** *that h-bounded-assms unfolding space-PiM* **by** *(intro merge-space-aux)*  
*(auto simp: PiE-def mem-Times-iff)*  
**hence** *bounded*  $((\lambda x. f (\text{merge } \{..\<i\} \{i..\<n\} x)) \text{ ' } ((\Pi_E i \in I. \text{space } (M i)) \times (\Pi_E i \in \{i..\<n\}. \text{space } (M i))))$   
**by** *(subst image-image[of f, symmetric])*  
*(intro bounded-subset[OF f-bounded] image-mono image-subsetI)*  
**thus** *?thesis*  
**using** *that unfolding h-def*  
**by** *(intro prob-space.finite-measure finite-measure.bounded-int-int-rules)*  
*(auto simp:space-PiM PiE-def)*  
**qed**

**have** *V-bounded*: *bounded*  $(V i \text{ 'space } (PiM I M))$

**if**  $i < n$   $\{..<i+1\} \subseteq I$  **for**  $i \in I$   
**using that unfolding**  $V\text{-def}$  **by** (*intro bounded-intros h-bounded*)  
*auto*

**have**  $V\text{-upd-bounded: bounded } ((\lambda x. V j (\xi(j := x))) \text{ 'space } (M j))$   
**if**  $V\text{-upd-bounded-assms: } \xi \in \text{space } (Pi_M \{..<j\} M)$   $j < n$  **for**  $j \in \xi$   
**proof** –  
**have**  $\xi(j := v) \in \text{space } (Pi_M \{..<j + 1\} M)$  **if**  $v \in \text{space } (M j)$   
**for**  $v$   
**using**  $V\text{-upd-bounded-assms}$  **that unfolding**  $\text{space-PiM PiE-def}$   
*extensional-def Pi-def* **by** *auto*  
**thus** *?thesis*  
**using that unfolding**  $\text{image-image[of } V j (\lambda x. (\xi(j := x))), \text{symmetric]}$   
**by** (*intro bounded-subset[OF V-bounded[of j {..<j+1}]*) *that*  
*image-mono*) *auto*  
**qed**

**have**  $h\text{-step: } h j \omega = \int \tau. h (j+1) (\omega (j := \tau)) \partial M j$  (**is**  $?L1 = ?R1$ )  
**if**  $\omega \in \text{space } (Pi_M \{..<j\} M)$   $j < n$  **for**  $j \in \omega$   
**proof** –  
**have**  $0: (\lambda x. f (\text{merge } \{..<j\} \{j..<n\} (\omega, x))) \in \text{borel-measurable}$   
 $(Pi_M \{j..<n\} M)$   
**using that unfolding**  $\text{space-PiM}$  **by** (*intro f-merge-meas*) *auto*

**have**  $1: \text{insert } j \{Suc j..<n\} = \{j..<n\}$   
**using that** *by auto*

**have**  $2: \text{bounded } ((\lambda x. (f (\text{merge } \{..<j\} \{j..<n\} (\omega, x)))) \text{ 'space}$   
 $(Pi_M \{j..<n\} M))$   
**using that by** (*intro f-merge-bounded*) (*simp-all add: space-PiM*)

**have**  $?L1 = (\int \xi. f (\text{merge } \{..<j\} \{j..<n\} (\omega, \xi)) \partial Pi_M (\text{insert } j$   
 $\{j+1..<n\} M))$   
**unfolding**  $h\text{-def}$  **using that by** (*intro Bochner-Integration.integral-cong*  
*refl PiM-cong*) *auto*  
**also have**  $... = (\int \tau. (\int \xi. f (\text{merge } \{..<j\} \{j..<n\} (\omega, (\xi(j := \tau))))$   
 $\partial Pi_M \{j+1..<n\} M) \partial M j)$   
**using that(1,2)**  $0 \ 1 \ 2$  **by** (*intro product-integral-insert prob-space-imp-sigma-finite*  
*assms(1)*  
*int-rules f-merge-meas*) (*simp-all*)  
**also have**  $... = ?R1$   
**using that(2)** **unfolding**  $h\text{-def}$   
**by** (*intro Bochner-Integration.integral-cong arg-cong[where f=f]*  
*ext*) (*auto simp:merge-def*)  
**finally show** *?thesis*  
**by** *simp*  
**qed**

**have**  $V$ -meas:  $V i \in \text{borel-measurable } (PiM I M)$  **if**  $i < n \{..<i+1\}$   
 $\subseteq I$  **for**  $i I$   
**unfolding**  $V$ -def **using that by** (intro borel-measurable-diff h-meas)  
*auto*

**have**  $V$ -upd-meas:  $(\lambda x. V j (\xi(j := x))) \in \text{borel-measurable } (M j)$   
**if**  $j < n \xi \in \text{space } (PiM \{..<j\} M)$  **for**  $j \xi$   
**using that by** (intro measurable-compose[ $OF - V$ -meas[**where**  
 $I = \text{insert } j \{..<j\}$ ]]  
*measurable-component-update*) *auto*

**have**  $V$ -cong:  
 $V j \omega = V j \xi$  **if**  $\text{restrict } \omega \{..<(j+1)\} = \text{restrict } \xi \{..<(j+1)\}$  **for**  
 $j \omega \xi$   
**using that** *restrict-subset-eq*[ $OF - \text{that}$ ] **unfolding**  $V$ -def  
**by** (intro *arg-cong2*[**where**  $f = (-)$ ] *h-cong*) *simp-all*

**have**  $\text{exp-}V$ :  $(\int \omega. V j (\xi(j := \omega)) \partial M j) = 0$  (**is** ? $L1 = 0$ )  
**if**  $j < n \xi \in \text{space } (PiM \{..<j\} M)$  **for**  $j \xi$   
**proof** –

**have**  $\text{fun-upd } \xi j$  ‘  $\text{space } (M j) \subseteq \text{space } (PiM (\text{insert } j \{..<j\}) M)$   
**using that** **unfolding** *space-PiM* **by** (intro *image-subsetI PiE-fun-upd*)  
*auto*  
**hence**  $0$ :bounded  $((\lambda x. h (\text{Suc } j) (\xi(j := x)))$  ‘  $\text{space } (M j)$ )  
**unfolding** *image-image*[of  $h (\text{Suc } j) \lambda x. \xi(j := x)$ , *symmetric*]  
**using that**  
**by** (intro *bounded-subset*[ $OF$  *h-bounded*[**where**  $i = j + 1$  **and**  $I = \{..<j+1\}$ ]])  
*image-mono*)  
*(auto simp:lessThan-Suc)*

**have**  $1$ : $(\lambda x. h (\text{Suc } j) (\xi(j := x))) \in \text{borel-measurable } (M j)$   
**using**  $h$ -meas **that by** (intro measurable-compose[ $OF - h$ -meas[**where**  
 $I = \text{insert } j \{..<j\}$ ]]  
*measurable-component-update*) *auto*

**have** ? $L1 = (\int \omega. h (\text{Suc } j) (\xi(j := \omega)) - h j \xi \partial M j)$   
**unfolding**  $V$ -def  
**by** (intro *Bochner-Integration.integral-cong arg-cong2*[**where**  
 $f = (-)$ ] *refl h-cong*) *auto*  
**also have** ... =  $(\int \omega. h (\text{Suc } j) (\xi(j := \omega)) \partial M j) - (\int \omega. h j \xi \partial M j)$   
**using that by** (intro *Bochner-Integration.integral-diff int-rules 0 1*) *auto*  
**also have** ... = 0  
**using that** (1) *assms(1) prob-space.prob-space* **unfolding**  $h$ -step[ $OF$   
*that(2,1)*] **by** *auto*  
**finally show** ?*thesis*  
**by** *simp*

qed

**have**  $\text{var-}V: |V j x - V j y| \leq c j$  (**is**  $?L1 \leq ?R1$ )  
**if**  $\text{var-}V\text{-assms}: j < n \{x, y\} \subseteq \text{space } (PiM \{..<j+1\} M)$   
 $\text{restrict } x \{..<j\} = \text{restrict } y \{..<j\}$  **for**  $x y j$

**proof** –

**have**  $x\text{-ran}: x \in PiE \{..<j+1\} (\lambda i. \text{space } (M i))$  **and**  $y\text{-ran}: y \in PiE \{..<j+1\} (\lambda i. \text{space } (M i))$   
**using**  $\text{that}(2)$  **by**  $(\text{simp-all add:space-PiM})$

**have**  $0: j+1 \leq n$   
**using**  $\text{that}$  **by**  $\text{simp}$

**have**  $?L1 = |h (Suc j) x - h j y - (h (Suc j) y - h j y)|$   
**unfolding**  $V\text{-def}$  **by**  $(\text{intro arg-cong}[\text{where } f=abs] \text{arg-cong2}[\text{where } f=(-)] \text{refl h-cong that})$

**also have**  $\dots = |h (j+1) x - h (j+1) y|$   
**by**  $\text{simp}$

**also have**  $\dots =$

$|(\int \omega. f(\text{merge } \{..<j+1\} \{j+1..<n\} (x, \omega)) - f(\text{merge } \{..<j+1\} \{j+1..<n\} (y, \omega))) \partial PiM \{j+1..<n\} M|$

**using**  $\text{that unfolding h-def by (intro arg-cong[where } f=abs] f\text{-merge-meas}[OF 0] x\text{-ran})$

$\text{Bochner-Integration.integral-diff[symmetric] int-rules f-merge-bounded}[OF 0] y\text{-ran}) \text{auto}$

**also have**  $\dots \leq$

$(\int \omega. |f(\text{merge } \{..<j+1\} \{j+1..<n\} (x, \omega)) - f(\text{merge } \{..<j+1\} \{j+1..<n\} (y, \omega))| \partial PiM \{j+1..<n\} M)$

**by**  $(\text{intro integral-abs-bound})$

**also have**  $\dots \leq (\int \omega. c j \partial PiM \{j+1..<n\} M)$

**proof**  $(\text{intro Bochner-Integration.integral-mono}' \text{delta int-rules c-ge-0 ballI merge-space } 0)$

**fix**  $\omega$  **assume**  $\omega \in \text{space } (PiM \{j+1..<n\} M)$

**have**  $\{..<j+1\} \cap (\{..<n\} - \{j\}) = \{..<j\}$

**using**  $\text{that by auto}$

**thus**  $\text{restrict } (\text{merge } \{..<j+1\} \{j+1..<n\} (x, \omega)) (\{..<n\} - \{j\})$

$=$

$\text{restrict } (\text{merge } \{..<j+1\} \{j+1..<n\} (y, \omega)) (\{..<n\} - \{j\})$

**using**  $\text{that}(1,3) \text{less-antisym unfolding restrict-merge by (intro merge-cong refl) auto}$

**qed**  $(\text{simp-all add:space-PiM that}(1) x\text{-ran}[\text{simplified}] y\text{-ran}[\text{simplified}])$

**also have**  $\dots = c j$

**by**  $(\text{auto intro!:prob-space.prob-space prob-space-PiM assms}(1))$

**finally show**  $?thesis$  **by**  $\text{simp}$

qed

**have**  $f \xi - (\int \omega. f \omega \partial (PiM \{..<n\} M)) = (\sum i < n. V i \xi)$  **if**  $\xi \in \text{space } (PiM \{..<n\} M)$  **for**  $\xi$

**using**  $\text{that unfolding V-def by (subst sum-lessThan-telescope)}$

*(simp add: h-0 h-n)*  
**hence**  $?L = \mathcal{P}(\xi \text{ in } PiM \{..<n\} M. (\sum i < n. V i \xi) \geq \varepsilon)$   
**by** (*intro arg-cong2[where f=measure] refl Collect-restr-cong arg-cong2[where f=(≤)] auto*)  
**also have**  $\dots \leq \mathcal{P}(\xi \text{ in } PiM \{..<n\} M. \exp(t * (\sum i < n. V i \xi)) \geq \exp(t * \varepsilon))$   
**proof** (*intro finite-measure.finite-measure-mono subsetI prob-space.finite-measure int-rules*)  
**show**  $\{\xi \in \text{space } (PiM \{..<n\} M). \exp(t * \varepsilon) \leq \exp(t * (\sum i < n. V i \xi))\} \in \text{sets } (PiM \{..<n\} M)$   
**using** *V-meas by measurable*  
**qed** (*auto intro!:mult-left-mono[OF - t-ge-0]*)  
**also have**  $\dots \leq (\int \xi. \exp(t * (\sum i < n. V i \xi)) \partial PiM \{..<n\} M) / \exp(t * \varepsilon)$   
**by** (*intro integral-Markov-inequality-measure[where A={}] int-rules V-bounded V-meas*) *auto*  
**also have**  $\dots = \exp(t^2 * (\sum i \in \{n..<n\}. c i^2) / 8 - t * \varepsilon) * (\int \xi. \exp(t * (\sum i < n. V i \xi)) \partial PiM \{..<n\} M)$   
**by** (*simp add:exp-minus inverse-eq-divide*)  
**also have**  $\dots \leq \exp(t^2 * (\sum i \in \{0..<n\}. c i^2) / 8 - t * \varepsilon) * (\int \xi. \exp(t * (\sum i < 0. V i \xi)) \partial PiM \{..<0\} M)$   
**proof** (*rule ineq-chain*)  
**fix** *j* **assume** *a:j < n*  
**let**  $?L1 = \exp(t^2 * (\sum i=j+1..<n. (c i)^2) / 8 - t * \varepsilon)$   
**let**  $?L2 = ?L1 * (\int \xi. \exp(t * (\sum i < j+1. V i \xi)) \partial PiM \{..<j+1\} M)$   
**note**  $V\text{-upd-meas} = V\text{-upd-meas}[OF a]$   
**have**  $?L2 = ?L1 * (\int \xi. \exp(t * (\sum i < j. V i \xi)) * \exp(t * V j \xi) \partial PiM (\text{insert } j \{..<j\}) M)$   
**by** (*simp add:algebra-simps exp-add lessThan-Suc*)  
**also have**  $\dots = ?L1 * (\int \xi. (\int \omega. \exp(t * (\sum i < j. V i (\xi(j := \omega)))) * \exp(t * V j (\xi(j := \omega))) \partial M j) \partial PiM \{..<j\} M)$   
**using** *a* **by** (*intro product-integral-insert-rev arg-cong2[where f=(\*)] int-rules prob-space-imp-sigma-finite V-bounded V-meas*) *auto*  
**also have**  $\dots = ?L1 * (\int \xi. (\int \omega. \exp(t * (\sum i < j. V i \xi)) * \exp(t * V j (\xi(j := \omega))) \partial M j) \partial PiM \{..<j\} M)$   
**by** (*intro arg-cong2[where f=(\*)] Bochner-Integration.integral-cong arg-cong[where f=exp] sum.cong V-cong restrict-fupd*) *auto*  
**also have**  $\dots = ?L1 * (\int \xi. \exp(t * (\sum i < j. V i \xi)) * (\int \omega. \exp(t * V j (\xi(j := \omega))) \partial M j) \partial PiM \{..<j\} M)$   
**using** *a* **by** (*intro arg-cong2[where f=(\*)] Bochner-Integration.integral-cong refl Bochner-Integration.integral-mult-right V-upd-meas V-upd-bounded int-rules*) *auto*  
**also have**  $\dots \leq ?L1 * \int \xi. \exp(t * (\sum i < j. V i \xi)) * \exp(t^2 * c$

$j^2/8) \partial PiM \{..<j\} M$   
**proof** (intro mult-left-mono integral-mono)  
**fix**  $\xi$  **assume**  $c:\xi \in \text{space } (PiM \{..<j\} M)$   
**hence**  $b:\xi \in PiE \{..<j\} (\lambda i. \text{space } (M i))$   
**unfolding** space-PiM **by** simp  
**moreover** **have**  $\xi(j := v) \in PiE \{..<j+1\} (\lambda i. \text{space } (M i))$  **if**  
 $v \in \text{space } (M j)$  **for**  $v$   
**using**  $b$  **that** **unfolding** PiE-def extensional-def Pi-def **by** auto  
**ultimately** **show**  $LINT \omega | M j. \exp(t * V j (\xi(j := \omega))) \leq \exp$   
 $(t^2 * (c j)^2 / 8)$   
**using** V-upd-meas[OF c]  
**by** (intro prob-space.Hoeffdings-lemma-bochner-3 exp-V var-V a  
int-rules)  
(auto simp: space-PiM)  
**next**  
**show** integrable  $(PiM \{..<j\} M) (\lambda x. \exp(t * (\sum i < j. V i x))) * \exp$   
 $(t^2 * (c j)^2 / 8)$   
**using**  $a$  **by** (intro int-rules V-bounded V-meas) auto  
**qed** auto  
**also** **have**  $... = ?L1 * ((\int \xi. \exp(t * (\sum i < j. V i \xi))) \partial PiM \{..<j\} M) * \exp$   
 $(t^2 * (c j)^2 / 8)$   
**proof** (subst Bochner-Integration.integral-mult-left)  
**show** integrable  $(PiM \{..<j\} M) (\lambda \xi. \exp(t * (\sum i < j. V i \xi)))$   
**using**  $a$  **by** (intro int-rules V-bounded V-meas) auto  
**qed** auto  
**also** **have**  $... =$   
 $\exp(t^2 * (\sum i \in \text{insert } j \{j+1..<n\}. (c i)^2) / 8 - t * \varepsilon) * (\int \xi. \exp(t * (\sum i < j. V i \xi))) \partial PiM \{..<j\} M)$   
**by** (simp-all add:exp-add[symmetric] field-simps)  
**also** **have**  $... = \exp(t^2 * (\sum i = j..<n. (c i)^2) / 8 - t * \varepsilon) * (\int \xi. \exp(t * (\sum i < j. V i \xi))) \partial PiM \{..<j\} M)$   
**using**  $a$  **by** (intro arg-cong2[where f=(\*)] arg-cong[where f=exp] refl arg-cong2  
[where f=(-)] arg-cong2[where f=(/)] sum.cong) auto  
**finally** **show**  $?L2 \leq \exp(t^2 * (\sum i = j..<n. (c i)^2) / 8 - t * \varepsilon) * (\int \xi. \exp$   
 $(t * (\sum i < j. V i \xi))) \partial PiM \{..<j\} M)$   
**by** simp  
**qed**  
**also** **have**  $... = \exp(t^2 * (\sum i < n. c i^2) / 8 - t * \varepsilon)$  **by** (simp add:PiM-empty  
atLeast0LessThan)  
**also** **have**  $... = \exp(t * ((t * (\sum i < n. c i^2) / 8) - \varepsilon))$  **by** (simp  
add:algebra-simps power2-eq-square)  
**also** **have**  $... = \exp(t * (-\varepsilon/2))$  **using** sum-c-ge-0 **by** (auto simp  
add:divide-simps t-def)  
**also** **have**  $... = ?R$  **unfolding** t-def **by** (simp add:field-simps power2-eq-square)  
**finally** **show** ?thesis **by** simp  
**qed**

**theorem** mc-diarmid-inequality-distr:

**fixes**  $f :: ('i \Rightarrow 'a) \Rightarrow \text{real}$   
**assumes**  $\text{finite } I$   
**assumes**  $\bigwedge i. i \in I \Longrightarrow \text{prob-space } (M\ i)$   
**assumes**  $\bigwedge i\ x\ y. i \in I \Longrightarrow \{x, y\} \subseteq \text{space } (PiM\ I\ M) \Longrightarrow (\forall j \in I - \{i\}. x\ j = y\ j) \Longrightarrow |f\ x - f\ y| \leq c\ i$   
**assumes**  $f\text{-meas}: f \in \text{borel-measurable } (PiM\ I\ M)$  **and**  $\varepsilon\text{-gt-0}: \varepsilon > 0$   
**shows**  $\mathcal{P}(\omega \text{ in } PiM\ I\ M. f\ \omega - (\int \xi. f\ \xi\ \partial PiM\ I\ M) \geq \varepsilon) \leq \exp(-(\mathcal{L} * \varepsilon^{\wedge 2}) / (\sum i \in I. (c\ i)^{\wedge 2}))$   
**(is ?L ≤ ?R)**  
**proof** –  
**define**  $n$  **where**  $n = \text{card } I$   
**let**  $?q = \text{from-nat-into } I$   
**let**  $?r = \text{to-nat-on } I$   
**let**  $?f = (\lambda \xi. f\ (\lambda i \in I. \xi\ (?r\ i)))$   
  
**have**  $q: \text{bij-betw } ?q\ \{\dots < n\}\ I$  **unfolding**  $n\text{-def}$  **by**  $(\text{intro } \text{bij-betw-from-nat-into-finite } \text{assms}(1))$   
**have**  $r: \text{bij-betw } ?r\ I\ \{\dots < n\}$  **unfolding**  $n\text{-def}$  **by**  $(\text{intro } \text{to-nat-on-finite } \text{assms}(1))$   
  
**have**  $[\text{simp}]: ?q\ (?r\ x) = x$  **if**  $x \in I$  **for**  $x$   
**by**  $(\text{intro } \text{from-nat-into-to-nat-on that countable-finite } \text{assms}(1))$   
  
**have**  $[\text{simp}]: ?r\ (?q\ x) = x$  **if**  $x < n$  **for**  $x$   
**using**  $\text{bij-betw-imp-surj-on}[OF\ r]$  **that by**  $(\text{intro } \text{to-nat-on-from-nat-into } \text{auto})$   
  
**have**  $a: \bigwedge i. i \in \{\dots < n\} \Longrightarrow \text{prob-space } ((M \circ ?q)\ i)$   
**unfolding**  $\text{comp-def}$  **by**  $(\text{intro } \text{assms}(2)\ \text{bij-betw-apply}[OF\ q])$   
  
**have**  $b: PiM\ I\ M = PiM\ I\ (\lambda i. (M \circ ?q)\ (?r\ i))$  **by**  $(\text{intro } PiM\text{-cong})$   
 $(\text{simp-all } \text{add:comp-def})$   
**also have**  $\dots = \text{distr } (PiM\ \{\dots < n\}\ (M \circ ?q))\ (PiM\ I\ (\lambda i. (M \circ ?q)\ (?r\ i)))\ (\lambda \omega. \lambda n \in I. \omega\ (?r\ n))$   
**using**  $r$  **unfolding**  $\text{bij-betw-def}$  **by**  $(\text{intro } \text{distr-PiM-reindex}[\text{symmetric}])$   
 $a)$   $\text{auto}$   
**finally have**  $c: PiM\ I\ M = \text{distr } (PiM\ \{\dots < n\}\ (M \circ ?q))\ (PiM\ I\ (\lambda i. (M \circ ?q)\ (?r\ i)))\ (\lambda \omega. \lambda n \in I. \omega\ (?r\ n))$   
**by**  $\text{simp}$   
  
**have**  $d: (\lambda n \in I. x\ (?r\ n)) \in \text{space } (PiM\ I\ M)$  **if**  $\lambda x \in \text{space } (PiM\ \{\dots < n\}\ (M \circ ?q))$  **for**  $x$   
**proof** –  
**have**  $x\ (?r\ i) \in \text{space } (M\ i)$  **if**  $i \in I$  **for**  $i$   
**proof** –  
**have**  $?r\ i \in \{\dots < n\}$  **using**  $\text{bij-betw-apply}[OF\ r]$  **that by**  $\text{simp}$   
**hence**  $x\ (?r\ i) \in \text{space } ((M \circ ?q)\ (?r\ i))$  **using**  $\text{that } \lambda PiE\text{-mem}$   
**unfolding**  $\text{space-PiM}$  **by**  $\text{blast}$   
**thus**  $?thesis$  **using**  $\text{that unfolding comp-def by simp}$

```

qed
thus ?thesis unfolding space-PiM PiE-def by auto
qed

have ?L =  $\mathcal{P}(\omega \text{ in } PiM \{..<n\} (M \circ ?q). ?f \omega - (\int \xi. f \xi \partial PiM I M) \geq \varepsilon)$ 
proof (subst c, subst measure-distr, goal-cases)
  case 1 thus ?case
    by (intro measurable-restrict measurable-component-singleton
    bij-betw-apply[OF r])
  next
    case 2 thus ?case unfolding b[symmetric] by (intro measurable-sets-Collect[OF f-meas]) auto
  next
    case 3 thus ?case using d by (intro arg-cong2[where f=measure] refl) (auto simp:vimage-def)
qed
also have ... =  $\mathcal{P}(\omega \text{ in } PiM \{..<n\} (M \circ ?q). ?f \omega - (\int \xi. ?f \xi \partial PiM \{..<n\} (M \circ ?q)) \geq \varepsilon)$ 
proof (subst c, subst integral-distr, goal-cases)
  case (1  $\omega$ ) thus ?case
    by (intro measurable-restrict measurable-component-singleton
    bij-betw-apply[OF r])
  next
    case (2  $\omega$ ) thus ?case unfolding b[symmetric] by (rule f-meas)
  next
    case 3 thus ?case by simp
qed
also have ...  $\leq \exp(-(\mathcal{L} * \varepsilon^{\wedge 2}) / (\sum i < n. (c (?q i))^{\wedge 2}))$ 
proof (intro mc-diarmid-inequality-aux  $\varepsilon$ -gt-0, goal-cases)
  case (1  $i$ ) thus ?case by (intro a) auto
next
  case (2  $i$   $x$   $y$ )
  have  $x (?r j) = y (?r j)$  if  $j \in I - \{?q i\}$  for  $j$ 
  proof -
    have  $?r j \in \{..<n\} - \{i\}$  using that bij-betw-apply[OF r] by auto
  thus ?thesis using 2 by simp
qed
hence  $\forall j \in I - \{?q i\}. (\lambda i \in I. x (?r i)) j = (\lambda i \in I. y (?r i)) j$  by auto
thus ?case using 2 d by (intro assms(3) bij-betw-apply[OF q]) auto
next
  case 3
  have  $(\lambda x. x (?r i)) \in PiM \{..<n\} (M \circ ?q) \rightarrow_M M i$  if  $i \in I$  for  $i$ 
  proof -
    have  $0 : M i = (M \circ ?q) (?r i)$  using that by (simp add: comp-def)
    show ?thesis unfolding 0 by (intro measurable-component-singleton

```

*bij-betw-apply*[*OF r*] *that*)  
**qed**  
**thus** *?case* **by** (*intro measurable-compose*[*OF - f-meas*] *measurable-restrict*)  
**qed**  
**also have** ... = *?R* **by** (*subst sum.reindex-bij-betw*[*OF q*]) *simp*  
**finally show** *?thesis* **by** *simp*  
**qed**

**lemma** (*in prob-space*) *mc-diarmid-inequality-classic*:

**fixes** *f* :: (*'i*  $\Rightarrow$  *'a*)  $\Rightarrow$  *real*  
**assumes** *finite I*  
**assumes** *indep-vars N X I*  
**assumes**  $\bigwedge i x y. i \in I \implies \{x, y\} \subseteq \text{space } (PiM I N) \implies (\forall j \in I - \{i\}. x j = y j) \implies |f x - f y| \leq c i$   
**assumes** *f-meas*: *f*  $\in$  *borel-measurable* (*PiM I N*) **and**  *$\varepsilon$ -gt-0*:  $\varepsilon > 0$   
**shows**  $\mathcal{P}(\omega \text{ in } M. f (\lambda i \in I. X i \omega) - (\int \xi. f (\lambda i \in I. X i \xi) \partial M) \geq \varepsilon) \leq \exp(-2 * \varepsilon^2 / (\sum i \in I. (c i)^2))$   
**(is ?L  $\leq$  ?R)**

**proof** -

**note** *indep-imp* = *iffD1*[*OF indep-vars-iff-distr-eq-PiM'*]  
**let** *?O* =  $\lambda i. \text{distr } M (N i) (X i)$   
**have** *a*:  $\text{distr } M (PiM I N) (\lambda x. \lambda i \in I. X i x) = PiM I ?O$   
**using** *assms(2)* **unfolding** *indep-vars-def* **by** (*intro indep-imp*[*OF - assms(2)*]) *auto*

**have** *b*:  $\text{space } (PiM I ?O) = \text{space } (PiM I N)$   
**by** (*metis* (*no-types, lifting*) *a space-distr*)

**have**  $(\lambda i \in I. X i \omega) \in \text{space } (PiM I N)$  **if**  $\omega \in \text{space } M$  **for**  $\omega$   
**using** *assms(2)* **that** **unfolding** *indep-vars-def measurable-def space-PiM* **by** *auto*

**hence**  $?L = \mathcal{P}(\omega \text{ in } M. (\lambda i \in I. X i \omega) \in \text{space } (PiM I N) \wedge f (\lambda i \in I. X i \omega) - (\int \xi. f (\lambda i \in I. X i \xi) \partial M) \geq \varepsilon)$

**by** (*intro arg-cong2*[**where** *f=measure*] *Collect-restr-cong refl*) *auto*  
**also have** ... =  $\mathcal{P}(\omega \text{ in } \text{distr } M (PiM I N) (\lambda x. \lambda i \in I. X i x). f \omega - (\int \xi. f (\lambda i \in I. X i \xi) \partial M) \geq \varepsilon)$

**proof** (*subst measure-distr, goal-cases*)

**case 1** **thus** *?case* **using** *assms(2)* **unfolding** *indep-vars-def* **by** (*intro measurable-restrict*) *auto*

**next**

**case 2** **thus** *?case* **unfolding** *space-distr* **by** (*intro measurable-sets-Collect*[*OF f-meas*]) *auto*

**next**

**case 3** **thus** *?case* **by** (*simp-all add:Int-def conj-commute*)

**qed**

**also have** ... =  $\mathcal{P}(\omega \text{ in } PiM I ?O. f \omega - (\int \xi. f (\lambda i \in I. X i \xi) \partial M) \geq \varepsilon)$

```

    unfolding a by simp
  also have ... =  $\mathcal{P}(\omega \text{ in } PiM I ?O. f \omega - (\int \xi. f \xi \partial \text{ distr } M (PiM I N) (\lambda x. \lambda i \in I. X i x)) \geq \varepsilon)$ 
  proof (subst integral-distr[OF - f-meas], goal-cases)
    case (1  $\omega$ ) thus ?case using assms(2) unfolding indep-vars-def
    by (intro measurable-restrict) auto
  next
    case 2 thus ?case by simp
  qed
  also have ... =  $\mathcal{P}(\omega \text{ in } PiM I ?O. f \omega - (\int \xi. f \xi \partial PiM I ?O) \geq \varepsilon)$ 
  unfolding a by simp
  also have ...  $\leq ?R$ 
    using f-meas assms(2) b unfolding indep-vars-def
    by (intro mc-diarmid-inequality-distr prob-space-distr assms(1)
     $\varepsilon$ -gt-0 assms(3)) auto
  finally show ?thesis by simp
qed

end

```

## 7 Paley-Zygmund Inequality

This section proves slight improvements of the Paley-Zygmund Inequality [7]. Unfortunately, the improvements are on Wikipedia with no citation.

```

theory Paley-Zygmund-Inequality
  imports Lp.Lp
begin

```

```

context prob-space
begin

```

```

theorem paley-zygmund-inequality-holder:

```

```

  assumes p:  $1 < (p::real)$ 
  assumes rv: random-variable borel Z
  assumes intZp: integrable M ( $\lambda z. |Z z| \text{ powr } p$ )
  assumes t:  $\vartheta \leq 1$ 
  assumes ZAEpos:  $\text{AE } z \text{ in } M. Z z \geq 0$ 
  shows
    ( $\text{expectation } (\lambda x. |Z x - \vartheta * \text{expectation } Z| \text{ powr } p) \text{ powr } (1 / (p-1))) *
    \text{prob } \{z \in \text{space } M. Z z > \vartheta * \text{expectation } Z\}
    \geq ((1-\vartheta) \text{ powr } (p / (p-1))) * \text{expectation } Z \text{ powr } (p / (p-1))$ 

```

```

proof -

```

```

  have intZ: integrable M Z
  apply (subst bound-L1-Lp[OF - rv intZp])
  using p by auto

```

```

define eZ where eZ = expectation Z
have eZ ≥ 0
  unfolding eZ-def
  using ZAEpos intZ integral-ge-const prob-Collect-eq-1 by auto

have ezp: expectation (λx. |Z x - ∅ * eZ| powr p) ≥ 0
  by (meson Bochner-Integration.integral-nonneg powr-ge-zero)

have expectation (λz. Z z - ∅ * eZ) = expectation (λz. Z z + (- ∅
* eZ))
  by auto
moreover have ... = expectation Z + expectation (λz. - ∅ * eZ)
  apply (subst Bochner-Integration.integral-add)
  using intZ by auto
moreover have ... = eZ + (- ∅ * eZ)
  apply (subst lebesgue-integral-const)
  using eZ-def prob-space by auto
ultimately have *: expectation (λz. Z z - ∅ * eZ) = eZ - ∅ * eZ
  by linarith

have ev: {z ∈ space M. ∅ * eZ < Z z} ∈ events
  using rv unfolding borel-measurable-iff-greater
  by auto

define q where q = p / (p-1)

have sqI:(indicat-real E x) powr q = indicat-real E (x::'a) for E x
  unfolding q-def
  by (metis indicator-simps(1) indicator-simps(2) powr-0 powr-one-eq-one)

have bm1: (λz. (Z z - ∅ * eZ)) ∈ borel-measurable M
  using borel-measurable-const borel-measurable-diff rv by blast
have bm2: (λz. indicat-real {z ∈ space M. Z z > ∅ * eZ} z) ∈
borel-measurable M
  using borel-measurable-indicator ev by blast
have integrable M (λx. |Z x + (-∅ * eZ)| powr p)
  apply (intro Minkowski-inequality[OF - rv - intZp])
  using p by auto
then have int1: integrable M (λx. |Z x - ∅ * eZ| powr p)
  by auto

have integrable M
(λx. 1 * indicat-real {z ∈ space M. ∅ * eZ < Z z} x)
  apply (intro integrable-real-mult-indicator[OF ev])
  by auto

then have int2: integrable M
(λx. |indicat-real {z ∈ space M. ∅ * eZ < Z z} x| powr q)
  by (auto simp add: sqI )

```

```

have pq:p > (0::real) q > 0 1/p + 1/q = 1
  unfolding q-def using p by (auto simp:divide-simps)
from Holder-inequality[OF pq bm1 bm2 int1 int2]
have hi: expectation (λx. (Z x - v * eZ) * indicat-real {z ∈ space
M. v * eZ < Z z} x)
  ≤ expectation (λx. |Z x - v * eZ| powr p) powr (1 / p) *
  expectation (λx. |indicat-real {z ∈ space M. v * eZ < Z z} x|
powr q) powr (1 / q)
  by auto

have eZ - v * eZ ≤
  expectation (λz. (Z z - v * eZ) * indicat-real {z ∈ space M. Z z
> v * eZ} z)
  unfolding *[symmetric]
  apply (intro integral-mono)
  using intZ ev apply auto[1]
  apply (auto intro!: integrable-real-mult-indicator simp add: intZ
ev)[1]
  unfolding indicator-def of-bool-def
  by (auto simp add: mult-nonneg-nonpos2)

also have ... ≤
  expectation (λx. |Z x - v * eZ| powr p) powr (1 / p) *
  expectation (λx. indicat-real {z ∈ space M. v * eZ < Z z} x)
powr (1 / q)
  using hi by (auto simp add: sqI)

finally have eZ - v * eZ ≤
  expectation (λx. |Z x - v * eZ| powr p) powr (1 / p) *
  expectation (λx. indicat-real {z ∈ space M. v * eZ < Z z} x) powr
(1 / q)
  by auto

then have (eZ - v * eZ) powr q ≤
  (expectation (λx. |Z x - v * eZ| powr p) powr (1 / p) *
  expectation (λx. indicat-real {z ∈ space M. v * eZ < Z z} x) powr
(1 / q)) powr q
  by (smt (verit, ccfv-SIG) ⟨0 ≤ eZ⟩ mult-left-le-one-le powr-mono2
pq(2) right-diff-distrib' t zero-le-mult-iff)

also have ... =
  (expectation (λx. |Z x - v * eZ| powr p) powr (1 / p)) powr q *
  (expectation (λx. indicat-real {z ∈ space M. v * eZ < Z z} x)
powr (1 / q)) powr q
  using powr-ge-zero powr-mult by presburger
also have ... =
  (expectation (λx. |Z x - v * eZ| powr p) powr (1 / p)) powr q *
  (expectation (λx. indicat-real {z ∈ space M. v * eZ < Z z} x))

```

**by** (*smt* (*verit*, *ccfv-SIG*) *Bochner-Integration.integral-nonneg divide-le-eq-1-pos indicator-pos-le nonzero-eq-divide-eq p powr-one powr-powr q-def*)  
**also have** ... =  
 (*expectation* ( $\lambda x. |Z x - \vartheta * eZ| \text{ powr } p$ ) *powr* ( $1 / (p-1)$ )) \*  
 (*expectation* ( $\lambda x. \text{indicat-real } \{z \in \text{space } M. \vartheta * eZ < Z z\} x$ ))  
**by** (*smt* (*verit*, *ccfv-threshold*) *divide-divide-eq-right divide-self-if p powr-powr q-def times-divide-eq-left*)  
**also have** ... =  
 (*expectation* ( $\lambda x. |Z x - \vartheta * eZ| \text{ powr } p$ ) *powr* ( $1 / (p-1)$ )) \*  
*prob*  $\{z \in \text{space } M. Z z > \vartheta * eZ\}$   
**by** (*simp add: ev*)  
  
**finally have 1:** ( $eZ - \vartheta * eZ$ ) *powr*  $q \leq$   
 (*expectation* ( $\lambda x. |Z x - \vartheta * eZ| \text{ powr } p$ ) *powr* ( $1 / (p-1)$ )) \*  
*prob*  $\{z \in \text{space } M. Z z > \vartheta * eZ\}$  **by** *linarith*  
  
**have** ( $eZ - \vartheta * eZ$ ) *powr*  $q = ((1 - \vartheta) * eZ)$  *powr*  $q$   
**by** (*simp add: mult.commute right-diff-distrib*)  
**also have** ... =  $(1 - \vartheta)$  *powr*  $q * eZ$  *powr*  $q$   
**by** (*simp add: <0 ≤ eZ> powr-mult t*)  
**finally show** *?thesis using 1 eZ-def q-def by force*  
**qed**

**corollary** *paley-zygmund-inequality:*

**assumes** *rv: random-variable borel Z*

**assumes** *intZsq: integrable M* ( $\lambda z. (Z z)^2$ )

**assumes** *t:  $\vartheta \leq 1$*

**assumes** *Zpos:  $\bigwedge z. z \in \text{space } M \implies Z z \geq 0$*

**shows**

(*variance*  $Z + (1-\vartheta)^2 * (\text{expectation } Z)^2$ ) \*  
*prob*  $\{z \in \text{space } M. Z z > \vartheta * \text{expectation } Z\}$   
 $\geq (1-\vartheta)^2 * (\text{expectation } Z)^2$

**proof** –

**have** *ZAEpos: AE z in M. Z z ≥ 0*

**by** (*simp add: Zpos*)

**define** *p where*  $p = (2::\text{real})$

**have** *p1: 1 < p using p-def by auto*

**have** *integrable M* ( $\lambda z. |Z z| \text{ powr } p$ ) **unfolding** *p-def*

**using** *intZsq by auto*

**from** *paley-zygmund-inequality-holder[OF p1 rv this t ZAEpos]*

**have**  $(1 - \vartheta) \text{ powr } (p / (p - 1)) * (\text{expectation } Z \text{ powr } (p / (p - 1)))$

$\leq \text{expectation } (\lambda x. |Z x - \vartheta * \text{expectation } Z| \text{ powr } p) \text{ powr } (1 / (p - 1)) *$

*prob*  $\{z \in \text{space } M. \vartheta * \text{expectation } Z < Z z\} .$

```

then have hi:  $(1 - \vartheta)^2 * (\text{expectation } Z)^2$ 
   $\leq \text{expectation } (\lambda x. (Z x - \vartheta * \text{expectation } Z)^2) *$ 
   $\text{prob } \{z \in \text{space } M. \vartheta * \text{expectation } Z < Z z\}$ 
unfolding p-def by (auto simp add: Zpos t)

have intZ: integrable M Z
  apply (subst square-integrable-imp-integrable[OF rv intZsq])
  by auto

define eZ where eZ = expectation Z
have eZ  $\geq 0$ 
  unfolding eZ-def
  using Bochner-Integration.integral-nonneg Zpos by blast

have exp: expectation  $(\lambda x. |Z x - \vartheta * eZ|^p)$   $\geq 0$ 
  by (meson Bochner-Integration.integral-nonneg powr-ge-zero)

have expectation  $(\lambda z. Z z - \vartheta * eZ) = \text{expectation } (\lambda z. Z z + (- \vartheta$ 
* eZ))
  by auto
also have ... = expectation Z + expectation  $(\lambda z. - \vartheta * eZ)$ 
  apply (subst Bochner-Integration.integral-add)
  using intZ by auto
also have ... = eZ +  $(- \vartheta * eZ)$ 
  apply (subst lebesgue-integral-const)
  using eZ-def prob-space by auto
finally have *: expectation  $(\lambda z. Z z - \vartheta * eZ) = eZ - \vartheta * eZ$ 
  by linarith
have variance Z =
  variance  $(\lambda z. (Z z - \vartheta * eZ))$ 
  using * eZ-def by auto
also have ... =
  expectation  $(\lambda z. (Z z - \vartheta * eZ)^2)$ 
   $- (\text{expectation } (\lambda x. Z x - \vartheta * eZ))^2$ 
  apply (subst variance-eq)
  by (auto simp add: intZ power2-diff intZsq)
also have ... = expectation  $(\lambda z. (Z z - \vartheta * eZ)^2) - ((1-\vartheta)^2 * eZ^2)$ 
  unfolding * by (auto simp: algebra-simps power2-eq-square)
finally have veq: expectation  $(\lambda z. (Z z - \vartheta * eZ)^2) = (\text{variance } Z$ 
+  $(1-\vartheta)^2 * eZ^2)$ 
  by linarith
thus ?thesis
  using hi by (simp add: eZ-def)
qed

end

end

```

## References

- [1] G. Bennett. Probability inequalities for the sum of independent random variables. *Journal of the American Statistical Association*, 57(297):33–45, 1962.
- [2] S. Boucheron, G. Lugosi, and O. Bousquet. Concentration inequalities. In O. Bousquet, U. von Luxburg, and G. Rätsch, editors, *Advanced Lectures on Machine Learning, ML Summer Schools 2003, Canberra, Australia, February 2-14, 2003, Tübingen, Germany, August 4-16, 2003, Revised Lectures*, volume 3176 of *Lecture Notes in Computer Science*, pages 208–240. Springer, 2003.
- [3] F. P. Cantelli. Sui confini della probabilita. In *Atti del Congresso Internazionale dei Matematici: Bologna del 3 al 10 de settembre di 1928*, pages 47–60, 1929.
- [4] B. Efron and C. Stein. The Jackknife Estimate of Variance. *The Annals of Statistics*, 9(3):586 – 596, 1981.
- [5] M. Loève. *Probability Theory I*, chapter Sums of Independent Random Variables, pages 235–279. Springer New York, New York, NY, 1977.
- [6] C. McDiarmid. *Surveys in Combinatorics, 1989: Invited Papers at the Twelfth British Combinatorial Conference*, chapter On the method of bounded differences, pages 148 – 188. London Mathematical Society Lecture Note Series. Cambridge University Press, 1989.
- [7] R. E. Paley and A. Zygmund. A note on analytic functions in the unit circle. In *Mathematical Proceedings of the Cambridge Philosophical Society*, volume 28, pages 266–272. Cambridge University Press, 1932.
- [8] J. M. Steele. An Efron-Stein Inequality for Nonsymmetric Statistics. *The Annals of Statistics*, 14(2):753 – 758, 1986.