

Quasi-Borel Spaces

Michikazu Hirata, Yasuhiko Minamide, Tetsuya Sato

February 4, 2022

Abstract

The notion of quasi-Borel spaces was introduced by Heunen et al. [1]. The theory provides a suitable denotational model for higher-order probabilistic programming languages with continuous distributions.

This entry is a formalization of the theory of quasi-Borel spaces, including construction of quasi-Borel spaces (product, coproduct, function spaces), the adjunction between the category of measurable spaces and the category of quasi-Borel spaces, and the probability monad on quasi-Borel spaces. This entry also contains the formalization of the Bayesian regression presented in the work of Heunen et al.

This work is a part of the work by same authors, *Program Logic for Higher-Order Probabilistic Programs in Isabelle/HOL*, which will be published in proceedings of the 16th International Symposium on Functional and Logic Programming (FLOPS 2022).

Contents

1	Standard Borel Spaces	2
1.1	Definition	3
1.2	\mathbb{R} , \mathbb{N} , Boolean, $[0, \infty]$	4
1.3	$\mathbb{R} \times \mathbb{R}$	5
1.4	$\mathbb{N} \times \mathbb{R}$	14
2	Quasi-Borel Spaces	14
2.1	Definitions	15
2.1.1	Quasi-Borel Spaces	15
2.1.2	Morphism of Quasi-Borel Spaces	17
2.1.3	Empty Space	18
2.1.4	Unit Space	19
2.1.5	Subspaces	19
2.1.6	Image Spaces	20
2.1.7	Ordering of Quasi-Borel Spaces	21
2.2	Relation to Measurable Spaces	23
2.2.1	The Functor R	23

2.2.2	The Functor L	23
2.2.3	The Adjunction	25
2.2.4	The Adjunction w.r.t. Ordering	26
2.3	Product Spaces	27
2.3.1	Binary Product Spaces	27
2.3.2	Product Spaces	30
2.4	Coproduct Spaces	32
2.4.1	Binary Coproduct Spaces	32
2.4.2	Countable Coproduct Spaces	34
2.4.3	Lists	35
2.5	Function Spaces	38
2.5.1	Function Spaces	38
3	Probability Spaces	41
3.1	Probability Measures	41
3.1.1	Probability Measures	41
3.2	The Probability Monad	55
3.2.1	The Probability Monad P	55
3.2.2	Return	58
3.2.3	Bind	58
3.2.4	The Functorial Action $P(f)$	59
3.2.5	Join	60
3.2.6	Strength	60
3.2.7	Properties of Return and Bind	64
3.2.8	Properties of Integrals	64
3.3	Binary Product Measure	67
3.3.1	Binary Product Measure	67
3.3.2	Fubini Theorem	68
3.4	Measure as QBS Measure	72
3.5	Example of Probability Measures	73
3.5.1	Normal Distribution	73
3.5.2	Uniform Distribution	74
3.5.3	Bernoulli Distribution	75
3.6	Bayesian Linear Regression	76
3.6.1	Prior	76
3.6.2	Likelihood	77
3.6.3	Posterior	77
3.6.4	Normalizer	78

1 Standard Borel Spaces

```

theory StandardBorel
  imports HOL-Probability.Probability
begin

```

A standard Borel space is the Borel space associated with a Polish space. Here, we define standard Borel spaces in another, but equivalent, way. See [1] Proposition 5.

abbreviation $real\text{-}borel \equiv borel :: real\ measure$

abbreviation $nat\text{-}borel \equiv borel :: nat\ measure$

abbreviation $ennreal\text{-}borel \equiv borel :: ennreal\ measure$

abbreviation $bool\text{-}borel \equiv borel :: bool\ measure$

1.1 Definition

locale $standard\text{-}borel =$

fixes $M :: 'a\ measure$

assumes $exist\text{-}fg: \exists f \in M \rightarrow_M real\text{-}borel. \exists g \in real\text{-}borel \rightarrow_M M.$
 $\forall x \in space\ M. (g \circ f)\ x = x$

begin

abbreviation $fg \equiv (SOME\ k. (fst\ k) \in M \rightarrow_M real\text{-}borel \wedge$
 $(snd\ k) \in real\text{-}borel \rightarrow_M M \wedge$
 $(\forall x \in space\ M. ((snd\ k) \circ (fst\ k))\ x = x))$

definition $f \equiv (fst\ fg)$

definition $g \equiv (snd\ fg)$

lemma

shows $f\text{-}meas[simp,measurable] : f \in M \rightarrow_M real\text{-}borel$
and $g\text{-}meas[simp,measurable] : g \in real\text{-}borel \rightarrow_M M$
and $gf\text{-}comp\text{-}id[simp]: \bigwedge x. x \in space\ M \implies (g \circ f)\ x = x$
 $\bigwedge x. x \in space\ M \implies g\ (f\ x) = x$

$\langle proof \rangle$

lemma $standard\text{-}borel\text{-}sets[simp]:$

assumes $sets\ M = sets\ Y$

shows $standard\text{-}borel\ Y$

$\langle proof \rangle$

lemma $f\text{-}inj:$

$inj\text{-}on\ f\ (space\ M)$

$\langle proof \rangle$

lemma $singleton\text{-}sets:$

assumes $x \in space\ M$

shows $\{x\} \in sets\ M$

$\langle proof \rangle$

lemma $countable\text{-}space\text{-}discrete:$

assumes $countable\ (space\ M)$

shows $sets\ M = sets\ (count\text{-}space\ (space\ M))$

$\langle proof \rangle$

end

lemma *standard-borelI*:

assumes $f \in Y \rightarrow_M \text{real-borel}$
 $g \in \text{real-borel} \rightarrow_M Y$
and $\bigwedge y. y \in \text{space } Y \implies (g \circ f) y = y$
shows *standard-borel* Y
<proof>

locale *standard-borel-space-UNIV* = *standard-borel* +

assumes *space-UNIV*: $\text{space } M = \text{UNIV}$

begin

lemma *gf-comp-id'[simp]*:

$g \circ f = \text{id}$ $g (f x) = x$
<proof>

lemma *f-inj'*:

inj f
<proof>

lemma *g-surj'*:

surj g
<proof>

end

lemma *standard-borel-space-UNIV*:

assumes $f \in Y \rightarrow_M \text{real-borel}$
 $g \in \text{real-borel} \rightarrow_M Y$
 $(g \circ f) = \text{id}$
and $\text{space } Y = \text{UNIV}$
shows *standard-borel-space-UNIV* Y
<proof>

lemma *standard-borel-space-UNIV'*:

assumes *standard-borel* Y
and $\text{space } Y = \text{UNIV}$
shows *standard-borel-space-UNIV* Y
<proof>

1.2 \mathbb{R} , \mathbb{N} , Boolean, $[0, \infty]$

\mathbb{R} is a standard Borel space.

interpretation *real* : *standard-borel-space-UNIV* *real-borel*

<proof>

A non-empty Borel subspace of \mathbb{R} is also a standard Borel space.

lemma *real-standard-borel-subset*:
assumes $U \in \text{sets real-borel}$
and $U \neq \{\}$
shows *standard-borel (restrict-space real-borel U)*
<proof>

A non-empty measurable subset of a standard Borel space is also a standard Borel space.

lemma(*in standard-borel*) *standard-borel-subset*:
assumes $U \in \text{sets } M$
 $U \neq \{\}$
shows *standard-borel (restrict-space M U)*
<proof>

\mathbb{N} is a standard Borel space.

interpretation *nat : standard-borel-space-UNIV nat-borel*
<proof>

For a countable space X , X is a standard Borel space iff X is a discrete space.

lemma *countable-standard-iff*:
assumes $\text{space } X \neq \{\}$
and *countable (space X)*
shows *standard-borel X \longleftrightarrow sets X = sets (count-space (space X))*
<proof>

\mathbb{B} is a standard Borel space.

lemma *to-bool-measurable*:
assumes $f - \{ \text{True} \} \cap \text{space } M \in \text{sets } M$
shows $f \in M \rightarrow_M \text{bool-borel}$
<proof>

interpretation *bool : standard-borel-space-UNIV bool-borel*
<proof>

$[0, \infty]$ (the set of extended non-negative real numbers) is a standard Borel space.

interpretation *ennreal : standard-borel-space-UNIV ennreal-borel*
<proof>

1.3 $\mathbb{R} \times \mathbb{R}$

definition *real-to-01open* :: *real \Rightarrow real* **where**
real-to-01open r \equiv arctan r / pi + 1 / 2

definition *real-to-01open-inverse* :: *real \Rightarrow real* **where**
*real-to-01open-inverse r \equiv tan (pi * r - (pi / 2))*

lemma *real-to-01open-inverse-correct*:
real-to-01open-inverse \circ *real-to-01open* = *id*
 ⟨*proof*⟩

lemma *real-to-01open-inverse-correct'*:
assumes $0 < r < 1$
shows *real-to-01open* (*real-to-01open-inverse* r) = r
 ⟨*proof*⟩

lemma *real-to-01open-01* :
 $0 < \text{real-to-01open } r \wedge \text{real-to-01open } r < 1$
 ⟨*proof*⟩

lemma *real-to-01open-continuous*:
continuous-on UNIV real-to-01open
 ⟨*proof*⟩

lemma *real-to-01open-inverse-continuous*:
continuous-on $\{0 < .. < 1\}$ *real-to-01open-inverse*
 ⟨*proof*⟩

lemma *real-to-01open-inverse-measurable*:
real-to-01open-inverse \in *restrict-space real-borel* $\{0 < .. < 1\} \rightarrow_M$ *real-borel*
 ⟨*proof*⟩

fun *r01-binary-expansion''* :: *real* \Rightarrow *nat* \Rightarrow *nat* \times *real* \times *real* **where**
r01-binary-expansion'' r 0 = (if $1/2 \leq r$ then (1,1,1/2)
 else (0,1/2,0)) |
r01-binary-expansion'' r (Suc n) = (let (-,ur,lr) = *r01-binary-expansion''* r n ;
 $k = (ur + lr)/2$ in
 (if $k \leq r$ then (1,ur,k)
 else (0,k,lr)))

a_n where $r = 0.a_0a_1a_2\dots$ for $0 < r < 1$.

definition *r01-binary-expansion'* :: *real* \Rightarrow *nat* \Rightarrow *nat* **where**
r01-binary-expansion' r $n \equiv$ *fst* (*r01-binary-expansion''* r n)

$a_n = 0$ or 1 .

lemma *real01-binary-expansion'-0or1*:
r01-binary-expansion' r $n \in \{0,1\}$
 ⟨*proof*⟩

definition *r01-binary-sum* :: (*nat* \Rightarrow *nat*) \Rightarrow *nat* \Rightarrow *real* **where**
r01-binary-sum a $n \equiv$ ($\sum_{i=0..n}$ *real* (a i) * $((1/2) \wedge$ (Suc i)))

definition *r01-binary-sum-lim* :: (*nat* \Rightarrow *nat*) \Rightarrow *real* **where**
r01-binary-sum-lim \equiv *lim* \circ *r01-binary-sum*

definition $r01\text{-binary-expression} :: \text{real} \Rightarrow \text{nat} \Rightarrow \text{real}$ **where**
 $r01\text{-binary-expression} \equiv r01\text{-binary-sum} \circ r01\text{-binary-expansion}'$

lemma $r01\text{-binary-expansion-lr-r-ur}$:

assumes $0 < r \ r < 1$

shows $(\text{snd} (\text{snd} (r01\text{-binary-expansion}'' r n))) \leq r \wedge$
 $r < (\text{fst} (\text{snd} (r01\text{-binary-expansion}'' r n)))$

$\langle \text{proof} \rangle$

$0 \leq lr \wedge lr < ur \wedge ur \leq 1.$

lemma $r01\text{-binary-expansion-lr-ur-nn}$:

shows $0 \leq \text{snd} (\text{snd} (r01\text{-binary-expansion}'' r n)) \wedge$

$\text{snd} (\text{snd} (r01\text{-binary-expansion}'' r n)) < \text{fst} (\text{snd} (r01\text{-binary-expansion}'' r$
 $n)) \wedge$

$\text{fst} (\text{snd} (r01\text{-binary-expansion}'' r n)) \leq 1$

$\langle \text{proof} \rangle$

lemma $r01\text{-binary-expansion-diff}$:

shows $(\text{fst} (\text{snd} (r01\text{-binary-expansion}'' r n))) - (\text{snd} (\text{snd} (r01\text{-binary-expansion}''$
 $r n))) = (1/2)^\wedge(\text{Suc } n)$

$\langle \text{proof} \rangle$

$lrn = Sn.$

lemma $r01\text{-binary-expression-eq-lr}$:

$\text{snd} (\text{snd} (r01\text{-binary-expansion}'' r n)) = r01\text{-binary-expression } r \ n$
 $\langle \text{proof} \rangle$

lemma $r01\text{-binary-expansion}'\text{-sum-range}$:

$\exists k::\text{nat}. (\text{snd} (\text{snd} (r01\text{-binary-expansion}'' r n))) = \text{real } k/2^\wedge(\text{Suc } n) \wedge$
 $k < 2^\wedge(\text{Suc } n) \wedge$
 $((r01\text{-binary-expansion}' r n) = 0 \longrightarrow \text{even } k) \wedge$
 $((r01\text{-binary-expansion}' r n) = 1 \longrightarrow \text{odd } k)$

$\langle \text{proof} \rangle$

$an = bn \leftrightarrow Sn = S'n.$

lemma $r01\text{-binary-expansion}'\text{-expression-eq}$:

$r01\text{-binary-expansion}' r1 = r01\text{-binary-expansion}' r2 \longleftrightarrow$
 $r01\text{-binary-expression } r1 = r01\text{-binary-expression } r2$

$\langle \text{proof} \rangle$

lemma power2-e :

$\bigwedge e::\text{real}. 0 < e \implies \exists n::\text{nat}. \text{real-of-rat } (1/2)^\wedge n < e$
 $\langle \text{proof} \rangle$

lemma $r01\text{-binary-expression-converges-to-r}$:

assumes $0 < r$

and $r < 1$

shows $\text{LIMSEQ } (r01\text{-binary-expression } r) \ r$

<proof>

lemma *r01-binary-expression-correct:*

assumes $0 < r$

and $r < 1$

shows $r = (\sum n. \text{real } (r01\text{-binary-expansion}' r n) * (1/2)^\wedge(Suc n))$

<proof>

$S0 \leq S1 \leq S2 \leq \dots$

lemma *binary-sum-incseq:*

incseq (*r01-binary-sum* *a*)

<proof>

lemma *r01-eq-iff:*

assumes $0 < r1$ $r1 < 1$

$0 < r2$ $r2 < 1$

shows $r1 = r2 \iff r01\text{-binary-expansion}' r1 = r01\text{-binary-expansion}' r2$

<proof>

lemma *power-half-summable:*

summable $(\lambda n. ((1::\text{real}) / 2)^\wedge Suc n)$

<proof>

lemma *binary-expression-summable:*

assumes $\bigwedge n. a n \in \{0,1 :: \text{nat}\}$

shows *summable* $(\lambda n. \text{real } (a n) * (1/2)^\wedge(Suc n))$

<proof>

lemma *binary-expression-gteq0:*

assumes $\bigwedge n. a n \in \{0,1 :: \text{nat}\}$

shows $0 \leq (\sum n. \text{real } (a (n + k)) * (1 / 2)^\wedge Suc (n + k))$

<proof>

lemma *binary-expression-leeq1:*

assumes $\bigwedge n. a n \in \{0,1 :: \text{nat}\}$

shows $(\sum n. \text{real } (a (n + k)) * (1 / 2)^\wedge Suc (n + k)) \leq 1$

<proof>

lemma *binary-expression-less-than:*

assumes $\bigwedge n. a n \in \{0,1 :: \text{nat}\}$

shows $(\sum n. \text{real } (a (n + k)) * (1 / 2)^\wedge Suc (n + k)) \leq (\sum n. (1 / 2)^\wedge Suc (n + k))$

<proof>

lemma *lim-sum-ai:*

assumes $\bigwedge n. a n \in \{0,1 :: \text{nat}\}$

shows $\text{lim } (\lambda n. (\sum i=0..n. \text{real } (a i) * (1/2)^\wedge(Suc i))) = (\sum n::\text{nat}. \text{real } (a n) * (1/2)^\wedge(Suc n))$

<proof>

lemma *half-1-minus-sum:*

$$1 - (\sum_{i < k}. ((1::real) / 2) ^ Suc i) = (1/2) ^ k$$

<proof>

lemma *half-sum:*

$$(\sum n. ((1::real) / 2) ^ (Suc (n + k))) = (1/2) ^ k$$

<proof>

lemma *ai-exists0-less-than-sum:*

assumes $\bigwedge n. a n \in \{0,1\}$

$$i \geq m$$

and $a i = 0$

shows $(\sum n::nat. real (a (n + m)) * (1/2) ^ (Suc (n + m))) < (1 / 2) ^ m$

<proof>

lemma *ai-exists0-less-than1:*

assumes $\bigwedge n. a n \in \{0,1\}$

and $\exists i. a i = 0$

shows $(\sum n::nat. real (a n) * (1/2) ^ (Suc n)) < 1$

<proof>

lemma *ai-1-gt:*

assumes $\bigwedge n. a n \in \{0,1\}$

and $a i = 1$

shows $(1/2) ^ (Suc i) \leq (\sum n::nat. real (a (n+i)) * (1/2) ^ (Suc (n+i)))$

<proof>

lemma *ai-exists1-gt0:*

assumes $\bigwedge n. a n \in \{0,1\}$

and $\exists i. a i = 1$

shows $0 < (\sum n::nat. real (a n) * (1/2) ^ (Suc n))$

<proof>

lemma *r01-binary-expression-ex0:*

assumes $0 < r < 1$

shows $\exists i. r01-binary-expansion' r i = 0$

<proof>

lemma *r01-binary-expression-ex1:*

assumes $0 < r < 1$

shows $\exists i. r01-binary-expansion' r i = 1$

<proof>

lemma *r01-binary-expansion'-gt1:*

$1 \leq r \iff (\forall n. r01-binary-expansion' r n = 1)$

<proof>

lemma *r01-binary-expansion'-lt0*:
 $r \leq 0 \iff (\forall n. r01\text{-binary-expansion}' r n = 0)$
 ⟨proof⟩

The sequence 111111... does not appear in $r = 0.a_1a_2\dots$

lemma *r01-binary-expression-ex0-strong*:
assumes $0 < r \ r < 1$
shows $\exists i \geq n. r01\text{-binary-expansion}' r i = 0$
 ⟨proof⟩

A binary expression is well-formed when 111... does not appear in the tail of the sequence

definition *biexp01-well-formed* :: $(nat \Rightarrow nat) \Rightarrow bool$ **where**
 $biexp01\text{-well-formed } a \equiv (\forall n. a n \in \{0,1\}) \wedge (\forall n. \exists m \geq n. a m = 0)$

lemma *biexp01-well-formedE*:
assumes *biexp01-well-formed* a
shows $(\forall n. a n \in \{0,1\}) \wedge (\forall n. \exists m \geq n. a m = 0)$
 ⟨proof⟩

lemma *biexp01-well-formedI*:
assumes $\bigwedge n. a n \in \{0,1\}$
and $\bigwedge n. \exists m \geq n. a m = 0$
shows *biexp01-well-formed* a
 ⟨proof⟩

lemma *r01-binary-expansion-well-formed*:
assumes $0 < r \ r < 1$
shows *biexp01-well-formed* $(r01\text{-binary-expansion}' r)$
 ⟨proof⟩

lemma *biexp01-well-formed-comb*:
assumes *biexp01-well-formed* a
and *biexp01-well-formed* b
shows *biexp01-well-formed* $(\lambda n. \text{if even } n \text{ then } a (n \text{ div } 2) \text{ else } b ((n-1) \text{ div } 2))$
 ⟨proof⟩

lemma *nat-complete-induction*:
assumes $P (0 :: nat)$
and $\bigwedge m. m \leq n \implies P m \implies P (Suc n)$
shows $P n$
 ⟨proof⟩

$(\sum m. \text{real } (a m) * (1 / 2) ^ Suc m) n = a n.$

lemma *biexp01-well-formed-an*:

assumes *biexp01-well-formed a*
shows *r01-binary-expansion' ($\sum m. \text{real } (a \ m) * (1 / 2) ^ \wedge \text{Suc } m) \ n = a \ n$*
<proof>

lemma *f01-borel-measurable:*
assumes *f - ' {0::real} ∈ sets real-borel*
f - ' {1} ∈ sets borel
and *∧r::real. f r ∈ {0,1}*
shows *f ∈ borel-measurable real-borel*
<proof>

lemma *r01-binary-expansion'-measurable:*
(λr. real (r01-binary-expansion' r n)) ∈ borel-measurable (borel :: real measure)
<proof>

definition *r01-to-r01-r01-fst' :: real ⇒ nat ⇒ nat where*
*r01-to-r01-r01-fst' r n ≡ r01-binary-expansion' r (2*n)*

lemma *r01-to-r01-r01-fst'in01:*
∧n. r01-to-r01-r01-fst' r n ∈ {0,1}
<proof>

definition *r01-to-r01-r01-fst-sum :: real ⇒ nat ⇒ real where*
r01-to-r01-r01-fst-sum ≡ r01-binary-sum ◦ r01-to-r01-r01-fst'

definition *r01-to-r01-r01-fst :: real ⇒ real where*
r01-to-r01-r01-fst = lim ◦ r01-to-r01-r01-fst-sum

lemma *r01-to-r01-r01-fst-def':*
*r01-to-r01-r01-fst r = ($\sum n. \text{real } (r01-binary-expansion' r (2*n)) * (1/2) ^ \wedge (n+1)$)*
<proof>

lemma *r01-to-r01-r01-fst-measurable:*
r01-to-r01-r01-fst ∈ borel-measurable borel
<proof>

definition *r01-to-r01-r01-snd' :: real ⇒ nat ⇒ nat where*
*r01-to-r01-r01-snd' r n = r01-binary-expansion' r (2*n + 1)*

lemma *r01-to-r01-r01-snd'in01:*
∧n. r01-to-r01-r01-snd' r n ∈ {0,1}
<proof>

definition $r01\text{-to-}r01\text{-}r01\text{-snd-sum} :: \text{real} \Rightarrow \text{nat} \Rightarrow \text{real}$ **where**
 $r01\text{-to-}r01\text{-}r01\text{-snd-sum} \equiv r01\text{-binary-sum} \circ r01\text{-to-}r01\text{-}r01\text{-snd}'$

definition $r01\text{-to-}r01\text{-}r01\text{-snd} :: \text{real} \Rightarrow \text{real}$ **where**
 $r01\text{-to-}r01\text{-}r01\text{-snd} = \text{lim} \circ r01\text{-to-}r01\text{-}r01\text{-snd-sum}$

lemma $r01\text{-to-}r01\text{-}r01\text{-snd-def}'$:
 $r01\text{-to-}r01\text{-}r01\text{-snd} \ r = (\sum n. \text{real} (r01\text{-binary-expansion}' \ r \ (2*n + 1)) * (1/2) \ \sim (n+1))$
 $\langle \text{proof} \rangle$

lemma $r01\text{-to-}r01\text{-}r01\text{-snd-measurable}$:
 $r01\text{-to-}r01\text{-}r01\text{-snd} \in \text{borel-measurable borel}$
 $\langle \text{proof} \rangle$

definition $r01\text{-to-}r01\text{-}r01 :: \text{real} \Rightarrow \text{real} \times \text{real}$ **where**
 $r01\text{-to-}r01\text{-}r01 \ r = (r01\text{-to-}r01\text{-}r01\text{-fst} \ r, r01\text{-to-}r01\text{-}r01\text{-snd} \ r)$

lemma $r01\text{-to-}r01\text{-}r01\text{-image}$:
 $r01\text{-to-}r01\text{-}r01 \ r \in \{0..1\} \times \{0..1\}$
 $\langle \text{proof} \rangle$

lemma $r01\text{-to-}r01\text{-}r01\text{-measurable}$:
 $r01\text{-to-}r01\text{-}r01 \in \text{real-borel} \rightarrow_M \text{real-borel} \otimes_M \text{real-borel}$
 $\langle \text{proof} \rangle$

lemma $r01\text{-to-}r01\text{-}r01\text{-3over4}$:
 $r01\text{-to-}r01\text{-}r01 \ (3/4) = (1/2, 1/2)$
 $\langle \text{proof} \rangle$

definition $r01\text{-}r01\text{-to-}r01' :: \text{real} \times \text{real} \Rightarrow \text{nat} \Rightarrow \text{nat}$ **where**
 $r01\text{-}r01\text{-to-}r01' \ rs \equiv (\lambda n. \text{if even } n \text{ then } r01\text{-binary-expansion}' \ (\text{fst } rs) \ (n \text{ div } 2)$
 $\text{else } r01\text{-binary-expansion}' \ (\text{snd } rs) \ ((n-1) \text{ div } 2))$

lemma $r01\text{-}r01\text{-to-}r01'\text{in}01$:
 $\bigwedge n. r01\text{-}r01\text{-to-}r01' \ rs \ n \in \{0,1\}$
 $\langle \text{proof} \rangle$

lemma $r01\text{-}r01\text{-to-}r01'\text{-well-formed}$:
assumes $0 < r1 \ r1 < 1$
and $0 < r2 \ r2 < 1$
shows $\text{biexp}01\text{-well-formed} \ (r01\text{-}r01\text{-to-}r01' \ (r1, r2))$
 $\langle \text{proof} \rangle$

definition $r01\text{-}r01\text{-to-}r01\text{-sum} :: \text{real} \times \text{real} \Rightarrow \text{nat} \Rightarrow \text{real}$ **where**
 $r01\text{-}r01\text{-to-}r01\text{-sum} \equiv r01\text{-binary-sum} \circ r01\text{-}r01\text{-to-}r01'$

definition $r01-r01-to-r01 :: real \times real \Rightarrow real$ **where**

$r01-r01-to-r01 \equiv lim \circ r01-r01-to-r01-sum$

lemma $r01-r01-to-r01-def'$:

$r01-r01-to-r01 (r1,r2) = (\sum n. real (r01-r01-to-r01' (r1,r2) n) * (1/2) \wedge (n+1))$
 $\langle proof \rangle$

lemma $r01-r01-to-r01-measurable$:

$r01-r01-to-r01 \in real-borel \otimes_M real-borel \rightarrow_M real-borel$
 $\langle proof \rangle$

lemma $r01-r01-to-r01-image$:

assumes $0 < r1$ $r1 < 1$

shows $r01-r01-to-r01 (r1,r2) \in \{0 < .. < 1\}$

$\langle proof \rangle$

lemma $r01-r01-to-r01-image'$:

assumes $0 < r2$ $r2 < 1$

shows $r01-r01-to-r01 (r1,r2) \in \{0 < .. < 1\}$

$\langle proof \rangle$

lemma $r01-r01-to-r01-binary-nth$:

assumes $0 < r1$ $r1 < 1$

and $0 < r2$ $r2 < 1$

shows $r01-binary-expansion' r1 n = r01-binary-expansion' (r01-r01-to-r01 (r1,r2)) (2*n) \wedge$

$r01-binary-expansion' r2 n = r01-binary-expansion' (r01-r01-to-r01 (r1,r2)) (2*n + 1)$

$\langle proof \rangle$

lemma $r01-r01--r01--r01-r01-id$:

assumes $0 < r1$ $r1 < 1$

$0 < r2$ $r2 < 1$

shows $(r01-to-r01-r01 \circ r01-r01-to-r01) (r1,r2) = (r1,r2)$

$\langle proof \rangle$

We first show that $M \otimes_M N$ is a standard Borel space for standard Borel spaces M and N .

lemma $pair-measurable[measurable]$:

assumes $f \in X \rightarrow_M Y$

and $g \in X' \rightarrow_M Y'$

shows $map-prod f g \in X \otimes_M X' \rightarrow_M Y \otimes_M Y'$

$\langle proof \rangle$

lemma $pair-standard-borel-standard$:

assumes $standard-borel M$

and $standard-borel N$

shows *standard-borel* ($M \otimes_M N$)
 ⟨*proof*⟩

lemma *pair-standard-borel-space-UNIV*:
assumes *standard-borel-space-UNIV* M
and *standard-borel-space-UNIV* N
shows *standard-borel-space-UNIV* ($M \otimes_M N$)
 ⟨*proof*⟩

locale *pair-standard-borel* = $s1$: *standard-borel* M + $s2$: *standard-borel* N
for $M :: 'a$ *measure* **and** $N :: 'b$ *measure*
begin

sublocale *standard-borel* $M \otimes_M N$
 ⟨*proof*⟩

end

locale *pair-standard-borel-space-UNIV* = $s1$: *standard-borel-space-UNIV* M + $s2$:
standard-borel-space-UNIV N
for $M :: 'a$ *measure* **and** $N :: 'b$ *measure*
begin

sublocale *pair-standard-borel* $M N$
 ⟨*proof*⟩

sublocale *standard-borel-space-UNIV* $M \otimes_M N$
 ⟨*proof*⟩

end

$\mathbb{R} \times \mathbb{R}$ is a standard Borel space.

interpretation *real-real* : *pair-standard-borel-space-UNIV* *real-borel* *real-borel*
 ⟨*proof*⟩

1.4 $\mathbb{N} \times \mathbb{R}$

$\mathbb{N} \times \mathbb{R}$ is a standard Borel space.

interpretation *nat-real*: *pair-standard-borel-space-UNIV* *nat-borel* *real-borel*
 ⟨*proof*⟩

end

2 Quasi-Borel Spaces

theory *QuasiBorel*
imports *StandardBorel*

begin

2.1 Definitions

We formalize quasi-Borel spaces introduced by Heunen et al. [1].

2.1.1 Quasi-Borel Spaces

definition *qbs-closed1* :: (real \Rightarrow 'a) set \Rightarrow bool

where *qbs-closed1* *Mx* \equiv ($\forall a \in Mx. \forall f \in \text{real-borel} \rightarrow_M \text{real-borel}. a \circ f \in Mx$)

definition *qbs-closed2* :: ['a set, (real \Rightarrow 'a) set] \Rightarrow bool

where *qbs-closed2* *X Mx* \equiv ($\forall x \in X. (\lambda r. x) \in Mx$)

definition *qbs-closed3* :: (real \Rightarrow 'a) set \Rightarrow bool

where *qbs-closed3* *Mx* \equiv ($\forall P::\text{real} \Rightarrow \text{nat}. \forall Fi::\text{nat} \Rightarrow \text{real} \Rightarrow 'a.$
 $(\forall i. P - \{i\} \in \text{sets real-borel})$
 $\rightarrow (\forall i. Fi i \in Mx)$
 $\rightarrow (\lambda r. Fi (P r) r) \in Mx$)

lemma *separate-measurable*:

fixes *P* :: real \Rightarrow nat

assumes $\bigwedge i. P - \{i\} \in \text{sets real-borel}$

shows $P \in \text{real-borel} \rightarrow_M \text{nat-borel}$

<proof>

lemma *measurable-separate*:

fixes *P* :: real \Rightarrow nat

assumes $P \in \text{real-borel} \rightarrow_M \text{nat-borel}$

shows $P - \{i\} \in \text{sets real-borel}$

<proof>

definition *is-quasi-borel* *X Mx* $\longleftrightarrow Mx \subseteq UNIV \rightarrow X \wedge \text{qbs-closed1 } Mx \wedge$
qbs-closed2 *X Mx* $\wedge \text{qbs-closed3 } Mx$

lemma *is-quasi-borel-intro[simp]*:

assumes $Mx \subseteq UNIV \rightarrow X$

and *qbs-closed1* *Mx* *qbs-closed2* *X Mx* *qbs-closed3* *Mx*

shows *is-quasi-borel* *X Mx*

<proof>

typedef 'a *quasi-borel* = {(*X*::'a set, *Mx*). *is-quasi-borel* *X Mx*}

<proof>

definition *qbs-space* :: 'a *quasi-borel* \Rightarrow 'a set **where**

qbs-space *X* $\equiv \text{fst } (\text{Rep-quasi-borel } X)$

definition *qbs-Mx* :: 'a *quasi-borel* \Rightarrow (real \Rightarrow 'a) set **where**

qbs-Mx *X* $\equiv \text{snd } (\text{Rep-quasi-borel } X)$

lemma *qbs-decomp* :
 $(qbs\text{-space } X, qbs\text{-Mx } X) \in \{(X :: 'a \text{ set}, Mx). \text{is-quasi-borel } X \text{ Mx}\}$
 ⟨proof⟩

lemma *qbs-Mx-to-X[dest]*:
assumes $\alpha \in qbs\text{-Mx } X$
shows $\alpha \in UNIV \rightarrow qbs\text{-space } X$
 $\alpha \ r \in qbs\text{-space } X$
 ⟨proof⟩

lemma *qbs-closed1I*:
assumes $\bigwedge \alpha \ f. \alpha \in Mx \implies f \in \text{real-borel} \rightarrow_M \text{real-borel} \implies \alpha \circ f \in Mx$
shows *qbs-closed1* Mx
 ⟨proof⟩

lemma *qbs-closed1-dest[simp]*:
assumes $\alpha \in qbs\text{-Mx } X$
and $f \in \text{real-borel} \rightarrow_M \text{real-borel}$
shows $\alpha \circ f \in qbs\text{-Mx } X$
 ⟨proof⟩

lemma *qbs-closed2I*:
assumes $\bigwedge x. x \in X \implies (\lambda r. x) \in Mx$
shows *qbs-closed2* $X \text{ Mx}$
 ⟨proof⟩

lemma *qbs-closed2-dest[simp]*:
assumes $x \in qbs\text{-space } X$
shows $(\lambda r. x) \in qbs\text{-Mx } X$
 ⟨proof⟩

lemma *qbs-closed3I*:
assumes $\bigwedge (P :: \text{real} \Rightarrow \text{nat}) \ Fi. (\bigwedge i. P \text{ - ' } \{i\} \in \text{sets real-borel}) \implies (\bigwedge i. Fi \ i \in Mx)$
 $\implies (\lambda r. Fi \ (P \ r) \ r) \in Mx$
shows *qbs-closed3* Mx
 ⟨proof⟩

lemma *qbs-closed3I'*:
assumes $\bigwedge (P :: \text{real} \Rightarrow \text{nat}) \ Fi. P \in \text{real-borel} \rightarrow_M \text{nat-borel} \implies (\bigwedge i. Fi \ i \in Mx)$
 $\implies (\lambda r. Fi \ (P \ r) \ r) \in Mx$
shows *qbs-closed3* Mx
 ⟨proof⟩

lemma *qbs-closed3-dest[simp]*:
fixes $P :: \text{real} \Rightarrow \text{nat}$ **and** $Fi :: \text{nat} \Rightarrow \text{real} \Rightarrow -$

assumes $\bigwedge i. P - \{i\} \in \text{sets real-borel}$
and $\bigwedge i. Fi\ i \in \text{qbs-Mx } X$
shows $(\lambda r. Fi\ (P\ r)\ r) \in \text{qbs-Mx } X$
 $\langle \text{proof} \rangle$

lemma *qbs-closed3-dest'*:
fixes $P :: \text{real} \Rightarrow \text{nat}$ **and** $Fi :: \text{nat} \Rightarrow \text{real} \Rightarrow -$
assumes $P \in \text{real-borel} \rightarrow_M \text{nat-borel}$
and $\bigwedge i. Fi\ i \in \text{qbs-Mx } X$
shows $(\lambda r. Fi\ (P\ r)\ r) \in \text{qbs-Mx } X$
 $\langle \text{proof} \rangle$

lemma *qbs-closed3-dest2*:
assumes *countable* I
and [*measurable*]: $P \in \text{real-borel} \rightarrow_M \text{count-space } I$
and $\bigwedge i. i \in I \implies Fi\ i \in \text{qbs-Mx } X$
shows $(\lambda r. Fi\ (P\ r)\ r) \in \text{qbs-Mx } X$
 $\langle \text{proof} \rangle$

lemma *qbs-closed3-dest2'*:
assumes *countable* I
and [*measurable*]: $P \in \text{real-borel} \rightarrow_M \text{count-space } I$
and $\bigwedge i. i \in \text{range } P \implies Fi\ i \in \text{qbs-Mx } X$
shows $(\lambda r. Fi\ (P\ r)\ r) \in \text{qbs-Mx } X$
 $\langle \text{proof} \rangle$

lemma *qbs-space-Mx*:
 $\text{qbs-space } X = \{\alpha\ x \mid x\ \alpha. \alpha \in \text{qbs-Mx } X\}$
 $\langle \text{proof} \rangle$

lemma *qbs-space-eq-Mx*:
assumes $\text{qbs-Mx } X = \text{qbs-Mx } Y$
shows $\text{qbs-space } X = \text{qbs-space } Y$
 $\langle \text{proof} \rangle$

lemma *qbs-eqI*:
assumes $\text{qbs-Mx } X = \text{qbs-Mx } Y$
shows $X = Y$
 $\langle \text{proof} \rangle$

2.1.2 Morphism of Quasi-Borel Spaces

definition *qbs-morphism* :: [*'a quasi-borel, 'b quasi-borel*] \Rightarrow (*'a* \Rightarrow *'b*) *set* (**infix** \rightarrow_Q 60) **where**
 $X \rightarrow_Q Y \equiv \{f \in \text{qbs-space } X \rightarrow \text{qbs-space } Y. \forall \alpha \in \text{qbs-Mx } X. f \circ \alpha \in \text{qbs-Mx } Y\}$

lemma *qbs-morphismI*:
assumes $\bigwedge \alpha. \alpha \in \text{qbs-Mx } X \implies f \circ \alpha \in \text{qbs-Mx } Y$

shows $f \in X \rightarrow_Q Y$
<proof>

lemma *qbs-morphismE[dest]*:
assumes $f \in X \rightarrow_Q Y$
shows $f \in \text{qbs-space } X \rightarrow \text{qbs-space } Y$
 $\bigwedge x. x \in \text{qbs-space } X \implies f x \in \text{qbs-space } Y$
 $\bigwedge \alpha. \alpha \in \text{qbs-Mx } X \implies f \circ \alpha \in \text{qbs-Mx } Y$
<proof>

lemma *qbs-morphism-ident[simp]*:
 $id \in X \rightarrow_Q X$
<proof>

lemma *qbs-morphism-ident'[simp]*:
 $(\lambda x. x) \in X \rightarrow_Q X$
<proof>

lemma *qbs-morphism-comp*:
assumes $f \in X \rightarrow_Q Y$ $g \in Y \rightarrow_Q Z$
shows $g \circ f \in X \rightarrow_Q Z$
<proof>

lemma *qbs-morphism-cong*:
assumes $\bigwedge x. x \in \text{qbs-space } X \implies f x = g x$
and $f \in X \rightarrow_Q Y$
shows $g \in X \rightarrow_Q Y$
<proof>

lemma *qbs-morphism-const*:
assumes $y \in \text{qbs-space } Y$
shows $(\lambda \cdot. y) \in X \rightarrow_Q Y$
<proof>

2.1.3 Empty Space

definition *empty-quasi-borel* :: 'a quasi-borel **where**
 $\text{empty-quasi-borel} \equiv \text{Abs-quasi-borel } (\{\}, \{\})$

lemma *eqb-correct*: $\text{Rep-quasi-borel empty-quasi-borel} = (\{\}, \{\})$
<proof>

lemma *eqb-space[simp]*: $\text{qbs-space empty-quasi-borel} = \{\}$
<proof>

lemma *eqb-Mx[simp]*: $\text{qbs-Mx empty-quasi-borel} = \{\}$
<proof>

lemma *qbs-empty-equiv*: $qbs\text{-space } X = \{\} \longleftrightarrow qbs\text{-Mx } X = \{\}$
 ⟨proof⟩

lemma *empty-quasi-borel-iff*:
 $qbs\text{-space } X = \{\} \longleftrightarrow X = \text{empty-quasi-borel}$
 ⟨proof⟩

2.1.4 Unit Space

definition *unit-quasi-borel* :: *unit quasi-borel* ($!Q$) **where**
unit-quasi-borel \equiv *Abs-quasi-borel* ($UNIV, UNIV$)

lemma *uqb-correct*: *Rep-quasi-borel unit-quasi-borel* = ($UNIV, UNIV$)
 ⟨proof⟩

lemma *uqb-space[simp]*: *qbs-space unit-quasi-borel* = $\{\}$
 ⟨proof⟩

lemma *uqb-Mx[simp]*: *qbs-Mx unit-quasi-borel* = $\{\lambda r. ()\}$
 ⟨proof⟩

lemma *unit-quasi-borel-terminal*:
 $\exists! f. f \in X \rightarrow_Q \text{unit-quasi-borel}$
 ⟨proof⟩

definition *to-unit-quasi-borel* :: '*a* \Rightarrow *unit* ($!Q$) **where**
to-unit-quasi-borel \equiv ($\lambda \cdot. ()$)

lemma *to-unit-quasi-borel-morphism* :
 $!Q \in X \rightarrow_Q \text{unit-quasi-borel}$
 ⟨proof⟩

2.1.5 Subspaces

definition *sub-qbs* :: [*'a quasi-borel, 'a set*] \Rightarrow '*a quasi-borel* **where**
sub-qbs $X U \equiv \text{Abs-quasi-borel } (qbs\text{-space } X \cap U, \{f \in UNIV \rightarrow qbs\text{-space } X \cap U. f \in qbs\text{-Mx } X\})$

lemma *sub-qbs-closed*:
 $qbs\text{-closed1 } \{f \in UNIV \rightarrow qbs\text{-space } X \cap U. f \in qbs\text{-Mx } X\}$
 $qbs\text{-closed2 } (qbs\text{-space } X \cap U) \{f \in UNIV \rightarrow qbs\text{-space } X \cap U. f \in qbs\text{-Mx } X\}$
 $qbs\text{-closed3 } \{f \in UNIV \rightarrow qbs\text{-space } X \cap U. f \in qbs\text{-Mx } X\}$
 ⟨proof⟩

lemma *sub-qbs-correct[simp]*: *Rep-quasi-borel (sub-qbs X U)* = ($qbs\text{-space } X \cap U, \{f \in UNIV \rightarrow qbs\text{-space } X \cap U. f \in qbs\text{-Mx } X\}$)
 ⟨proof⟩

lemma *sub-qbs-space[simp]*: *qbs-space (sub-qbs X U)* = $qbs\text{-space } X \cap U$
 ⟨proof⟩

lemma *sub-qbs-Mx[simp]*: $qbs\text{-}Mx (sub\text{-}qbs\ X\ U) = \{f \in UNIV \rightarrow qbs\text{-}space\ X \cap U. f \in qbs\text{-}Mx\ X\}$
 ⟨proof⟩

lemma *sub-qbs*:

assumes $U \subseteq qbs\text{-}space\ X$

shows $(qbs\text{-}space (sub\text{-}qbs\ X\ U), qbs\text{-}Mx (sub\text{-}qbs\ X\ U)) = (U, \{f \in UNIV \rightarrow U. f \in qbs\text{-}Mx\ X\})$

⟨proof⟩

2.1.6 Image Spaces

definition *map-qbs* :: $['a \Rightarrow 'b] \Rightarrow 'a\ quasi\text{-}borel \Rightarrow 'b\ quasi\text{-}borel$ **where**
 $map\text{-}qbs\ f\ X = Abs\text{-}quasi\text{-}borel (f\ ' (qbs\text{-}space\ X), \{\beta. \exists \alpha \in qbs\text{-}Mx\ X. \beta = f \circ \alpha\})$

lemma *map-qbs-f*:

$\{\beta. \exists \alpha \in qbs\text{-}Mx\ X. \beta = f \circ \alpha\} \subseteq UNIV \rightarrow f\ ' (qbs\text{-}space\ X)$
 ⟨proof⟩

lemma *map-qbs-closed1*:

$qbs\text{-}closed1\ \{\beta. \exists \alpha \in qbs\text{-}Mx\ X. \beta = f \circ \alpha\}$
 ⟨proof⟩

lemma *map-qbs-closed2*:

$qbs\text{-}closed2\ (f\ ' (qbs\text{-}space\ X))\ \{\beta. \exists \alpha \in qbs\text{-}Mx\ X. \beta = f \circ \alpha\}$
 ⟨proof⟩

lemma *map-qbs-closed3*:

$qbs\text{-}closed3\ \{\beta. \exists \alpha \in qbs\text{-}Mx\ X. \beta = f \circ \alpha\}$
 ⟨proof⟩

lemma *map-qbs-correct[simp]*:

$Rep\text{-}quasi\text{-}borel (map\text{-}qbs\ f\ X) = (f\ ' (qbs\text{-}space\ X), \{\beta. \exists \alpha \in qbs\text{-}Mx\ X. \beta = f \circ \alpha\})$
 ⟨proof⟩

lemma *map-qbs-space[simp]*:

$qbs\text{-}space (map\text{-}qbs\ f\ X) = f\ ' (qbs\text{-}space\ X)$
 ⟨proof⟩

lemma *map-qbs-Mx[simp]*:

$qbs\text{-}Mx (map\text{-}qbs\ f\ X) = \{\beta. \exists \alpha \in qbs\text{-}Mx\ X. \beta = f \circ \alpha\}$
 ⟨proof⟩

inductive-set *generating-Mx* :: $'a\ set \Rightarrow (real \Rightarrow 'a)\ set \Rightarrow (real \Rightarrow 'a)\ set$
for $X :: 'a\ set$ **and** $Mx :: (real \Rightarrow 'a)\ set$
where

Basic: $\alpha \in Mx \implies \alpha \in \text{generating-Mx } X \ Mx$
 | *Const*: $x \in X \implies (\lambda r. x) \in \text{generating-Mx } X \ Mx$
 | *Comp*: $f \in \text{real-borel} \rightarrow_M \text{real-borel} \implies \alpha \in \text{generating-Mx } X \ Mx \implies \alpha \circ f \in \text{generating-Mx } X \ Mx$
 | *Part*: $(\bigwedge i. Fi \ i \in \text{generating-Mx } X \ Mx) \implies P \in \text{real-borel} \rightarrow_M \text{nat-borel} \implies (\lambda r. Fi \ (P \ r) \ r) \in \text{generating-Mx } X \ Mx$

lemma *generating-Mx-to-space*:
assumes $Mx \subseteq UNIV \rightarrow X$
shows $\text{generating-Mx } X \ Mx \subseteq UNIV \rightarrow X$
<proof>

lemma *generating-Mx-closed1*:
qbs-closed1 ($\text{generating-Mx } X \ Mx$)
<proof>

lemma *generating-Mx-closed2*:
qbs-closed2 X ($\text{generating-Mx } X \ Mx$)
<proof>

lemma *generating-Mx-closed3*:
qbs-closed3 ($\text{generating-Mx } X \ Mx$)
<proof>

lemma *generating-Mx-Mx*:
 $\text{generating-Mx} \ (\text{qbs-space } X) \ (\text{qbs-Mx } X) = \text{qbs-Mx } X$
<proof>

2.1.7 Ordering of Quasi-Borel Spaces

instantiation *quasi-borel* :: (*type*) *order-bot*
begin

inductive *less-eq-quasi-borel* :: '*a* *quasi-borel* \Rightarrow '*a* *quasi-borel* \Rightarrow *bool* **where**
 $\text{qbs-space } X \subset \text{qbs-space } Y \implies \text{less-eq-quasi-borel } X \ Y$
 | $\text{qbs-space } X = \text{qbs-space } Y \implies \text{qbs-Mx } Y \subseteq \text{qbs-Mx } X \implies \text{less-eq-quasi-borel } X \ Y$

lemma *le-quasi-borel-iff*:
 $X \leq Y \iff (\text{if } \text{qbs-space } X = \text{qbs-space } Y \text{ then } \text{qbs-Mx } Y \subseteq \text{qbs-Mx } X \text{ else } \text{qbs-space } X \subset \text{qbs-space } Y)$
<proof>

definition *less-quasi-borel* :: '*a* *quasi-borel* \Rightarrow '*a* *quasi-borel* \Rightarrow *bool* **where**
 $\text{less-quasi-borel } X \ Y \iff (X \leq Y \wedge \neg Y \leq X)$

definition *bot-quasi-borel* :: '*a* *quasi-borel* **where**
 $\text{bot-quasi-borel} = \text{empty-quasi-borel}$

instance

<proof>

end

definition *inf-quasi-borel* :: [*'a quasi-borel, 'a quasi-borel*] \Rightarrow *'a quasi-borel* **where**
inf-quasi-borel *X X'* = *Abs-quasi-borel* (*qbs-space* *X* \cap *qbs-space* *X'*, *qbs-Mx* *X* \cap
qbs-Mx *X'*)

lemma *inf-quasi-borel-correct*: *Rep-quasi-borel* (*inf-quasi-borel* *X X'*) = (*qbs-space*
X \cap *qbs-space* *X'*, *qbs-Mx* *X* \cap *qbs-Mx* *X'*)
<proof>

lemma *inf-qbs-space[simp]*: *qbs-space* (*inf-quasi-borel* *X X'*) = *qbs-space* *X* \cap *qbs-space*
X'
<proof>

lemma *inf-qbs-Mx[simp]*: *qbs-Mx* (*inf-quasi-borel* *X X'*) = *qbs-Mx* *X* \cap *qbs-Mx* *X'*
<proof>

definition *max-quasi-borel* :: *'a set* \Rightarrow *'a quasi-borel* **where**
max-quasi-borel *X* = *Abs-quasi-borel* (*X*, *UNIV* \rightarrow *X*)

lemma *max-quasi-borel-correct*: *Rep-quasi-borel* (*max-quasi-borel* *X*) = (*X*, *UNIV*
 \rightarrow *X*)
<proof>

lemma *max-qbs-space[simp]*: *qbs-space* (*max-quasi-borel* *X*) = *X*
<proof>

lemma *max-qbs-Mx[simp]*: *qbs-Mx* (*max-quasi-borel* *X*) = *UNIV* \rightarrow *X*
<proof>

instantiation *quasi-borel* :: (*type*) *semilattice-sup*
begin

definition *sup-quasi-borel* :: *'a quasi-borel* \Rightarrow *'a quasi-borel* \Rightarrow *'a quasi-borel* **where**
sup-quasi-borel *X Y* \equiv (*if* *qbs-space* *X* = *qbs-space* *Y* *then* *inf-quasi-borel* *X Y*
else if *qbs-space* *X* \subset *qbs-space* *Y* *then* *Y*
else if *qbs-space* *Y* \subset *qbs-space* *X* *then* *X*
else *max-quasi-borel* (*qbs-space* *X* \cup *qbs-space* *Y*))

instance

<proof>

end

end

2.2 Relation to Measurable Spaces

```
theory Measure-QuasiBorel-Adjunction
  imports QuasiBorel
begin
```

We construct the adjunction between **Meas** and **QBS**, where **Meas** is the category of measurable spaces and measurable functions and **QBS** is the category of quasi-Borel spaces and morphisms.

2.2.1 The Functor R

```
definition measure-to-qbs :: 'a measure  $\Rightarrow$  'a quasi-borel where
  measure-to-qbs  $X \equiv$  Abs-quasi-borel (space  $X$ , real-borel  $\rightarrow_M X$ )
```

```
lemma R-Mx-correct: real-borel  $\rightarrow_M X \subseteq UNIV \rightarrow$  space  $X$ 
  <proof>
```

```
lemma R-qbs-closed1: qbs-closed1 (real-borel  $\rightarrow_M X$ )
  <proof>
```

```
lemma R-qbs-closed2: qbs-closed2 (space  $X$ ) (real-borel  $\rightarrow_M X$ )
  <proof>
```

```
lemma R-qbs-closed3: qbs-closed3 (real-borel  $\rightarrow_M (X :: 'a$  measure))
  <proof>
```

```
lemma R-correct[simp]: Rep-quasi-borel (measure-to-qbs  $X$ ) = (space  $X$ , real-borel
 $\rightarrow_M X$ )
  <proof>
```

```
lemma qbs-space-R[simp]: qbs-space (measure-to-qbs  $X$ ) = space  $X$ 
  <proof>
```

```
lemma qbs-Mx-R[simp]: qbs-Mx (measure-to-qbs  $X$ ) = real-borel  $\rightarrow_M X$ 
  <proof>
```

The following lemma says that *measure-to-qbs* is a functor from **Meas** to **QBS**.

```
lemma r-preserves-morphisms:
   $X \rightarrow_M Y \subseteq$  (measure-to-qbs  $X$ )  $\rightarrow_Q$  (measure-to-qbs  $Y$ )
  <proof>
```

2.2.2 The Functor L

```
definition sigma-Mx :: 'a quasi-borel  $\Rightarrow$  'a set set where
  sigma-Mx  $X \equiv$  {  $U \cap$  qbs-space  $X \mid U. \forall \alpha \in$  qbs-Mx  $X. \alpha - ' U \in$  sets real-borel }
```

```
definition qbs-to-measure :: 'a quasi-borel  $\Rightarrow$  'a measure where
```

qbs-to-measure $X \equiv \text{Abs-measure } (qbs\text{-space } X, \text{sigma-Mx } X, \lambda A. (\text{if } A = \{\} \text{ then } 0 \text{ else if } A \in - \text{sigma-Mx } X \text{ then } 0 \text{ else } \infty))$

lemma *measure-space-L*: *measure-space* (*qbs-space* X) (*sigma-Mx* X) ($\lambda A. (\text{if } A = \{\} \text{ then } 0 \text{ else if } A \in - \text{sigma-Mx } X \text{ then } 0 \text{ else } \infty)$)
 ⟨*proof*⟩

lemma *L-correct[simp]*: *Rep-measure* (*qbs-to-measure* X) = (*qbs-space* X , *sigma-Mx* X , $\lambda A. (\text{if } A = \{\} \text{ then } 0 \text{ else if } A \in - \text{sigma-Mx } X \text{ then } 0 \text{ else } \infty)$)
 ⟨*proof*⟩

lemma *space-L[simp]*: *space* (*qbs-to-measure* X) = *qbs-space* X
 ⟨*proof*⟩

lemma *sets-L[simp]*: *sets* (*qbs-to-measure* X) = *sigma-Mx* X
 ⟨*proof*⟩

lemma *emeasure-L[simp]*: *emeasure* (*qbs-to-measure* X) = ($\lambda A. \text{if } A = \{\} \vee A \notin \text{sigma-Mx } X \text{ then } 0 \text{ else } \infty$)
 ⟨*proof*⟩

lemma *qbs-Mx-sigma-Mx-contr*:
assumes *qbs-space* $X = \text{qbs-space } Y$
and *qbs-Mx* $X \subseteq \text{qbs-Mx } Y$
shows *sigma-Mx* $Y \subseteq \text{sigma-Mx } X$
 ⟨*proof*⟩

The following lemma says that *qbs-to-measure* is a functor from **QBS** to **Meas**.

lemma *l-preserved-morphisms*:
 $X \rightarrow_Q Y \subseteq (\text{qbs-to-measure } X) \rightarrow_M (\text{qbs-to-measure } Y)$
 ⟨*proof*⟩

abbreviation *qbs-borel* $\equiv \text{measure-to-qbs borel}$

declare [[*coercion measure-to-qbs*]]

abbreviation *real-quasi-borel* $:: \text{real quasi-borel } (\mathbb{R}_Q)$ **where**
real-quasi-borel $\equiv \text{qbs-borel}$

abbreviation *nat-quasi-borel* $:: \text{nat quasi-borel } (\mathbb{N}_Q)$ **where**
nat-quasi-borel $\equiv \text{qbs-borel}$

abbreviation *ennreal-quasi-borel* $:: \text{ennreal quasi-borel } (\mathbb{R}_{Q \geq 0})$ **where**
ennreal-quasi-borel $\equiv \text{qbs-borel}$

abbreviation *bool-quasi-borel* $:: \text{bool quasi-borel } (\mathbb{B}_Q)$ **where**
bool-quasi-borel $\equiv \text{qbs-borel}$

lemma *qbs-Mx-is-morphisms*:
 $qbs-Mx X = real-quasi-borel \rightarrow_Q X$
 ⟨proof⟩

lemma *qbs-Mx-subset-of-measurable*:
 $qbs-Mx X \subseteq real-borel \rightarrow_M qbs-to-measure X$
 ⟨proof⟩

lemma *L-max-of-measurables*:
assumes *space* $M = qbs-space X$
and $qbs-Mx X \subseteq real-borel \rightarrow_M M$
shows $sets M \subseteq sets (qbs-to-measure X)$
 ⟨proof⟩

lemma *qbs-Mx-are-measurable[simp,measurable]*:
assumes $\alpha \in qbs-Mx X$
shows $\alpha \in real-borel \rightarrow_M qbs-to-measure X$
 ⟨proof⟩

lemma *measure-to-qbs-cong-sets*:
assumes $sets M = sets N$
shows $measure-to-qbs M = measure-to-qbs N$
 ⟨proof⟩

lemma *lr-sets[simp,measurable-cong]*:
 $sets X \subseteq sets (qbs-to-measure (measure-to-qbs X))$
 ⟨proof⟩

lemma(in *standard-borel*) *standard-borel-lr-sets-ident[simp, measurable-cong]*:
 $sets (qbs-to-measure (measure-to-qbs M)) = sets M$
 ⟨proof⟩

2.2.3 The Adjunction

lemma *lr-adjunction-correspondence* :
 $X \rightarrow_Q (measure-to-qbs Y) = (qbs-to-measure X) \rightarrow_M Y$
 ⟨proof⟩

lemma(in *standard-borel*) *standard-borel-r-full-faithful*:
 $M \rightarrow_M Y = measure-to-qbs M \rightarrow_Q measure-to-qbs Y$
 ⟨proof⟩

lemma *qbs-morphism-dest[dest]*:
assumes $f \in X \rightarrow_Q measure-to-qbs Y$
shows $f \in qbs-to-measure X \rightarrow_M Y$
 ⟨proof⟩

lemma(in *standard-borel*) *qbs-morphism-dest[dest]*:

assumes $k \in \text{measure-to-qbs } M \rightarrow_Q \text{ measure-to-qbs } Y$
shows $k \in M \rightarrow_M Y$
 $\langle \text{proof} \rangle$

lemma *qbs-morphism-measurable-intro*[intro!]:

assumes $f \in \text{qbs-to-measure } X \rightarrow_M Y$
shows $f \in X \rightarrow_Q \text{ measure-to-qbs } Y$
 $\langle \text{proof} \rangle$

lemma(in *standard-borel*) *qbs-morphism-measurable-intro*[intro!]:

assumes $k \in M \rightarrow_M Y$
shows $k \in \text{measure-to-qbs } M \rightarrow_Q \text{ measure-to-qbs } Y$
 $\langle \text{proof} \rangle$

We can use the measurability prover when we reason about morphisms.

lemma

assumes $f \in \mathbb{R}_Q \rightarrow_Q \mathbb{R}_Q$
shows $(\lambda x. 2 * f x + (f x) \hat{~} 2) \in \mathbb{R}_Q \rightarrow_Q \mathbb{R}_Q$
 $\langle \text{proof} \rangle$

lemma

assumes $f \in X \rightarrow_Q \mathbb{R}_Q$
and $\alpha \in \text{qbs-Mx } X$
shows $(\lambda x. 2 * f (\alpha x) + (f (\alpha x)) \hat{~} 2) \in \mathbb{R}_Q \rightarrow_Q \mathbb{R}_Q$
 $\langle \text{proof} \rangle$

lemma *qbs-morphisn-from-countable*:

fixes $X :: 'a \text{ quasi-borel}$
assumes *countable* (*qbs-space* X)
 $\text{qbs-Mx } X \subseteq \text{real-borel} \rightarrow_M \text{count-space } (\text{qbs-space } X)$
and $\bigwedge i. i \in \text{qbs-space } X \implies f i \in \text{qbs-space } Y$
shows $f \in X \rightarrow_Q Y$
 $\langle \text{proof} \rangle$

corollary *nat-qbs-morphism*:

assumes $\bigwedge n. f n \in \text{qbs-space } Y$
shows $f \in \mathbb{N}_Q \rightarrow_Q Y$
 $\langle \text{proof} \rangle$

corollary *bool-qbs-morphism*:

assumes $\bigwedge b. f b \in \text{qbs-space } Y$
shows $f \in \mathbb{B}_Q \rightarrow_Q Y$
 $\langle \text{proof} \rangle$

2.2.4 The Adjunction w.r.t. Ordering

lemma *l-mono*:

mono qbs-to-measure

<proof>

lemma *r-mono*:

mono measure-to-qbs

<proof>

lemma *rl-order-adjunction*:

$X \leq \text{qbs-to-measure } Y \iff \text{measure-to-qbs } X \leq Y$
<proof>

end

2.3 Product Spaces

theory *Binary-Product-QuasiBorel*

imports *Measure-QuasiBorel-Adjunction*

begin

2.3.1 Binary Product Spaces

definition *pair-qbs-Mx* :: [*'a quasi-borel, 'b quasi-borel*] \Rightarrow (*real* \Rightarrow *'a* \times *'b*) *set*
where

pair-qbs-Mx X Y \equiv {*f. fst* \circ *f* \in *qbs-Mx X* \wedge *snd* \circ *f* \in *qbs-Mx Y*}

definition *pair-qbs* :: [*'a quasi-borel, 'b quasi-borel*] \Rightarrow (*'a* \times *'b*) *quasi-borel* (**infix**
 \otimes_Q *80*) **where**

pair-qbs X Y = *Abs-quasi-borel (qbs-space X* \times *qbs-space Y, pair-qbs-Mx X Y)*

lemma *pair-qbs-f[simp]*: *pair-qbs-Mx X Y* \subseteq *UNIV* \rightarrow *qbs-space X* \times *qbs-space Y*
<proof>

lemma *pair-qbs-closed1*: *qbs-closed1 (pair-qbs-Mx (X::'a quasi-borel) (Y::'b quasi-borel))*
<proof>

lemma *pair-qbs-closed2*: *qbs-closed2 (qbs-space X* \times *qbs-space Y) (pair-qbs-Mx X*
Y)
<proof>

lemma *pair-qbs-closed3*: *qbs-closed3 (pair-qbs-Mx (X::'a quasi-borel) (Y::'b quasi-borel))*
<proof>

lemma *pair-qbs-correct*: *Rep-quasi-borel (X* \otimes_Q *Y) = (qbs-space X* \times *qbs-space*
Y, pair-qbs-Mx X Y)
<proof>

lemma *pair-qbs-space[simp]*: *qbs-space (X* \otimes_Q *Y) = qbs-space X* \times *qbs-space Y*
<proof>

lemma *pair-qbs-Mx[simp]*: *qbs-Mx (X* \otimes_Q *Y) = pair-qbs-Mx X Y*

<proof>

lemma *pair-qbs-morphismI*:

assumes $\bigwedge \alpha \beta. \alpha \in \text{qbs-Mx } X \implies \beta \in \text{qbs-Mx } Y$
 $\implies f \circ (\lambda r. (\alpha r, \beta r)) \in \text{qbs-Mx } Z$

shows $f \in (X \otimes_Q Y) \rightarrow_Q Z$

<proof>

lemma *fst-qbs-morphism*:

$\text{fst} \in X \otimes_Q Y \rightarrow_Q X$

<proof>

lemma *snd-qbs-morphism*:

$\text{snd} \in X \otimes_Q Y \rightarrow_Q Y$

<proof>

lemma *qbs-morphism-pair-iff*:

$f \in X \rightarrow_Q Y \otimes_Q Z \iff \text{fst} \circ f \in X \rightarrow_Q Y \wedge \text{snd} \circ f \in X \rightarrow_Q Z$

<proof>

lemma *qbs-morphism-Pair1*:

assumes $x \in \text{qbs-space } X$

shows $\text{Pair } x \in Y \rightarrow_Q X \otimes_Q Y$

<proof>

lemma *qbs-morphism-Pair1'*:

assumes $x \in \text{qbs-space } X$

and $f \in X \otimes_Q Y \rightarrow_Q Z$

shows $(\lambda y. f (x,y)) \in Y \rightarrow_Q Z$

<proof>

lemma *qbs-morphism-Pair2*:

assumes $y \in \text{qbs-space } Y$

shows $(\lambda x. (x,y)) \in X \rightarrow_Q X \otimes_Q Y$

<proof>

lemma *qbs-morphism-Pair2'*:

assumes $y \in \text{qbs-space } Y$

and $f \in X \otimes_Q Y \rightarrow_Q Z$

shows $(\lambda x. f (x,y)) \in X \rightarrow_Q Z$

<proof>

lemma *qbs-morphism-fst''*:

assumes $f \in X \rightarrow_Q Y$

shows $(\lambda k. f (\text{fst } k)) \in X \otimes_Q Z \rightarrow_Q Y$

<proof>

lemma *qbs-morphism-snd''*:

assumes $f \in X \rightarrow_Q Y$

shows $(\lambda k. f (snd k)) \in Z \otimes_Q X \rightarrow_Q Y$
<proof>

lemma *qbs-morphism-tuple*:

assumes $f \in Z \rightarrow_Q X$

and $g \in Z \rightarrow_Q Y$

shows $(\lambda z. (f z, g z)) \in Z \rightarrow_Q X \otimes_Q Y$
<proof>

lemma *qbs-morphism-map-prod*:

assumes $f \in X \rightarrow_Q Y$

and $g \in X' \rightarrow_Q Y'$

shows $map-prod f g \in X \otimes_Q X' \rightarrow_Q Y \otimes_Q Y'$
<proof>

lemma *qbs-morphism-pair-swap'*:

$(\lambda(x,y). (y,x)) \in (X::'a \text{ quasi-borel}) \otimes_Q (Y::'b \text{ quasi-borel}) \rightarrow_Q Y \otimes_Q X$
<proof>

lemma *qbs-morphism-pair-swap*:

assumes $f \in X \otimes_Q Y \rightarrow_Q Z$

shows $(\lambda(x,y). f (y,x)) \in Y \otimes_Q X \rightarrow_Q Z$
<proof>

lemma *qbs-morphism-pair-assoc1*:

$(\lambda((x,y),z). (x,(y,z))) \in (X \otimes_Q Y) \otimes_Q Z \rightarrow_Q X \otimes_Q (Y \otimes_Q Z)$
<proof>

lemma *qbs-morphism-pair-assoc2*:

$(\lambda(x,(y,z)). ((x,y),z)) \in X \otimes_Q (Y \otimes_Q Z) \rightarrow_Q (X \otimes_Q Y) \otimes_Q Z$
<proof>

lemma *pair-qbs-fst*:

assumes $qbs-space Y \neq \{\}$

shows $map-qbs fst (X \otimes_Q Y) = X$
<proof>

lemma *pair-qbs-snd*:

assumes $qbs-space X \neq \{\}$

shows $map-qbs snd (X \otimes_Q Y) = Y$
<proof>

The following lemma corresponds to [1] Proposition 19(1).

lemma *r-preserves-product* :

$measure-to-qbs (X \otimes_M Y) = measure-to-qbs X \otimes_Q measure-to-qbs Y$
<proof>

lemma *l-product-sets*[*simp,measurable-cong*]:
 $sets (qbs\text{-to-measure } X \otimes_M qbs\text{-to-measure } Y) \subseteq sets (qbs\text{-to-measure } (X \otimes_Q Y))$
 ⟨*proof*⟩

lemma(**in** *pair-standard-borel*) *l-r-r-sets*[*simp,measurable-cong*]:
 $sets (qbs\text{-to-measure } (measure\text{-to-qbs } M \otimes_Q measure\text{-to-qbs } N)) = sets (M \otimes_M N)$
 ⟨*proof*⟩

end

2.3.2 Product Spaces

theory *Product-QuasiBorel*

imports *Binary-Product-QuasiBorel*

begin

definition *prod-qbs-Mx* :: [*'a set, 'a \Rightarrow 'b quasi-borel*] \Rightarrow (*real \Rightarrow 'a \Rightarrow 'b*) *set*
where
 $prod\text{-qbs-Mx } I M \equiv \{ \alpha \mid \alpha. \forall i. (i \in I \longrightarrow (\lambda r. \alpha r i) \in qbs\text{-Mx } (M i)) \wedge (i \notin I \longrightarrow (\lambda r. \alpha r i) = (\lambda r. undefined)) \}$

lemma *prod-qbs-MxI*:
assumes $\bigwedge i. i \in I \Longrightarrow (\lambda r. \alpha r i) \in qbs\text{-Mx } (M i)$
and $\bigwedge i. i \notin I \Longrightarrow (\lambda r. \alpha r i) = (\lambda r. undefined)$
shows $\alpha \in prod\text{-qbs-Mx } I M$
 ⟨*proof*⟩

lemma *prod-qbs-MxE*:
assumes $\alpha \in prod\text{-qbs-Mx } I M$
shows $\bigwedge i. i \in I \Longrightarrow (\lambda r. \alpha r i) \in qbs\text{-Mx } (M i)$
and $\bigwedge i. i \notin I \Longrightarrow (\lambda r. \alpha r i) = (\lambda r. undefined)$
and $\bigwedge i r. i \notin I \Longrightarrow \alpha r i = undefined$
 ⟨*proof*⟩

definition *PiQ* :: [*'a set \Rightarrow ('a \Rightarrow 'b quasi-borel) \Rightarrow ('a \Rightarrow 'b) quasi-borel*] **where**
 $PiQ I M \equiv Abs\text{-quasi-borel } (\Pi_E i \in I. qbs\text{-space } (M i), prod\text{-qbs-Mx } I M)$

syntax

-PiQ :: *pttrn \Rightarrow 'i set \Rightarrow 'a quasi-borel \Rightarrow ('i \Rightarrow 'a) quasi-borel* (($\exists \Pi_Q -\in - / -$)
 10)

translations

$\Pi_Q x \in I. M == CONST PiQ I (\lambda x. M)$

lemma *PiQ-f*: $prod\text{-qbs-Mx } I M \subseteq UNIV \rightarrow (\Pi_E i \in I. qbs\text{-space } (M i))$

<proof>

lemma *PiQ-closed1: qbs-closed1 (prod-qbs-Mx I M)*
<proof>

lemma *PiQ-closed2: qbs-closed2 ($\prod_E i \in I. \text{qbs-space } (M i)$) (prod-qbs-Mx I M)*
<proof>

lemma *PiQ-closed3: qbs-closed3 (prod-qbs-Mx I M)*
<proof>

lemma *PiQ-correct: Rep-quasi-borel (PiQ I M) = ($\prod_E i \in I. \text{qbs-space } (M i)$), prod-qbs-Mx I M)*
<proof>

lemma *PiQ-space[simp]: qbs-space (PiQ I M) = ($\prod_E i \in I. \text{qbs-space } (M i)$)*
<proof>

lemma *PiQ-Mx[simp]: qbs-Mx (PiQ I M) = prod-qbs-Mx I M*
<proof>

lemma *qbs-morphism-component-singleton:*
assumes $i \in I$
shows $(\lambda x. x i) \in (\prod_Q i \in I. (M i)) \rightarrow_Q M i$
<proof>

lemma *product-qbs-canonical1:*
assumes $\bigwedge i. i \in I \implies f i \in Y \rightarrow_Q X i$
and $\bigwedge i. i \notin I \implies f i = (\lambda y. \text{undefined})$
shows $(\lambda y i. f i y) \in Y \rightarrow_Q (\prod_Q i \in I. X i)$
<proof>

lemma *product-qbs-canonical2:*
assumes $\bigwedge i. i \in I \implies f i \in Y \rightarrow_Q X i$
 $\bigwedge i. i \notin I \implies f i = (\lambda y. \text{undefined})$
 $g \in Y \rightarrow_Q (\prod_Q i \in I. X i)$
 $\bigwedge i. i \in I \implies f i = (\lambda x. x i) \circ g$
and $y \in \text{qbs-space } Y$
shows $g y = (\lambda i. f i y)$
<proof>

lemma *merge-qbs-morphism:*
 $\text{merge } I J \in (\prod_Q i \in I. (M i)) \otimes_Q (\prod_Q j \in J. (M j)) \rightarrow_Q (\prod_Q i \in I \cup J. (M i))$
<proof>

The following lemma corresponds to [1] Proposition 19(1).

lemma *r-preserves-product':*
 $\text{measure-to-qbs } (\prod_M i \in I. M i) = (\prod_Q i \in I. \text{measure-to-qbs } (M i))$

<proof>

$$\prod_{i=0,1} X_i \cong X_1 \times X_2.$$

lemma *product-binary-product:*

$$\begin{aligned} & \exists f g. f \in (\prod_Q i \in UNIV. \text{if } i \text{ then } X \text{ else } Y) \rightarrow_Q X \otimes_Q Y \wedge g \in X \otimes_Q Y \rightarrow_Q \\ & (\prod_Q i \in UNIV. \text{if } i \text{ then } X \text{ else } Y) \wedge \\ & \quad g \circ f = id \wedge f \circ g = id \end{aligned}$$

<proof>

end

2.4 Coproduct Spaces

theory *Binary-CoProduct-QuasiBorel*

imports *Measure-QuasiBorel-Adjunction*

begin

2.4.1 Binary Coproduct Spaces

definition *copair-qbs-Mx* :: [*'a quasi-borel, 'b quasi-borel*] \Rightarrow (*real* \Rightarrow *'a + 'b*) *set*
where

$$\begin{aligned} & \text{copair-qbs-Mx } X \ Y \equiv \\ & \{g. \exists S \in \text{sets real-borel.} \\ & (S = \{\} \longrightarrow (\exists \alpha 1 \in \text{qbs-Mx } X. g = (\lambda r. \text{Inl } (\alpha 1 \ r)))) \wedge \\ & (S = UNIV \longrightarrow (\exists \alpha 2 \in \text{qbs-Mx } Y. g = (\lambda r. \text{Inr } (\alpha 2 \ r)))) \wedge \\ & ((S \neq \{\} \wedge S \neq UNIV) \longrightarrow \\ & (\exists \alpha 1 \in \text{qbs-Mx } X. \\ & \exists \alpha 2 \in \text{qbs-Mx } Y. \\ & \quad g = (\lambda r::\text{real. (if } (r \in S) \text{ then Inl } (\alpha 1 \ r) \text{ else Inr } (\alpha 2 \ r)))))) \} \end{aligned}$$

definition *copair-qbs* :: [*'a quasi-borel, 'b quasi-borel*] \Rightarrow (*'a + 'b*) *quasi-borel*
(**infixr** $\langle + \rangle_Q$ 65) **where**

$$\text{copair-qbs } X \ Y \equiv \text{Abs-quasi-borel (qbs-space } X \ \langle + \rangle \ \text{qbs-space } Y, \text{copair-qbs-Mx } X \ Y)$$

The followin is an equivalent definition of *copair-qbs-Mx*.

definition *copair-qbs-Mx2* :: [*'a quasi-borel, 'b quasi-borel*] \Rightarrow (*real* \Rightarrow *'a + 'b*)
set **where**

$$\begin{aligned} & \text{copair-qbs-Mx2 } X \ Y \equiv \\ & \{g. (\text{if qbs-space } X = \{\} \wedge \text{qbs-space } Y = \{\} \text{ then False} \\ & \quad \text{else if qbs-space } X \neq \{\} \wedge \text{qbs-space } Y = \{\} \text{ then} \\ & \quad \quad (\exists \alpha 1 \in \text{qbs-Mx } X. g = (\lambda r. \text{Inl } (\alpha 1 \ r))) \\ & \quad \text{else if qbs-space } X = \{\} \wedge \text{qbs-space } Y \neq \{\} \text{ then} \\ & \quad \quad (\exists \alpha 2 \in \text{qbs-Mx } Y. g = (\lambda r. \text{Inr } (\alpha 2 \ r))) \\ & \quad \text{else} \\ & \quad (\exists S \in \text{sets real-borel. } \exists \alpha 1 \in \text{qbs-Mx } X. \exists \alpha 2 \in \text{qbs-Mx } Y. \\ & \quad \quad g = (\lambda r::\text{real. (if } (r \in S) \text{ then Inl } (\alpha 1 \ r) \text{ else Inr } (\alpha 2 \ r)))) \} \end{aligned}$$

lemma *copair-qbs-Mx-equiv*: $\text{copair-qbs-Mx } (X :: 'a \text{ quasi-borel}) (Y :: 'b \text{ quasi-borel})$
 $= \text{copair-qbs-Mx2 } X Y$
 $\langle \text{proof} \rangle$

lemma *copair-qbs-f[simp]*: $\text{copair-qbs-Mx } X Y \subseteq \text{UNIV} \rightarrow \text{qbs-space } X \langle + \rangle$
 $\text{qbs-space } Y$
 $\langle \text{proof} \rangle$

lemma *copair-qbs-closed1*: $\text{qbs-closed1 } (\text{copair-qbs-Mx } X Y)$
 $\langle \text{proof} \rangle$

lemma *copair-qbs-closed2*: $\text{qbs-closed2 } (\text{qbs-space } X \langle + \rangle \text{qbs-space } Y) (\text{copair-qbs-Mx } X Y)$
 $\langle \text{proof} \rangle$

lemma *copair-qbs-closed3*: $\text{qbs-closed3 } (\text{copair-qbs-Mx } X Y)$
 $\langle \text{proof} \rangle$

lemma *copair-qbs-correct*: $\text{Rep-quasi-borel } (\text{copair-qbs } X Y) = (\text{qbs-space } X \langle + \rangle$
 $\text{qbs-space } Y, \text{copair-qbs-Mx } X Y)$
 $\langle \text{proof} \rangle$

lemma *copair-qbs-space[simp]*: $\text{qbs-space } (\text{copair-qbs } X Y) = \text{qbs-space } X \langle + \rangle$
 $\text{qbs-space } Y$
 $\langle \text{proof} \rangle$

lemma *copair-qbs-Mx[simp]*: $\text{qbs-Mx } (\text{copair-qbs } X Y) = \text{copair-qbs-Mx } X Y$
 $\langle \text{proof} \rangle$

lemma *Inl-qbs-morphism*:
 $\text{Inl} \in X \rightarrow_Q X \langle + \rangle_Q Y$
 $\langle \text{proof} \rangle$

lemma *Inr-qbs-morphism*:
 $\text{Inr} \in Y \rightarrow_Q X \langle + \rangle_Q Y$
 $\langle \text{proof} \rangle$

lemma *case-sum-preserves-morphisms*:
assumes $f \in X \rightarrow_Q Z$
and $g \in Y \rightarrow_Q Z$
shows $\text{case-sum } f g \in X \langle + \rangle_Q Y \rightarrow_Q Z$
 $\langle \text{proof} \rangle$

lemma *map-sum-preserves-morphisms*:
assumes $f \in X \rightarrow_Q Y$
and $g \in X' \rightarrow_Q Y'$

shows $\text{map-sum } f g \in X \langle + \rangle_Q X' \rightarrow_Q Y \langle + \rangle_Q Y'$
 $\langle \text{proof} \rangle$

end

2.4.2 Countable Coproduct Spaces

theory *CoProduct-QuasiBorel*

imports

Product-QuasiBorel

Binary-CoProduct-QuasiBorel

begin

definition *coprod-qbs-Mx* :: [*'a set, 'a \Rightarrow 'b quasi-borel*] \Rightarrow (*real \Rightarrow 'a \times 'b*) *set*
where

coprod-qbs-Mx I X \equiv { $\lambda r. (f r, \alpha (f r) r) \mid f \alpha. f \in \text{real-borel} \rightarrow_M \text{count-space } I$
 $\wedge (\forall i \in \text{range } f. \alpha i \in \text{qbs-Mx } (X i))$ }

lemma *coprod-qbs-MxI*:

assumes $f \in \text{real-borel} \rightarrow_M \text{count-space } I$

and $\bigwedge i. i \in \text{range } f \implies \alpha i \in \text{qbs-Mx } (X i)$

shows $(\lambda r. (f r, \alpha (f r) r)) \in \text{coprod-qbs-Mx } I X$

$\langle \text{proof} \rangle$

definition *coprod-qbs-Mx'* :: [*'a set, 'a \Rightarrow 'b quasi-borel*] \Rightarrow (*real \Rightarrow 'a \times 'b*) *set*
where

coprod-qbs-Mx' I X \equiv { $\lambda r. (f r, \alpha (f r) r) \mid f \alpha. f \in \text{real-borel} \rightarrow_M \text{count-space } I$
 $\wedge (\forall i. (i \in \text{range } f \vee \text{qbs-space } (X i) \neq \{\}) \longrightarrow \alpha i \in \text{qbs-Mx } (X i))$ }

lemma *coproduct-qbs-Mx-eq*:

$\text{coprod-qbs-Mx } I X = \text{coprod-qbs-Mx}' I X$

$\langle \text{proof} \rangle$

definition *coprod-qbs* :: [*'a set, 'a \Rightarrow 'b quasi-borel*] \Rightarrow (*'a \times 'b*) *quasi-borel* **where**
coprod-qbs I X $\equiv \text{Abs-quasi-borel } (\text{SIGMA } i:I. \text{qbs-space } (X i), \text{coprod-qbs-Mx } I X)$

syntax

-coprod-qbs :: *pttrn \Rightarrow 'i set \Rightarrow 'a quasi-borel \Rightarrow ('i \times 'a) quasi-borel* (($\exists \Pi_Q \text{-}\in\text{-}/$
 $\text{-})$ 10)

translations

$\Pi_Q x \in I. M \equiv \text{CONST } \text{coprod-qbs } I (\lambda x. M)$

lemma *coprod-qbs-f[simp]*: $\text{coprod-qbs-Mx } I X \subseteq \text{UNIV} \rightarrow (\text{SIGMA } i:I. \text{qbs-space } (X i))$

$\langle \text{proof} \rangle$

lemma *coprod-qbs-closed1*: *qbs-closed1 (coprod-qbs-Mx I X)*

<proof>

lemma *coprod-qbs-closed2*: *qbs-closed2* (SIGMA *i*:*I*. *qbs-space* (*X i*)) (*coprod-qbs-Mx I X*)

<proof>

lemma *coprod-qbs-closed3*:
qbs-closed3 (*coprod-qbs-Mx I X*)
<proof>

lemma *coprod-qbs-correct*: *Rep-quasi-borel* (*coprod-qbs I X*) = (SIGMA *i*:*I*. *qbs-space* (*X i*), *coprod-qbs-Mx I X*)
<proof>

lemma *coproduct-qbs-space[simp]*: *qbs-space* (*coprod-qbs I X*) = (SIGMA *i*:*I*. *qbs-space* (*X i*))
<proof>

lemma *coproduct-qbs-Mx[simp]*: *qbs-Mx* (*coprod-qbs I X*) = *coprod-qbs-Mx I X*
<proof>

lemma *ini-morphism*:
assumes *j* ∈ *I*
shows (λ*x*. (*j,x*)) ∈ *X j* →_Q (Π_Q *i* ∈ *I*. *X i*)
<proof>

lemma *coprod-qbs-canonical1*:
assumes *countable I*
and ∧*i*. *i* ∈ *I* ⇒ *f i* ∈ *X i* →_Q *Y*
shows (λ(*i,x*). *f i x*) ∈ (Π_Q *i* ∈ *I*. *X i*) →_Q *Y*
<proof>

lemma *coprod-qbs-canonical1'*:
assumes *countable I*
and ∧*i*. *i* ∈ *I* ⇒ (λ*x*. *f (i,x)*) ∈ *X i* →_Q *Y*
shows *f* ∈ (Π_Q *i* ∈ *I*. *X i*) →_Q *Y*
<proof>

Π_{*i*=0,1} *X_i* ≅ *X₁* + *X₂*.

lemma *coproduct-binary-coproduct*:
∃ *f g*. *f* ∈ (Π_Q *i* ∈ UNIV. if *i* then *X* else *Y*) →_Q *X* <+>_Q *Y* ∧ *g* ∈ *X* <+>_Q *Y*
→_Q (Π_Q *i* ∈ UNIV. if *i* then *X* else *Y*) ∧
g ∘ *f* = *id* ∧ *f* ∘ *g* = *id*
<proof>

2.4.3 Lists

abbreviation *list-of X* ≡ Π_Q *n* ∈ (UNIV :: nat set). (Π_Q *i* ∈ {..*n*}. *X*)

abbreviation $list-nil :: nat \times (nat \Rightarrow 'a)$ **where**

$list-nil \equiv (0, \lambda n. undefined)$

abbreviation $list-cons :: ['a, nat \times (nat \Rightarrow 'a)] \Rightarrow nat \times (nat \Rightarrow 'a)$ **where**

$list-cons\ x\ l \equiv (Suc\ (fst\ l), (\lambda n. if\ n = 0\ then\ x\ else\ (snd\ l)\ (n - 1)))$

definition $list-head :: nat \times (nat \Rightarrow 'a) \Rightarrow 'a$ **where**

$list-head\ l = snd\ l\ 0$

definition $list-tail :: nat \times (nat \Rightarrow 'a) \Rightarrow nat \times (nat \Rightarrow 'a)$ **where**

$list-tail\ l = (fst\ l - 1, \lambda m. (snd\ l)\ (Suc\ m))$

lemma $list-simp1$:

$list-nil \neq list-cons\ x\ l$

$\langle proof \rangle$

lemma $list-simp2$:

assumes $list-cons\ a\ al = list-cons\ b\ bl$

shows $a = b\ al = bl$

$\langle proof \rangle$

lemma $list-simp3$:

shows $list-head\ (list-cons\ a\ l) = a$

$\langle proof \rangle$

lemma $list-simp4$:

assumes $l \in qbs-space\ (list-of\ X)$

shows $list-tail\ (list-cons\ a\ l) = l$

$\langle proof \rangle$

lemma $list-decomp1$:

assumes $l \in qbs-space\ (list-of\ X)$

shows $l = list-nil \vee$

$(\exists a\ l'. a \in qbs-space\ X \wedge l' \in qbs-space\ (list-of\ X) \wedge l = list-cons\ a\ l')$

$\langle proof \rangle$

lemma $list-simp5$:

assumes $l \in qbs-space\ (list-of\ X)$

and $l \neq list-nil$

shows $l = list-cons\ (list-head\ l)\ (list-tail\ l)$

$\langle proof \rangle$

lemma $list-simp6$:

$list-nil \in qbs-space\ (list-of\ X)$

$\langle proof \rangle$

lemma $list-simp7$:

assumes $a \in qbs-space\ X$

and $l \in qbs-space\ (list-of\ X)$

shows $list-cons\ a\ l \in qbs-space\ (list-of\ X)$

$\langle \text{proof} \rangle$

lemma *list-destruct-rule*:

assumes $l \in \text{qbs-space } (\text{list-of } X)$
 $P \text{ list-nil}$
and $\bigwedge a l'. a \in \text{qbs-space } X \implies l' \in \text{qbs-space } (\text{list-of } X) \implies P (\text{list-cons } a$
 $l')$
shows $P l$
 $\langle \text{proof} \rangle$

lemma *list-induct-rule*:

assumes $l \in \text{qbs-space } (\text{list-of } X)$
 $P \text{ list-nil}$
and $\bigwedge a l'. a \in \text{qbs-space } X \implies l' \in \text{qbs-space } (\text{list-of } X) \implies P l' \implies P$
 $(\text{list-cons } a l')$
shows $P l$
 $\langle \text{proof} \rangle$

fun *from-list* :: $'a \text{ list} \Rightarrow \text{nat} \times (\text{nat} \Rightarrow 'a)$ **where**

from-list [] = *list-nil* |
from-list (a#l) = *list-cons* a (*from-list* l)

fun *to-list'* :: $\text{nat} \Rightarrow (\text{nat} \Rightarrow 'a) \Rightarrow 'a \text{ list}$ **where**

to-list' 0 - = [] |
to-list' (Suc n) f = f 0 # *to-list'* n ($\lambda n. f (\text{Suc } n)$)

definition *to-list* :: $\text{nat} \times (\text{nat} \Rightarrow 'a) \Rightarrow 'a \text{ list}$ **where**

to-list $\equiv \text{case-prod } \text{to-list}'$

lemma *to-list-simp1*:

shows *to-list* *list-nil* = []
 $\langle \text{proof} \rangle$

lemma *to-list-simp2*:

assumes $l \in \text{qbs-space } (\text{list-of } X)$
shows *to-list* (*list-cons* a l) = a # *to-list* l
 $\langle \text{proof} \rangle$

lemma *from-list-length*:

fst (*from-list* l) = *length* l
 $\langle \text{proof} \rangle$

lemma *from-list-in-list-of*:

assumes $set l \subseteq \text{qbs-space } X$
shows *from-list* l $\in \text{qbs-space } (\text{list-of } X)$
 $\langle \text{proof} \rangle$

lemma *from-list-in-list-of'*:

shows $from-list\ l \in qbs-space\ (list-of\ (Abs-quasi-borel\ (UNIV, UNIV)))$
 $\langle proof \rangle$

lemma *list-cons-in-list-of*:
assumes $set\ (a\#\ l) \subseteq qbs-space\ X$
shows $list-cons\ a\ (from-list\ l) \in qbs-space\ (list-of\ X)$
 $\langle proof \rangle$

lemma *from-list-to-list-ident*:
 $(to-list\ \circ\ from-list)\ l = l$
 $\langle proof \rangle$

lemma *to-list-from-list-ident*:
assumes $l \in qbs-space\ (list-of\ X)$
shows $(from-list\ \circ\ to-list)\ l = l$
 $\langle proof \rangle$

definition *rec-list'* $:: 'b \Rightarrow ('a \Rightarrow (nat \times (nat \Rightarrow 'a))) \Rightarrow 'b \Rightarrow 'b \Rightarrow (nat \times (nat \Rightarrow 'a)) \Rightarrow 'b$ **where**
 $rec-list'\ t0\ f\ l \equiv (rec-list\ t0\ (\lambda x\ l'.\ f\ x\ (from-list\ l'))\ (to-list\ l))$

lemma *rec-list'-simp1*:
 $rec-list'\ t\ f\ list-nil = t$
 $\langle proof \rangle$

lemma *rec-list'-simp2*:
assumes $l \in qbs-space\ (list-of\ X)$
shows $rec-list'\ t\ f\ (list-cons\ x\ l) = f\ x\ l\ (rec-list'\ t\ f\ l)$
 $\langle proof \rangle$

end

2.5 Function Spaces

theory *Exponent-QuasiBorel*
imports *CoProduct-QuasiBorel*
begin

2.5.1 Function Spaces

definition *exp-qbs-Mx* $:: ['a\ quasi-borel, 'b\ quasi-borel] \Rightarrow (real \Rightarrow 'a \Rightarrow 'b)$ **set**
where
 $exp-qbs-Mx\ X\ Y \equiv \{g :: real \Rightarrow 'a \Rightarrow 'b.\ case-prod\ g \in \mathbf{R}_Q \otimes_Q X \rightarrow_Q Y\}$

definition *exp-qbs* $:: ['a\ quasi-borel, 'b\ quasi-borel] \Rightarrow ('a \Rightarrow 'b)$ **quasi-borel** (**infixr**
 \Rightarrow_Q 61) **where**
 $X \Rightarrow_Q Y \equiv Abs-quasi-borel\ (X \rightarrow_Q Y, exp-qbs-Mx\ X\ Y)$

lemma *exp-qbs-f[simp]*: $\text{exp-qbs-Mx } X \ Y \subseteq \text{UNIV} \rightarrow (X :: 'a \text{ quasi-borel}) \rightarrow_Q (Y :: 'b \text{ quasi-borel})$

<proof>

lemma *exp-qbs-closed1*: $\text{qbs-closed1 } (\text{exp-qbs-Mx } X \ Y)$

<proof>

lemma *exp-qbs-closed2*: $\text{qbs-closed2 } (X \rightarrow_Q Y) (\text{exp-qbs-Mx } X \ Y)$

<proof>

lemma *exp-qbs-closed3*: $\text{qbs-closed3 } (\text{exp-qbs-Mx } X \ Y)$

<proof>

lemma *exp-qbs-correct*: $\text{Rep-quasi-borel } (\text{exp-qbs } X \ Y) = (X \rightarrow_Q Y, \text{exp-qbs-Mx } X \ Y)$

<proof>

lemma *exp-qbs-space[simp]*: $\text{qbs-space } (\text{exp-qbs } X \ Y) = X \rightarrow_Q Y$

<proof>

lemma *exp-qbs-Mx[simp]*: $\text{qbs-Mx } (\text{exp-qbs } X \ Y) = \text{exp-qbs-Mx } X \ Y$

<proof>

lemma *qbs-exp-morphismI*:

assumes $\bigwedge \alpha \ \beta. \ \alpha \in \text{qbs-Mx } X \implies$

$\beta \in \text{pair-qbs-Mx real-quasi-borel } Y \implies$

$(\lambda(r,x). (f \circ \alpha) \ r \ x) \circ \beta \in \text{qbs-Mx } Z$

shows $f \in X \rightarrow_Q \text{exp-qbs } Y \ Z$

<proof>

definition *qbs-eval* :: $(('a \Rightarrow 'b) \times 'a) \Rightarrow 'b$ **where**

$\text{qbs-eval } a \equiv (\text{fst } a) (\text{snd } a)$

lemma *qbs-eval-morphism*:

$\text{qbs-eval} \in (\text{exp-qbs } X \ Y) \otimes_Q X \rightarrow_Q Y$

<proof>

lemma *curry-morphism*:

$\text{curry} \in \text{exp-qbs } (X \otimes_Q Y) \ Z \rightarrow_Q \text{exp-qbs } X \ (\text{exp-qbs } Y \ Z)$

<proof>

lemma *curry-preserves-morphisms*:

assumes $f \in X \otimes_Q Y \rightarrow_Q Z$

shows $\text{curry } f \in X \rightarrow_Q \text{exp-qbs } Y \ Z$

<proof>

lemma *uncurry-morphism*:

$case\text{-}prod \in exp\text{-}qbs\ X\ (exp\text{-}qbs\ Y\ Z) \rightarrow_Q exp\text{-}qbs\ (X \otimes_Q Y)\ Z$
 $\langle proof \rangle$

lemma *uncurry-preserves-morphisms*:
assumes $f \in X \rightarrow_Q exp\text{-}qbs\ Y\ Z$
shows $case\text{-}prod\ f \in X \otimes_Q Y \rightarrow_Q Z$
 $\langle proof \rangle$

lemma *arg-swap-morphism*:
assumes $f \in X \rightarrow_Q exp\text{-}qbs\ Y\ Z$
shows $(\lambda y\ x. f\ x\ y) \in Y \rightarrow_Q exp\text{-}qbs\ X\ Z$
 $\langle proof \rangle$

lemma *exp-qbs-comp-morphism*:
assumes $f \in W \rightarrow_Q exp\text{-}qbs\ X\ Y$
and $g \in W \rightarrow_Q exp\text{-}qbs\ Y\ Z$
shows $(\lambda w. g\ w \circ f\ w) \in W \rightarrow_Q exp\text{-}qbs\ X\ Z$
 $\langle proof \rangle$

lemma *case-sum-morphism*:
 $case\text{-}prod\ case\text{-}sum \in exp\text{-}qbs\ X\ Z \otimes_Q exp\text{-}qbs\ Y\ Z \rightarrow_Q exp\text{-}qbs\ (X <+>_Q Y)\ Z$
 $\langle proof \rangle$

lemma *not-qbs-morphism*:
 $Not \in \mathbb{B}_Q \rightarrow_Q \mathbb{B}_Q$
 $\langle proof \rangle$

lemma *or-qbs-morphism*:
 $(\vee) \in \mathbb{B}_Q \rightarrow_Q exp\text{-}qbs\ \mathbb{B}_Q\ \mathbb{B}_Q$
 $\langle proof \rangle$

lemma *and-qbs-morphism*:
 $(\wedge) \in \mathbb{B}_Q \rightarrow_Q exp\text{-}qbs\ \mathbb{B}_Q\ \mathbb{B}_Q$
 $\langle proof \rangle$

lemma *implies-qbs-morphism*:
 $(\longrightarrow) \in \mathbb{B}_Q \rightarrow_Q \mathbb{B}_Q \Rightarrow_Q \mathbb{B}_Q$
 $\langle proof \rangle$

lemma *less-nat-qbs-morphism*:
 $(<) \in \mathbb{N}_Q \rightarrow_Q exp\text{-}qbs\ \mathbb{N}_Q\ \mathbb{B}_Q$
 $\langle proof \rangle$

lemma *less-real-qbs-morphism*:
 $(<) \in \mathbb{R}_Q \rightarrow_Q exp\text{-}qbs\ \mathbb{R}_Q\ \mathbb{B}_Q$
 $\langle proof \rangle$

lemma *rec-list-morphism'*:
rec-list' ∈ qbs-space (exp-qbs Y (exp-qbs (exp-qbs X (exp-qbs (list-of X) (exp-qbs Y Y))) (exp-qbs (list-of X) Y)))
 ⟨*proof*⟩

end

3 Probability Spaces

3.1 Probability Measures

theory *Probability-Space-QuasiBorel*
imports *Exponent-QuasiBorel*
begin

3.1.1 Probability Measures

type-synonym *'a qbs-prob-t = 'a quasi-borel * (real ⇒ 'a) * real measure*

locale *in-Mx =*
fixes *X :: 'a quasi-borel*
and *α :: real ⇒ 'a*
assumes *in-Mx[simp]: α ∈ qbs-Mx X*

locale *qbs-prob = in-Mx X α + real-distribution μ*
for *X :: 'a quasi-borel and α and μ*
begin
declare *prob-space-axioms[simp]*

lemma *m-in-space-prob-algebra[simp]:*
μ ∈ space (prob-algebra real-borel)
 ⟨*proof*⟩
end

locale *pair-qbs-probs = qp1:qbs-prob X α μ + qp2:qbs-prob Y β ν*
for *X :: 'a quasi-borel and α μ and Y :: 'b quasi-borel and β ν*
begin

sublocale *pair-prob-space μ ν*
 ⟨*proof*⟩

lemma *ab-measurable[measurable]:*
map-prod α β ∈ real-borel ⊗_M real-borel →_M qbs-to-measure (X ⊗_Q Y)
 ⟨*proof*⟩

lemma *ab-g-in-Mx[simp]:*
map-prod α β ∘ real-real.g ∈ pair-qbs-Mx X Y

<proof>

sublocale *qbs-prob* $X \otimes_Q Y$ *map-prod* $\alpha \beta \circ \text{real-real.g distr } (\mu \otimes_M \nu)$ *real-borel*
real-real.f
<proof>

end

locale *pair-qbs-prob* = *qp1:qbs-prob* $X \alpha \mu$ + *qp2:qbs-prob* $Y \beta \nu$
for $X :: 'a \text{ quasi-borel}$ **and** $\alpha \mu$ **and** $Y :: 'a \text{ quasi-borel}$ **and** $\beta \nu$
begin

sublocale *pair-qbs-probs*
<proof>

lemma *same-spaces[simp]*:
assumes $Y = X$
shows $\beta \in \text{qbs-Mx } X$
<proof>

end

lemma *prob-algebra-real-prob-measure*:
 $p \in \text{space } (\text{prob-algebra } (\text{real-borel})) = \text{real-distribution } p$
<proof>

lemma *qbs-probI*:
assumes $\alpha \in \text{qbs-Mx } X$
and *sets* $\mu = \text{sets borel}$
and *prob-space* μ
shows *qbs-prob* $X \alpha \mu$
<proof>

lemma *qbs-empty-not-qbs-prob* : $\neg \text{qbs-prob } (\text{empty-quasi-borel}) f M$
<proof>

definition *qbs-prob-eq* :: [$'a \text{ qbs-prob-t}$, $'a \text{ qbs-prob-t}$] $\Rightarrow \text{bool}$ **where**
qbs-prob-eq $p1 p2 \equiv$
(let $(qbs1, a1, m1) = p1;$
 $(qbs2, a2, m2) = p2$ *in*
 $\text{qbs-prob } qbs1 a1 m1 \wedge \text{qbs-prob } qbs2 a2 m2 \wedge qbs1 = qbs2 \wedge$
 $\text{distr } m1 (\text{qbs-to-measure } qbs1) a1 = \text{distr } m2 (\text{qbs-to-measure } qbs2) a2)$

definition *qbs-prob-eq2* :: [$'a \text{ qbs-prob-t}$, $'a \text{ qbs-prob-t}$] $\Rightarrow \text{bool}$ **where**
qbs-prob-eq2 $p1 p2 \equiv$
(let $(qbs1, a1, m1) = p1;$
 $(qbs2, a2, m2) = p2$ *in*
 $\text{qbs-prob } qbs1 a1 m1 \wedge \text{qbs-prob } qbs2 a2 m2 \wedge qbs1 = qbs2 \wedge$
 $(\forall f \in qbs1 \rightarrow_Q \text{real-quasi-borel}.$

$$(\int x. f (a1 x) \partial m1) = (\int x. f (a2 x) \partial m2))$$

definition *qbs-prob-eq3* :: [*'a qbs-prob-t, 'a qbs-prob-t*] \Rightarrow *bool* **where**
qbs-prob-eq3 *p1 p2* \equiv
 (let (*qbs1, a1, m1*) = *p1*;
 (*qbs2, a2, m2*) = *p2* in
 (*qbs-prob qbs1 a1 m1* \wedge *qbs-prob qbs2 a2 m2* \wedge *qbs1* = *qbs2* \wedge
 ($\forall f \in$ *qbs1* \rightarrow_Q *real-quasi-borel*.
 ($\forall k \in$ *qbs-space qbs1*. $0 \leq f k$) \longrightarrow
 ($\int x. f (a1 x) \partial m1$) = ($\int x. f (a2 x) \partial m2$))))

definition *qbs-prob-eq4* :: [*'a qbs-prob-t, 'a qbs-prob-t*] \Rightarrow *bool* **where**
qbs-prob-eq4 *p1 p2* \equiv
 (let (*qbs1, a1, m1*) = *p1*;
 (*qbs2, a2, m2*) = *p2* in
 (*qbs-prob qbs1 a1 m1* \wedge *qbs-prob qbs2 a2 m2* \wedge *qbs1* = *qbs2* \wedge
 ($\forall f \in$ *qbs1* \rightarrow_Q $\mathbb{R}_{\geq 0}$.
 ($\int ^+ x. f (a1 x) \partial m1$) = ($\int ^+ x. f (a2 x) \partial m2$))))

lemma(in *qbs-prob*) *qbs-prob-eq-refl[simp]*:
qbs-prob-eq (*X, α, μ*) (*X, α, μ*)
<proof>

lemma(in *qbs-prob*) *qbs-prob-eq2-refl[simp]*:
qbs-prob-eq2 (*X, α, μ*) (*X, α, μ*)
<proof>

lemma(in *qbs-prob*) *qbs-prob-eq3-refl[simp]*:
qbs-prob-eq3 (*X, α, μ*) (*X, α, μ*)
<proof>

lemma(in *qbs-prob*) *qbs-prob-eq4-refl[simp]*:
qbs-prob-eq4 (*X, α, μ*) (*X, α, μ*)
<proof>

lemma(in *pair-qbs-prob*) *qbs-prob-eq-intro*:
assumes *X = Y*
and *distr μ (qbs-to-measure X) α = distr ν (qbs-to-measure X) β*
shows *qbs-prob-eq* (*X, α, μ*) (*Y, β, ν*)
<proof>

lemma(in *pair-qbs-prob*) *qbs-prob-eq2-intro*:
assumes *X = Y*
and $\bigwedge f. f \in$ *qbs-to-measure X* \rightarrow_M *real-borel*
 \implies ($\int x. f (\alpha x) \partial \mu$) = ($\int x. f (\beta x) \partial \nu$)
shows *qbs-prob-eq2* (*X, α, μ*) (*Y, β, ν*)
<proof>

lemma(in *pair-qbs-prob*) *qbs-prob-eq3-intro*:

assumes $X = Y$
and $\bigwedge f. f \in \text{qbs-to-measure } X \rightarrow_M \text{real-borel} \implies (\forall k \in \text{qbs-space } X. 0 \leq f k)$
 $\implies (\int x. f (\alpha x) \partial \mu) = (\int x. f (\beta x) \partial \nu)$
shows $\text{qbs-prob-eq3 } (X, \alpha, \mu) (Y, \beta, \nu)$
 $\langle \text{proof} \rangle$

lemma(in *pair-qbs-prob*) *qbs-prob-eq4-intro*:
assumes $X = Y$
and $\bigwedge f. f \in \text{qbs-to-measure } X \rightarrow_M \text{ennreal-borel}$
 $\implies (\int^{+x}. f (\alpha x) \partial \mu) = (\int^{+x}. f (\beta x) \partial \nu)$
shows $\text{qbs-prob-eq4 } (X, \alpha, \mu) (Y, \beta, \nu)$
 $\langle \text{proof} \rangle$

lemma *qbs-prob-eq-dest*:
assumes $\text{qbs-prob-eq } (X, \alpha, \mu) (Y, \beta, \nu)$
shows $\text{qbs-prob } X \alpha \mu$
 $\text{qbs-prob } Y \beta \nu$
 $Y = X$
and $\text{distr } \mu (\text{qbs-to-measure } X) \alpha = \text{distr } \nu (\text{qbs-to-measure } X) \beta$
 $\langle \text{proof} \rangle$

lemma *qbs-prob-eq2-dest*:
assumes $\text{qbs-prob-eq2 } (X, \alpha, \mu) (Y, \beta, \nu)$
shows $\text{qbs-prob } X \alpha \mu$
 $\text{qbs-prob } Y \beta \nu$
 $Y = X$
and $\bigwedge f. f \in \text{qbs-to-measure } X \rightarrow_M \text{real-borel}$
 $\implies (\int x. f (\alpha x) \partial \mu) = (\int x. f (\beta x) \partial \nu)$
 $\langle \text{proof} \rangle$

lemma *qbs-prob-eq3-dest*:
assumes $\text{qbs-prob-eq3 } (X, \alpha, \mu) (Y, \beta, \nu)$
shows $\text{qbs-prob } X \alpha \mu$
 $\text{qbs-prob } Y \beta \nu$
 $Y = X$
and $\bigwedge f. f \in \text{qbs-to-measure } X \rightarrow_M \text{real-borel} \implies (\forall k \in \text{qbs-space } X. 0 \leq f k)$
 $\implies (\int x. f (\alpha x) \partial \mu) = (\int x. f (\beta x) \partial \nu)$
 $\langle \text{proof} \rangle$

lemma *qbs-prob-eq4-dest*:
assumes $\text{qbs-prob-eq4 } (X, \alpha, \mu) (Y, \beta, \nu)$
shows $\text{qbs-prob } X \alpha \mu$
 $\text{qbs-prob } Y \beta \nu$
 $Y = X$
and $\bigwedge f. f \in \text{qbs-to-measure } X \rightarrow_M \text{ennreal-borel}$
 $\implies (\int^{+x}. f (\alpha x) \partial \mu) = (\int^{+x}. f (\beta x) \partial \nu)$
 $\langle \text{proof} \rangle$

definition *qbs-prob-t-ennintegral* :: [*'a qbs-prob-t, 'a \Rightarrow ennreal*] \Rightarrow *ennreal* **where**
qbs-prob-t-ennintegral *p f* \equiv
 (if *f* \in (*fst p*) \rightarrow_Q *ennreal-quasi-borel*
 then ($\int^+ x. f$ (*fst* (*snd p*) *x*) ∂ (*snd* (*snd p*))) else 0)

definition *qbs-prob-t-integral* :: [*'a qbs-prob-t, 'a \Rightarrow real*] \Rightarrow *real* **where**
qbs-prob-t-integral *p f* \equiv
 (if *f* \in (*fst p*) \rightarrow_Q \mathbf{R}_Q
 then ($\int x. f$ (*fst* (*snd p*) *x*) ∂ (*snd* (*snd p*)))
 else 0)

definition *qbs-prob-t-integrable* :: [*'a qbs-prob-t, 'a \Rightarrow real*] \Rightarrow *bool* **where**
qbs-prob-t-integrable *p f* $\equiv f \in \text{fst } p \rightarrow_Q \text{real-quasi-borel} \wedge \text{integrable } (\text{snd } (\text{snd } p))$
 (*f* \circ (*fst* (*snd p*)))

definition *qbs-prob-t-measure* :: *'a qbs-prob-t \Rightarrow 'a measure* **where**
qbs-prob-t-measure *p* $\equiv \text{distr } (\text{snd } (\text{snd } p)) (\text{qbs-to-measure } (\text{fst } p)) (\text{fst } (\text{snd } p))$

lemma *qbs-prob-eq-symp*:
symp qbs-prob-eq
<proof>

lemma *qbs-prob-eq-transp*:
transp qbs-prob-eq
<proof>

quotient-type *'a qbs-prob-space = 'a qbs-prob-t / partial: qbs-prob-eq*
morphisms *rep-qbs-prob-space qbs-prob-space*
<proof>

interpretation *qbs-prob-space : quot-type qbs-prob-eq Abs-qbs-prob-space Rep-qbs-prob-space*
<proof>

lemma *qbs-prob-space-induct*:
assumes $\bigwedge X \alpha \mu. \text{qbs-prob } X \alpha \mu \implies P (\text{qbs-prob-space } (X, \alpha, \mu))$
shows *P s*
<proof>

lemma *qbs-prob-space-induct'*:
assumes $\bigwedge X \alpha \mu. \text{qbs-prob } X \alpha \mu \implies s = \text{qbs-prob-space } (X, \alpha, \mu) \implies P$
 (*qbs-prob-space* (*X, α, μ*))
shows *P s*
<proof>

lemma *rep-qbs-prob-space*:
 $\exists X \alpha \mu. p = \text{qbs-prob-space } (X, \alpha, \mu) \wedge \text{qbs-prob } X \alpha \mu$
<proof>

lemma(in *qbs-prob*) *in-Rep*:
 $(X, \alpha, \mu) \in \text{Rep-qbs-prob-space } (\text{qbs-prob-space } (X, \alpha, \mu))$
 $\langle \text{proof} \rangle$

lemma(in *qbs-prob*) *if-in-Rep*:
assumes $(X', \alpha', \mu') \in \text{Rep-qbs-prob-space } (\text{qbs-prob-space } (X, \alpha, \mu))$
shows $X' = X$
 $\text{qbs-prob } X' \alpha' \mu'$
 $\text{qbs-prob-eq } (X, \alpha, \mu) (X', \alpha', \mu')$
 $\langle \text{proof} \rangle$

lemma(in *qbs-prob*) *in-Rep-induct*:
assumes $\bigwedge Y \beta \nu. (Y, \beta, \nu) \in \text{Rep-qbs-prob-space } (\text{qbs-prob-space } (X, \alpha, \mu)) \implies P$
 (Y, β, ν)
shows $P (\text{rep-qbs-prob-space } (\text{qbs-prob-space } (X, \alpha, \mu)))$
 $\langle \text{proof} \rangle$

lemma *qbs-prob-eq-2-implies-3* :
assumes *qbs-prob-eq2* $p1 \ p2$
shows *qbs-prob-eq3* $p1 \ p2$
 $\langle \text{proof} \rangle$

lemma *qbs-prob-eq-3-implies-1* :
assumes *qbs-prob-eq3* $(p1 :: 'a \text{ qbs-prob-t}) \ p2$
shows *qbs-prob-eq* $p1 \ p2$
 $\langle \text{proof} \rangle$

lemma *qbs-prob-eq-1-implies-2* :
assumes *qbs-prob-eq* $p1 \ (p2 :: 'a \text{ qbs-prob-t})$
shows *qbs-prob-eq2* $p1 \ p2$
 $\langle \text{proof} \rangle$

lemma *qbs-prob-eq-1-implies-4* :
assumes *qbs-prob-eq* $p1 \ p2$
shows *qbs-prob-eq4* $p1 \ p2$
 $\langle \text{proof} \rangle$

lemma *qbs-prob-eq-4-implies-3* :
assumes *qbs-prob-eq4* $p1 \ p2$
shows *qbs-prob-eq3* $p1 \ p2$
 $\langle \text{proof} \rangle$

lemma *qbs-prob-eq-equiv12* :
 $\text{qbs-prob-eq} = \text{qbs-prob-eq2}$
 $\langle \text{proof} \rangle$

lemma *qbs-prob-eq-equiv13* :
 $\text{qbs-prob-eq} = \text{qbs-prob-eq3}$

<proof>

lemma *qbs-prob-eq-equiv14* :
qbs-prob-eq = *qbs-prob-eq4*
<proof>

lemma *qbs-prob-eq-equiv23* :
qbs-prob-eq2 = *qbs-prob-eq3*
<proof>

lemma *qbs-prob-eq-equiv24* :
qbs-prob-eq2 = *qbs-prob-eq4*
<proof>

lemma *qbs-prob-eq-equiv34* :
qbs-prob-eq3 = *qbs-prob-eq4*
<proof>

lemma *qbs-prob-eq-equiv31* :
qbs-prob-eq = *qbs-prob-eq3*
<proof>

lemma *qbs-prob-space-eq* :
assumes *qbs-prob-eq* (*X*, α , μ) (*Y*, β , ν)
shows *qbs-prob-space* (*X*, α , μ) = *qbs-prob-space* (*Y*, β , ν)
<proof>

lemma(**in** *pair-qbs-prob*) *qbs-prob-space-eq* :
assumes *Y* = *X*
and *distr* μ (*qbs-to-measure* *X*) α = *distr* ν (*qbs-to-measure* *X*) β
shows *qbs-prob-space* (*X*, α , μ) = *qbs-prob-space* (*Y*, β , ν)
<proof>

lemma(**in** *pair-qbs-prob*) *qbs-prob-space-eq2* :
assumes *Y* = *X*
and $\bigwedge f. f \in \text{qbs-to-measure } X \rightarrow_M \text{real-borel}$
 $\implies (\int x. f (\alpha x) \partial \mu) = (\int x. f (\beta x) \partial \nu)$
shows *qbs-prob-space* (*X*, α , μ) = *qbs-prob-space* (*Y*, β , ν)
<proof>

lemma(**in** *pair-qbs-prob*) *qbs-prob-space-eq3* :
assumes *Y* = *X*
and $\bigwedge f. f \in \text{qbs-to-measure } X \rightarrow_M \text{real-borel} \implies (\forall k \in \text{qbs-space } X. 0 \leq f k)$
 $\implies (\int x. f (\alpha x) \partial \mu) = (\int x. f (\beta x) \partial \nu)$
shows *qbs-prob-space* (*X*, α , μ) = *qbs-prob-space* (*Y*, β , ν)
<proof>

lemma(**in** *pair-qbs-prob*) *qbs-prob-space-eq4* :
assumes *Y* = *X*

and $\bigwedge f. f \in \text{qbs-to-measure } X \rightarrow_M \text{ennreal-borel}$
 $\implies (\int^{+x}. f (\alpha x) \partial \mu) = (\int^{+x}. f (\beta x) \partial \nu)$
shows $\text{qbs-prob-space } (X, \alpha, \mu) = \text{qbs-prob-space } (Y, \beta, \nu)$
 $\langle \text{proof} \rangle$

lemma(**in** *pair-qbs-prob*) *qbs-prob-space-eq-inverse*:
assumes $\text{qbs-prob-space } (X, \alpha, \mu) = \text{qbs-prob-space } (Y, \beta, \nu)$
shows $\text{qbs-prob-eq } (X, \alpha, \mu) (Y, \beta, \nu)$
and $\text{qbs-prob-eq2 } (X, \alpha, \mu) (Y, \beta, \nu)$
and $\text{qbs-prob-eq3 } (X, \alpha, \mu) (Y, \beta, \nu)$
and $\text{qbs-prob-eq4 } (X, \alpha, \mu) (Y, \beta, \nu)$
 $\langle \text{proof} \rangle$

lift-definition *qbs-prob-space-qbs* :: $'a \text{ qbs-prob-space} \Rightarrow 'a \text{ quasi-borel}$
is *fst* $\langle \text{proof} \rangle$

lemma(**in** *qbs-prob*) *qbs-prob-space-qbs-computation[simp]*:
 $\text{qbs-prob-space-qbs } (\text{qbs-prob-space } (X, \alpha, \mu)) = X$
 $\langle \text{proof} \rangle$

lemma *rep-qbs-prob-space'*:
assumes $\text{qbs-prob-space-qbs } s = X$
shows $\exists \alpha \mu. s = \text{qbs-prob-space } (X, \alpha, \mu) \wedge \text{qbs-prob } X \alpha \mu$
 $\langle \text{proof} \rangle$

lift-definition *qbs-prob-ennintegral* :: $['a \text{ qbs-prob-space}, 'a \Rightarrow \text{ennreal}] \Rightarrow \text{ennreal}$
is *qbs-prob-t-ennintegral*
 $\langle \text{proof} \rangle$

lift-definition *qbs-prob-integral* :: $['a \text{ qbs-prob-space}, 'a \Rightarrow \text{real}] \Rightarrow \text{real}$
is *qbs-prob-t-integral*
 $\langle \text{proof} \rangle$

syntax
 $\text{-qbs-prob-ennintegral} :: \text{pttrn} \Rightarrow \text{ennreal} \Rightarrow 'a \text{ qbs-prob-space} \Rightarrow \text{ennreal} (\int^+_Q ((2 \text{-./ -}) / \partial \text{-}) [60,61] 110)$

translations
 $\int^+_Q x. f \partial p \equiv \text{CONST } \text{qbs-prob-ennintegral } p (\lambda x. f)$

syntax
 $\text{-qbs-prob-integral} :: \text{pttrn} \Rightarrow \text{real} \Rightarrow 'a \text{ qbs-prob-space} \Rightarrow \text{real} (\int_Q ((2 \text{-./ -}) / \partial \text{-}) [60,61] 110)$

translations
 $\int_Q x. f \partial p \equiv \text{CONST } \text{qbs-prob-integral } p (\lambda x. f)$

We define the function $l_X \in L(P(X)) \rightarrow_M G(X)$.

lift-definition $qbs\text{-}prob\text{-}measure :: 'a\ qbs\text{-}prob\text{-}space \Rightarrow 'a\ measure$
is $qbs\text{-}prob\text{-}t\text{-}measure$
 $\langle proof \rangle$

declare $[[coercion\ qbs\text{-}prob\text{-}measure]]$

lemma(in $qbs\text{-}prob$) $qbs\text{-}prob\text{-}measure\text{-}computation[simp]$:
 $qbs\text{-}prob\text{-}measure\ (qbs\text{-}prob\text{-}space\ (X,\alpha,\mu)) = distr\ \mu\ (qbs\text{-}to\text{-}measure\ X)\ \alpha$
 $\langle proof \rangle$

definition $qbs\text{-}emeasure :: 'a\ qbs\text{-}prob\text{-}space \Rightarrow 'a\ set \Rightarrow ennreal$ **where**
 $qbs\text{-}emeasure\ s \equiv emeasure\ (qbs\text{-}prob\text{-}measure\ s)$

lemma(in $qbs\text{-}prob$) $qbs\text{-}emeasure\text{-}computation[simp]$:
assumes $U \in sets\ (qbs\text{-}to\text{-}measure\ X)$
shows $qbs\text{-}emeasure\ (qbs\text{-}prob\text{-}space\ (X,\alpha,\mu))\ U = emeasure\ \mu\ (\alpha - ' U)$
 $\langle proof \rangle$

definition $qbs\text{-}measure :: 'a\ qbs\text{-}prob\text{-}space \Rightarrow 'a\ set \Rightarrow real$ **where**
 $qbs\text{-}measure\ s \equiv measure\ (qbs\text{-}prob\text{-}measure\ s)$

interpretation $qbs\text{-}prob\text{-}measure\text{-}prob\text{-}space : prob\text{-}space\ qbs\text{-}prob\text{-}measure\ (s::'a\ qbs\text{-}prob\text{-}space)$ **for** s
 $\langle proof \rangle$

lemma $qbs\text{-}prob\text{-}measure\text{-}space$:
 $qbs\text{-}space\ (qbs\text{-}prob\text{-}space\text{-}qbs\ s) = space\ (qbs\text{-}prob\text{-}measure\ s)$
 $\langle proof \rangle$

lemma $qbs\text{-}prob\text{-}measure\text{-}sets[measurable\text{-}cong]$:
 $sets\ (qbs\text{-}to\text{-}measure\ (qbs\text{-}prob\text{-}space\text{-}qbs\ s)) = sets\ (qbs\text{-}prob\text{-}measure\ s)$
 $\langle proof \rangle$

lemma(in $qbs\text{-}prob$) $qbs\text{-}prob\text{-}ennintegral\text{-}def$:
assumes $f \in X \rightarrow_Q \mathbb{R}_{Q \geq 0}$
shows $qbs\text{-}prob\text{-}ennintegral\ (qbs\text{-}prob\text{-}space\ (X,\alpha,\mu))\ f = (\int^+ x. f\ (\alpha\ x)\ \partial\ \mu)$
 $\langle proof \rangle$

lemma(in $qbs\text{-}prob$) $qbs\text{-}prob\text{-}ennintegral\text{-}def2$:
assumes $f \in X \rightarrow_Q \mathbb{R}_{Q \geq 0}$
shows $qbs\text{-}prob\text{-}ennintegral\ (qbs\text{-}prob\text{-}space\ (X,\alpha,\mu))\ f = integral^N\ (distr\ \mu\ (qbs\text{-}to\text{-}measure\ X)\ \alpha)\ f$
 $\langle proof \rangle$

lemma (in $qbs\text{-}prob$) $qbs\text{-}prob\text{-}ennintegral\text{-}not\text{-}morphism$:
assumes $f \notin X \rightarrow_Q \mathbb{R}_{Q \geq 0}$

shows $qbs\text{-}prob\text{-}ennintegral (qbs\text{-}prob\text{-}space (X, \alpha, \mu)) f = 0$
 ⟨proof⟩

lemma $qbs\text{-}prob\text{-}ennintegral\text{-}def2$:

assumes $qbs\text{-}prob\text{-}space\text{-}qbs s = (X :: 'a\ quasi\text{-}borel)$
and $f \in X \rightarrow_Q \mathbb{R}_{Q \geq 0}$
shows $qbs\text{-}prob\text{-}ennintegral s f = integral^N (qbs\text{-}prob\text{-}measure s) f$
 ⟨proof⟩

lemma(in $qbs\text{-}prob$) $qbs\text{-}prob\text{-}integral\text{-}def$:

assumes $f \in X \rightarrow_Q real\text{-}quasi\text{-}borel$
shows $qbs\text{-}prob\text{-}integral (qbs\text{-}prob\text{-}space (X, \alpha, \mu)) f = (\int x. f (\alpha x) \partial \mu)$
 ⟨proof⟩

lemma(in $qbs\text{-}prob$) $qbs\text{-}prob\text{-}integral\text{-}def2$:

$qbs\text{-}prob\text{-}integral (qbs\text{-}prob\text{-}space (X, \alpha, \mu)) f = integral^L (distr \mu (qbs\text{-}to\text{-}measure X) \alpha) f$
 ⟨proof⟩

lemma $qbs\text{-}prob\text{-}integral\text{-}def2$:

$qbs\text{-}prob\text{-}integral (s :: 'a\ qbs\text{-}prob\text{-}space) f = integral^L (qbs\text{-}prob\text{-}measure s) f$
 ⟨proof⟩

definition $qbs\text{-}prob\text{-}var :: 'a\ qbs\text{-}prob\text{-}space \Rightarrow ('a \Rightarrow real) \Rightarrow real$ **where**
 $qbs\text{-}prob\text{-}var s f \equiv qbs\text{-}prob\text{-}integral s (\lambda x. (f x - qbs\text{-}prob\text{-}integral s f)^2)$

lemma(in $qbs\text{-}prob$) $qbs\text{-}prob\text{-}var\text{-}computation$:

assumes $f \in X \rightarrow_Q real\text{-}quasi\text{-}borel$
shows $qbs\text{-}prob\text{-}var (qbs\text{-}prob\text{-}space (X, \alpha, \mu)) f = (\int x. (f (\alpha x) - (\int x. f (\alpha x) \partial \mu))^2 \partial \mu)$
 ⟨proof⟩

lift-definition $qbs\text{-}integrable :: ['a\ qbs\text{-}prob\text{-}space, 'a \Rightarrow real] \Rightarrow bool$

is $qbs\text{-}prob\text{-}t\text{-}integrable$

⟨proof⟩

lemma(in $qbs\text{-}prob$) $qbs\text{-}integrable\text{-}def$:

$qbs\text{-}integrable (qbs\text{-}prob\text{-}space (X, \alpha, \mu)) f = (f \in X \rightarrow_Q \mathbb{R}_Q \wedge integrable \mu (f \circ \alpha))$
 ⟨proof⟩

lemma $qbs\text{-}integrable\text{-}morphism$:

assumes $qbs\text{-}prob\text{-}space\text{-}qbs s = X$

and $qbs\text{-}integrable s f$

shows $f \in X \rightarrow_Q \mathbb{R}_Q$

⟨proof⟩

lemma(in $qbs\text{-}prob$) $qbs\text{-}integrable\text{-}measurable[simp, measurable]$:

assumes $qbs\text{-}integrable (qbs\text{-}prob\text{-}space (X, \alpha, \mu)) f$

shows $f \in \text{qbs-to-measure } X \rightarrow_M \text{real-borel}$
 ⟨proof⟩

lemma *qbs-integrable-iff-integrable*:

$(\text{qbs-integrable } (s::'a \text{ qbs-prob-space}) f) = (\text{integrable } (\text{qbs-prob-measure } s) f)$
 ⟨proof⟩

lemma(in *qbs-prob*) *qbs-integrable-iff-integrable-distr*:

$\text{qbs-integrable } (\text{qbs-prob-space } (X, \alpha, \mu)) f = \text{integrable } (\text{distr } \mu (\text{qbs-to-measure } X) \alpha) f$
 ⟨proof⟩

lemma(in *qbs-prob*) *qbs-integrable-iff-integrable*:

assumes $f \in \text{qbs-to-measure } X \rightarrow_M \text{real-borel}$
shows $\text{qbs-integrable } (\text{qbs-prob-space } (X, \alpha, \mu)) f = \text{integrable } \mu (\lambda x. f (\alpha x))$
 ⟨proof⟩

lemma *qbs-integrable-if-integrable*:

assumes $\text{integrable } (\text{qbs-prob-measure } s) f$
shows $\text{qbs-integrable } (s::'a \text{ qbs-prob-space}) f$
 ⟨proof⟩

lemma *integrable-if-qbs-integrable*:

assumes $\text{qbs-integrable } (s::'a \text{ qbs-prob-space}) f$
shows $\text{integrable } (\text{qbs-prob-measure } s) f$
 ⟨proof⟩

lemma *qbs-integrable-iff-bounded*:

assumes $\text{qbs-prob-space-qbs } s = X$
shows $\text{qbs-integrable } s f \longleftrightarrow f \in X \rightarrow_Q \mathbb{R}_Q \wedge \text{qbs-prob-ennintegral } s (\lambda x. \text{ennreal } |f x|) < \infty$
 (is ?lhs = ?rhs)
 ⟨proof⟩

lemma *qbs-integrable-cong*:

assumes $\text{qbs-prob-space-qbs } s = X$
 $\bigwedge x. x \in \text{qbs-space } X \implies f x = g x$
and $\text{qbs-integrable } s f$
shows $\text{qbs-integrable } s g$
 ⟨proof⟩

lemma *qbs-integrable-const[simp]*:

$\text{qbs-integrable } s (\lambda x. c)$
 ⟨proof⟩

lemma *qbs-integrable-add[simp]*:

assumes $\text{qbs-integrable } s f$
and $\text{qbs-integrable } s g$
shows $\text{qbs-integrable } s (\lambda x. f x + g x)$

$\langle proof \rangle$

lemma *qbs-integrable-diff[simp]*:
 assumes *qbs-integrable s f*
 and *qbs-integrable s g*
 shows *qbs-integrable s ($\lambda x. f x - g x$)*
 $\langle proof \rangle$

lemma *qbs-integrable-mult-iff[simp]*:
 $(qbs-integrable s (\lambda x. c * f x)) = (c = 0 \vee qbs-integrable s f)$
 $\langle proof \rangle$

lemma *qbs-integrable-mult[simp]*:
 assumes *qbs-integrable s f*
 shows *qbs-integrable s ($\lambda x. c * f x$)*
 $\langle proof \rangle$

lemma *qbs-integrable-abs[simp]*:
 assumes *qbs-integrable s f*
 shows *qbs-integrable s ($\lambda x. |f x|$)*
 $\langle proof \rangle$

lemma *qbs-integrable-sq[simp]*:
 assumes *qbs-integrable s f*
 and *qbs-integrable s ($\lambda x. (f x)^2$)*
 shows *qbs-integrable s ($\lambda x. (f x - c)^2$)*
 $\langle proof \rangle$

lemma *qbs-ennintegral-eq-qbs-integral*:
 assumes *qbs-prob-space-qbs s = X*
 qbs-integrable s f
 and $\bigwedge x. x \in qbs-space X \implies 0 \leq f x$
 shows *qbs-prob-ennintegral s ($\lambda x. ennreal (f x)$) = ennreal (qbs-prob-integral s f)*
 $\langle proof \rangle$

lemma *qbs-prob-ennintegral-cong*:
 assumes *qbs-prob-space-qbs s = X*
 and $\bigwedge x. x \in qbs-space X \implies f x = g x$
 shows *qbs-prob-ennintegral s f = qbs-prob-ennintegral s g*
 $\langle proof \rangle$

lemma *qbs-prob-ennintegral-const*:
qbs-prob-ennintegral (s::'a qbs-prob-space) ($\lambda x. c$) = c
 $\langle proof \rangle$

lemma *qbs-prob-ennintegral-add*:
 assumes *qbs-prob-space-qbs s = X*

$f \in (X :: 'a \text{ quasi-borel}) \rightarrow_Q \mathbb{R}_{Q \geq 0}$
and $g \in X \rightarrow_Q \mathbb{R}_{Q \geq 0}$
shows $\text{qbs-prob-ennintegral } s (\lambda x. f x + g x) = \text{qbs-prob-ennintegral } s f + \text{qbs-prob-ennintegral } s g$
 $\langle \text{proof} \rangle$

lemma *qbs-prob-ennintegral-cmult*:
assumes $\text{qbs-prob-space-qbs } s = X$
and $f \in X \rightarrow_Q \mathbb{R}_{Q \geq 0}$
shows $\text{qbs-prob-ennintegral } s (\lambda x. c * f x) = c * \text{qbs-prob-ennintegral } s f$
 $\langle \text{proof} \rangle$

lemma *qbs-prob-ennintegral-cmult-noninfty*:
assumes $c \neq \infty$
shows $\text{qbs-prob-ennintegral } s (\lambda x. c * f x) = c * \text{qbs-prob-ennintegral } s f$
 $\langle \text{proof} \rangle$

lemma *qbs-prob-integral-cong*:
assumes $\text{qbs-prob-space-qbs } s = X$
and $\bigwedge x. x \in \text{qbs-space } X \implies f x = g x$
shows $\text{qbs-prob-integral } s f = \text{qbs-prob-integral } s g$
 $\langle \text{proof} \rangle$

lemma *qbs-prob-integral-nonneg*:
assumes $\text{qbs-prob-space-qbs } s = X$
and $\bigwedge x. x \in \text{qbs-space } X \implies 0 \leq f x$
shows $0 \leq \text{qbs-prob-integral } s f$
 $\langle \text{proof} \rangle$

lemma *qbs-prob-integral-mono*:
assumes $\text{qbs-prob-space-qbs } s = X$
 $\text{qbs-integrable } (s :: 'a \text{ qbs-prob-space}) f$
 $\text{qbs-integrable } s g$
and $\bigwedge x. x \in \text{qbs-space } X \implies f x \leq g x$
shows $\text{qbs-prob-integral } s f \leq \text{qbs-prob-integral } s g$
 $\langle \text{proof} \rangle$

lemma *qbs-prob-integral-const*:
 $\text{qbs-prob-integral } (s :: 'a \text{ qbs-prob-space}) (\lambda x. c) = c$
 $\langle \text{proof} \rangle$

lemma *qbs-prob-integral-add*:
assumes $\text{qbs-integrable } (s :: 'a \text{ qbs-prob-space}) f$
and $\text{qbs-integrable } s g$
shows $\text{qbs-prob-integral } s (\lambda x. f x + g x) = \text{qbs-prob-integral } s f + \text{qbs-prob-integral } s g$
 $\langle \text{proof} \rangle$

lemma *qbs-prob-integral-diff*:

assumes *qbs-integrable* (*s*::'a *qbs-prob-space*) *f*
and *qbs-integrable* *s g*
shows $qbs\text{-prob-integral } s (\lambda x. f x - g x) = qbs\text{-prob-integral } s f - qbs\text{-prob-integral } s g$
<proof>

lemma *qbs-prob-integral-cmult*:
 $qbs\text{-prob-integral } s (\lambda x. c * f x) = c * qbs\text{-prob-integral } s f$
<proof>

lemma *real-qbs-prob-integral-def*:
assumes *qbs-integrable* (*s*::'a *qbs-prob-space*) *f*
shows $qbs\text{-prob-integral } s f = enn2real (qbs\text{-prob-ennintegral } s (\lambda x. ennreal (f x))) - enn2real (qbs\text{-prob-ennintegral } s (\lambda x. ennreal (- f x)))$
<proof>

lemma *qbs-prob-var-eq*:
assumes *qbs-integrable* (*s*::'a *qbs-prob-space*) *f*
and *qbs-integrable* *s* ($\lambda x. (f x)^2$)
shows $qbs\text{-prob-var } s f = qbs\text{-prob-integral } s (\lambda x. (f x)^2) - (qbs\text{-prob-integral } s f)^2$
<proof>

lemma *qbs-prob-var-affine*:
assumes *qbs-integrable* *s f*
shows $qbs\text{-prob-var } s (\lambda x. a * f x + b) = a^2 * qbs\text{-prob-var } s f$
(is ?lhs = ?rhs)
<proof>

lemma *qbs-prob-integral-Markov-inequality*:
assumes *qbs-prob-space-qbs* *s = X*
and *qbs-integrable* *s f*
 $\bigwedge x. x \in qbs\text{-space } X \implies 0 \leq f x$
and $0 < c$
shows $qbs\text{-emeasure } s \{x \in qbs\text{-space } X. c \leq f x\} \leq ennreal (1/c * qbs\text{-prob-integral } s f)$
<proof>

lemma *qbs-prob-integral-Markov-inequality'*:
assumes *qbs-prob-space-qbs* *s = X*
qbs-integrable *s f*
 $\bigwedge x. x \in qbs\text{-space } (qbs\text{-prob-space-qbs } s) \implies 0 \leq f x$
and $0 < c$
shows $qbs\text{-measure } s \{x \in qbs\text{-space } (qbs\text{-prob-space-qbs } s). c \leq f x\} \leq (1/c * qbs\text{-prob-integral } s f)$
<proof>

lemma *qbs-prob-integral-Markov-inequality-abs*:
assumes *qbs-prob-space-qbs* *s = X*

$qbs\text{-integrable } s f$
and $0 < c$
shows $qbs\text{-emeasure } s \{x \in qbs\text{-space } X. c \leq |f x|\} \leq \text{ennreal } (1/c * qbs\text{-prob-integral } s (\lambda x. |f x|))$
 $\langle \text{proof} \rangle$

lemma *qbs-prob-integral-Markov-inequality-abs'*:
assumes $qbs\text{-prob-space-qbs } s = X$
 $qbs\text{-integrable } s f$
and $0 < c$
shows $qbs\text{-measure } s \{x \in qbs\text{-space } X. c \leq |f x|\} \leq (1/c * qbs\text{-prob-integral } s (\lambda x. |f x|))$
 $\langle \text{proof} \rangle$

lemma *qbs-prob-integral-real-Markov-inequality*:
assumes $qbs\text{-prob-space-qbs } s = \mathbb{R}_Q$
 $qbs\text{-integrable } s f$
and $0 < c$
shows $qbs\text{-emeasure } s \{r. c \leq |f r|\} \leq \text{ennreal } (1/c * qbs\text{-prob-integral } s (\lambda x. |f x|))$
 $\langle \text{proof} \rangle$

lemma *qbs-prob-integral-real-Markov-inequality'*:
assumes $qbs\text{-prob-space-qbs } s = \mathbb{R}_Q$
 $qbs\text{-integrable } s f$
and $0 < c$
shows $qbs\text{-measure } s \{r. c \leq |f r|\} \leq 1/c * qbs\text{-prob-integral } s (\lambda x. |f x|)$
 $\langle \text{proof} \rangle$

lemma *qbs-prob-integral-Chebyshev-inequality*:
assumes $qbs\text{-prob-space-qbs } s = X$
 $qbs\text{-integrable } s f$
 $qbs\text{-integrable } s (\lambda x. (f x)^2)$
and $0 < b$
shows $qbs\text{-measure } s \{x \in qbs\text{-space } X. b \leq |f x - qbs\text{-prob-integral } s f|\} \leq 1 / b^2 * qbs\text{-prob-var } s f$
 $\langle \text{proof} \rangle$

end

3.2 The Probability Monad

theory *Monad-QuasiBorel*
imports *Probability-Space-QuasiBorel*
begin

3.2.1 The Probability Monad P

definition *monadP-qbs-Px* :: 'a quasi-borel \Rightarrow 'a qbs-prob-space set **where**
 $monadP\text{-qbs-Px } X \equiv \{s. qbs\text{-prob-space-qbs } s = X\}$

locale *in-Px* =
fixes $X :: 'a \text{ quasi-borel}$ **and** $s :: 'a \text{ qbs-prob-space}$
assumes $\text{in-Px}: s \in \text{monadP-qbs-Px } X$
begin

lemma *qbs-prob-space-X[simp]*:
 $\text{qbs-prob-space-qbs } s = X$
 $\langle \text{proof} \rangle$

end

locale *in-MPx* =
fixes $X :: 'a \text{ quasi-borel}$ **and** $\beta :: \text{real} \Rightarrow 'a \text{ qbs-prob-space}$
assumes $\text{ex}: \exists \alpha \in \text{qbs-Mx } X. \exists g \in \text{real-borel} \rightarrow_M \text{prob-algebra real-borel}.$
 $\forall r. \beta \ r = \text{qbs-prob-space } (X, \alpha, g \ r)$
begin

lemma *rep-inMPx*:
 $\exists \alpha \ g. \alpha \in \text{qbs-Mx } X \wedge g \in \text{real-borel} \rightarrow_M \text{prob-algebra real-borel} \wedge$
 $\beta = (\lambda r. \text{qbs-prob-space } (X, \alpha, g \ r))$
 $\langle \text{proof} \rangle$

end

definition *monadP-qbs-MPx* :: $'a \text{ quasi-borel} \Rightarrow (\text{real} \Rightarrow 'a \text{ qbs-prob-space}) \text{ set}$
where
 $\text{monadP-qbs-MPx } X \equiv \{\beta. \text{in-MPx } X \ \beta\}$

definition *monadP-qbs* :: $'a \text{ quasi-borel} \Rightarrow 'a \text{ qbs-prob-space quasi-borel}$ **where**
 $\text{monadP-qbs } X \equiv \text{Abs-quasi-borel } (\text{monadP-qbs-Px } X, \text{monadP-qbs-MPx } X)$

lemma(**in** *qbs-prob*) *qbs-prob-space-in-Px*:
 $\text{qbs-prob-space } (X, \alpha, \mu) \in \text{monadP-qbs-Px } X$
 $\langle \text{proof} \rangle$

lemma *rep-monadP-qbs-Px*:
assumes $s \in \text{monadP-qbs-Px } X$
shows $\exists \alpha \ \mu. s = \text{qbs-prob-space } (X, \alpha, \mu) \wedge \text{qbs-prob } X \ \alpha \ \mu$
 $\langle \text{proof} \rangle$

lemma *rep-monadP-qbs-MPx*:
assumes $\beta \in \text{monadP-qbs-MPx } X$
shows $\exists \alpha \ g. \alpha \in \text{qbs-Mx } X \wedge g \in \text{real-borel} \rightarrow_M \text{prob-algebra real-borel} \wedge$
 $\beta = (\lambda r. \text{qbs-prob-space } (X, \alpha, g \ r))$
 $\langle \text{proof} \rangle$

lemma *qbs-prob-MPx*:
assumes $\alpha \in \text{qbs-Mx } X$

and $g \in \text{real-borel} \rightarrow_M \text{prob-algebra real-borel}$
shows $\text{qbs-prob } X \alpha (g \ r)$
 $\langle \text{proof} \rangle$

lemma $\text{monadP-qbs-f[simp]}$: $\text{monadP-qbs-MPx } X \subseteq \text{UNIV} \rightarrow \text{monadP-qbs-Px } X$
 $\langle \text{proof} \rangle$

lemma $\text{monadP-qbs-closed1}$: $\text{qbs-closed1 } (\text{monadP-qbs-MPx } X)$
 $\langle \text{proof} \rangle$

lemma $\text{monadP-qbs-closed2}$: $\text{qbs-closed2 } (\text{monadP-qbs-Px } X) (\text{monadP-qbs-MPx } X)$
 $\langle \text{proof} \rangle$

lemma $\text{monadP-qbs-closed3}$: $\text{qbs-closed3 } (\text{monadP-qbs-MPx } (X :: 'a \text{ quasi-borel}))$
 $\langle \text{proof} \rangle$

lemma $\text{monadP-qbs-correct}$: $\text{Rep-quasi-borel } (\text{monadP-qbs } X) = (\text{monadP-qbs-Px } X, \text{monadP-qbs-MPx } X)$
 $\langle \text{proof} \rangle$

lemma $\text{monadP-qbs-space[simp]}$: $\text{qbs-space } (\text{monadP-qbs } X) = \text{monadP-qbs-Px } X$
 $\langle \text{proof} \rangle$

lemma $\text{monadP-qbs-Mx[simp]}$: $\text{qbs-Mx } (\text{monadP-qbs } X) = \text{monadP-qbs-MPx } X$
 $\langle \text{proof} \rangle$

lemma $\text{monadP-qbs-empty-iff}$:
 $\text{qbs-space } X = \{\} \longleftrightarrow \text{qbs-space } (\text{monadP-qbs } X) = \{\}$
 $\langle \text{proof} \rangle$

If $\beta \in \text{MPx}$, there exists $X \alpha g$ s.t. $\beta \ r = [X, \alpha, g \ r]$. We define a function which picks $X \alpha g$ from $\beta \in \text{MPx}$.

definition $\text{rep-monadP-qbs-MPx} :: (\text{real} \Rightarrow 'a \text{ qbs-prob-space}) \Rightarrow 'a \text{ quasi-borel} \times (\text{real} \Rightarrow 'a) \times (\text{real} \Rightarrow \text{real measure})$ **where**
 $\text{rep-monadP-qbs-MPx } \beta \equiv \text{let } X = \text{qbs-prob-space-qbs } (\beta \ \text{undefined});$
 $\alpha g = (\text{SOME } k. (\text{fst } k) \in \text{qbs-Mx } X \wedge (\text{snd } k) \in \text{real-borel}$
 $\rightarrow_M \text{prob-algebra real-borel}$
 $\wedge \beta = (\lambda r. \text{qbs-prob-space } (X, \text{fst } k, \text{snd } k \ r)))$
 $\text{in } (X, \alpha g)$

lemma $\text{qbs-prob-measure-measurable[measurable]}$:
 $\text{qbs-prob-measure} \in \text{qbs-to-measure } (\text{monadP-qbs } (X :: 'a \text{ quasi-borel})) \rightarrow_M \text{prob-algebra}$
 $(\text{qbs-to-measure } X)$
 $\langle \text{proof} \rangle$

lemma qbs-l-inj :
 $\text{inj-on qbs-prob-measure } (\text{monadP-qbs-Px } X)$
 $\langle \text{proof} \rangle$

lemma *qbs-prob-measure-measurable*'[*measurable*]:
qbs-prob-measure \in *qbs-to-measure* (*monadP-qbs* ($X :: 'a$ *quasi-borel*)) \rightarrow_M *sub-prob-algebra* (*qbs-to-measure* X)
 ⟨*proof*⟩

3.2.2 Return

definition *qbs-return* :: [$'a$ *quasi-borel*, $'a$] \Rightarrow $'a$ *qbs-prob-space* **where**
qbs-return X $x \equiv$ *qbs-prob-space* ($X, \lambda r. x, Eps$ *real-distribution*)

lemma(*in real-distribution*) *qbs-return-qbs-prob*:
assumes $x \in$ *qbs-space* X
shows *qbs-prob* X ($\lambda r. x$) M
 ⟨*proof*⟩

lemma(*in real-distribution*) *qbs-return-computation* :
assumes $x \in$ *qbs-space* X
shows *qbs-return* X $x =$ *qbs-prob-space* ($X, \lambda r. x, M$)
 ⟨*proof*⟩

lemma *qbs-return-morphism*:
qbs-return $X \in X \rightarrow_Q$ *monadP-qbs* X
 ⟨*proof*⟩

lemma *qbs-return-morphism'*:
assumes $f \in X \rightarrow_Q Y$
shows ($\lambda x. \text{qbs-return } Y (f x)$) $\in X \rightarrow_Q$ *monadP-qbs* Y
 ⟨*proof*⟩

3.2.3 Bind

definition *qbs-bind* :: $'a$ *qbs-prob-space* \Rightarrow ($'a \Rightarrow 'b$ *qbs-prob-space*) \Rightarrow $'b$ *qbs-prob-space*
where
qbs-bind s $f \equiv$ (*let* (*qbsx*, α, μ) = *rep-qbs-prob-space* s ;
 (*qbsy*, β, g) = *rep-monadP-qbs-MPx* ($f \circ \alpha$)
 in *qbs-prob-space* (*qbsy*, $\beta, \mu \ggg g$))

adhoc-overloading *Monad-Syntax.bind* *qbs-bind*

lemma(*in qbs-prob*) *qbs-bind-computation*:
assumes $s =$ *qbs-prob-space* (X, α, μ)
 $f \in X \rightarrow_Q$ *monadP-qbs* Y
 $\beta \in$ *qbs-Mx* Y
and [*measurable*]: $g \in$ *real-borel* \rightarrow_M *prob-algebra real-borel*
 and ($f \circ \alpha$) = ($\lambda r. \text{qbs-prob-space } (Y, \beta, g r)$)
shows *qbs-prob* Y β ($\mu \ggg g$)
 $s \ggg f =$ *qbs-prob-space* ($Y, \beta, \mu \ggg g$)
 ⟨*proof*⟩

lemma *qbs-bind-morphism'*:

assumes $f \in X \rightarrow_Q \text{monadP-qbs } Y$

shows $(\lambda x. x \gg\equiv f) \in \text{monadP-qbs } X \rightarrow_Q \text{monadP-qbs } Y$

<proof>

lemma *qbs-return-comp*:

assumes $\alpha \in \text{qbs-Mx } X$

shows $(\text{qbs-return } X \circ \alpha) = (\lambda r. \text{qbs-prob-space } (X, \alpha, \text{return real-borel } r))$

<proof>

lemma *qbs-bind-return'*:

assumes $x \in \text{monadP-qbs-Px } X$

shows $x \gg\equiv \text{qbs-return } X = x$

<proof>

lemma *qbs-bind-return*:

assumes $f \in X \rightarrow_Q \text{monadP-qbs } Y$

and $x \in \text{qbs-space } X$

shows $\text{qbs-return } X x \gg\equiv f = f x$

<proof>

lemma *qbs-bind-assoc*:

assumes $s \in \text{monadP-qbs-Px } X$

$f \in X \rightarrow_Q \text{monadP-qbs } Y$

and $g \in Y \rightarrow_Q \text{monadP-qbs } Z$

shows $s \gg\equiv (\lambda x. f x \gg\equiv g) = (s \gg\equiv f) \gg\equiv g$

<proof>

lemma *qbs-bind-cong*:

assumes $s \in \text{monadP-qbs-Px } X$

$\bigwedge x. x \in \text{qbs-space } X \implies f x = g x$

and $f \in X \rightarrow_Q \text{monadP-qbs } Y$

shows $s \gg\equiv f = s \gg\equiv g$

<proof>

3.2.4 The Functorial Action $P(f)$

definition *monadP-qbs-Pf* :: [*'a quasi-borel, 'b quasi-borel, 'a \Rightarrow 'b, 'a qbs-prob-space*]

\Rightarrow *'b qbs-prob-space* **where**

$\text{monadP-qbs-Pf } - Y f s x \equiv s x \gg\equiv \text{qbs-return } Y \circ f$

lemma *monadP-qbs-Pf-morphism*:

assumes $f \in X \rightarrow_Q Y$

shows $\text{monadP-qbs-Pf } X Y f \in \text{monadP-qbs } X \rightarrow_Q \text{monadP-qbs } Y$

<proof>

lemma(in *qbs-prob*) *monadP-qbs-Pf-computation*:

assumes $s = \text{qbs-prob-space } (X, \alpha, \mu)$

and $f \in X \rightarrow_Q Y$

shows $qbs\text{-}prob\ Y\ (f \circ \alpha)\ \mu$
and $monadP\text{-}qbs\text{-}Pf\ X\ Y\ f\ s = qbs\text{-}prob\text{-}space\ (Y, f \circ \alpha, \mu)$
 $\langle proof \rangle$

We show that P is a functor i.e. P preserves identity and composition.

lemma $monadP\text{-}qbs\text{-}Pf\text{-}id$:
assumes $s \in monadP\text{-}qbs\text{-}Px\ X$
shows $monadP\text{-}qbs\text{-}Pf\ X\ X\ id\ s = s$
 $\langle proof \rangle$

lemma $monadP\text{-}qbs\text{-}Pf\text{-}comp$:
assumes $s \in monadP\text{-}qbs\text{-}Px\ X$
 $f \in X \rightarrow_Q Y$
and $g \in Y \rightarrow_Q Z$
shows $((monadP\text{-}qbs\text{-}Pf\ Y\ Z\ g) \circ (monadP\text{-}qbs\text{-}Pf\ X\ Y\ f))\ s = monadP\text{-}qbs\text{-}Pf\ X\ Z\ (g \circ f)\ s$
 $\langle proof \rangle$

3.2.5 Join

definition $qbs\text{-}join :: 'a\ qbs\text{-}prob\text{-}space\ qbs\text{-}prob\text{-}space \Rightarrow 'a\ qbs\text{-}prob\text{-}space$ **where**
 $qbs\text{-}join \equiv (\lambda sst.\ sst \ggg id)$

lemma $qbs\text{-}join\text{-}morphism$:
 $qbs\text{-}join \in monadP\text{-}qbs\ (monadP\text{-}qbs\ X) \rightarrow_Q monadP\text{-}qbs\ X$
 $\langle proof \rangle$

lemma $qbs\text{-}join\text{-}computation$:
assumes $qbs\text{-}prob\ (monadP\text{-}qbs\ X)\ \beta\ \mu$
 $ssx = qbs\text{-}prob\text{-}space\ (monadP\text{-}qbs\ X, \beta, \mu)$
 $\alpha \in qbs\text{-}Mx\ X$
 $g \in real\text{-}borel \rightarrow_M prob\text{-}algebra\ real\text{-}borel$
and $\beta = (\lambda r.\ qbs\text{-}prob\text{-}space\ (X, \alpha, g\ r))$
shows $qbs\text{-}prob\ X\ \alpha\ (\mu \ggg g)\ qbs\text{-}join\ ssx = qbs\text{-}prob\text{-}space\ (X, \alpha, \mu \ggg g)$
 $\langle proof \rangle$

3.2.6 Strength

definition $qbs\text{-}strength :: ['a\ quasi\text{-}borel, 'b\ quasi\text{-}borel, 'a \times 'b\ qbs\text{-}prob\text{-}space] \Rightarrow ('a \times 'b)\ qbs\text{-}prob\text{-}space$ **where**
 $qbs\text{-}strength\ W\ X = (\lambda (w, sx).\ let\ (-, \alpha, \mu) = rep\text{-}qbs\text{-}prob\text{-}space\ sx$
 $in\ qbs\text{-}prob\text{-}space\ (W \otimes_Q X, \lambda r.\ (w, \alpha\ r), \mu))$

lemma(in $qbs\text{-}prob$) $qbs\text{-}strength\text{-}computation$:
assumes $w \in qbs\text{-}space\ W$
and $sx = qbs\text{-}prob\text{-}space\ (X, \alpha, \mu)$
shows $qbs\text{-}prob\ (W \otimes_Q X)\ (\lambda r.\ (w, \alpha\ r))\ \mu$
 $qbs\text{-}strength\ W\ X\ (w, sx) = qbs\text{-}prob\text{-}space\ (W \otimes_Q X, \lambda r.\ (w, \alpha\ r), \mu)$
 $\langle proof \rangle$

lemma *qbs-strength-natural*:

assumes $f \in X \rightarrow_Q X'$
 $g \in Y \rightarrow_Q Y'$
 $x \in \text{qbs-space } X$
and $sy \in \text{monadP-qbs-Px } Y$
shows $(\text{monadP-qbs-Pf } (X \otimes_Q Y) (X' \otimes_Q Y') (\text{map-prod } f \ g) \circ \text{qbs-strength } X \ Y) (x, sy) = (\text{qbs-strength } X' \ Y' \circ \text{map-prod } f \ (\text{monadP-qbs-Pf } Y \ Y' \ g)) (x, sy)$
(is ?lhs = ?rhs)
 $\langle \text{proof} \rangle$

lemma *qbs-strength-ab-r*:

assumes $\alpha \in \text{qbs-Mx } X$
 $\beta \in \text{monadP-qbs-MPx } Y$
 $\gamma \in \text{qbs-Mx } Y$
and $[\text{measurable}]: g \in \text{real-borel} \rightarrow_M \text{prob-algebra real-borel}$
and $\beta = (\lambda r. \text{qbs-prob-space } (Y, \gamma, g \ r))$
shows $\text{qbs-prob } (X \otimes_Q Y) (\text{map-prod } \alpha \ \gamma \circ \text{real-real.g}) (\text{distr } (\text{return real-borel } r \otimes_M g \ r) \text{ real-borel real-real.f})$
 $\text{qbs-strength } X \ Y (\alpha \ r, \beta \ r) = \text{qbs-prob-space } (X \otimes_Q Y, \text{map-prod } \alpha \ \gamma \circ \text{real-real.g}, \text{distr } (\text{return real-borel } r \otimes_M g \ r) \text{ real-borel real-real.f})$
 $\langle \text{proof} \rangle$

lemma *qbs-strength-morphism*:

$\text{qbs-strength } X \ Y \in X \otimes_Q \text{monadP-qbs } Y \rightarrow_Q \text{monadP-qbs } (X \otimes_Q Y)$
 $\langle \text{proof} \rangle$

lemma *qbs-bind-morphism''*:

$(\lambda(f, x). x \ggg f) \in \text{exp-qbs } X (\text{monadP-qbs } Y) \otimes_Q (\text{monadP-qbs } X) \rightarrow_Q (\text{monadP-qbs } Y)$
 $\langle \text{proof} \rangle$

lemma *qbs-bind-morphism'''*:

$(\lambda f x. x \ggg f) \in \text{exp-qbs } X (\text{monadP-qbs } Y) \rightarrow_Q \text{exp-qbs } (\text{monadP-qbs } X)$
 $(\text{monadP-qbs } Y)$
 $\langle \text{proof} \rangle$

lemma *qbs-bind-morphism*:

assumes $f \in X \rightarrow_Q \text{monadP-qbs } Y$
and $g \in X \rightarrow_Q \text{exp-qbs } Y (\text{monadP-qbs } Z)$
shows $(\lambda x. f \ x \ggg g \ x) \in X \rightarrow_Q \text{monadP-qbs } Z$
 $\langle \text{proof} \rangle$

lemma *qbs-bind-morphism''''*:

assumes $x \in \text{monadP-qbs-Px } X$
shows $(\lambda f. x \ggg f) \in \text{exp-qbs } X (\text{monadP-qbs } Y) \rightarrow_Q \text{monadP-qbs } Y$
 $\langle \text{proof} \rangle$

lemma *qbs-strength-law1*:

assumes $x \in \text{qbs-space } (\text{unit-quasi-borel } \otimes_Q \text{ monadP-qbs } X)$
shows $\text{snd } x = (\text{monadP-qbs-Pf } (\text{unit-quasi-borel } \otimes_Q X) X \text{ snd } \circ \text{qbs-strength } \text{unit-quasi-borel } X) x$
 $\langle \text{proof} \rangle$

lemma *qbs-strength-law2*:

assumes $x \in \text{qbs-space } ((X \otimes_Q Y) \otimes_Q \text{ monadP-qbs } Z)$
shows $(\text{qbs-strength } X (Y \otimes_Q Z) \circ (\text{map-prod id } (\text{qbs-strength } Y Z)) \circ (\lambda((x,y),z). (x,(y,z)))) x =$
 $(\text{monadP-qbs-Pf } ((X \otimes_Q Y) \otimes_Q Z) (X \otimes_Q (Y \otimes_Q Z)) (\lambda((x,y),z). (x,(y,z))) \circ \text{qbs-strength } (X \otimes_Q Y) Z) x$
(is ?lhs = ?rhs)
 $\langle \text{proof} \rangle$

lemma *qbs-strength-law3*:

assumes $x \in \text{qbs-space } (X \otimes_Q Y)$
shows $\text{qbs-return } (X \otimes_Q Y) x = (\text{qbs-strength } X Y \circ (\text{map-prod id } (\text{qbs-return } Y))) x$
 $\langle \text{proof} \rangle$

lemma *qbs-strength-law4*:

assumes $x \in \text{qbs-space } (X \otimes_Q \text{ monadP-qbs } (\text{monadP-qbs } Y))$
shows $(\text{qbs-strength } X Y \circ \text{map-prod id } \text{qbs-join}) x = (\text{qbs-join } \circ \text{monadP-qbs-Pf } (X \otimes_Q \text{ monadP-qbs } Y) (\text{monadP-qbs } (X \otimes_Q Y)) (\text{qbs-strength } X Y) \circ \text{qbs-strength } X (\text{monadP-qbs } Y)) x$
(is ?lhs = ?rhs)
 $\langle \text{proof} \rangle$

lemma *qbs-return-Mxpair*:

assumes $\alpha \in \text{qbs-Mx } X$
and $\beta \in \text{qbs-Mx } Y$
shows $\text{qbs-return } (X \otimes_Q Y) (\alpha r, \beta k) = \text{qbs-prob-space } (X \otimes_Q Y, \text{map-prod } \alpha \beta \circ \text{real-real.g}, \text{distr } (\text{return real-borel } r \otimes_M \text{return real-borel } k) \text{ real-borel real-real.f})$
 $\text{qbs-prob } (X \otimes_Q Y) (\text{map-prod } \alpha \beta \circ \text{real-real.g}) (\text{distr } (\text{return real-borel } r \otimes_M \text{return real-borel } k) \text{ real-borel real-real.f})$
 $\langle \text{proof} \rangle$

lemma *pair-return-return*:

assumes $l \in \text{space } M$
and $r \in \text{space } N$
shows $\text{return } M l \otimes_M \text{return } N r = \text{return } (M \otimes_M N) (l,r)$
 $\langle \text{proof} \rangle$

lemma *bind-bind-return-distr*:

assumes *real-distribution* μ
and *real-distribution* ν

shows $\mu \gg (\lambda r. \nu \gg (\lambda l. \text{distr} (\text{return real-borel } r \otimes_M \text{return real-borel } l) \text{ real-borel real-real.f}))$
 $= \text{distr} (\mu \otimes_M \nu) \text{ real-borel real-real.f}$
(is ?lhs = ?rhs)
 <proof>

lemma(in pair-qbs-probs) qbs-bind-return-qp:

shows $\text{qbs-prob-space} (Y, \beta, \nu) \gg (\lambda y. \text{qbs-prob-space} (X, \alpha, \mu) \gg (\lambda x. \text{qbs-return} (X \otimes_Q Y) (x,y))) = \text{qbs-prob-space} (X \otimes_Q Y, \text{map-prod } \alpha \beta \circ \text{real-real.g}, \text{distr} (\mu \otimes_M \nu) \text{ real-borel real-real.f})$
 $\text{qbs-prob} (X \otimes_Q Y) (\text{map-prod } \alpha \beta \circ \text{real-real.g}) (\text{distr} (\mu \otimes_M \nu) \text{ real-borel real-real.f})$
 <proof>

lemma(in pair-qbs-probs) qbs-bind-return-pq:

shows $\text{qbs-prob-space} (X, \alpha, \mu) \gg (\lambda x. \text{qbs-prob-space} (Y, \beta, \nu) \gg (\lambda y. \text{qbs-return} (X \otimes_Q Y) (x,y))) = \text{qbs-prob-space} (X \otimes_Q Y, \text{map-prod } \alpha \beta \circ \text{real-real.g}, \text{distr} (\mu \otimes_M \nu) \text{ real-borel real-real.f})$
 $\text{qbs-prob} (X \otimes_Q Y) (\text{map-prod } \alpha \beta \circ \text{real-real.g}) (\text{distr} (\mu \otimes_M \nu) \text{ real-borel real-real.f})$
 <proof>

lemma qbs-bind-return-rotate:

assumes $p \in \text{monadP-qbs-Px } X$
and $q \in \text{monadP-qbs-Px } Y$
shows $q \gg (\lambda y. p \gg (\lambda x. \text{qbs-return} (X \otimes_Q Y) (x,y))) = p \gg (\lambda x. q \gg (\lambda y. \text{qbs-return} (X \otimes_Q Y) (x,y)))$
 <proof>

lemma qbs-pair-bind-return1:

assumes $f \in X \otimes_Q Y \rightarrow_Q \text{monadP-qbs } Z$
 $p \in \text{monadP-qbs-Px } X$
and $q \in \text{monadP-qbs-Px } Y$
shows $q \gg (\lambda y. p \gg (\lambda x. f (x,y))) = (q \gg (\lambda y. p \gg (\lambda x. \text{qbs-return} (X \otimes_Q Y) (x,y)))) \gg f$
(is ?lhs = ?rhs)
 <proof>

lemma qbs-pair-bind-return2:

assumes $f \in X \otimes_Q Y \rightarrow_Q \text{monadP-qbs } Z$
 $p \in \text{monadP-qbs-Px } X$
and $q \in \text{monadP-qbs-Px } Y$
shows $p \gg (\lambda x. q \gg (\lambda y. f (x,y))) = (p \gg (\lambda x. q \gg (\lambda y. \text{qbs-return} (X \otimes_Q Y) (x,y)))) \gg f$
(is ?lhs = ?rhs)
 <proof>

lemma qbs-bind-rotate:

assumes $f \in X \otimes_Q Y \rightarrow_Q \text{monadP-qbs } Z$

$p \in \text{monadP-qbs-Px } X$
and $q \in \text{monadP-qbs-Px } Y$
shows $q \gg (\lambda y. p \gg (\lambda x. f(x,y))) = p \gg (\lambda x. q \gg (\lambda y. f(x,y)))$
 ⟨proof⟩

lemma(in *pair-qbs-probs*) *qbs-bind-bind-return*:
assumes $f \in X \otimes_Q Y \rightarrow_Q Z$
shows $\text{qbs-prob } Z (f \circ (\text{map-prod } \alpha \beta \circ \text{real-real.g})) (\text{distr } (\mu \otimes_M \nu) \text{ real-borel } \text{real-real.f})$
and $\text{qbs-prob-space } (X, \alpha, \mu) \gg (\lambda x. \text{qbs-prob-space } (Y, \beta, \nu) \gg (\lambda y. \text{qbs-return } Z (f(x,y)))) = \text{qbs-prob-space } (Z, f \circ (\text{map-prod } \alpha \beta \circ \text{real-real.g}), \text{distr } (\mu \otimes_M \nu) \text{ real-borel } \text{real-real.f})$
 (is ?lhs = ?rhs)
 ⟨proof⟩

3.2.7 Properties of Return and Bind

lemma *qbs-prob-measure-return*:
assumes $x \in \text{qbs-space } X$
shows $\text{qbs-prob-measure } (\text{qbs-return } X x) = \text{return } (\text{qbs-to-measure } X) x$
 ⟨proof⟩

lemma *qbs-prob-measure-bind*:
assumes $s \in \text{monadP-qbs-Px } X$
and $f \in X \rightarrow_Q \text{monadP-qbs } Y$
shows $\text{qbs-prob-measure } (s \gg f) = \text{qbs-prob-measure } s \gg \text{qbs-prob-measure } \circ f$
 (is ?lhs = ?rhs)
 ⟨proof⟩

lemma *qbs-of-return*:
assumes $x \in \text{qbs-space } X$
shows $\text{qbs-prob-space-qbs } (\text{qbs-return } X x) = X$
 ⟨proof⟩

lemma *qbs-of-bind*:
assumes $s \in \text{monadP-qbs-Px } X$
and $f \in X \rightarrow_Q \text{monadP-qbs } Y$
shows $\text{qbs-prob-space-qbs } (s \gg f) = Y$
 ⟨proof⟩

3.2.8 Properties of Integrals

lemma *qbs-integrable-return*:
assumes $x \in \text{qbs-space } X$
and $f \in X \rightarrow_Q \mathbb{R}_Q$
shows $\text{qbs-integrable } (\text{qbs-return } X x) f$
 ⟨proof⟩

lemma *qbs-integrable-bind-return:*

assumes $s \in \text{monadP-qbs-Px } Y$

$f \in Z \rightarrow_Q \mathbb{R}_Q$

and $g \in Y \rightarrow_Q Z$

shows $\text{qbs-integrable } (s \gg (\lambda y. \text{qbs-return } Z (g y))) f = \text{qbs-integrable } s (f \circ g)$
 $\langle \text{proof} \rangle$

lemma *qbs-prob-ennintegral-morphism:*

assumes $L \in X \rightarrow_Q \text{monadP-qbs } Y$

and $f \in X \rightarrow_Q \text{exp-qbs } Y \mathbb{R}_{Q \geq 0}$

shows $(\lambda x. \text{qbs-prob-ennintegral } (L x) (f x)) \in X \rightarrow_Q \mathbb{R}_{Q \geq 0}$
 $\langle \text{proof} \rangle$

lemma *qbs-morphism-ennintegral-fst:*

assumes $q \in \text{monadP-qbs-Px } Y$

and $f \in X \otimes_Q Y \rightarrow_Q \mathbb{R}_{Q \geq 0}$

shows $(\lambda x. \int^+_Q y. f (x, y) \partial q) \in X \rightarrow_Q \mathbb{R}_{Q \geq 0}$
 $\langle \text{proof} \rangle$

lemma *qbs-morphism-ennintegral-snd:*

assumes $p \in \text{monadP-qbs-Px } X$

and $f \in X \otimes_Q Y \rightarrow_Q \mathbb{R}_{Q \geq 0}$

shows $(\lambda y. \int^+_Q x. f (x, y) \partial p) \in Y \rightarrow_Q \mathbb{R}_{Q \geq 0}$
 $\langle \text{proof} \rangle$

lemma *qbs-prob-ennintegral-morphism':*

assumes $f \in X \rightarrow_Q \mathbb{R}_{Q \geq 0}$

shows $(\lambda s. \text{qbs-prob-ennintegral } s f) \in \text{monadP-qbs } X \rightarrow_Q \mathbb{R}_{Q \geq 0}$
 $\langle \text{proof} \rangle$

lemma *qbs-prob-ennintegral-return:*

assumes $f \in X \rightarrow_Q \mathbb{R}_{Q \geq 0}$

and $x \in \text{qbs-space } X$

shows $\text{qbs-prob-ennintegral } (\text{qbs-return } X x) f = f x$
 $\langle \text{proof} \rangle$

lemma *qbs-prob-ennintegral-bind:*

assumes $s \in \text{monadP-qbs-Px } X$

$f \in X \rightarrow_Q \text{monadP-qbs } Y$

and $g \in Y \rightarrow_Q \mathbb{R}_{Q \geq 0}$

shows $\text{qbs-prob-ennintegral } (s \gg f) g = \text{qbs-prob-ennintegral } s (\lambda y. (\text{qbs-prob-ennintegral } (f y) g))$
 $(\text{is } ?lhs = ?rhs)$
 $\langle \text{proof} \rangle$

lemma *qbs-prob-ennintegral-bind-return:*

assumes $s \in \text{monadP-qbs-Px } Y$

$f \in Z \rightarrow_Q \mathbb{R}_{Q \geq 0}$
and $g \in Y \rightarrow_Q Z$
shows $qbs\text{-prob-ennintegral } (s \gg (\lambda y. qbs\text{-return } Z (g y))) f = qbs\text{-prob-ennintegral } s (f \circ g)$
 $\langle proof \rangle$

lemma *qbs-prob-integral-morphism'*:
assumes $f \in X \rightarrow_Q \mathbb{R}_Q$
shows $(\lambda s. qbs\text{-prob-integral } s f) \in monadP\text{-qbs } X \rightarrow_Q \mathbb{R}_Q$
 $\langle proof \rangle$

lemma *qbs-morphism-integral-fst*:
assumes $q \in monadP\text{-qbs-}Px Y$
and $f \in X \otimes_Q Y \rightarrow_Q \mathbb{R}_Q$
shows $(\lambda x. \int_Q y. f (x, y) \partial q) \in X \rightarrow_Q \mathbb{R}_Q$
 $\langle proof \rangle$

lemma *qbs-morphism-integral-snd*:
assumes $p \in monadP\text{-qbs-}Px X$
and $f \in X \otimes_Q Y \rightarrow_Q \mathbb{R}_Q$
shows $(\lambda y. \int_Q x. f (x, y) \partial p) \in Y \rightarrow_Q \mathbb{R}_Q$
 $\langle proof \rangle$

lemma *qbs-prob-integral-morphism*:
assumes $L \in X \rightarrow_Q monadP\text{-qbs } Y$
 $f \in X \rightarrow_Q exp\text{-qbs } Y \mathbb{R}_Q$
and $\bigwedge x. x \in qbs\text{-space } X \implies qbs\text{-integrable } (L x) (f x)$
shows $(\lambda x. qbs\text{-prob-integral } (L x) (f x)) \in X \rightarrow_Q \mathbb{R}_Q$
 $\langle proof \rangle$

lemma *qbs-prob-integral-morphism''*:
assumes $f \in X \rightarrow_Q \mathbb{R}_Q$
and $L \in Y \rightarrow_Q monadP\text{-qbs } X$
shows $(\lambda y. qbs\text{-prob-integral } (L y) f) \in Y \rightarrow_Q \mathbb{R}_Q$
 $\langle proof \rangle$

lemma *qbs-prob-integral-return*:
assumes $f \in X \rightarrow_Q \mathbb{R}_Q$
and $x \in qbs\text{-space } X$
shows $qbs\text{-prob-integral } (qbs\text{-return } X x) f = f x$
 $\langle proof \rangle$

lemma *qbs-prob-integral-bind*:
assumes $s \in monadP\text{-qbs-}Px X$
 $f \in X \rightarrow_Q monadP\text{-qbs } Y$
 $g \in Y \rightarrow_Q \mathbb{R}_Q$
and $\exists K. \forall y \in qbs\text{-space } Y. |g y| \leq K$
shows $qbs\text{-prob-integral } (s \gg f) g = qbs\text{-prob-integral } s (\lambda y. (qbs\text{-prob-integral } (f y) g))$

(is ?lhs = ?rhs)
 ⟨proof⟩

lemma *qbs-prob-integral-bind-return*:

assumes $s \in \text{monadP-qbs-Px } Y$
 $f \in Z \rightarrow_Q \mathbb{R}_Q$
 and $g \in Y \rightarrow_Q Z$
 shows $\text{qbs-prob-integral } (s \gg (\lambda y. \text{qbs-return } Z (g y))) f = \text{qbs-prob-integral } s$
 $(f \circ g)$
 ⟨proof⟩

lemma *qbs-prob-var-bind-return*:

assumes $s \in \text{monadP-qbs-Px } Y$
 $f \in Z \rightarrow_Q \mathbb{R}_Q$
 and $g \in Y \rightarrow_Q Z$
 shows $\text{qbs-prob-var } (s \gg (\lambda y. \text{qbs-return } Z (g y))) f = \text{qbs-prob-var } s (f \circ g)$
 ⟨proof⟩

end

3.3 Binary Product Measure

theory *Pair-QuasiBorel-Measure*

imports *Monad-QuasiBorel*

begin

3.3.1 Binary Product Measure

Special case of [1] Proposition 23 where $\Omega = \mathbb{R} \times \mathbb{R}$ and $X = X \times Y$. Let $[\alpha, \mu] \in P(X)$ and $[\beta, \nu] \in P(Y)$. $\alpha \times \beta$ is the α in Proposition 23.

definition *qbs-prob-pair-measure-t* :: [*'a qbs-prob-t, 'b qbs-prob-t*] \Rightarrow (*'a* \times *'b*)
qbs-prob-t **where**

qbs-prob-pair-measure-t $p \ q \equiv$ (let $(X, \alpha, \mu) = p$;
 $(Y, \beta, \nu) = q$ in
 $(X \otimes_Q Y, \text{map-prod } \alpha \ \beta \circ \text{real-real.g}, \text{distr } (\mu \otimes_M \nu)$
real-borel real-real.f))

lift-definition *qbs-prob-pair-measure* :: [*'a qbs-prob-space, 'b qbs-prob-space*] \Rightarrow (*'a*
 \times *'b*) *qbs-prob-space* (**infix** \otimes_{Qmes} 80)

is *qbs-prob-pair-measure-t*

⟨proof⟩

lemma(in *pair-qbs-probs*) *qbs-prob-pair-measure-computation*:

$(\text{qbs-prob-space } (X, \alpha, \mu)) \otimes_{Qmes} (\text{qbs-prob-space } (Y, \beta, \nu)) = \text{qbs-prob-space } (X$
 $\otimes_Q Y, \text{map-prod } \alpha \ \beta \circ \text{real-real.g}, \text{distr } (\mu \otimes_M \nu) \text{ real-borel real-real.f})$
 $\text{qbs-prob } (X \otimes_Q Y) (\text{map-prod } \alpha \ \beta \circ \text{real-real.g}) (\text{distr } (\mu \otimes_M \nu) \text{ real-borel}$
real-real.f)
 ⟨proof⟩

lemma *qbs-prob-pair-measure-qbs*:

qbs-prob-space-qbs ($p \otimes_{Qmes} q$) = *qbs-prob-space-qbs* $p \otimes_Q$ *qbs-prob-space-qbs* q
 ⟨proof⟩

lemma(in *pair-qbs-probs*) *qbs-prob-pair-measure-measure*:

shows *qbs-prob-measure* (*qbs-prob-space* (X, α, μ) \otimes_{Qmes} *qbs-prob-space* (Y, β, ν))
 = *distr* ($\mu \otimes_M \nu$) (*qbs-to-measure* ($X \otimes_Q Y$)) (*map-prod* $\alpha \beta$)
 ⟨proof⟩

lemma *qbs-prob-pair-measure-morphism*:

case-prod *qbs-prob-pair-measure* \in *monadP-qbs* $X \otimes_Q$ *monadP-qbs* $Y \rightarrow_Q$ *monadP-qbs* ($X \otimes_Q Y$)
 ⟨proof⟩

lemma(in *pair-qbs-probs*) *qbs-prob-pair-measure-nnintegral*:

assumes $f \in X \otimes_Q Y \rightarrow_Q \mathbb{R}_{Q \geq 0}$
shows ($\int^+_Q z. f z \partial(\text{qbs-prob-space } (X, \alpha, \mu) \otimes_{Qmes} \text{qbs-prob-space } (Y, \beta, \nu))$)
 = ($\int^+ z. (f \circ \text{map-prod } \alpha \beta) z \partial(\mu \otimes_M \nu)$)
 (is ?lhs = ?rhs)
 ⟨proof⟩

lemma(in *pair-qbs-probs*) *qbs-prob-pair-measure-integral*:

assumes $f \in X \otimes_Q Y \rightarrow_Q \mathbb{R}_Q$
shows ($\int_Q z. f z \partial(\text{qbs-prob-space } (X, \alpha, \mu) \otimes_{Qmes} \text{qbs-prob-space } (Y, \beta, \nu))$)
 = ($\int z. (f \circ \text{map-prod } \alpha \beta) z \partial(\mu \otimes_M \nu)$)
 (is ?lhs = ?rhs)
 ⟨proof⟩

lemma *qbs-prob-pair-measure-eq-bind*:

assumes $p \in \text{monadP-qbs-Px } X$
and $q \in \text{monadP-qbs-Px } Y$
shows $p \otimes_{Qmes} q = p \gg (\lambda x. q \gg (\lambda y. \text{qbs-return } (X \otimes_Q Y) (x, y)))$
 ⟨proof⟩

3.3.2 Fubini Theorem

lemma *qbs-prob-ennintegral-Fubini-fst*:

assumes $p \in \text{monadP-qbs-Px } X$
 $q \in \text{monadP-qbs-Px } Y$
and $f \in X \otimes_Q Y \rightarrow_Q \mathbb{R}_{Q \geq 0}$
shows ($\int^+_Q x. \int^+_Q y. f (x, y) \partial q \partial p$) = ($\int^+_Q z. f z \partial(p \otimes_{Qmes} q)$)
 (is ?lhs = ?rhs)
 ⟨proof⟩

lemma *qbs-prob-ennintegral-Fubini-snd*:

assumes $p \in \text{monadP-qbs-Px } X$
 $q \in \text{monadP-qbs-Px } Y$
and $f \in X \otimes_Q Y \rightarrow_Q \mathbb{R}_{Q \geq 0}$

shows $(\int^{+Q} y. \int^{+Q} x. f(x,y) \partial p \partial q) = (\int^{+Q} x. f x \partial(p \otimes_{Qmes} q))$
(is ?lhs = ?rhs)
 <proof>

lemma *qbs-prob-ennintegral-indep1:*

assumes $p \in \text{monadP-qbs-Px } X$

and $f \in X \rightarrow_Q \mathbb{R}_{Q \geq 0}$

shows $(\int^{+Q} z. f(\text{fst } z) \partial(p \otimes_{Qmes} q)) = (\int^{+Q} x. f x \partial p)$
(is ?lhs = -)

<proof>

lemma *qbs-prob-ennintegral-indep2:*

assumes $q \in \text{monadP-qbs-Px } Y$

and $f \in Y \rightarrow_Q \mathbb{R}_{Q \geq 0}$

shows $(\int^{+Q} z. f(\text{snd } z) \partial(p \otimes_{Qmes} q)) = (\int^{+Q} y. f y \partial q)$
(is ?lhs = -)

<proof>

lemma *qbs-ennintegral-indep-mult:*

assumes $p \in \text{monadP-qbs-Px } X$

$q \in \text{monadP-qbs-Px } Y$

$f \in X \rightarrow_Q \mathbb{R}_{Q \geq 0}$

and $g \in Y \rightarrow_Q \mathbb{R}_{Q \geq 0}$

shows $(\int^{+Q} z. f(\text{fst } z) * g(\text{snd } z) \partial(p \otimes_{Qmes} q)) = (\int^{+Q} x. f x \partial p) * (\int^{+Q} y. g y \partial q)$
(is ?lhs = ?rhs)

<proof>

lemma(**in** *pair-qbs-probs*) *qbs-prob-pair-measure-integrable:*

assumes $\text{qbs-integrable } (\text{qbs-prob-space } (X, \alpha, \mu) \otimes_{Qmes} \text{qbs-prob-space } (Y, \beta, \nu))$
 f

shows $f \in X \otimes_Q Y \rightarrow_Q \mathbb{R}_Q$
 $\text{integrable } (\mu \otimes_M \nu) (f \circ (\text{map-prod } \alpha \beta))$

<proof>

lemma(**in** *pair-qbs-probs*) *qbs-prob-pair-measure-integrable':*

assumes $f \in X \otimes_Q Y \rightarrow_Q \mathbb{R}_Q$

and $\text{integrable } (\mu \otimes_M \nu) (f \circ (\text{map-prod } \alpha \beta))$

shows $\text{qbs-integrable } (\text{qbs-prob-space } (X, \alpha, \mu) \otimes_{Qmes} \text{qbs-prob-space } (Y, \beta, \nu))$

f

<proof>

lemma *qbs-integrable-pair-swap:*

assumes $\text{qbs-integrable } (p \otimes_{Qmes} q) f$

shows $\text{qbs-integrable } (q \otimes_{Qmes} p) (\lambda(x,y). f(y,x))$

<proof>

lemma *qbs-integrable-pair1:*

assumes $p \in \text{monadP-qbs-Px } X$
 $q \in \text{monadP-qbs-Px } Y$
 $f \in X \otimes_Q Y \rightarrow_Q \mathbb{R}_Q$
 $\text{qbs-integrable } p (\lambda x. \int_Q y. |f(x,y)| \partial q)$
and $\bigwedge x. x \in \text{qbs-space } X \implies \text{qbs-integrable } q (\lambda y. f(x,y))$
shows $\text{qbs-integrable } (p \otimes_{Qmes} q) f$
 <proof>

lemma *qbs-integrable-pair2*:
assumes $p \in \text{monadP-qbs-Px } X$
 $q \in \text{monadP-qbs-Px } Y$
 $f \in X \otimes_Q Y \rightarrow_Q \mathbb{R}_Q$
 $\text{qbs-integrable } q (\lambda y. \int_Q x. |f(x,y)| \partial p)$
and $\bigwedge y. y \in \text{qbs-space } Y \implies \text{qbs-integrable } p (\lambda x. f(x,y))$
shows $\text{qbs-integrable } (p \otimes_{Qmes} q) f$
 <proof>

lemma *qbs-integrable-fst*:
assumes $\text{qbs-integrable } (p \otimes_{Qmes} q) f$
shows $\text{qbs-integrable } p (\lambda x. \int_Q y. f(x,y) \partial q)$
 <proof>

lemma *qbs-integrable-snd*:
assumes $\text{qbs-integrable } (p \otimes_{Qmes} q) f$
shows $\text{qbs-integrable } q (\lambda y. \int_Q x. f(x,y) \partial p)$
 <proof>

lemma *qbs-integrable-indep-mult*:
assumes $\text{qbs-integrable } p f$
and $\text{qbs-integrable } q g$
shows $\text{qbs-integrable } (p \otimes_{Qmes} q) (\lambda x. f(\text{fst } x) * g(\text{snd } x))$
 <proof>

lemma *qbs-integrable-indep1*:
assumes $\text{qbs-integrable } p f$
shows $\text{qbs-integrable } (p \otimes_{Qmes} q) (\lambda x. f(\text{fst } x))$
 <proof>

lemma *qbs-integrable-indep2*:
assumes $\text{qbs-integrable } q g$
shows $\text{qbs-integrable } (p \otimes_{Qmes} q) (\lambda x. g(\text{snd } x))$
 <proof>

lemma *qbs-prob-integral-Fubini-fst*:
assumes $\text{qbs-integrable } (p \otimes_{Qmes} q) f$
shows $(\int_Q x. \int_Q y. f(x,y) \partial q \partial p) = (\int_Q z. f z \partial(p \otimes_{Qmes} q))$
 (is ?lhs = ?rhs)
 <proof>

lemma *qbs-prob-integral-Fubini-snd:*

assumes *qbs-integrable* $(p \otimes_{Qmes} q) f$
shows $(\int_Q y. \int_Q x. f(x,y) \partial p \partial q) = (\int_Q z. f z \partial(p \otimes_{Qmes} q))$
(is ?lhs = ?rhs)

<proof>

lemma *qbs-prob-integral-indep1:*

assumes *qbs-integrable* $p f$
shows $(\int_Q z. f(fst z) \partial(p \otimes_{Qmes} q)) = (\int_Q x. f x \partial p)$
<proof>

lemma *qbs-prob-integral-indep2:*

assumes *qbs-integrable* $q g$
shows $(\int_Q z. g(snd z) \partial(p \otimes_{Qmes} q)) = (\int_Q y. g y \partial q)$
<proof>

lemma *qbs-prob-integral-indep-mult:*

assumes *qbs-integrable* $p f$
and *qbs-integrable* $q g$
shows $(\int_Q z. f(fst z) * g(snd z) \partial(p \otimes_{Qmes} q)) = (\int_Q x. f x \partial p) * (\int_Q y. g y \partial q)$
(is ?lhs = ?rhs)
<proof>

lemma *qbs-prob-var-indep-plus:*

assumes *qbs-integrable* $(p \otimes_{Qmes} q) f$
qbs-integrable $(p \otimes_{Qmes} q) (\lambda z. (f z)^2)$
qbs-integrable $(p \otimes_{Qmes} q) g$
qbs-integrable $(p \otimes_{Qmes} q) (\lambda z. (g z)^2)$
qbs-integrable $(p \otimes_{Qmes} q) (\lambda z. (f z) * (g z))$
and $(\int_Q z. f z * g z \partial(p \otimes_{Qmes} q)) = (\int_Q z. f z \partial(p \otimes_{Qmes} q)) * (\int_Q z. g z \partial(p \otimes_{Qmes} q))$
shows *qbs-prob-var* $(p \otimes_{Qmes} q) (\lambda z. f z + g z) = \text{qbs-prob-var } (p \otimes_{Qmes} q) f + \text{qbs-prob-var } (p \otimes_{Qmes} q) g$
<proof>

lemma *qbs-prob-var-indep-plus':*

assumes *qbs-integrable* $p f$
qbs-integrable $p (\lambda x. (f x)^2)$
qbs-integrable $q g$
and *qbs-integrable* $q (\lambda x. (g x)^2)$
shows *qbs-prob-var* $(p \otimes_{Qmes} q) (\lambda z. f(fst z) + g(snd z)) = \text{qbs-prob-var } p f + \text{qbs-prob-var } q g$
<proof>

end

3.4 Measure as QBS Measure

theory *Measure-as-QuasiBorel-Measure*
imports *Pair-QuasiBorel-Measure*

begin

lemma *distr-id'*:

assumes *sets N = sets M*

$f \in N \rightarrow_M N$

and $\bigwedge x. x \in \text{space } N \implies f x = x$

shows $\text{distr } N M f = N$

<proof>

Every probability measure on a standard Borel space can be represented as a measure on a quasi-Borel space [1], Proposition 23.

locale *standard-borel-prob-space = standard-borel P + p:prob-space P*

for $P :: 'a \text{ measure}$

begin

sublocale *qbs-prob measure-to-qbs P g distr P real-borel f*

<proof>

lift-definition *as-qbs-measure :: 'a qbs-prob-space is*

(measure-to-qbs P, g, distr P real-borel f)

<proof>

lemma *as-qbs-measure-retract:*

assumes *[measurable]: a ∈ P →_M real-borel*

and *[measurable]: b ∈ real-borel →_M P*

and *[simp]: ⋀ x. x ∈ space P ⇒ (b ∘ a) x = x*

shows *qbs-prob (measure-to-qbs P) b (distr P real-borel a)*

as-qbs-measure = qbs-prob-space (measure-to-qbs P, b, distr P real-borel a)

<proof>

lemma *measure-as-qbs-measure-qbs:*

qbs-prob-space-qbs as-qbs-measure = measure-to-qbs P

<proof>

lemma *measure-as-qbs-measure-image:*

as-qbs-measure ∈ monadP-qbs-Px (measure-to-qbs P)

<proof>

lemma *as-qbs-measure-as-measure[simp]:*

distr (distr P real-borel f) (qbs-to-measure (measure-to-qbs P)) g = P

<proof>

lemma *measure-as-qbs-measure-recover:*

qbs-prob-measure as-qbs-measure = P

<proof>

end

lemma(in *standard-borel*) *qbs-prob-measure-recover*:
 assumes $q \in \text{monadP-qbs-Px}$ (*measure-to-qbs M*)
 shows *standard-borel-prob-space.as-qbs-measure* (*qbs-prob-measure q*) = q
<proof>

lemma(in *standard-borel-prob-space*) *ennintegral-as-qbs-ennintegral*:
 assumes $k \in \text{borel-measurable P}$
 shows $(\int^+_{\mathcal{Q}} x. k x \partial \text{as-qbs-measure}) = (\int^+ x. k x \partial P)$
<proof>

lemma(in *standard-borel-prob-space*) *integral-as-qbs-integral*:
 $(\int_{\mathcal{Q}} x. k x \partial \text{as-qbs-measure}) = (\int x. k x \partial P)$
<proof>

lemma(in *standard-borel*) *measure-with-args-morphism*:
 assumes [*measurable*]: $\mu \in X \rightarrow_M \text{prob-algebra M}$
 shows *standard-borel-prob-space.as-qbs-measure* $\circ \mu \in \text{measure-to-qbs } X \rightarrow_{\mathcal{Q}}$
monadP-qbs (measure-to-qbs M)
<proof>

lemma(in *standard-borel*) *measure-with-args-recover*:
 assumes $\mu \in \text{space } X \rightarrow \text{space (prob-algebra M)}$
 and $x \in \text{space } X$
 shows *qbs-prob-measure* (*standard-borel-prob-space.as-qbs-measure* (μx)) = μ
 x
<proof>

3.5 Example of Probability Measures

Probability measures on \mathbb{R} can be represented as probability measures on the quasi-Borel space \mathbb{R} .

3.5.1 Normal Distribution

definition *normal-distribution* :: *real* \times *real* \Rightarrow *real measure* **where**
normal-distribution $\mu\sigma = (\text{if } 0 < (\text{snd } \mu\sigma) \text{ then density lborel } (\lambda x. \text{ennreal (normal-density (fst } \mu\sigma) (\text{snd } \mu\sigma) x))$
 else return lborel 0)

lemma *normal-distribution-measurable*:
 normal-distribution $\in \text{real-borel} \otimes_M \text{real-borel} \rightarrow_M \text{prob-algebra real-borel}$
<proof>

definition *qbs-normal-distribution* :: *real* \Rightarrow *real* \Rightarrow *real qbs-prob-space* **where**

qbs-normal-distribution \equiv *curry* (*standard-borel-prob-space.as-qbs-measure* \circ *normal-distribution*)

lemma *qbs-normal-distribution-morphism*:

qbs-normal-distribution $\in \mathbf{R}_Q \rightarrow_Q \text{exp-qbs } \mathbf{R}_Q$ (*monadP-qbs* \mathbf{R}_Q)
 ⟨*proof*⟩

context

fixes $\mu \sigma :: \text{real}$
assumes *sigma*: $\sigma > 0$

begin

interpretation *n-dist:standard-borel-prob-space normal-distribution* (μ, σ)

⟨*proof*⟩

lemma *qbs-normal-distribution-def2*:

qbs-normal-distribution $\mu \sigma = \text{n-dist.as-qbs-measure}$
 ⟨*proof*⟩

lemma *qbs-normal-distribution-integral*:

$(\int_Q x. f x \partial (\text{qbs-normal-distribution } \mu \sigma)) = (\int x. f x \partial (\text{density lborel } (\lambda x. \text{ennreal } (\text{normal-density } \mu \sigma x))))$
 ⟨*proof*⟩

lemma *qbs-normal-distribution-expectation*:

assumes $f \in \text{real-borel} \rightarrow_M \text{real-borel}$
shows $(\int_Q x. f x \partial (\text{qbs-normal-distribution } \mu \sigma)) = (\int x. \text{normal-density } \mu \sigma x * f x \partial \text{lborel})$
 ⟨*proof*⟩

end

3.5.2 Uniform Distribution

definition *interval-uniform-distribution* $:: \text{real} \Rightarrow \text{real} \Rightarrow \text{real measure}$ **where**

interval-uniform-distribution $a b \equiv$ (if $a < b$ then *uniform-measure lborel* $\{a <..<b\}$
 else *return lborel* 0)

lemma *sets-interval-uniform-distribution[measurable-cong]*:

sets (*interval-uniform-distribution* $a b$) = *borel*
 ⟨*proof*⟩

lemma *interval-uniform-distribution-measurable*:

$(\lambda r. \text{interval-uniform-distribution } (\text{fst } r) (\text{snd } r)) \in \text{real-borel} \otimes_M \text{real-borel} \rightarrow_M \text{prob-algebra real-borel}$
 ⟨*proof*⟩

definition *qbs-interval-uniform-distribution* $:: \text{real} \Rightarrow \text{real} \Rightarrow \text{real qbs-prob-space}$

where

$qbs\text{-interval-uniform-distribution} \equiv \text{curry } (\text{standard-borel-prob-space.as-qbs-measure} \circ (\lambda r. \text{interval-uniform-distribution } (\text{fst } r) (\text{snd } r)))$

lemma $qbs\text{-interval-uniform-distribution-morphism}$:

$qbs\text{-interval-uniform-distribution} \in \mathbb{R}_Q \rightarrow_Q \text{exp-qbs } \mathbb{R}_Q \text{ (monadP-qbs } \mathbb{R}_Q)$
 $\langle \text{proof} \rangle$

context

fixes $a \ b :: \text{real}$

assumes $a\text{-less-than-}b : a < b$

begin

definition $ab\text{-qbs-uniform-distribution} \equiv qbs\text{-interval-uniform-distribution } a \ b$

interpretation $ab\text{-u-dist}$: $\text{standard-borel-prob-space interval-uniform-distribution } a \ b$

$\langle \text{proof} \rangle$

lemma $qbs\text{-interval-uniform-distribution-def2}$:

$ab\text{-qbs-uniform-distribution} = ab\text{-u-dist.as-qbs-measure}$
 $\langle \text{proof} \rangle$

lemma $qbs\text{-uniform-distribution-expectation}$:

assumes $f \in \mathbb{R}_Q \rightarrow_Q \mathbb{R}_{Q \geq 0}$

shows $(\int^+ x. f \ x \ \partial ab\text{-qbs-uniform-distribution}) = (\int^+ x \in \{a < .. < b\}. f \ x \ \partial lborel)$
 $/ (b - a)$

(**is** $?lhs = ?rhs$)

$\langle \text{proof} \rangle$

end

3.5.3 Bernoulli Distribution

definition $qbs\text{-bernoulli} :: \text{real} \Rightarrow \text{bool qbs-prob-space where}$

$qbs\text{-bernoulli} \equiv \text{standard-borel-prob-space.as-qbs-measure} \circ (\lambda x. \text{measure-pmf } (\text{bernoulli-pmf } x))$

lemma $bernoulli\text{-measurable}$:

$(\lambda x. \text{measure-pmf } (\text{bernoulli-pmf } x)) \in \text{real-borel} \rightarrow_M \text{prob-algebra bool-borel}$
 $\langle \text{proof} \rangle$

lemma $qbs\text{-bernoulli-morphism}$:

$qbs\text{-bernoulli} \in \mathbb{R}_Q \rightarrow_Q \text{monadP-qbs } \mathbb{B}_Q$
 $\langle \text{proof} \rangle$

lemma $qbs\text{-bernoulli-measure}$:

$qbs\text{-prob-measure } (qbs\text{-bernoulli } p) = \text{measure-pmf } (\text{bernoulli-pmf } p)$

<proof>

context

fixes $p :: \text{real}$

assumes $p \geq 0$ and $p \leq 1$

begin

lemma *qbs-bernoulli-expectation*:

$(\int_Q x. f x \text{ } \partial \text{qbs-bernoulli } p) = f \text{ True} * p + f \text{ False} * (1 - p)$

<proof>

end

end

3.6 Bayesian Linear Regression

theory *Bayesian-Linear-Regression*

imports *Measure-as-QuasiBorel-Measure*

begin

We formalize the Bayesian linear regression presented in [1] section VI.

3.6.1 Prior

abbreviation $\nu \equiv \text{density lborel } (\lambda x. \text{ennreal } (\text{normal-density } 0 \ 3 \ x))$

interpretation ν : *standard-borel-prob-space* ν

<proof>

term $\nu.\text{as-qbs-measure} :: \text{real qbs-prob-space}$

definition *prior* :: $(\text{real} \Rightarrow \text{real}) \text{ qbs-prob-space}$ **where**

$\text{prior} \equiv \text{do } \{ s \leftarrow \nu.\text{as-qbs-measure} ;$

$b \leftarrow \nu.\text{as-qbs-measure} ;$

$\text{qbs-return } (\mathbb{R}_Q \Rightarrow_Q \mathbb{R}_Q) (\lambda r. s * r + b) \}$

lemma *ν -as-qbs-measure-eq*:

$\nu.\text{as-qbs-measure} = \text{qbs-prob-space } (\mathbb{R}_Q, \text{id}, \nu)$

<proof>

interpretation ν -qp: *pair-qbs-prob* $\mathbb{R}_Q \text{ id } \nu \ \mathbb{R}_Q \text{ id } \nu$

<proof>

lemma *ν -as-qbs-measure-in-Pr*:

$\nu.\text{as-qbs-measure} \in \text{monadP-qbs-Px } \mathbb{R}_Q$

<proof>

lemma *sets-real-real-real[measurable-cong]*:

$\text{sets } (\text{qbs-to-measure } ((\mathbb{R}_Q \otimes_Q \mathbb{R}_Q) \otimes_Q \mathbb{R}_Q)) = \text{sets } ((\text{borel} \otimes_M \text{borel}) \otimes_M \text{borel})$

<proof>

lemma *lin-morphism*:

$(\lambda(s, b) \ r. \ s * r + b) \in \mathbb{R}_Q \otimes_Q \mathbb{R}_Q \rightarrow_Q \mathbb{R}_Q \Rightarrow_Q \mathbb{R}_Q$
<proof>

lemma *lin-measurable[measurable]*:

$(\lambda(s, b) \ r. \ s * r + b) \in \text{real-borel} \otimes_M \text{real-borel} \rightarrow_M \text{qbs-to-measure} (\mathbb{R}_Q \Rightarrow_Q \mathbb{R}_Q)$
<proof>

lemma *prior-computation*:

$\text{qbs-prob} (\mathbb{R}_Q \Rightarrow_Q \mathbb{R}_Q) ((\lambda(s, b) \ r. \ s * r + b) \circ \text{real-real.g}) (\text{distr} (\nu \otimes_M \nu) \text{real-borel real-real.f})$
 $\text{prior} = \text{qbs-prob-space} (\mathbb{R}_Q \Rightarrow_Q \mathbb{R}_Q, (\lambda(s, b) \ r. \ s * r + b) \circ \text{real-real.g}, \text{distr} (\nu \otimes_M \nu) \text{real-borel real-real.f})$
<proof>

The following lemma corresponds to the equation (5).

lemma *prior-measure*:

$\text{qbs-prob-measure prior} = \text{distr} (\nu \otimes_M \nu) (\text{qbs-to-measure} (\text{exp-qbs } \mathbb{R}_Q \ \mathbb{R}_Q))$
 $(\lambda(s, b) \ r. \ s * r + b)$
<proof>

lemma *prior-in-space*:

$\text{prior} \in \text{qbs-space} (\text{monadP-qbs} (\mathbb{R}_Q \Rightarrow_Q \mathbb{R}_Q))$
<proof>

3.6.2 Likelihood

abbreviation $d \ \mu \ x \equiv \text{normal-density } \mu \ (1/2) \ x$

lemma *d-positive* : $0 < d \ \mu \ x$

<proof>

definition *obs* :: $(\text{real} \Rightarrow \text{real}) \Rightarrow \text{ennreal}$ **where**

$\text{obs } f \equiv d \ (f \ 1) \ 2.5 * d \ (f \ 2) \ 3.8 * d \ (f \ 3) \ 4.5 * d \ (f \ 4) \ 6.2 * d \ (f \ 5) \ 8$

lemma *obs-morphism*:

$\text{obs} \in \mathbb{R}_Q \Rightarrow_Q \mathbb{R}_Q \rightarrow_Q \mathbb{R}_{Q \geq 0}$
<proof>

lemma *obs-measurable[measurable]*:

$\text{obs} \in \text{qbs-to-measure} (\text{exp-qbs } \mathbb{R}_Q \ \mathbb{R}_Q) \rightarrow_M \text{ennreal-borel}$
<proof>

3.6.3 Posterior

lemma *id-obs-morphism*:

$(\lambda f. \ (f, \text{obs } f)) \in \mathbb{R}_Q \Rightarrow_Q \mathbb{R}_Q \rightarrow_Q (\mathbb{R}_Q \Rightarrow_Q \mathbb{R}_Q) \otimes_Q \mathbb{R}_{Q \geq 0}$

<proof>

lemma *push-forward-measure-in-space:*

monadP-qbs-Pf ($\mathbb{R}_Q \Rightarrow_Q \mathbb{R}_Q$) ($(\mathbb{R}_Q \Rightarrow_Q \mathbb{R}_Q) \otimes_Q \mathbb{R}_{Q \geq 0}$) ($\lambda f. (f, \text{obs } f)$) *prior* \in
qbs-space (*monadP-qbs* ($(\mathbb{R}_Q \Rightarrow_Q \mathbb{R}_Q) \otimes_Q \mathbb{R}_{Q \geq 0}$))

<proof>

lemma *push-forward-measure-computation:*

qbs-prob ($(\mathbb{R}_Q \Rightarrow_Q \mathbb{R}_Q) \otimes_Q \mathbb{R}_{Q \geq 0}$) ($\lambda l. (((\lambda(s, b) r. s * r + b) \circ \text{real-real.g})$
 $l, ((\text{obs} \circ (\lambda(s, b) r. s * r + b)) \circ \text{real-real.g}) l)$) (*distr* ($\nu \otimes_M \nu$) *real-borel*
real-real.f)

monadP-qbs-Pf ($\mathbb{R}_Q \Rightarrow_Q \mathbb{R}_Q$) ($(\mathbb{R}_Q \Rightarrow_Q \mathbb{R}_Q) \otimes_Q \mathbb{R}_{Q \geq 0}$) ($\lambda f. (f, \text{obs } f)$) *prior* =
qbs-prob-space ($(\mathbb{R}_Q \Rightarrow_Q \mathbb{R}_Q) \otimes_Q \mathbb{R}_{Q \geq 0}$, ($\lambda l. (((\lambda(s, b) r. s * r + b) \circ \text{real-real.g})$
 $l, ((\text{obs} \circ (\lambda(s, b) r. s * r + b)) \circ \text{real-real.g}) l)$), *distr* ($\nu \otimes_M \nu$) *real-borel*
real-real.f)

<proof>

3.6.4 Normalizer

We use the unit space for an error.

definition *norm-qbs-measure* :: ($'a \times \text{ennreal}$) *qbs-prob-space* \Rightarrow $'a$ *qbs-prob-space*
+ *unit* **where**

norm-qbs-measure $p \equiv$ (*let* ($XR, \alpha\beta, \nu$) = *rep-qbs-prob-space* p *in*
 if *emeasure* (*density* ν (*snd* $\circ \alpha\beta$)) *UNIV* = 0 *then* *Inr* ()
 else if *emeasure* (*density* ν (*snd* $\circ \alpha\beta$)) *UNIV* = ∞ *then* *Inr* ()
 else *Inl* (*qbs-prob-space* (*map-qbs* *fst* XR , *fst* $\circ \alpha\beta$, *density* ν
($\lambda r. \text{snd} (\alpha\beta r) / \text{emeasure} (\text{density } \nu (\text{snd} \circ \alpha\beta)) \text{ UNIV}$))))

lemma *norm-qbs-measure-qbs-prob:*

assumes *qbs-prob* ($X \otimes_Q \mathbb{R}_{Q \geq 0}$) ($\lambda r. (\alpha r, \beta r)$) μ
emeasure (*density* μ β) *UNIV* $\neq 0$

and *emeasure* (*density* μ β) *UNIV* $\neq \infty$

shows *qbs-prob* $X \alpha$ (*density* μ ($\lambda r. (\beta r) / \text{emeasure} (\text{density } \mu \beta) \text{ UNIV}$))

<proof>

lemma *norm-qbs-measure-computation:*

assumes *qbs-prob* ($X \otimes_Q \mathbb{R}_{Q \geq 0}$) ($\lambda r. (\alpha r, \beta r)$) μ

shows *norm-qbs-measure* (*qbs-prob-space* ($X \otimes_Q \mathbb{R}_{Q \geq 0}$, ($\lambda r. (\alpha r, \beta r)$), μ)) =
(*if* *emeasure* (*density* μ β) *UNIV* = 0 *then* *Inr* ()

else if *emeasure*
(*density* μ β) *UNIV* = ∞ *then* *Inr* ()

else *Inl* (*qbs-prob-space*
(X, α , *density* μ ($\lambda r. (\beta r) / \text{emeasure} (\text{density } \mu \beta) \text{ UNIV}$))))

<proof>

lemma *norm-qbs-measure-morphism:*

norm-qbs-measure \in *monadP-qbs* ($X \otimes_Q \mathbb{R}_{Q \geq 0}$) \rightarrow_Q *monadP-qbs* $X <+>_Q 1_Q$
<proof>

The following is the semantics of the entire program.

definition *program* :: (real \Rightarrow real) qbs-prob-space + unit **where**
program \equiv norm-qbs-measure (monadP-qbs-Pf ($\mathbb{R}_Q \Rightarrow_Q \mathbb{R}_Q$)) (($\mathbb{R}_Q \Rightarrow_Q \mathbb{R}_Q$) \otimes_Q $\mathbb{R}_{Q \geq 0}$) ($\lambda f. (f, \text{obs } f)$) prior

lemma *program-in-space*:

program \in qbs-space (monadP-qbs ($\mathbb{R}_Q \Rightarrow_Q \mathbb{R}_Q$)) $\langle + \rangle_Q 1_Q$
 \langle proof \rangle

We calculate the normalizing constant.

lemma *complete-the-square*:

fixes a b c x :: real
assumes a \neq 0
shows $a*x^2 + b*x + c = a * (x + (b / (2*a)))^2 - ((b^2 - 4*a*c)/(4*a))$
 \langle proof \rangle

lemma *complete-the-square2'*:

fixes a b c x :: real
assumes a \neq 0
shows $a*x^2 - 2*b*x + c = a * (x - (b / a))^2 - ((b^2 - a*c)/a)$
 \langle proof \rangle

lemma *normal-density-mu-x-swap*:

normal-density $\mu \sigma x =$ normal-density $x \sigma \mu$
 \langle proof \rangle

lemma *normal-density-plus-shift*:

normal-density $\mu \sigma (x + y) =$ normal-density $(\mu - x) \sigma y$
 \langle proof \rangle

lemma *normal-density-times*:

assumes $\sigma > 0 \sigma' > 0$
shows normal-density $\mu \sigma x * \text{normal-density } \mu' \sigma' x = (1 / \text{sqrt } (2 * \text{pi} * (\sigma^2 + \sigma'^2))) * \text{exp } (- (\mu - \mu')^2 / (2 * (\sigma^2 + \sigma'^2))) * \text{normal-density } ((\mu * \sigma'^2 + \mu' * \sigma^2) / (\sigma^2 + \sigma'^2)) (\sigma * \sigma' / \text{sqrt } (\sigma^2 + \sigma'^2)) x$
(is ?lhs = ?rhs)
 \langle proof \rangle

lemma *normal-density-times'*:

assumes $\sigma > 0 \sigma' > 0$
shows $a * \text{normal-density } \mu \sigma x * \text{normal-density } \mu' \sigma' x = a * (1 / \text{sqrt } (2 * \text{pi} * (\sigma^2 + \sigma'^2))) * \text{exp } (- (\mu - \mu')^2 / (2 * (\sigma^2 + \sigma'^2))) * \text{normal-density } ((\mu * \sigma'^2 + \mu' * \sigma^2) / (\sigma^2 + \sigma'^2)) (\sigma * \sigma' / \text{sqrt } (\sigma^2 + \sigma'^2)) x$
 \langle proof \rangle

lemma *normal-density-times-minusx*:

assumes $\sigma > 0 \sigma' > 0 a \neq a'$
shows normal-density $(\mu - a*x) \sigma y * \text{normal-density } (\mu' - a'*x) \sigma' y = (1$

$/ |a' - a| * \text{normal-density } ((\mu' - \mu)/(a' - a)) (\text{sqrt } ((\sigma^2 + \sigma'^2)/(a' - a)^2)) x * \text{normal-density } (((\mu - a*x)*\sigma'^2 + (\mu' - a'*x)*\sigma^2)/(\sigma^2 + \sigma'^2)) (\sigma * \sigma' / \text{sqrt } (\sigma^2 + \sigma'^2)) y$
 <proof>

The following is the normalizing constant of the program.

abbreviation $C \equiv \text{ennreal } ((4 * \text{sqrt } 2 / (\text{pi}^2 * \text{sqrt } (66961 * \text{pi}))) * (\text{exp } (- (1674761 / 1674025))))$

lemma *program-normalizing-constant:*

$\text{emeasure } (\text{density } (\text{distr } (\nu \otimes_M \nu) \text{ real-borel real-real.f}) (\text{obs } \circ (\lambda(s, b) r. s * r + b) \circ \text{real-real.g})) \text{ UNIV} = C$
 (is ?lhs = ?rhs)
 <proof>

The program returns a probability measure, rather than error.

lemma *program-result:*

$\text{qbs-prob } (\mathbb{R}_Q \Rightarrow_Q \mathbb{R}_Q) ((\lambda(s, b) r. s * r + b) \circ \text{real-real.g}) (\text{density } (\text{distr } (\nu \otimes_M \nu) \text{ real-borel real-real.f}) (\lambda r. (\text{obs } \circ (\lambda(s, b) r. s * r + b) \circ \text{real-real.g}) r / C))$
 $\text{program} = \text{Inl } (\text{qbs-prob-space } (\mathbb{R}_Q \Rightarrow_Q \mathbb{R}_Q, (\lambda(s, b) r. s * r + b) \circ \text{real-real.g}, \text{density } (\text{distr } (\nu \otimes_M \nu) \text{ real-borel real-real.f}) (\lambda r. (\text{obs } \circ (\lambda(s, b) r. s * r + b) \circ \text{real-real.g}) r / C)))$
 <proof>

lemma *program-inl:*

$\text{program} \in \text{Inl } ' (\text{qbs-space } (\text{monadP-qbs } (\mathbb{R}_Q \Rightarrow_Q \mathbb{R}_Q)))$
 <proof>

lemma *program-result-measure:*

$\text{qbs-prob-measure } (\text{qbs-prob-space } (\mathbb{R}_Q \Rightarrow_Q \mathbb{R}_Q, (\lambda(s, b) r. s * r + b) \circ \text{real-real.g}, \text{density } (\text{distr } (\nu \otimes_M \nu) \text{ real-borel real-real.f}) (\lambda r. (\text{obs } \circ (\lambda(s, b) r. s * r + b) \circ \text{real-real.g}) r / C)))$
 $= \text{density } (\text{qbs-prob-measure prior}) (\lambda k. \text{obs } k / C)$
 (is ?lhs = ?rhs)
 <proof>

lemma *program-result-measure':*

$\text{qbs-prob-measure } (\text{qbs-prob-space } (\text{exp-qbs } \mathbb{R}_Q \mathbb{R}_Q, (\lambda(s, b) r. s * r + b) \circ \text{real-real.g}, \text{density } (\text{distr } (\nu \otimes_M \nu) \text{ real-borel real-real.f}) (\lambda r. (\text{obs } \circ (\lambda(s, b) r. s * r + b) \circ \text{real-real.g}) r / C)))$
 $= \text{distr } (\text{density } (\nu \otimes_M \nu) (\lambda(s, b). \text{obs } (\lambda r. s * r + b) / C)) (\text{qbs-to-measure } (\text{exp-qbs } \mathbb{R}_Q \mathbb{R}_Q)) (\lambda(s, b) r. s * r + b)$
 <proof>

end

References

- [1] C. Heunen, O. Kammar, S. Staton, and H. Yang. A convenient category for higher-order probability theory. In *Proceedings of the 32nd Annual ACM/IEEE Symposium on Logic in Computer Science, LICS '17*. IEEE Press, 2017.