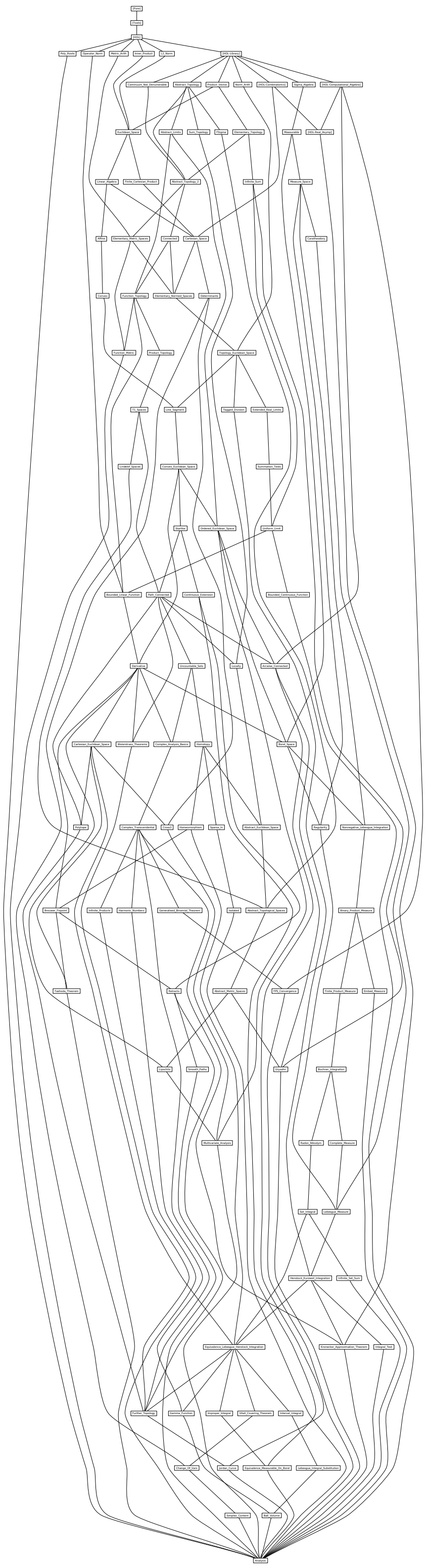
Analysis

March 13, 2025

Contents



Chapter 1

Linear Algebra

theory *L2_Norm*
imports *Complex_Main*
begin

1.1 L2 Norm

definition *L2_set* :: $('a \Rightarrow real) \Rightarrow 'a\ set \Rightarrow real$ **where**
$L2\_set\ f\ A = sqrt\ (\sum i{\in}A.\ (f\ i)^2)$

proposition *L2_set_triangle_ineq*:
$L2\_set\ (\lambda i.\ f\ i + g\ i)\ A \leq L2\_set\ f\ A + L2\_set\ g\ A$

end

1.2 Inner Product Spaces and Gradient Derivative

theory *Inner_Product*
imports *Complex_Main*
begin

1.2.1 Real inner product spaces

class *real_inner* = *real_vector* + *sgn_div_norm* + *dist_norm* + *uniformity_dist* + *open_uniformity* +
fixes *inner* :: $'a \Rightarrow 'a \Rightarrow real$
assumes *inner_commute*: $inner\ x\ y = inner\ y\ x$
and *inner_add_left*: $inner\ (x + y)\ z = inner\ x\ z + inner\ y\ z$
and *inner_scaleR_left* [*simp*]: $inner\ (scaleR\ r\ x)\ y = r * (inner\ x\ y)$
and *inner_ge_zero* [*simp*]: $0 \leq inner\ x\ x$
and *inner_eq_zero_iff* [*simp*]: $inner\ x\ x = 0 \longleftrightarrow x = 0$
and *norm_eq_sqrt_inner*: $norm\ x = sqrt\ (inner\ x\ x)$
begin

1.2.2 Class instances

instantiation *real* :: *real_inner*
begin

instantiation *complex* :: *real_inner*
begin

1.2.3 Gradient derivative

definition
gderiv :: [*'a*::*real_inner* ⇒ *real*, *'a*, *'a*] ⇒ *bool*
(‹(‹*notation*=‹*mixfix GDERIV*››*GDERIV* (_)/ (_)/ :> (_))› [*1000*, *1000*, *60*] *60*)
where
GDERIV f x :> *D* ⟷ *FDERIV f x* :> (λ*h*. *inner h D*)

end

1.3 Cartesian Products as Vector Spaces

theory *Product_Vector*
imports
Complex_Main
HOL−Library.Product_Plus
begin

1.3.1 Product is a Module

lemma *scale_prod*: *scale x* (*a*, *b*) = (*s1 x a*, *s2 x b*)

sublocale *p*: *module scale*

1.3.2 Product is a Real Vector Space

instantiation *prod* :: (*real_vector*, *real_vector*) *real_vector*
begin

proposition *scaleR_Pair* [*simp*]: *scaleR r* (*a*, *b*) = (*scaleR r a*, *scaleR r b*)

1.3.3 Product is a Metric Space

class *uniform_topological_monoid_add* = *topological_monoid_add* + *uniform_space* +
  assumes *uniformly_continuous_add'*:
    filterlim $(\lambda((a,b), (c,d)).\ (a + c,\ b + d))$ *uniformity* (*uniformity* $\times_F$ *uniformity*)

class *uniform_topological_group_add* = *topological_group_add* + *uniform_topological_monoid_add* +
  assumes *uniformly_continuous_uminus'*: *filterlim* $(\lambda(a, b).\ (-a, -b))$ *uniformity* *uniformity*
begin

instantiation *prod* :: (*metric_space*, *metric_space*) *metric_space*
begin

proposition *dist_Pair_Pair*: *dist* (*a*, *b*) (*c*, *d*) = *sqrt* ((*dist a c*)^2 + (*dist b d*)^2)

1.3.4 Product is a Complete Metric Space

instance *prod* :: (*complete_space*, *complete_space*) *complete_space*

1.3.5 Product is a Normed Vector Space

instantiation *prod* :: (*real_normed_vector*, *real_normed_vector*) *real_normed_vector*
begin

proposition *norm_Pair*: *norm* (*a*, *b*) = *sqrt* ((*norm a*)^2 + (*norm b*)^2)

instance *prod* :: (*banach*, *banach*) *banach*

proposition *has_derivative_Pair* [*derivative_intros*]:
  assumes *f*: (*f has_derivative f'*) (*at x within s*)
    and *g*: (*g has_derivative g'*) (*at x within s*)
  shows ((λx. (*f x*, *g x*)) *has_derivative* (λh. (*f' h*, *g' h*))) (*at x within s*)

1.3.6 Product is Finite Dimensional

proposition *dim_Times*:
  assumes *vs1.subspace S vs2.subspace T*
  shows *p.dim*(*S* × *T*) = *vs1.dim S* + *vs2.dim T*

end

1.4 Finite-Dimensional Inner Product Spaces

theory *Euclidean_Space*
imports
  L2_Norm
  Inner_Product
  Product_Vector
begin

1.4.1 Type class of Euclidean spaces

class *euclidean_space* = *real_inner* +
  fixes *Basis* :: *'a set*
  assumes *nonempty_Basis* [*simp*]: *Basis* ≠ {}
  assumes *finite_Basis* [*simp*]: *finite Basis*
  assumes *inner_Basis*:
    ⟦*u* ∈ *Basis*; *v* ∈ *Basis*⟧ ⟹ *inner u v* = (*if u* = *v then 1 else 0*)
  assumes *euclidean_all_zero_iff*:
    (∀ *u*∈*Basis*. *inner x u* = *0*) ⟷ (*x* = *0*)

1.4.2 Class instances

instantiation *real* :: *euclidean_space*
begin
instantiation *complex* :: *euclidean_space*
begin
instantiation *prod* :: (*real_inner*, *real_inner*) *real_inner*
begin

instantiation *prod* :: (*euclidean_space*, *euclidean_space*) *euclidean_space*
begin

1.4.3 Locale instances

end

1.5 Elementary Linear Algebra on Euclidean Spaces

theory *Linear_Algebra*
imports
  Euclidean_Space
  HOL−Library.Infinite_Set
begin

1.5.1 Substandard Basis

1.5.2 Orthogonality

definition (**in** *real_inner*) *orthogonal x y* $\longleftrightarrow$ $x \cdot y = 0$

1.5.3 Orthogonality of a transformation

definition *orthogonal_transformation f* $\longleftrightarrow$ *linear f* $\wedge$ $(\forall v\ w.\ f\ v \cdot f\ w = v \cdot w)$

1.5.4 Bilinear functions

definition
bilinear :: ('*a*::*real_vector* ⇒ '*b*::*real_vector* ⇒ '*c*::*real_vector*) ⇒ *bool* **where**
bilinear f $\longleftrightarrow$ $(\forall x.$ *linear* $(\lambda y.\ f\ x\ y)) \wedge (\forall y.$ *linear* $(\lambda x.\ f\ x\ y))$

1.5.5 Adjoints

definition *adjoint* :: (('*a*::*real_inner*) ⇒ ('*b*::*real_inner*)) ⇒ '*b* ⇒ '*a* **where**
adjoint f = (*SOME f'*. $\forall x\ y.\ f\ x \cdot y = x \cdot f'\ y$)

1.5.6 Infinity norm

definition *infnorm* (*x*::'*a*::*euclidean_space*) = *Sup* $\{|x \cdot b|\ |b.\ b \in$ *Basis*$\}$

1.5.7 Collinearity

definition *collinear* :: '*a*::*real_vector set* ⇒ *bool*
where *collinear S* $\longleftrightarrow$ $(\exists u.\ \forall x \in S.\ \forall\ y \in S.\ \exists c.\ x - y = c *_R u)$

1.5.8 Properties of special hyperplanes

proposition *dim_hyperplane*:
fixes *a* :: '*a*::*euclidean_space*
assumes $a \neq 0$
shows *dim* $\{x.\ a \cdot x = 0\}$ = *DIM*('*a*) − *1*

1.5.9 Orthogonal bases and Gram-Schmidt process

proposition *Gram_Schmidt_step*:
fixes *S* :: '*a*::*euclidean_space set*
assumes *S*: *pairwise orthogonal S* **and** *x*: $x \in$ *span S*

shows *orthogonal x* $(a - (\sum b{\in}S.\ (b \cdot a\ /\ (b \cdot b)) *_R\ b))$

proposition *orthogonal_extension*:
fixes *S* :: *'a::euclidean_space set*
assumes *S*: *pairwise orthogonal S*
obtains *U* **where** *pairwise orthogonal* $(S \cup U)$ *span* $(S \cup U) =$ *span* $(S \cup T)$

1.5.10 Decomposing a vector into parts in orthogonal subspaces

proposition *orthonormal_basis_subspace*:
fixes *S* :: *'a :: euclidean_space set*
assumes *subspace S*
obtains *B* **where** $B \subseteq S$ *pairwise orthogonal B*
and $\bigwedge x.\ x \in B \Longrightarrow$ *norm* $x = 1$
and *independent B card B* = *dim S span B* = *S*

proposition *dim_orthogonal_sum*:
fixes *A* :: *'a::euclidean_space set*
assumes $\bigwedge x\ y.\ [\![x \in A;\ y \in B]\!] \Longrightarrow x \cdot y = 0$
shows *dim*$(A \cup B)$ = *dim A* + *dim B*

1.5.11 Linear functions are (uniformly) continuous on any set

end

1.6 Affine Sets

theory *Affine*
imports *Linear_Algebra*
begin

1.6.1 Affine set and affine hull

definition *affine* :: *'a::real_vector set* ⇒ *bool*
where *affine S* $\longleftrightarrow (\forall x{\in}S.\ \forall y{\in}S.\ \forall u\ v.\ u + v = 1 \longrightarrow u *_R x + v *_R y \in S)$

1.6.2 Affine Dependence

definition *affine_dependent* :: *'a::real_vector set* ⇒ *bool*
where *affine_dependent S* ⟷ (∃*x*∈*S*. *x* ∈ *affine hull* (*S* − {*x*}))

proposition *affine_dependent_explicit*:
affine_dependent p ⟷
(∃*S U*. *finite S* ∧ *S* ⊆ *p* ∧ *sum U S* = *0* ∧ (∃*v*∈*S*. *U v* ≠ *0*) ∧ *sum* (λ*v*. *U v* $*_R$ *v*) *S* = *0*)

proposition *extend_to_affine_basis*:
fixes *S V* :: *'n::real_vector set*
assumes ¬ *affine_dependent S S* ⊆ *V*
obtains *T* **where** ¬ *affine_dependent T S* ⊆ *T T* ⊆ *V affine hull T* = *affine hull V*

1.6.3 Affine Dimension of a Set

definition *aff_dim* :: (*'a::euclidean_space*) *set* ⇒ *int*
where *aff_dim V* =
(*SOME d* :: *int*.
∃*B*. *affine hull B* = *affine hull V* ∧ ¬ *affine_dependent B* ∧ *of_nat* (*card B*) = *d* + *1*)

end

1.7 Convex Sets and Functions

theory *Convex*
imports
Affine HOL−Library.Set_Algebras HOL−Library.FuncSet
begin

1.7.1 Convex Sets

definition *convex* :: *'a::real_vector set* ⇒ *bool*
where *convex s* ⟷ (∀*x*∈*s*. ∀*y*∈*s*. ∀*u*≥*0*. ∀*v*≥*0*. *u* + *v* = *1* ⟶ *u* $*_R$ *x* + *v* $*_R$ *y* ∈ *s*)

1.7.2 Convex Functions on a Set

definition *convex_on* :: *'a::real_vector set* ⇒ (*'a* ⇒ *real*) ⇒ *bool*
where *convex_on S f* ⟷ *convex S* ∧
(∀*x*∈*S*. ∀*y*∈*S*. ∀*u*≥*0*. ∀*v*≥*0*. *u* + *v* = *1* ⟶ *f* (*u* $*_R$ *x* + *v* $*_R$ *y*) ≤ *u* * *f x* + *v* * *f y*)

definition *concave_on* :: *'a::real_vector set* ⇒ (*'a* ⇒ *real*) ⇒ *bool*
where *concave_on S f* ≡ *convex_on S* (λ*x*. − *f x*)

1.7.3 Convexity of the generalised binomial

1.7.4 Some inequalities: Applications of convexity

1.7.5 Misc related lemmas

1.7.6 Cones

definition *cone* :: *'a::real_vector set* ⇒ *bool*
where *cone s* ⟷ (∀ *x*∈*s*. ∀ *c*≥*0*. *c* $*_R$ *x* ∈ *s*)

proposition *cone_hull_expl*: *cone hull S* = {*c* $*_R$ *x* | *c x*. *c* ≥ *0* ∧ *x* ∈ *S*}
(**is** *?lhs* = *?rhs*)

1.7.7 Convex hull

proposition *convex_hull_indexed*:
fixes *S* :: *'a::real_vector set*
shows *convex hull S* =
{*y*. ∃ *k u x*. (∀ *i*∈{*1*::*nat* .. *k*}. *0* ≤ *u i* ∧ *x i* ∈ *S*) ∧
(*sum u* {*1*..*k*} = *1*) ∧ (∑ *i* = *1*..*k*. *u i* $*_R$ *x i*) = *y*}
(**is** *?xyz* = *?hull*)

1.7.8 Caratheodory's theorem

theorem *caratheodory*:
convex hull p =
{*x*::*'a::euclidean_space*. ∃ *S*. *finite S* ∧ *S* ⊆ *p* ∧ *card S* ≤ *DIM*(*'a*) + *1* ∧ *x* ∈ *convex hull S*}

1.8 Conic sets and conic hull

1.9 Convex cones and corresponding hulls

1.9.1 Radon's theorem

theorem *Radon*:
assumes *affine_dependent c*

obtains $M\ P$ **where** $M \subseteq c\ P \subseteq c\ M \cap P = \{\}\ (convex\ hull\ M) \cap (convex\ hull\ P) \neq \{\}$

1.9.2 Helly's theorem

theorem *Helly*:
fixes $\mathcal{F}$ :: *'a::euclidean_space set set*
assumes $card\ \mathcal{F} \geq DIM('a) + 1\ \forall s{\in}\mathcal{F}.\ convex\ s$
and $\bigwedge t.\ [\![t{\subseteq}\mathcal{F};\ card\ t = DIM('a) + 1]\!] \Longrightarrow \bigcap t \neq \{\}$
shows $\bigcap \mathcal{F} \neq \{\}$

1.9.3 Epigraphs of convex functions

definition $epigraph\ S\ (f :: \_ \Rightarrow real) = \{xy.\ fst\ xy \in S \wedge f\ (fst\ xy) \leq snd\ xy\}$

end

1.10 Definition of Finite Cartesian Product Type

theory *Finite_Cartesian_Product*
imports
Euclidean_Space
L2_Norm
HOL−Library.Numeral_Type
HOL−Library.Countable_Set
HOL−Library.FuncSet
begin

1.10.1 Cardinality of vectors

proposition *CARD_vec* [*simp*]:
$CARD('a^{\frown}'b) = CARD('a)\ \hat{}\ CARD('b)$
instantiation *vec* :: (*zero*, *finite*) *zero*
begin

instantiation *vec* :: (*plus*, *finite*) *plus*
begin

instantiation *vec* :: (*minus*, *finite*) *minus*
begin

instantiation *vec* :: (*uminus*, *finite*) *uminus*
begin
instantiation *vec* :: (*times*, *finite*) *times*
begin

instantiation *vec* :: (*one*, *finite*) *one*
begin

instantiation *vec* :: (*ord*, *finite*) *ord*
begin

1.10.2 Real vector space

definition $scaleR \equiv (\lambda\ r\ x.\ (\chi\ i.\ scaleR\ r\ (x\$i)))$

1.10.3 Topological space

definition [*code del*]:
  $open\ (S :: ('a\ \hat{}\ 'b)\ set) \longleftrightarrow$
    $(\forall\, x{\in}S.\ \exists\, A.\ (\forall\, i.\ open\ (A\ i) \wedge x\$i \in A\ i) \wedge$
      $(\forall\, y.\ (\forall\, i.\ y\$i \in A\ i) \longrightarrow y \in S))$

1.10.4 Metric space

definition
  $dist\ x\ y = L2\_set\ (\lambda i.\ dist\ (x\$i)\ (y\$i))\ UNIV$

definition [*code del*]:
  $(uniformity :: (('a\hat{}'b::\_) \times ('a\hat{}'b::\_))\ filter) =$
    $(INF\ e{\in}\{0 <..\}.\ principal\ \{(x, y).\ dist\ x\ y < e\})$

proposition *dist_vec_nth_le*: $dist\ (x\ \$\ i)\ (y\ \$\ i) \leq dist\ x\ y$

1.10.5 Normed vector space

definition $norm\ x = L2\_set\ (\lambda i.\ norm\ (x\$i))\ UNIV$

definition $sgn\ (x::'a\hat{}'b) = scaleR\ (inverse\ (norm\ x))\ x$

1.10.6 Inner product space

definition $inner\ x\ y = sum\ (\lambda i.\ inner\ (x\$i)\ (y\$i))\ UNIV$

1.10.7 Euclidean space

definition *axis k x* = (χ *i*. *if i* = *k then x else 0*)

definition *Basis* = ($\bigcup$ *i*. $\bigcup$ *u*∈*Basis*. {*axis i u*})

proposition *DIM_cart* [*simp*]: *DIM*('*a*^'*b*) = *CARD*('*b*) ∗ *DIM*('*a*)

1.10.8 Matrix operations

definition *map_matrix*::('*a* ⇒ '*b*) ⇒ (('*a*, '*i*::*finite*)*vec*, '*j*::*finite*) *vec* ⇒ (('*b*, '*i*)*vec*, '*j*) *vec* **where**
map_matrix f x = (χ *i j*. *f* (*x* $ *i* $ *j*))

definition *matrix_matrix_mult* :: ('*a*::*semiring_1*) ^'*n*^'*m* ⇒ '*a* ^'*p*^'*n* ⇒ '*a* ^ '*p* ^'*m*
(**infixl** ‹∗∗› *70*)
where *m* ∗∗ *m*' == (χ *i j*. *sum* (λ*k*. ((*m*$*i*)$*k*) ∗ ((*m*'$*k*)$*j*)) (*UNIV* :: '*n set*)) ::'*a* ^ '*p* ^'*m*

definition *matrix_vector_mult* :: ('*a*::*semiring_1*) ^'*n*^'*m* ⇒ '*a* ^'*n* ⇒ '*a* ^ '*m*
(**infixl** ‹∗*v*› *70*)
where *m* ∗*v* *x* ≡ (χ *i*. *sum* (λ*j*. ((*m*$*i*)$*j*) ∗ (*x*$*j*)) (*UNIV* ::'*n set*)) :: '*a*^'*m*

definition *vector_matrix_mult* :: '*a* ^ '*m* ⇒ ('*a*::*semiring_1*) ^'*n*^'*m* ⇒ '*a* ^'*n*
(**infixl** ‹*v*∗› *70*)
where *v v*∗ *m* == (χ *j*. *sum* (λ*i*. ((*v*$*i*) ∗ (*m*$*i*)$*j*)) (*UNIV* :: '*m set*)) :: '*a*^'*n*
definition *matrix* :: ('*a*::{*plus*,*times*, *one*, *zero*}^'*m* ⇒ '*a* ^ '*n*) ⇒ '*a*^'*m*^'*n*
where *matrix f* = (χ *i j*. (*f*(*axis j 1*))$*i*)

1.10.9 Inverse matrices (not necessarily square)

definition
invertible(*A*::'*a*::*semiring_1*^'*n*^'*m*) ⟷ (∃ *A*'::'*a*^'*m*^'*n*. *A* ∗∗ *A*' = *mat 1* ∧ *A*' ∗∗ *A* = *mat 1*)

definition
matrix_inv(*A*:: '*a*::*semiring_1*^'*n*^'*m*) =
(*SOME A*'::'*a*^'*m*^'*n*. *A* ∗∗ *A*' = *mat 1* ∧ *A*' ∗∗ *A* = *mat 1*)

end

1.11 Linear Algebra on Finite Cartesian Products

theory *Cartesian_Space*
imports
HOL−*Combinatorics.Transposition*
Finite_Cartesian_Product

Linear_Algebra
begin

1.11.1 Some interesting theorems and interpretations

1.11.2 Rank of a matrix

definition *rank* :: $'a$::*field*$\widehat{\ }'n\widehat{\ }'m$=>*nat*
where *row_rank_def_gen*: *rank A* ≡ *vec.dim*(*rows A*)

1.11.3 Orthogonality of a matrix

definition *orthogonal_matrix* (*Q*::$'a$::*semiring_1*$\widehat{\ }'n\widehat{\ }'n$) ⟷
transpose Q ∗∗ *Q* = *mat 1* ∧ *Q* ∗∗ *transpose Q* = *mat 1*

proposition *orthogonal_matrix_mul*:
fixes *A* :: *real* $\widehat{\ }'n\widehat{\ }'n$
assumes *orthogonal_matrix A orthogonal_matrix B*
shows *orthogonal_matrix*(*A* ∗∗ *B*)

proposition *orthogonal_transformation_matrix*:
fixes *f*:: *real*$\widehat{\ }'n$ ⇒ *real*$\widehat{\ }'n$
shows *orthogonal_transformation f* ⟷ *linear f* ∧ *orthogonal_matrix*(*matrix f*)
(**is** *?lhs* ⟷ *?rhs*)

1.11.4 Finding an Orthogonal Matrix

proposition *orthogonal_matrix_exists_basis*:
fixes *a* :: *real*$\widehat{\ }'n$
assumes *norm a* = *1*
obtains *A* **where** *orthogonal_matrix A A* ∗*v* (*axis k 1*) = *a*

proposition *orthogonal_transformation_exists*:
fixes *a b* :: *real*$\widehat{\ }'n$
assumes *norm a* = *norm b*
obtains *f* **where** *orthogonal_transformation f f a* = *b*

1.11.5 Scaling and isometry

proposition *scaling_linear*:

fixes f :: $'a$::*real_inner* ⇒ $'a$::*real_inner*
assumes *f0*: $f\ 0 = 0$
and *fd*: $\forall\, x\ y.\ dist\ (f\ x)\ (f\ y) = c * dist\ x\ y$
shows *linear f*
proposition *orthogonal_transformation_isometry*:
orthogonal_transformation f ⟷ $f(0::'a::real\_inner) = (0::'a) \wedge (\forall\, x\ y.\ dist(f\ x)\ (f\ y) = dist\ x\ y)$

1.11.6 Induction on matrix row operations

end

1.12 Traces and Determinants of Square Matrices

theory *Determinants*
imports
HOL−Combinatorics.Permutations
Cartesian_Space
begin

1.12.1 Trace

definition *trace* :: $'a::semiring\_1\,\hat{}\,'n\,\hat{}\,'n \Rightarrow 'a$
where *trace* $A = sum\ (\lambda i.\ ((A\$i)\$i))\ (UNIV::'n\ set)$

Definition of determinant

definition *det*:: $'a::comm\_ring\_1\,\hat{}\,'n\,\hat{}\,'n \Rightarrow 'a$ **where**
det A =
sum $(\lambda p.\ of\_int\ (sign\ p) * prod\ (\lambda i.\ A\$i\$p\ i)\ (UNIV :: 'n\ set))$
$\{p.\ p\ permutes\ (UNIV :: 'n\ set)\}$
proposition *det_diagonal*:
fixes A :: $'a::comm\_ring\_1\,\hat{}\,'n\,\hat{}\,'n$
assumes *ld*: $\bigwedge i\ j.\ i \neq j \Longrightarrow A\$i\$j = 0$
shows $det\ A = prod\ (\lambda i.\ A\$i\$i)\ (UNIV::'n\ set)$

proposition *det_matrix_scaleR* [*simp*]: $det\ (matrix\ (((*_R)\ r)) :: real\,\hat{}\,'n\,\hat{}\,'n) = r\ \hat{}\ CARD('n::finite)$

proposition *det_mul*:
fixes $A\ B$ :: $'a::comm\_ring\_1\,\hat{}\,'n\,\hat{}\,'n$
shows $det\ (A ** B) = det\ A * det\ B$

1.12.2 Relation to invertibility

proposition *invertible_det_nz*:
fixes *A*::*'a*::{*field*}^*'n*^*'n*
shows *invertible A* ⟷ *det A* ≠ *0*

Invertibility of matrices and corresponding linear functions

1.12.3 Cramer's rule

proposition *cramer_lemma*:
fixes *A* :: *'a*::{*field*}^*'n*^*'n*
shows *det*((χ *i j*. *if j* = *k then* (*A* ∗*v x*)$*i else A*$*i*$*j*):: *'a*::{*field*}^*'n*^*'n*) = *x*$*k* ∗ *det A*

proposition *cramer*:
fixes *A* ::*'a*::{*field*}^*'n*^*'n*
assumes *d0*: *det A* ≠ *0*
shows *A* ∗*v x* = *b* ⟷ *x* = (χ *k*. *det*(χ *i j*. *if j*=*k then b*$*i else A*$*i*$*j*) / *det A*)

proposition *det_orthogonal_matrix*:
fixes *Q*:: *'a*::*linordered_idom*^*'n*^*'n*
assumes *oQ*: *orthogonal_matrix Q*
shows *det Q* = *1* ∨ *det Q* = − *1*

proposition *orthogonal_transformation_det* [*simp*]:
fixes *f* :: *real*^*'n* ⇒ *real*^*'n*
shows *orthogonal_transformation f* ⟹ |*det* (*matrix f*)| = *1*

1.12.4 Rotation, reflection, rotoinversion

definition *rotation_matrix Q* ⟷ *orthogonal_matrix Q* ∧ *det Q* = *1*
definition *rotoinversion_matrix Q* ⟷ *orthogonal_matrix Q* ∧ *det Q* = − *1*

end

1.13 Operators involving abstract topology

theory *Abstract_Topology*
imports
Complex_Main
HOL−Library.Set_Idioms
HOL−Library.FuncSet
begin

1.13.1 General notion of a topology as a value

definition *istopology* :: $('a\ set \Rightarrow bool) \Rightarrow bool$ **where**
istopology $L \equiv (\forall S\ T.\ L\ S \longrightarrow L\ T \longrightarrow L\ (S \cap T)) \wedge (\forall \mathcal{K}.\ (\forall K{\in}\mathcal{K}.\ L\ K) \longrightarrow L\ (\bigcup \mathcal{K}))$

typedef *'a topology* = $\{L::('a\ set) \Rightarrow bool.\ istopology\ L\}$
morphisms *openin topology*
proposition *openin_clauses*:
fixes U :: *'a topology*
shows
openin U $\{\}$
$\bigwedge S\ T.\ openin\ U\ S \Longrightarrow openin\ U\ T \Longrightarrow openin\ U\ (S{\cap}T)$
$\bigwedge K.\ (\forall S \in K.\ openin\ U\ S) \Longrightarrow openin\ U\ (\bigcup K)$
definition *closedin* :: *'a topology* $\Rightarrow$ *'a set* $\Rightarrow$ *bool* **where**
$closedin\ U\ S \longleftrightarrow S \subseteq topspace\ U \wedge openin\ U\ (topspace\ U - S)$

1.13.2 The discrete topology

1.13.3 Subspace topology

definition *subtopology* :: *'a topology* $\Rightarrow$ *'a set* $\Rightarrow$ *'a topology*
where $subtopology\ U\ V = topology\ (\lambda T.\ \exists S.\ T = S \cap V \wedge openin\ U\ S)$

1.13.4 The canonical topology from the underlying type class

abbreviation *euclidean* :: *'a::topological_space topology*
where $euclidean \equiv topology\ open$

1.13.5 Basic "localization" results are handy for connectedness.

1.13.6 Derived set (set of limit points)

1.13.7 Closure with respect to a topological space

1.13.8 Frontier with respect to topological space

1.13.9 Locally finite collections

1.13.10 Continuous maps

lemma *continuous_map_alt*:
continuous_map T1 T2 f
$= ((\forall\, U.\ openin\ T2\ U \longrightarrow openin\ T1\ (f -` U \cap topspace\ T1)) \wedge f \in topspace\ T1 \to topspace\ T2)$

1.13.11 Open and closed maps (not a priori assumed continuous)

1.13.12 Quotient maps

1.13.13 Separated Sets

1.13.14 Homeomorphisms

1.13.15 Relation of homeomorphism between topological spaces

1.13.16 Connected topological spaces

1.13.17 Compact sets

proposition *compact_space_fip*:
compact_space X $\longleftrightarrow$
$(\forall \mathcal{U}.\ (\forall\, C{\in}\mathcal{U}.\ closedin\ X\ C) \wedge (\forall \mathcal{F}.\ finite\ \mathcal{F} \wedge \mathcal{F} \subseteq \mathcal{U} \longrightarrow \bigcap\mathcal{F} \neq \{\}) \longrightarrow \bigcap\mathcal{U} \neq \{\})$
(**is** _ = *?rhs*)

corollary *compactin_fip*:
compactin X S $\longleftrightarrow$
$S \subseteq topspace\ X\ \wedge$
$(\forall \mathcal{U}.\ (\forall\, C{\in}\mathcal{U}.\ closedin\ X\ C) \wedge (\forall \mathcal{F}.\ finite\ \mathcal{F} \wedge \mathcal{F} \subseteq \mathcal{U} \longrightarrow S \cap \bigcap\mathcal{F} \neq \{\}) \longrightarrow S \cap \bigcap\mathcal{U} \neq \{\})$

corollary *compact_space_imp_nest*:
fixes $C :: nat \Rightarrow 'a\ set$
assumes *compact_space X* **and** *clo*: $\bigwedge n.\ closedin\ X\ (C\ n)$
and *ne*: $\bigwedge n.\ C\ n \neq \{\}$ **and** *dec*: *decseq C*
shows $(\bigcap n.\ C\ n) \neq \{\}$

1.13.18 Embedding maps

1.13.19 Retraction and section maps

1.13.20 Continuity

1.13.21 The topology generated by some (open) subsets

1.13.22 Topology bases and sub-bases

1.13.23 Continuity via bases/subbases, hence upper and lower semicontinuity

1.13.24 Pullback topology

definition *pullback_topology*::(′*a set*) ⇒ (′*a* ⇒ ′*b*) ⇒ (′*b topology*) ⇒ (′*a topology*)
 where *pullback_topology A f T* = *topology* (λ*S*. ∃ *U*. *openin T U* ∧ *S* = *f*−‘*U* ∩ *A*)

proposition *continuous_map_pullback* [*intro*]:
 assumes *continuous_map T1 T2 g*
 shows *continuous_map* (*pullback_topology A f T1*) *T2* (*g o f*)

proposition *continuous_map_pullback′* [*intro*]:
 assumes *continuous_map T1 T2* (*f o g*) *topspace T1* ⊆ *g*−‘*A*
 shows *continuous_map T1* (*pullback_topology A f T2*) *g*

1.13.25 Proper maps (not a priori assumed continuous)

1.13.26 Perfect maps (proper, continuous and surjective)

end

1.14 *F*-Sigma and *G*-Delta sets in a Topological Space

theory *FSigma*
 imports *Abstract_Topology*
begin

end

1.15 Disjoint sum of arbitarily many spaces

theory *Sum_Topology*
imports *Abstract_Topology*
begin

end

Chapter 2

Topology

theory *Elementary_Topology*
imports
  HOL−Library.Set_Idioms
  HOL−Library.Disjoint_Sets
  Product_Vector
begin

2.1 Elementary Topology

2.1.1 Topological Basis

definition *topological_basis B* $\longleftrightarrow$
  $(\forall\, b{\in}B.\ open\ b) \wedge (\forall\, x.\ open\ x \longrightarrow (\exists\, B'.\ B' \subseteq B \wedge \bigcup B' = x))$

2.1.2 Countable Basis

locale *countable_basis* = *topological_space p* **for** *p*::'*a set* ⇒ *bool* +
  fixes *B* :: '*a set set*
  assumes *is_basis*: *topological_basis B*
    and *countable_basis*: *countable B*
begin

class *second_countable_topology* = *topological_space* +
  assumes *ex_countable_subbasis*:
    $\exists\, B$::'*a set set*. *countable B* ∧ *open* = *generate_topology B*
begin

proposition *Lindelof*:
  fixes $\mathcal{F}$ :: '*a*::*second_countable_topology set set*
  assumes $\mathcal{F}$: $\bigwedge S.\ S \in \mathcal{F} \Longrightarrow open\ S$
  obtains $\mathcal{F}'$ **where** $\mathcal{F}' \subseteq \mathcal{F}$ *countable* $\mathcal{F}'$ $\bigcup \mathcal{F}' = \bigcup \mathcal{F}$

2.1.3 Polish spaces

class *polish_space* = *complete_space* + *second_countable_topology*

2.1.4 Limit Points

definition (**in** *topological_space*) *islimpt*:: $'a \Rightarrow 'a\ set \Rightarrow bool$ (**infixr** ‹*islimpt*› *60*)
where $x\ islimpt\ S \longleftrightarrow (\forall\, T.\ x{\in}T \longrightarrow open\ T \longrightarrow (\exists\, y{\in}S.\ y{\in}T \wedge y{\neq}x))$

2.1.5 Interior of a Set

definition *interior* :: $('a{::}topological\_space)\ set \Rightarrow 'a\ set$ **where**
$interior\ S = \bigcup\{T.\ open\ T \wedge T \subseteq S\}$

2.1.6 Closure of a Set

definition *closure* :: $('a{::}topological\_space)\ set \Rightarrow 'a\ set$ **where**
$closure\ S = S \cup \{x\ .\ x\ islimpt\ S\}$

2.1.7 Frontier (also known as boundary)

definition *frontier* :: $('a{::}topological\_space)\ set \Rightarrow 'a\ set$ **where**
$frontier\ S = closure\ S - interior\ S$

2.1.8 Limits

2.1.9 Compactness

proposition *Heine_Borel_imp_Bolzano_Weierstrass*:
assumes *compact S*
and *infinite T*
and $T \subseteq S$
shows $\exists\, x \in S.\ x\ islimpt\ T$

definition *countably_compact* :: $('a{::}topological\_space)\ set \Rightarrow bool$ **where**
$countably\_compact\ U \longleftrightarrow$
$(\forall\, A.\ countable\ A \longrightarrow (\forall\, a{\in}A.\ open\ a) \longrightarrow U \subseteq \bigcup A$
$\longrightarrow (\exists\, T{\subseteq}A.\ finite\ T \wedge U \subseteq \bigcup T))$

proposition *countably_compact_imp_compact_second_countable*:
$countably\_compact\ U \Longrightarrow compact\ (U :: 'a :: second\_countable\_topology\ set)$

definition *seq_compact* :: *'a::topological_space set* ⇒ *bool* **where**
seq_compact S ⟷
(∀*f*. (∀*n*. *f n* ∈ *S*) ⟶ (∃*l*∈*S*. ∃*r::nat⇒nat*. *strict_mono r* ∧ (*f* ∘ *r*) ⟶ *l*))

proposition *Bolzano_Weierstrass_imp_seq_compact*:
fixes *S* :: *'a::{t1_space, first_countable_topology} set*
shows (⋀*T*. ⟦*infinite T*; *T* ⊆ *S*⟧ ⟹ ∃ *x* ∈ *S*. *x islimpt T*) ⟹ *seq_compact S*

2.1.10 Continuity

2.1.11 Homeomorphisms

definition *homeomorphism S T f g* ⟷
(∀*x*∈*S*. (*g*(*f x*) = *x*)) ∧ (*f* ‘ *S* = *T*) ∧ *continuous_on S f* ∧
(∀*y*∈*T*. (*f*(*g y*) = *y*)) ∧ (*g* ‘ *T* = *S*) ∧ *continuous_on T g*

definition *homeomorphic* :: *'a::topological_space set* ⇒ *'b::topological_space set* ⇒ *bool*
(**infixr** ‹*homeomorphic*› *60*)
where *s homeomorphic t* ≡ (∃*f g*. *homeomorphism s t f g*)

end
theory *Abstract_Limits*
imports
Abstract_Topology
begin

2.1.12 nhdsin and atin

2.1.13 Limits in a topological space

2.1.14 Pointwise continuity in topological spaces

2.1.15 Combining theorems for continuous functions into the reals

end

2.2 Non-Denumerability of the Continuum

theory *Continuum_Not_Denumerable*
imports
Complex_Main

HOL−Library.Countable_Set
begin

theorem *real_non_denum*: $\nexists f :: nat \Rightarrow real.\ surj\ f$

corollary *complex_non_denum*: $\nexists f :: nat \Rightarrow complex.\ surj\ f$

end

2.3 Abstract Topology 2

theory *Abstract_Topology_2*
imports
Elementary_Topology Abstract_Topology Continuum_Not_Denumerable
HOL−Library.Indicator_Function
HOL−Library.Equipollence
begin

2.3.1 Closure

corollary *infinite_openin*:
fixes $S :: 'a :: t1\_space\ set$
shows $[\![openin\ (top\_of\_set\ U)\ S;\ x \in S;\ x\ islimpt\ U]\!] \Longrightarrow infinite\ S$

2.3.2 Frontier

2.3.3 Compactness

2.3.4 Continuity

2.3.5 Retractions

definition *retraction* $:: ('a::topological\_space)\ set \Rightarrow 'a\ set \Rightarrow ('a \Rightarrow 'a) \Rightarrow bool$
where $retraction\ S\ T\ r \longleftrightarrow$
$T \subseteq S \land continuous\_on\ S\ r \land r \in S \rightarrow T \land (\forall x{\in}T.\ r\ x = x)$

definition *retract_of* (**infixl** ‹*retract′_of*› *50*) **where**
$T\ retract\_of\ S \longleftrightarrow (\exists r.\ retraction\ S\ T\ r)$

2.3.6 Retractions on a topological space

2.3.7 Paths and path-connectedness

2.3.8 Connected components

2.3.9 Combining theorems for continuous functions into the reals

2.3.10 A few cardinality results

end

2.4 Connected Components

theory *Connected*
  imports
    Abstract_Topology_2
begin

2.4.1 Connected components, considered as a connectedness relation or a set

definition $connected\_component\ S\ x\ y \equiv \exists T.\ connected\ T \wedge T \subseteq S \wedge x \in T \wedge y \in T$

2.4.2 The set of connected components of a set

definition $components{::}\ 'a{::}topological\_space\ set \Rightarrow 'a\ set\ set$
  where $components\ S \equiv connected\_component\_set\ S\ `\ S$

2.4.3 Lemmas about components

proposition *component_diff_connected*:
  fixes $S :: 'a{::}metric\_space\ set$
  assumes $connected\ S\ connected\ U\ S \subseteq U$ **and** $C{:}\ C \in components\ (U - S)$
  shows $connected(U - C)$

end

theory *Function_Topology*

imports
Elementary_Topology
Abstract_Limits
Connected
begin

2.5 Function Topology

2.5.1 The product topology

definition *product_topology*::$('i \Rightarrow ('a\ topology)) \Rightarrow ('i\ set) \Rightarrow (('i \Rightarrow 'a)\ topology)$
where *product_topology T I* =
topology_generated_by $\{(\Pi_E\ i{\in}I.\ X\ i)\ |X.\ (\forall i.\ openin\ (T\ i)\ (X\ i)) \wedge finite\ \{i.\ X\ i \neq topspace\ (T\ i)\}\}$

proposition *product_topology*:
product_topology X I =
topology
(*arbitrary union_of*
((*finite intersection_of*
$(\lambda F.\ \exists i\ U.\ F = \{f.\ f\ i \in U\} \wedge i \in I \wedge openin\ (X\ i)\ U))$
relative_to $(\Pi_E\ i{\in}I.\ topspace\ (X\ i))))$
(**is** _ = *topology* (_ *union_of* ((_ *intersection_of* ?Ψ) *relative_to* ?TOP)))

proposition *product_topology_open_contains_basis*:
assumes *openin* (*product_topology T I*) $U\ x \in U$
shows $\exists X.\ x \in (\Pi_E\ i{\in}I.\ X\ i) \wedge (\forall i.\ openin\ (T\ i)\ (X\ i)) \wedge finite\ \{i.\ X\ i \neq topspace\ (T\ i)\} \wedge (\Pi_E\ i{\in}I.\ X\ i) \subseteq U$

corollary *openin_product_topology_alt*:
openin (*product_topology X I*) $S \longleftrightarrow$
$(\forall x \in S.\ \exists U.\ finite\ \{i \in I.\ U\ i \neq topspace(X\ i)\} \wedge$
$(\forall i \in I.\ openin\ (X\ i)\ (U\ i)) \wedge x \in Pi_E\ I\ U \wedge Pi_E\ I\ U \subseteq S)$

corollary *closedin_product_topology*:
closedin (*product_topology X I*) (*PiE I S*) $\longleftrightarrow$ *PiE I S* = {} ∨ $(\forall i \in I.\ closedin\ (X\ i)\ (S\ i))$

corollary *closedin_product_topology_singleton*:
$f \in extensional\ I \Longrightarrow$ *closedin* (*product_topology X I*) $\{f\} \longleftrightarrow (\forall i \in I.\ closedin\ (X\ i)\ \{f\ i\})$

Powers of a single topological space as a topological space, using type classes

instantiation *fun* :: (*type*, *topological_space*) *topological_space*
begin

definition *open_fun_def*:

$open\ U = openin\ (product\_topology\ (\lambda i.\ euclidean)\ UNIV)\ U$

proposition *product_topology_basis'*:
fixes $x{::}'i \Rightarrow 'a$ **and** $U{::}'i \Rightarrow ('b{::}topological\_space)\ set$
assumes $finite\ I\ \bigwedge i.\ i \in I \Longrightarrow open\ (U\ i)$
shows $open\ \{f.\ \forall i{\in}I.\ f\ (x\ i) \in U\ i\}$

Topological countability for product spaces

proposition *product_topology_countable_basis*:
shows $\exists K{::}(('a{::}countable \Rightarrow 'b{::}second\_countable\_topology)\ set\ set).$
$topological\_basis\ K \wedge countable\ K \wedge$
$(\forall k{\in}K.\ \exists X.\ (k = Pi_E\ UNIV\ X) \wedge (\forall i.\ open\ (X\ i)) \wedge finite\ \{i.\ X\ i \neq UNIV\})$

2.5.2 The Alexander subbase theorem

theorem *Alexander_subbase*:
assumes X: $topology\ (arbitrary\ union\_of\ (finite\ intersection\_of\ (\lambda x.\ x \in \mathcal{B})\ relative\_to\ \bigcup\mathcal{B})) = X$
and fin: $\bigwedge C.\ [\![C \subseteq \mathcal{B};\ \bigcup C = topspace\ X]\!] \Longrightarrow \exists C'.\ finite\ C' \wedge C' \subseteq C \wedge \bigcup C' = topspace\ X$
shows $compact\_space\ X$

corollary *Alexander_subbase_alt*:
assumes $U \subseteq \bigcup\mathcal{B}$
and fin: $\bigwedge C.\ [\![C \subseteq \mathcal{B};\ U \subseteq \bigcup C]\!] \Longrightarrow \exists C'.\ finite\ C' \wedge C' \subseteq C \wedge U \subseteq \bigcup C'$
and X: $topology$
$(arbitrary\ union\_of$
$(finite\ intersection\_of\ (\lambda x.\ x \in \mathcal{B})\ relative\_to\ U)) = X$
shows $compact\_space\ X$

proposition *continuous_map_componentwise*:
$continuous\_map\ X\ (product\_topology\ Y\ I)\ f \longleftrightarrow$
$f\ `\ (topspace\ X) \subseteq extensional\ I \wedge (\forall k \in I.\ continuous\_map\ X\ (Y\ k)\ (\lambda x.\ f\ x\ k))$
(**is** $?lhs \longleftrightarrow \_ \wedge ?rhs$)

proposition *open_map_product_projection*:
assumes $i \in I$
shows $open\_map\ (product\_topology\ Y\ I)\ (Y\ i)\ (\lambda f.\ f\ i)$

2.5.3 Open Pi-sets in the product topology

proposition *openin_PiE_gen*:
openin (*product_topology X I*) (*PiE I S*) ⟷
PiE I S = {} ∨
finite {*i* ∈ *I*. *S i* ≠ *topspace* (*X i*)} ∧ (∀ *i* ∈ *I*. *openin* (*X i*) (*S i*))
(**is** *?lhs* ⟷ _ ∨ *?rhs*)

corollary *openin_PiE*:
finite I ⟹ *openin* (*product_topology X I*) (*PiE I S*) ⟷ *PiE I S* = {} ∨ (∀ *i* ∈ *I*. *openin* (*X i*) (*S i*))

proposition *compact_space_product_topology*:
compact_space(*product_topology X I*) ⟷
(*product_topology X I*) = *trivial_topology* ∨ (∀ *i* ∈ *I*. *compact_space*(*X i*))
(**is** *?lhs* = *?rhs*)

corollary *compactin_PiE*:
compactin (*product_topology X I*) (*PiE I S*) ⟷
PiE I S = {} ∨ (∀ *i* ∈ *I*. *compactin* (*X i*) (*S i*))

2.5.4 Relationship with connected spaces, paths, etc.

proposition *connected_space_product_topology*:
connected_space(*product_topology X I*) ⟷
(∃ *i* ∈ *I*. *X i* = *trivial_topology*) ∨ (∀ *i* ∈ *I*. *connected_space*(*X i*))
(**is** *?lhs* ⟷ *?eq* ∨ *?rhs*)

2.5.5 Projections from a function topology to a component

2.5.6 Limits

end

2.6 The binary product topology

theory *Product_Topology*
imports *Function_Topology*
begin

2.7 Product Topology

2.7.1 Definition

2.7.2 Continuity

proposition *compact_space_prod_topology*:
compact_space(*prod_topology X Y*) ⟷ (*prod_topology X Y*) = *trivial_topology* ∨ *compact_space X* ∧ *compact_space Y*

2.7.3 Homeomorphic maps

proposition *connected_space_prod_topology*:
connected_space(*prod_topology X Y*) ⟷
(*prod_topology X Y*) = *trivial_topology* ∨ *connected_space X* ∧ *connected_space Y* (**is** *?lhs=?rhs*)

end

2.8 T1 and Hausdorff spaces

theory *T1_Spaces*
imports *Product_Topology*
begin

2.9 T1 spaces with equivalences to many naturally "nice" properties.

proposition *t1_space_product_topology*:
t1_space (*product_topology X I*)
⟷ (*product_topology X I*) = *trivial_topology* ∨ (∀ *i* ∈ *I*. *t1_space* (*X i*))

2.9.1 Hausdorff Spaces

end

2.10 Lindelöf spaces

theory *Lindelof_Spaces*
imports *T1_Spaces*
begin

end

Chapter 3

Functional Analysis

theory *Metric_Arith*
  imports *HOL.Real_Vector_Spaces*
begin
theorem *metric_eq_thm* [*THEN HOL.eq_reflection*]:
  $x \in s \Longrightarrow y \in s \Longrightarrow x = y \longleftrightarrow (\forall\, a{\in}s.\ dist\ x\ a = dist\ y\ a)$

end

Chapter 4

Elementary Metric Spaces

theory *Elementary_Metric_Spaces*
  imports
    Abstract_Topology_2
    Metric_Arith
begin

4.1 Open and closed balls

definition *ball* :: *'a::metric_space ⇒ real ⇒ 'a set*
  where *ball x e = {y. dist x y < e}*

definition *cball* :: *'a::metric_space ⇒ real ⇒ 'a set*
  where *cball x e = {y. dist x y ≤ e}*

definition *sphere* :: *'a::metric_space ⇒ real ⇒ 'a set*
  where *sphere x e = {y. dist x y = e}*

4.2 Limit Points

4.3 Perfect Metric Spaces

4.4 Finite and discrete

4.5 Interior

4.6 Frontier

4.7 Limits

proposition *Lim*: *(f ⟶ l) net ⟷ trivial_limit net ∨ (∀ e>0. eventually (λx. dist (f x) l < e) net)*

proposition *Lim_within_le*: $(f \longrightarrow l)(at\ a\ within\ S) \longleftrightarrow$
$(\forall e>0.\ \exists d>0.\ \forall x\in S.\ 0 < dist\ x\ a \wedge dist\ x\ a \le d \longrightarrow dist\ (f\ x)\ l < e)$

proposition *Lim_within*: $(f \longrightarrow l)\ (at\ a\ within\ S) \longleftrightarrow$
$(\forall e >0.\ \exists d>0.\ \forall x \in S.\ 0 < dist\ x\ a \wedge dist\ x\ a\ < d \longrightarrow dist\ (f\ x)\ l < e)$

corollary *Lim_withinI* [*intro?*]:
assumes $\bigwedge e.\ e > 0 \Longrightarrow \exists d>0.\ \forall x \in S.\ 0 < dist\ x\ a \wedge dist\ x\ a < d \longrightarrow dist\ (f\ x)\ l \le e$
shows $(f \longrightarrow l)\ (at\ a\ within\ S)$

proposition *Lim_at*: $(f \longrightarrow l)\ (at\ a) \longleftrightarrow$
$(\forall e >0.\ \exists d>0.\ \forall x.\ 0 < dist\ x\ a \wedge dist\ x\ a < d\ \longrightarrow dist\ (f\ x)\ l < e)$

4.8 Continuity

proposition *continuous_within_eps_delta*:
$continuous\ (at\ x\ within\ s)\ f \longleftrightarrow (\forall e>0.\ \exists d>0.\ \forall x'\in s.\ \ dist\ x'\ x < d\ {-}{-}{>}\ dist\ (f\ x')\ (f\ x) < e)$

corollary *continuous_at_eps_delta*:
$continuous\ (at\ x)\ f \longleftrightarrow (\forall e > 0.\ \exists d > 0.\ \forall x'.\ dist\ x'\ x < d \longrightarrow dist\ (f\ x')\ (f\ x) < e)$

4.9 Closure and Limit Characterization

4.10 Boundedness

definition (**in** *metric_space*) *bounded* :: *'a set* $\Rightarrow$ *bool*
where $bounded\ S \longleftrightarrow (\exists x\ e.\ \forall y\in S.\ dist\ x\ y \le e)$

4.11 Compactness

proposition *seq_compact_imp_totally_bounded*:
assumes *seq_compact S*
shows $\forall e>0.\ \exists k.\ finite\ k \wedge k \subseteq S \wedge S \subseteq (\bigcup x\in k.\ ball\ x\ e)$
proposition *seq_compact_imp_Heine_Borel*:
fixes *S* :: *'a* :: *metric_space set*
assumes *seq_compact S*
shows *compact S*

proposition *compact_eq_seq_compact_metric*:
compact (*S* :: *'a::metric_space set*) $\longleftrightarrow$ *seq_compact S*

proposition *compact_def*: — this is the definition of compactness in HOL Light
$compact\ (S :: 'a::metric\_space\ set) \longleftrightarrow$
$(\forall f.\ (\forall n.\ f\ n \in S) \longrightarrow (\exists l{\in}S.\ \exists r{::}nat{\Rightarrow}nat.\ strict\_mono\ r \wedge (f \circ r) \longlongrightarrow l))$
proposition *compact_eq_Bolzano_Weierstrass*:
fixes $S :: 'a::metric\_space\ set$
shows $compact\ S \longleftrightarrow (\forall T.\ infinite\ T \wedge T \subseteq S \longrightarrow (\exists x \in S.\ x\ islimpt\ T))$

proposition *Bolzano_Weierstrass_imp_bounded*:
$(\bigwedge T.\ [\![infinite\ T;\ T \subseteq S]\!] \Longrightarrow (\exists x \in S.\ x\ islimpt\ T)) \Longrightarrow bounded\ S$

4.12 Banach fixed point theorem

theorem *banach_fix*:— TODO: rename to *Banach_fix*
assumes s: $complete\ s\ s \neq \{\}$
and c: $0 \leq c\ c < 1$
and f: $f\ `\ s \subseteq s$
and *lipschitz*: $\forall x{\in}s.\ \forall y{\in}s.\ dist\ (f\ x)\ (f\ y) \leq c * dist\ x\ y$
shows $\exists !x{\in}s.\ f\ x = x$

4.13 Edelstein fixed point theorem

theorem *Edelstein_fix*:
fixes $S :: 'a::metric\_space\ set$
assumes S: $compact\ S\ S \neq \{\}$
and *gs*: $(g\ `\ S) \subseteq S$
and *dist*: $\forall x{\in}S.\ \forall y{\in}S.\ x \neq y \longrightarrow dist\ (g\ x)\ (g\ y) < dist\ x\ y$
shows $\exists !x{\in}S.\ g\ x = x$

4.14 The diameter of a set

definition $diameter :: 'a::metric\_space\ set \Rightarrow real$ **where**
$diameter\ S = (if\ S = \{\}\ then\ 0\ else\ SUP\ (x,y){\in}S{\times}S.\ dist\ x\ y)$

proposition *Lebesgue_number_lemma*:
assumes $compact\ S\ \mathcal{C} \neq \{\}\ S \subseteq \bigcup\mathcal{C}$ **and** *ope*: $\bigwedge B.\ B \in \mathcal{C} \Longrightarrow open\ B$
obtains δ **where** $0 < \delta\ \bigwedge T.\ [\![T \subseteq S;\ diameter\ T < \delta]\!] \Longrightarrow \exists B \in \mathcal{C}.\ T \subseteq B$

4.15 Metric spaces with the Heine-Borel property

class $heine\_borel = metric\_space +$
assumes *bounded_imp_convergent_subsequence*:

bounded (*range f*) $\Longrightarrow$ $\exists$ *l r*. *strict_mono* (*r*::*nat*$\Rightarrow$*nat*) $\wedge$ ((*f* $\circ$ *r*) $\longrightarrow$ *l*)
sequentially

proposition *bounded_closed_imp_seq_compact*:
fixes *S*::'*a*::*heine_borel set*
assumes *bounded S*
and *closed S*
shows *seq_compact S*

instance *real* :: *heine_borel*

instance *prod* :: (*heine_borel*, *heine_borel*) *heine_borel*

4.16 Completeness

proposition (**in** *metric_space*) *completeI*:
assumes $\bigwedge f.\ \forall n.\ f\ n \in s \Longrightarrow Cauchy\ f \Longrightarrow \exists l{\in}s.\ f \longrightarrow l$
shows *complete s*

proposition (**in** *metric_space*) *completeE*:
assumes *complete s* **and** $\forall n.\ f\ n \in s$ **and** *Cauchy f*
obtains *l* **where** $l \in s$ **and** $f \longrightarrow l$

proposition *compact_eq_totally_bounded*:
$compact\ s \longleftrightarrow complete\ s \wedge (\forall e{>}0.\ \exists k.\ finite\ k \wedge s \subseteq (\bigcup x{\in}k.\ ball\ x\ e))$
(**is** _ $\longleftrightarrow$ *?rhs*)

4.17 Cauchy continuity

4.18 Properties of Balls and Spheres

4.19 Distance from a Set

4.20 Infimum Distance

definition *infdist x A* = (*if A* = {} *then 0 else INF a*$\in$*A*. *dist x a*)

4.21 Separation between Points and Sets

proposition *separate_point_closed*:
fixes *S* :: '*a*::*heine_borel set*
assumes *closed S* **and** $a \notin S$
shows $\exists d{>}0.\ \forall x{\in}S.\ d \leq dist\ a\ x$

proposition *separate_compact_closed*:
  fixes *S T* :: *'a::heine_borel set*
  assumes *compact S*
    and *T*: *closed T S* ∩ *T* = {}
  shows ∃ *d*>*0*. ∀ *x*∈*S*. ∀ *y*∈*T*. *d* ≤ *dist x y*

proposition *separate_closed_compact*:
  fixes *S T* :: *'a::heine_borel set*
  assumes *S*: *closed S*
    and *T*: *compact T*
    and *dis*: *S* ∩ *T* = {}
  shows ∃ *d*>*0*. ∀ *x*∈*S*. ∀ *y*∈*T*. *d* ≤ *dist x y*

proposition *compact_in_open_separated*:
  fixes *A*::*'a::heine_borel set*
  assumes *A*: *A* ≠ {} *compact A*
  assumes *open B*
  assumes *A* ⊆ *B*
  obtains *e* **where** *e* > *0* {*x*. *infdist x A* ≤ *e*} ⊆ *B*

4.22 Uniform Continuity

4.23 Continuity on a Compact Domain Implies Uniform Continuity

corollary *compact_uniformly_continuous*:
  fixes *f* :: *'a* :: *metric_space* ⇒ *'b* :: *metric_space*
  assumes *f*: *continuous_on S f* **and** *S*: *compact S*
  shows *uniformly_continuous_on S f*

4.24 With Abstract Topology (TODO: move and remove dependency?)

4.25 Closed Nest

4.26 Consequences for Real Numbers

4.27 The infimum of the distance between two sets

definition *setdist* :: *'a::metric_space set* ⇒ *'a set* ⇒ *real* **where**
  setdist s t ≡
      (*if s* = {} ∨ *t* = {} *then 0*

else Inf $\{$*dist x y*$|$ *x y*. $x \in s \wedge y \in t\})$

proposition *setdist_attains_inf*:
assumes *compact B B* $\neq \{\}$
obtains *y* **where** $y \in B$ *setdist A B* = *infdist y A*

4.28 Diameter Lemma

end

4.29 Elementary Normed Vector Spaces

theory *Elementary_Normed_Spaces*
imports
HOL−Library.FuncSet
Elementary_Metric_Spaces Cartesian_Space
Connected
begin

4.29.1 Orthogonal Transformation of Balls

4.29.2 Support

4.29.3 Intervals

4.29.4 Limit Points

4.29.5 Balls and Spheres in Normed Spaces

corollary *compact_sphere* [*simp*]:
fixes *a* :: *'a*::{*real_normed_vector*,*perfect_space*,*heine_borel*}
shows *compact* (*sphere a r*)

corollary *bounded_sphere* [*simp*]:
fixes *a* :: *'a*::{*real_normed_vector*,*perfect_space*,*heine_borel*}
shows *bounded* (*sphere a r*)

corollary *closed_sphere* [*simp*]:
fixes *a* :: *'a*::{*real_normed_vector*,*perfect_space*,*heine_borel*}
shows *closed* (*sphere a r*)

4.29.6 Filters

4.29.7 Trivial Limits

4.29.8 Limits

proposition *Lim_at_infinity*: $(f \longrightarrow l)$ *at_infinity* $\longleftrightarrow (\forall e>0.\ \exists b.\ \forall x.\ norm\ x \geq b \longrightarrow dist\ (f\ x)\ l < e)$

corollary *Lim_at_infinityI* [*intro?*]:
assumes $\bigwedge e.\ e > 0 \Longrightarrow \exists B.\ \forall x.\ norm\ x \geq B \longrightarrow dist\ (f\ x)\ l \leq e$
shows $(f \longrightarrow l)$ *at_infinity*

4.29.9 Boundedness

corollary *cobounded_imp_unbounded*:
fixes S :: *'a*::{*real_normed_vector*, *perfect_space*} *set*
shows *bounded* $(- S) \Longrightarrow \neg$ *bounded* S

4.29.10 Normed spaces with the Heine-Borel property

4.29.11 Intersecting chains of compact sets and the Baire property

proposition *bounded_closed_chain*:
fixes $\mathcal{F}$ :: *'a::heine_borel set set*
assumes $B \in \mathcal{F}$ *bounded* B **and** $\mathcal{F}$: $\bigwedge S.\ S \in \mathcal{F} \Longrightarrow closed\ S$ **and** $\{\} \notin \mathcal{F}$
and *chain*: $\bigwedge S\ T.\ S \in \mathcal{F} \wedge T \in \mathcal{F} \Longrightarrow S \subseteq T \vee T \subseteq S$
shows $\bigcap \mathcal{F} \neq \{\}$

corollary *compact_chain*:
fixes $\mathcal{F}$ :: *'a::heine_borel set set*
assumes $\bigwedge S.\ S \in \mathcal{F} \Longrightarrow compact\ S\ \{\} \notin \mathcal{F}$
$\bigwedge S\ T.\ S \in \mathcal{F} \wedge T \in \mathcal{F} \Longrightarrow S \subseteq T \vee T \subseteq S$
shows $\bigcap\ \mathcal{F} \neq \{\}$
theorem *Baire*:
fixes S::*'a*::{*real_normed_vector*,*heine_borel*} *set*
assumes *closed* S *countable* $\mathcal{G}$
and *ope*: $\bigwedge T.\ T \in \mathcal{G} \Longrightarrow openin\ (top\_of\_set\ S)\ T \wedge S \subseteq closure\ T$
shows $S \subseteq closure(\bigcap \mathcal{G})$

4.29.12 Continuity

proposition *homeomorphic_ball_UNIV*:
fixes a ::*'a::real_normed_vector*

assumes *0 < r* **shows** *ball a r homeomorphic (UNIV:: 'a set)*

4.29.13 Connected Normed Spaces

end

4.30 Linear Decision Procedure for Normed Spaces

theory *Norm_Arith*
imports *HOL−Library.Sum_of_Squares*
begin

method_setup *norm* = ‹
Scan.succeed (SIMPLE_METHOD' o NormArith.norm_arith_tac)
› *prove simple linear statements about vector norms*

proposition *dist_triangle_add*:
fixes *x y x' y' :: 'a::real_normed_vector*
shows *dist (x + y) (x' + y') ≤ dist x x' + dist y y'*

end

Chapter 5

Vector Analysis

theory *Topology_Euclidean_Space*
  imports
    Elementary_Normed_Spaces
    Linear_Algebra
    Norm_Arith
begin

5.1 Elementary Topology in Euclidean Space

5.1.1 Boxes

abbreviation *One* :: *'a::euclidean_space* **where**
One ≡ $\sum$ *Basis*

definition (**in** *euclidean_space*) *eucl_less* (**infix** ‹<e› *50*) **where**
eucl_less a b ⟷ (∀ *i*∈*Basis*. *a* • *i* < *b* • *i*)

definition *box_eucl_less*: *box a b* = {*x*. *a* <*e x* ∧ *x* <*e b*}
definition *cbox a b* = {*x*. ∀ *i*∈*Basis*. *a* • *i* ≤ *x* • *i* ∧ *x* • *i* ≤ *b* • *i*}

corollary *open_countable_Union_open_box*:
  fixes *S* :: *'a* :: *euclidean_space set*
  assumes *open S*
  obtains 𝒟 **where** *countable* 𝒟 𝒟 ⊆ *Pow S* ⋀*X*. *X* ∈ 𝒟 ⟹ ∃ *a b*. *X* = *box a b*
⋃𝒟 = *S*

corollary *open_countable_Union_open_cbox*:
  fixes *S* :: *'a* :: *euclidean_space set*
  assumes *open S*
  obtains 𝒟 **where** *countable* 𝒟 𝒟 ⊆ *Pow S* ⋀*X*. *X* ∈ 𝒟 ⟹ ∃ *a b*. *X* = *cbox a*
b ⋃𝒟 = *S*

5.1.2 General Intervals

definition *is_interval* $(s::('a::euclidean\_space)\ set) \longleftrightarrow$
$(\forall a \in s.\ \forall b \in s.\ \forall x.\ (\forall i \in Basis.\ ((a \cdot i \le x \cdot i \wedge x \cdot i \le b \cdot i) \vee (b \cdot i \le x \cdot i \wedge x \cdot i \le a \cdot i)))$
$\longrightarrow x \in s)$

5.1.3 Limit Component Bounds

5.1.4 Class Instances

instance *euclidean_space* ⊆ *heine_borel*

instance *euclidean_space* ⊆ *banach*

5.1.5 Compact Boxes

proposition *is_interval_compact*:
$is\_interval\ S \wedge compact\ S \longleftrightarrow (\exists\, a\ b.\ S = cbox\ a\ b)$ (**is** *?lhs = ?rhs*)

proposition *tendsto_componentwise_iff*:
fixes $f :: \_ \Rightarrow 'b::euclidean\_space$
shows $(f \longrightarrow l)\ F \longleftrightarrow (\forall i \in Basis.\ ((\lambda x.\ (f\ x \cdot i)) \longrightarrow (l \cdot i))\ F)$
(**is** *?lhs = ?rhs*)

corollary *continuous_componentwise*:
$continuous\ F\ f \longleftrightarrow (\forall i \in Basis.\ continuous\ F\ (\lambda x.\ (f\ x \cdot i)))$

corollary *continuous_on_componentwise*:
fixes $S :: 'a :: t2\_space\ set$
shows $continuous\_on\ S\ f \longleftrightarrow (\forall i \in Basis.\ continuous\_on\ S\ (\lambda x.\ (f\ x \cdot i)))$

5.1.6 Separability

proposition *separable*:
fixes $S :: 'a::\{metric\_space, second\_countable\_topology\}\ set$
obtains T **where** $countable\ T\ T \subseteq S\ S \subseteq closure\ T$
proposition *open_surjective_linear_image*:
fixes $f :: 'a::real\_normed\_vector \Rightarrow 'b::euclidean\_space$
assumes $open\ A\ linear\ f\ surj\ f$
shows $open(f\ `\ A)$

corollary *open_bijective_linear_image_eq*:
  fixes *f* :: *'a::euclidean_space ⇒ 'b::euclidean_space*
  assumes *linear f bij f*
    shows *open(f ' A) ⟷ open A*

corollary *interior_bijective_linear_image*:
  fixes *f* :: *'a::euclidean_space ⇒ 'b::euclidean_space*
  assumes *linear f bij f*
  shows *interior (f ' S) = f ' interior S*
proposition *injective_imp_isometric*:
  fixes *f* :: *'a::euclidean_space ⇒ 'b::euclidean_space*
  assumes *s*: *closed s subspace s*
    and *f*: *bounded_linear f ∀ x∈s. f x = 0 ⟶ x = 0*
  shows *∃ e>0. ∀ x∈s. norm (f x) ≥ e * norm x*

proposition *closed_injective_image_subspace*:
  fixes *f* :: *'a::euclidean_space ⇒ 'b::euclidean_space*
  assumes *subspace s bounded_linear f ∀ x∈s. f x = 0 ⟶ x = 0 closed s*
  shows *closed(f ' s)*

5.1.7 Set Distance

corollary *setdist_gt_0_compact_closed*:
  assumes *S*: *compact S* **and** *T*: *closed T*
    shows *setdist S T > 0 ⟷ (S ≠ {} ∧ T ≠ {} ∧ S ∩ T = {})*

end

5.2 Line Segment

theory *Line_Segment*
imports
  Convex
  Topology_Euclidean_Space
begin

corollary *component_complement_connected*:
  fixes *S* :: *'a::real_normed_vector set*
  assumes *connected S C ∈ components (−S)*
  shows *connected(−C)*

proposition *clopen*:
  fixes *S* :: *'a :: real_normed_vector set*
  shows *closed S ∧ open S ⟷ S = {} ∨ S = UNIV*

corollary *compact_open*:

fixes *S* :: *'a* :: *euclidean_space set*
shows *compact S* ∧ *open S* ⟷ *S* = {}

corollary *finite_imp_not_open*:
fixes *S* :: *'a*::{*real_normed_vector*, *perfect_space*} *set*
shows ⟦*finite S*; *open S*⟧ ⟹ *S*={}

corollary *empty_interior_finite*:
fixes *S* :: *'a*::{*real_normed_vector*, *perfect_space*} *set*
shows *finite S* ⟹ *interior S* = {}

5.2.1 Midpoint

definition *midpoint* :: *'a*::*real_vector* ⇒ *'a* ⇒ *'a*
where *midpoint a b* = (*inverse* (*2*::*real*)) $*_R$ (*a* + *b*)

5.2.2 Open and closed segments

definition *closed_segment* :: *'a*::*real_vector* ⇒ *'a* ⇒ *'a set*
where *closed_segment a b* = {(*1* − *u*) $*_R$ *a* + *u* $*_R$ *b* | *u*::*real*. *0* ≤ *u* ∧ *u* ≤ *1*}

definition *open_segment* :: *'a*::*real_vector* ⇒ *'a* ⇒ *'a set* **where**
open_segment a b ≡ *closed_segment a b* − {*a*,*b*}

proposition *dist_decreases_open_segment*:
fixes *a* :: *'a* :: *euclidean_space*
assumes *x* ∈ *open_segment a b*
shows *dist c x* < *dist c a* ∨ *dist c x* < *dist c b*

corollary *open_segment_furthest_le*:
fixes *a b x y* :: *'a*::*euclidean_space*
assumes *x* ∈ *open_segment a b*
shows *norm* (*y* − *x*) < *norm* (*y* − *a*) ∨ *norm* (*y* − *x*) < *norm* (*y* − *b*)

corollary *dist_decreases_closed_segment*:
fixes *a* :: *'a* :: *euclidean_space*
assumes *x* ∈ *closed_segment a b*
shows *dist c x* ≤ *dist c a* ∨ *dist c x* ≤ *dist c b*

corollary *segment_furthest_le*:
fixes *a b x y* :: *'a*::*euclidean_space*
assumes *x* ∈ *closed_segment a b*
shows *norm* (*y* − *x*) ≤ *norm* (*y* − *a*) ∨ *norm* (*y* − *x*) ≤ *norm* (*y* − *b*)

5.2.3 Betweenness

definition *between* = (λ(*a*,*b*) *x*. *x* $\in$ *closed_segment a b*)

end

5.3 Convex Sets and Functions on (Normed) Euclidean Spaces

theory *Convex_Euclidean_Space*
imports
Convex Topology_Euclidean_Space Line_Segment
begin
corollary *empty_interior_lowdim*:
fixes *S* :: *'n*::*euclidean_space set*
shows *dim S* < *DIM* (*'n*) $\Longrightarrow$ *interior S* = {}

corollary *aff_dim_nonempty_interior*:
fixes *S* :: *'a*::*euclidean_space set*
shows *interior S* $\neq$ {} $\Longrightarrow$ *aff_dim S* = *DIM*(*'a*)

5.3.1 Relative interior of a set

definition *rel_interior S* =
{*x*. $\exists$ *T*. *openin* (*top_of_set* (*affine hull S*)) *T* $\wedge$ *x* $\in$ *T* $\wedge$ *T* $\subseteq$ *S*}
definition *rel_open S* $\longleftrightarrow$ *rel_interior S* = *S*

5.3.2 Closest point of a convex set is unique, with a continuous projection

definition *closest_point* :: *'a*::{*real_inner*,*heine_borel*} *set* $\Rightarrow$ *'a* $\Rightarrow$ *'a*
where *closest_point S a* = (*SOME x*. *x* $\in$ *S* $\wedge$ ($\forall$ *y*$\in$*S*. *dist a x* $\leq$ *dist a y*))

proposition *closest_point_in_rel_interior*:
assumes *closed S S* $\neq$ {} **and** *x*: *x* $\in$ *affine hull S*
shows *closest_point S x* $\in$ *rel_interior S* $\longleftrightarrow$ *x* $\in$ *rel_interior S*

end

Chapter 6

Unsorted

theory *Starlike*
  imports
    Convex_Euclidean_Space
    Line_Segment
begin

6.0.1 The relative frontier of a set

definition *rel_frontier S = closure S − rel_interior S*

proposition *ray_to_rel_frontier*:
  fixes *a* :: *'a::real_inner*
  assumes *bounded S*
    and *a*: *a ∈ rel_interior S*
    and *aff*: *(a + l) ∈ affine hull S*
    and *l ≠ 0*
  obtains *d* **where** *0 < d (a + d *R l) ∈ rel_frontier S*
    *⋀e. ⟦0 ≤ e; e < d⟧ ⟹ (a + e *R l) ∈ rel_interior S*

corollary *ray_to_frontier*:
  fixes *a* :: *'a::euclidean_space*
  assumes *bounded S*
    and *a*: *a ∈ interior S*
    and *l ≠ 0*
  obtains *d* **where** *0 < d (a + d *R l) ∈ frontier S*
    *⋀e. ⟦0 ≤ e; e < d⟧ ⟹ (a + e *R l) ∈ interior S*

proposition *rel_frontier_not_sing*:
  fixes *a* :: *'a::euclidean_space*
  assumes *bounded S*
    shows *rel_frontier S ≠ {a}*

6.0.2 Coplanarity, and collinearity in terms of affine hull

definition *coplanar* **where**
coplanar S $\equiv \exists$ *u v w. S* $\subseteq$ *affine hull* $\{u,v,w\}$

6.0.3 Connectedness of the intersection of a chain

proposition *connected_chain*:
fixes $\mathcal{F}$ :: *'a :: euclidean_space set set*
assumes *cc*: $\bigwedge S.\ S \in \mathcal{F} \Longrightarrow$ *compact S* $\wedge$ *connected S*
and *linear*: $\bigwedge S\ T.\ S \in \mathcal{F} \wedge T \in \mathcal{F} \Longrightarrow S \subseteq T \vee T \subseteq S$
shows *connected*$(\bigcap \mathcal{F})$

6.0.4 Proper maps, including projections out of compact sets

proposition *proper_map*:
fixes *f* :: *'a::heine_borel* $\Rightarrow$ *'b::heine_borel*
assumes *closedin* (*top_of_set S*) *K*
and *com*: $\bigwedge U.\ [\![U \subseteq T;\ \textit{compact}\ U]\!] \Longrightarrow$ *compact* $(S \cap f -\text{`}\ U)$
and *f* ‘ $S \subseteq T$
shows *closedin* (*top_of_set T*) (*f* ‘ *K*)

6.0.5 Closure of conic hulls

proposition *closedin_conic_hull*:
fixes *S* :: *'a::euclidean_space set*
assumes *compact T* $0 \notin T$ $T \subseteq S$
shows *closedin* (*top_of_set* (*conic hull S*)) (*conic hull T*)

corollary *affine_hull_convex_Int_open*:
fixes *S* :: *'a::real_normed_vector set*
assumes *convex S open T* $S \cap T \neq \{\}$
shows *affine hull* $(S \cap T)$ = *affine hull S*

corollary *affine_hull_affine_Int_nonempty_interior*:
fixes *S* :: *'a::real_normed_vector set*
assumes *affine S* $S \cap$ *interior* $T \neq \{\}$
shows *affine hull* $(S \cap T)$ = *affine hull S*

corollary *affine_hull_affine_Int_open*:
fixes *S* :: *'a::real_normed_vector set*

assumes *affine S open T S* $\cap$ *T* $\neq$ {}
shows *affine hull* (*S* $\cap$ *T*) = *affine hull S*

corollary *affine_hull_convex_Int_openin*:
fixes *S* :: *'a::real_normed_vector set*
assumes *convex S openin* (*top_of_set* (*affine hull S*)) *T S* $\cap$ *T* $\neq$ {}
shows *affine hull* (*S* $\cap$ *T*) = *affine hull S*

corollary *affine_hull_openin*:
fixes *S* :: *'a::real_normed_vector set*
assumes *openin* (*top_of_set* (*affine hull T*)) *S S* $\neq$ {}
shows *affine hull S* = *affine hull T*

corollary *affine_hull_open*:
fixes *S* :: *'a::real_normed_vector set*
assumes *open S S* $\neq$ {}
shows *affine hull S* = *UNIV*

proposition *aff_dim_eq_hyperplane*:
fixes *S* :: *'a::euclidean_space set*
shows *aff_dim S* = *DIM*(*'a*) − *1* $\longleftrightarrow$ ($\exists$ *a b*. *a* $\neq$ *0* $\wedge$ *affine hull S* = {*x*. *a* · *x* = *b*})
(**is** *?lhs* = *?rhs*)

corollary *aff_dim_hyperplane* [*simp*]:
fixes *a* :: *'a::euclidean_space*
shows *a* $\neq$ *0* $\Longrightarrow$ *aff_dim* {*x*. *a* · *x* = *r*} = *DIM*(*'a*) − *1*

proposition *aff_dim_sums_Int*:
assumes *affine S*
and *affine T*
and *S* $\cap$ *T* $\neq$ {}
shows *aff_dim* {*x* + *y*| *x y*. *x* $\in$ *S* $\wedge$ *y* $\in$ *T*} = (*aff_dim S* + *aff_dim T*) − *aff_dim*(*S* $\cap$ *T*)

6.0.6 Lower-dimensional affine subsets are nowhere dense

proposition *dense_complement_subspace*:
fixes *S* :: *'a* :: *euclidean_space set*
assumes *dim_less*: *dim T* < *dim S* **and** *subspace S* **shows** *closure*(*S* − *T*) = *S*

6.0.7 Paracompactness

proposition *paracompact*:
fixes $S :: 'a :: \{metric\_space,second\_countable\_topology\}\ set$
assumes $S \subseteq \bigcup\mathcal{C}$ **and** *opC*: $\bigwedge T.\ T \in \mathcal{C} \Longrightarrow open\ T$
obtains $\mathcal{C}'$ **where** $S \subseteq \bigcup\ \mathcal{C}'$
and $\bigwedge U.\ U \in \mathcal{C}' \Longrightarrow open\ U \wedge (\exists\, T.\ T \in \mathcal{C} \wedge U \subseteq T)$
and $\bigwedge x.\ x \in S$
$\Longrightarrow \exists\, V.\ open\ V \wedge x \in V \wedge finite\ \{U.\ U \in \mathcal{C}' \wedge (U \cap V \neq \{\})\}$

corollary *paracompact_closedin*:
fixes $S :: 'a :: \{metric\_space,second\_countable\_topology\}\ set$
assumes *cin*: $closedin\ (top\_of\_set\ U)\ S$
and *oin*: $\bigwedge T.\ T \in \mathcal{C} \Longrightarrow openin\ (top\_of\_set\ U)\ T$
and $S \subseteq \bigcup\mathcal{C}$
obtains $\mathcal{C}'$ **where** $S \subseteq \bigcup\ \mathcal{C}'$
and $\bigwedge V.\ V \in \mathcal{C}' \Longrightarrow openin\ (top\_of\_set\ U)\ V \wedge (\exists\, T.\ T \in \mathcal{C} \wedge V$
$\subseteq T)$
and $\bigwedge x.\ x \in U$
$\Longrightarrow \exists\, V.\ openin\ (top\_of\_set\ U)\ V \wedge x \in V \wedge$
$finite\ \{X.\ X \in \mathcal{C}' \wedge (X \cap V \neq \{\})\}$

6.0.8 Covering an open set by a countable chain of compact sets

proposition *open_Union_compact_subsets*:
fixes $S :: 'a::euclidean\_space\ set$
assumes $open\ S$
obtains C **where** $\bigwedge n.\ compact(C\ n)$ $\bigwedge n.\ C\ n \subseteq S$
$\bigwedge n.\ C\ n \subseteq interior(C(Suc\ n))$
$\bigcup(range\ C) = S$
$\bigwedge K.\ [\![compact\ K;\ K \subseteq S]\!] \Longrightarrow \exists\, N.\ \forall\, n{\geq}N.\ K \subseteq (C\ n)$

6.0.9 Orthogonal complement

definition *orthogonal_comp* (‹(‹open_block notation=‹postfix ⊥››_$^\perp$)› *[80] 80*)
where $orthogonal\_comp\ W \equiv \{x.\ \forall\, y \in W.\ orthogonal\ y\ x\}$

proposition *subspace_orthogonal_comp*: $subspace\ (W^\perp)$

proposition *subspace_sum_orthogonal_comp*:
fixes $U :: 'a :: euclidean\_space\ set$
assumes $subspace\ U$
shows $U + U^\perp = UNIV$

end

6.1 Path-Connectedness

theory *Path_Connected*
imports
  Starlike
  T1_Spaces
begin

6.1.1 Paths and Arcs

definition *path* :: (*real* ⇒ *'a::topological_space*) ⇒ *bool*
  where *path g* ≡ *continuous_on* {*0..1*} *g*

definition *pathstart* :: (*real* ⇒ *'a::topological_space*) ⇒ *'a*
  where *pathstart g* ≡ *g 0*

definition *pathfinish* :: (*real* ⇒ *'a::topological_space*) ⇒ *'a*
  where *pathfinish g* ≡ *g 1*

definition *path_image* :: (*real* ⇒ *'a::topological_space*) ⇒ *'a set*
  where *path_image g* ≡ *g* ‘ {*0* .. *1*}

definition *reversepath* :: (*real* ⇒ *'a::topological_space*) ⇒ *real* ⇒ *'a*
  where *reversepath g* ≡ (λ*x*. *g*(*1* − *x*))

definition *joinpaths* :: (*real* ⇒ *'a::topological_space*) ⇒ (*real* ⇒ *'a*) ⇒ *real* ⇒ *'a*
  (**infixr** ‹+++› *75*)
  where *g1* +++ *g2* ≡ (λ*x*. *if x* ≤ *1*/*2 then g1* (*2* ∗ *x*) *else g2* (*2* ∗ *x* − *1*))

definition *loop_free* :: (*real* ⇒ *'a::topological_space*) ⇒ *bool*
  where *loop_free g* ≡ ∀ *x*∈{*0..1*}. ∀ *y*∈{*0..1*}. *g x* = *g y* ⟶ *x* = *y* ∨ *x* = *0* ∧ *y* = *1* ∨ *x* = *1* ∧ *y* = *0*

definition *simple_path* :: (*real* ⇒ *'a::topological_space*) ⇒ *bool*
  where *simple_path g* ≡ *path g* ∧ *loop_free g*

definition *arc* :: (*real* ⇒ *'a* :: *topological_space*) ⇒ *bool*
  where *arc g* ≡ *path g* ∧ *inj_on g* {*0..1*}

6.1.2 Subpath

definition *subpath* :: *real* ⇒ *real* ⇒ (*real* ⇒ *'a*) ⇒ *real* ⇒ *'a::real_normed_vector*
  where *subpath a b g* ≡ λ*x*. *g*((*b* − *a*) ∗ *x* + *a*)

6.1.3 Shift Path to Start at Some Given Point

definition *shiftpath* :: *real* ⇒ (*real* ⇒ *'a::topological_space*) ⇒ *real* ⇒ *'a*
  where *shiftpath a f* = (λ*x*. *if* (*a* + *x*) ≤ *1 then f* (*a* + *x*) *else f* (*a* + *x* − *1*))

6.1.4 Straight-Line Paths

definition *linepath* :: $'a$::*real_normed_vector* $\Rightarrow$ $'a$ $\Rightarrow$ *real* $\Rightarrow$ $'a$
where *linepath a b* $= (\lambda x.\ (1 - x) *_R\ a + x *_R\ b)$
proposition *injective_eq_1d_open_map_UNIV*:
fixes f :: *real* $\Rightarrow$ *real*
assumes *contf*: *continuous_on S f* **and** *S*: *is_interval S*
shows *inj_on f S* $\longleftrightarrow$ $(\forall\, T.\ open\ T \wedge T \subseteq S \longrightarrow open(f\ `\ T))$
(**is** *?lhs = ?rhs*)

6.1.5 Path component

definition *path_component S x y* $\equiv$
$(\exists\, g.\ path\ g \wedge path\_image\ g \subseteq S \wedge pathstart\ g = x \wedge pathfinish\ g = y)$

abbreviation
path_component_set S x $\equiv$ *Collect* (*path_component S x*)

6.1.6 Path connectedness of a space

definition *path_connected S* $\longleftrightarrow$
$(\forall\, x{\in}S.\ \forall\, y{\in}S.\ \exists\, g.\ path\ g \wedge path\_image\ g \subseteq S \wedge pathstart\ g = x \wedge pathfinish\ g = y)$

6.1.7 Path components

6.1.8 Paths and path-connectedness

6.1.9 Path components

6.1.10 Sphere is path-connected

corollary *connected_punctured_universe*:
$2 \leq DIM('N::euclidean\_space) \Longrightarrow connected(- \{a::'N\})$

proposition *path_connected_sphere*:
fixes a :: $'a$ :: *euclidean_space*
assumes $2 \leq DIM('a)$
shows *path_connected*(*sphere a r*)

corollary *path_connected_complement_bounded_convex*:
fixes S :: $'a$ :: *euclidean_space set*
assumes *bounded S convex S* **and** *2*: $2 \leq DIM('a)$

shows $path\_connected\ (-\ S)$

proposition $connected\_open\_delete$:
assumes $open\ S\ connected\ S$ **and** 2: $2 \le DIM('N::euclidean\_space)$
shows $connected(S - \{a::'N\})$

corollary $path\_connected\_open\_delete$:
assumes $open\ S\ connected\ S$ **and** 2: $2 \le DIM('N::euclidean\_space)$
shows $path\_connected(S - \{a::'N\})$

corollary $path\_connected\_punctured\_ball$:
$2 \le DIM('N::euclidean\_space) \Longrightarrow path\_connected(ball\ a\ r - \{a::'N\})$

corollary $connected\_punctured\_ball$:
$2 \le DIM('N::euclidean\_space) \Longrightarrow connected(ball\ a\ r - \{a::'N\})$

corollary $connected\_open\_delete\_finite$:
fixes $S\ T::'a::euclidean\_space\ set$
assumes S: $open\ S\ connected\ S$ **and** 2: $2 \le DIM('a)$ **and** $finite\ T$
shows $connected(S - T)$

6.1.11 Every annulus is a connected set

proposition $path\_connected\_annulus$:
fixes $a :: 'N::euclidean\_space$
assumes $2 \le DIM('N)$
shows $path\_connected\ \{x.\ r1 < norm(x - a) \wedge norm(x - a) < r2\}$
$path\_connected\ \{x.\ r1 < norm(x - a) \wedge norm(x - a) \le r2\}$
$path\_connected\ \{x.\ r1 \le norm(x - a) \wedge norm(x - a) < r2\}$
$path\_connected\ \{x.\ r1 \le norm(x - a) \wedge norm(x - a) \le r2\}$

proposition $connected\_annulus$:
fixes $a :: 'N::euclidean\_space$
assumes $2 \le DIM('N::euclidean\_space)$
shows $connected\ \{x.\ r1 < norm(x - a) \wedge norm(x - a) < r2\}$
$connected\ \{x.\ r1 < norm(x - a) \wedge norm(x - a) \le r2\}$
$connected\ \{x.\ r1 \le norm(x - a) \wedge norm(x - a) < r2\}$
$connected\ \{x.\ r1 \le norm(x - a) \wedge norm(x - a) \le r2\}$

corollary $open\_components$:
fixes $S :: 'a::real\_normed\_vector\ set$
shows $[\![open\ u;\ S \in components\ u]\!] \Longrightarrow open\ S$

proposition $components\_open\_unique$:
fixes $S :: 'a::real\_normed\_vector\ set$
assumes $pairwise\ disjnt\ A\ \bigcup A = S$
$\bigwedge X.\ X \in A \Longrightarrow open\ X \wedge connected\ X \wedge X \ne \{\}$

shows *components* $S = A$

6.1.12 The *inside* and *outside* of a Set

The inside comprises the points in a bounded connected component of the set's complement. The outside comprises the points in unbounded connected component of the complement.

definition *inside* **where**
$inside\ S \equiv \{x.\ (x \notin S) \wedge bounded(connected\_component\_set\ (-\ S)\ x)\}$

definition *outside* **where**
$outside\ S \equiv -S \cap \{x.\ \neg\ bounded(connected\_component\_set\ (-\ S)\ x)\}$

6.1.13 Condition for an open map's image to contain a ball

proposition *ball_subset_open_map_image*:
fixes $f :: 'a::heine\_borel \Rightarrow 'b :: \{real\_normed\_vector,heine\_borel\}$
assumes *contf*: $continuous\_on\ (closure\ S)\ f$
and *oint*: $open\ (f\ `\ interior\ S)$
and *le_no*: $\bigwedge z.\ z \in frontier\ S \Longrightarrow r \leq norm(f\ z - f\ a)$
and $bounded\ S\ a \in S\ 0 < r$
shows $ball\ (f\ a)\ r \subseteq f\ `\ S$

proposition *embedding_map_into_euclideanreal*:
assumes $path\_connected\_space\ X$
shows $embedding\_map\ X\ euclideanreal\ f \longleftrightarrow$
$continuous\_map\ X\ euclideanreal\ f \wedge inj\_on\ f\ (topspace\ X)$

end

6.2 Neighbourhood bases and Locally path-connected spaces

theory *Locally*
imports
Path_Connected Function_Topology Sum_Topology
begin

6.2.1 Neighbourhood Bases

6.2.2 Locally path-connected spaces

6.2.3 Locally connected spaces

6.2.4 Dimension of a topological space

end

6.3 Some Uncountable Sets

theory *Uncountable_Sets*
  imports *Path_Connected Continuum_Not_Denumerable*
begin

end

6.4 Homotopy of Maps

theory *Homotopy*
  imports *Path_Connected Product_Topology Uncountable_Sets*
begin

definition *homotopic_with*
where
 $homotopic\_with\ P\ X\ Y\ f\ g \equiv$
  $(\exists h.\ continuous\_map\ (prod\_topology\ (top\_of\_set\ \{0..1::real\})\ X)\ Y\ h \wedge$
    $(\forall x.\ h(0,\ x) = f\ x) \wedge$
    $(\forall x.\ h(1,\ x) = g\ x) \wedge$
    $(\forall t \in \{0..1\}.\ P(\lambda x.\ h(t,x))))$

proposition *homotopic_with*:
  assumes $\bigwedge h\ k.\ (\bigwedge x.\ x \in topspace\ X \Longrightarrow h\ x = k\ x) \Longrightarrow (P\ h \longleftrightarrow P\ k)$
  shows $homotopic\_with\ P\ X\ Y\ p\ q \longleftrightarrow$
     $(\exists h.\ continuous\_map\ (prod\_topology\ (subtopology\ euclideanreal\ \{0..1\})$
$X)\ Y\ h \wedge$
      $(\forall x \in topspace\ X.\ h(0,x) = p\ x) \wedge$
      $(\forall x \in topspace\ X.\ h(1,x) = q\ x) \wedge$
      $(\forall t \in \{0..1\}.\ P(\lambda x.\ h(t,\ x))))$

6.4.1 Homotopy with P is an equivalence relation

proposition *homotopic_with_trans*:
   assumes $homotopic\_with\ P\ X\ Y\ f\ g\ \ homotopic\_with\ P\ X\ Y\ g\ h$

shows *homotopic_with P X Y f h*

6.4.2 Continuity lemmas

corollary *homotopic_compose*:
assumes *homotopic_with* ($\lambda x.$ *True*) *X Y f f′ homotopic_with* ($\lambda x.$ *True*) *Y Z g g′*
shows *homotopic_with* ($\lambda x.$ *True*) *X Z* ($g \circ f$) ($g' \circ f'$)

proposition *homotopic_with_compose_continuous_right*:
⟦*homotopic_with_canon* ($\lambda f.$ *p* ($f \circ h$)) *X Y f g*; *continuous_on W h*; $h \in W \to X$⟧
⟹ *homotopic_with_canon p W Y* ($f \circ h$) ($g \circ h$)

proposition *homotopic_with_compose_continuous_left*:
⟦*homotopic_with_canon* ($\lambda f.$ *p* ($h \circ f$)) *X Y f g*; *continuous_on Y h*; $h \in Y \to Z$⟧
⟹ *homotopic_with_canon p X Z* ($h \circ f$) ($h \circ g$)

proposition *homotopic_with_eq*:
assumes *h*: *homotopic_with P X Y f g*
and *f′*: $\bigwedge x.$ $x \in$ *topspace X* ⟹ $f'\ x = f\ x$
and *g′*: $\bigwedge x.$ $x \in$ *topspace X* ⟹ $g'\ x = g\ x$
and *P*: ($\bigwedge h\ k.$ ($\bigwedge x.$ $x \in$ *topspace X* ⟹ $h\ x = k\ x$) ⟹ *P h* ⟷ *P k*)
shows *homotopic_with P X Y f′ g′*

6.4.3 Homotopy of paths, maintaining the same endpoints

definition *homotopic_paths* :: [*′a set*, *real* ⇒ *′a*, *real* ⇒ *′a*::*topological_space*] ⇒ *bool*
where
homotopic_paths S p q ≡
homotopic_with_canon ($\lambda r.$ *pathstart r* = *pathstart p* ∧ *pathfinish r* = *pathfinish p*) {*0..1*} *S p q*

proposition *homotopic_paths_imp_pathstart*:
homotopic_paths S p q ⟹ *pathstart p* = *pathstart q*

proposition *homotopic_paths_imp_pathfinish*:
homotopic_paths S p q ⟹ *pathfinish p* = *pathfinish q*

proposition *homotopic_paths_refl* [*simp*]: *homotopic_paths S p p* ⟷ *path p* ∧ *path_image p* ⊆ *S*

proposition *homotopic_paths_sym*: *homotopic_paths S p q* ⟹ *homotopic_paths S q p*

proposition *homotopic_paths_sym_eq*: *homotopic_paths S p q* $\longleftrightarrow$ *homotopic_paths S q p*

proposition *homotopic_paths_trans* [*trans*]:
assumes *homotopic_paths S p q homotopic_paths S q r*
shows *homotopic_paths S p r*

proposition *homotopic_paths_eq*:
$[\![$*path p*; *path_image p* $\subseteq$ *S*; $\bigwedge t.\ t \in \{0..1\} \Longrightarrow p\ t = q\ t]\!]$ $\Longrightarrow$ *homotopic_paths S p q*

proposition *homotopic_paths_reparametrize*:
assumes *path p*
and *pips*: *path_image p* $\subseteq$ *S*
and *contf*: *continuous_on* $\{0..1\}$ *f*
and *f01* :$f \in \{0..1\} \rightarrow \{0..1\}$
and [*simp*]: $f(0) = 0\ f(1) = 1$
and *q*: $\bigwedge t.\ t \in \{0..1\} \Longrightarrow q(t) = p(f\ t)$
shows *homotopic_paths S p q*

proposition *homotopic_paths_reversepath*:
homotopic_paths S (*reversepath p*) (*reversepath q*) $\longleftrightarrow$ *homotopic_paths S p q*

proposition *homotopic_paths_join*:
$[\![$*homotopic_paths S p p'*; *homotopic_paths S q q'*; *pathfinish p* = *pathstart q*$]\!]$ $\Longrightarrow$ *homotopic_paths S* (*p* +++ *q*) (*p'* +++ *q'*)

proposition *homotopic_paths_continuous_image*:
$[\![$*homotopic_paths S f g*; *continuous_on S h*; $h \in S \rightarrow t]\!]$ $\Longrightarrow$ *homotopic_paths t* $(h \circ f)$ $(h \circ g)$

6.4.4 Group properties for homotopy of paths

So taking equivalence classes under homotopy would give the fundamental group

proposition *homotopic_paths_rid*:
assumes *path p path_image p* $\subseteq$ *S*
shows *homotopic_paths S* (*p* +++ *linepath* (*pathfinish p*) (*pathfinish p*)) *p*

proposition *homotopic_paths_lid*:
$[\![$*path p*; *path_image p* $\subseteq$ *S*$]\!]$ $\Longrightarrow$ *homotopic_paths S* (*linepath* (*pathstart p*) (*pathstart p*) +++ *p*) *p*

proposition *homotopic_paths_assoc*:
$[\![$*path p*; *path_image p* $\subseteq$ *S*; *path q*; *path_image q* $\subseteq$ *S*; *path r*; *path_image r* $\subseteq$

S; *pathfinish p* = *pathstart q*;
pathfinish q = *pathstart r*⟧
⟹ *homotopic_paths S* (*p* +++ (*q* +++ *r*)) ((*p* +++ *q*) +++ *r*)

proposition *homotopic_paths_rinv*:
assumes *path p path_image p* ⊆ *S*
shows *homotopic_paths S* (*p* +++ *reversepath p*) (*linepath* (*pathstart p*) (*pathstart p*))

proposition *homotopic_paths_linv*:
assumes *path p path_image p* ⊆ *S*
shows *homotopic_paths S* (*reversepath p* +++ *p*) (*linepath* (*pathfinish p*) (*pathfinish p*))

6.4.5 Homotopy of loops without requiring preservation of endpoints

definition *homotopic_loops* :: *'a::topological_space set* ⇒ (*real* ⇒ *'a*) ⇒ (*real* ⇒ *'a*) ⇒ *bool* **where**
homotopic_loops S p q ≡
homotopic_with_canon (λ*r*. *pathfinish r* = *pathstart r*) {*0..1*} *S p q*

proposition *homotopic_loops_imp_loop*:
homotopic_loops S p q ⟹ *pathfinish p* = *pathstart p* ∧ *pathfinish q* = *pathstart q*

proposition *homotopic_loops_imp_path*:
homotopic_loops S p q ⟹ *path p* ∧ *path q*

proposition *homotopic_loops_imp_subset*:
homotopic_loops S p q ⟹ *path_image p* ⊆ *S* ∧ *path_image q* ⊆ *S*

proposition *homotopic_loops_refl*:
homotopic_loops S p p ⟷
path p ∧ *path_image p* ⊆ *S* ∧ *pathfinish p* = *pathstart p*

proposition *homotopic_loops_sym*: *homotopic_loops S p q* ⟹ *homotopic_loops S q p*

proposition *homotopic_loops_sym_eq*: *homotopic_loops S p q* ⟷ *homotopic_loops S q p*

proposition *homotopic_loops_trans*:
⟦*homotopic_loops S p q*; *homotopic_loops S q r*⟧ ⟹ *homotopic_loops S p r*

proposition *homotopic_loops_subset*:
⟦*homotopic_loops S p q*; *S* ⊆ *t*⟧ ⟹ *homotopic_loops t p q*

proposition *homotopic_loops_eq*:
$\llbracket$*path p*; *path_image p* $\subseteq$ *S*; *pathfinish p* = *pathstart p*; $\bigwedge t.\ t \in \{0..1\} \Longrightarrow p(t) = q(t)\rrbracket$
$\Longrightarrow$ *homotopic_loops S p q*

proposition *homotopic_loops_continuous_image*:
$\llbracket$*homotopic_loops S f g*; *continuous_on S h*; $h \in S \to t\rrbracket \Longrightarrow$ *homotopic_loops t* $(h \circ f)$ $(h \circ g)$

6.4.6 Relations between the two variants of homotopy

proposition *homotopic_paths_imp_homotopic_loops*:
$\llbracket$*homotopic_paths S p q*; *pathfinish p* = *pathstart p*; *pathfinish q* = *pathstart p*$\rrbracket$ $\Longrightarrow$ *homotopic_loops S p q*

proposition *homotopic_loops_imp_homotopic_paths_null*:
assumes *homotopic_loops S p* (*linepath a a*)
shows *homotopic_paths S p* (*linepath* (*pathstart p*) (*pathstart p*))

proposition *homotopic_loops_conjugate*:
fixes *S* :: *'a::real_normed_vector set*
assumes *path p path q* **and** *pip*: *path_image p* $\subseteq$ *S* **and** *piq*: *path_image q* $\subseteq$ *S*
and *pq*: *pathfinish p* = *pathstart q* **and** *qloop*: *pathfinish q* = *pathstart q*
shows *homotopic_loops S* (*p* +++ *q* +++ *reversepath p*) *q*

6.4.7 Homotopy and subpaths

proposition *homotopic_join_subpaths*:
$\llbracket$*path g*; *path_image g* $\subseteq$ *S*; $u \in \{0..1\}$; $v \in \{0..1\}$; $w \in \{0..1\}\rrbracket$
$\Longrightarrow$ *homotopic_paths S* (*subpath u v g* +++ *subpath v w g*) (*subpath u w g*)

6.4.8 Simply connected sets

defined as "all loops are homotopic (as loops)

definition *simply_connected* **where**
simply_connected S $\equiv$
$\forall$ *p q*. *path p* $\wedge$ *pathfinish p* = *pathstart p* $\wedge$ *path_image p* $\subseteq$ *S* $\wedge$
path q $\wedge$ *pathfinish q* = *pathstart q* $\wedge$ *path_image q* $\subseteq$ *S*
$\longrightarrow$ *homotopic_loops S p q*

proposition *simply_connected_Times*:
fixes *S* :: *'a::real_normed_vector set* **and** *T* :: *'b::real_normed_vector set*
assumes *S*: *simply_connected S* **and** *T*: *simply_connected T*
shows *simply_connected*(*S* $\times$ *T*)

6.4.9 Contractible sets

definition *contractible* **where**
$contractible\ S \equiv \exists\, a.\ homotopic\_with\_canon\ (\lambda x.\ True)\ S\ S\ id\ (\lambda x.\ a)$

proposition *contractible_imp_simply_connected*:
fixes S :: _::*real_normed_vector set*
assumes *contractible* S **shows** *simply_connected* S

corollary *contractible_imp_connected*:
fixes S :: _::*real_normed_vector set*
shows $contractible\ S \Longrightarrow connected\ S$

6.4.10 Starlike sets

definition $starlike\ S \longleftrightarrow (\exists\, a{\in}S.\ \forall\, x{\in}S.\ closed\_segment\ a\ x \subseteq S)$

6.4.11 Local versions of topological properties in general

definition $locally :: ('a::topological\_space\ set \Rightarrow bool) \Rightarrow 'a\ set \Rightarrow bool$
where
$locally\ P\ S \equiv$
$\forall\, w\ x.\ openin\ (top\_of\_set\ S)\ w \wedge x \in w$
$\longrightarrow (\exists\, U\ V.\ openin\ (top\_of\_set\ S)\ U \wedge P\ V \wedge x \in U \wedge U \subseteq V \wedge V$
$\subseteq w)$

proposition *homeomorphism_locally_imp*:
fixes $S :: 'a::metric\_space\ set$ **and** $T :: 'b::t2\_space\ set$
assumes S: *locally* $P\ S$ **and** *hom*: *homeomorphism* $S\ T\ f\ g$
and Q: $\bigwedge S\ S'.\ [\![P\ S;\ homeomorphism\ S\ S'\ f\ g]\!] \Longrightarrow Q\ S'$
shows *locally* $Q\ T$

6.4.12 An induction principle for connected sets

proposition *connected_induction*:
assumes *connected* S
and *opD*: $\bigwedge T\ a.\ [\![openin\ (top\_of\_set\ S)\ T;\ a \in T]\!] \Longrightarrow \exists\, z.\ z \in T \wedge P\ z$
and *opI*: $\bigwedge a.\ a \in S$
$\Longrightarrow \exists\, T.\ openin\ (top\_of\_set\ S)\ T \wedge a \in T \wedge$
$(\forall\, x \in T.\ \forall\, y \in T.\ P\ x \wedge P\ y \wedge Q\ x \longrightarrow Q\ y)$
and *etc*: $a \in S\ b \in S\ P\ a\ P\ b\ Q\ a$
shows $Q\ b$

6.4.13 Basic properties of local compactness

proposition *locally_compact*:
fixes S :: *'a* :: *metric_space set*
shows
locally compact $S \longleftrightarrow$
$(\forall x \in S.\ \exists u\ v.\ x \in u \wedge u \subseteq v \wedge v \subseteq S \wedge$
openin (*top_of_set* S) $u \wedge$ *compact* v)
(**is** *?lhs* = *?rhs*)

6.4.14 Sura-Bura's results about compact components of sets

proposition *Sura_Bura_compact*:
fixes S :: *'a::euclidean_space set*
assumes *compact* S **and** C: $C \in$ *components* S
shows $C = \bigcap\{T.\ C \subseteq T \wedge$ *openin* (*top_of_set* S) $T \wedge$
closedin (*top_of_set* S) $T\}$
(**is** $C = \bigcap$ *?T*)

corollary *Sura_Bura_clopen_subset*:
fixes S :: *'a::euclidean_space set*
assumes S: *locally compact* S **and** C: $C \in$ *components* S **and** *compact* C
and U: *open* U $C \subseteq U$
obtains K **where** *openin* (*top_of_set* S) K *compact* K $C \subseteq K$ $K \subseteq U$

corollary *Sura_Bura_clopen_subset_alt*:
fixes S :: *'a::euclidean_space set*
assumes S: *locally compact* S **and** C: $C \in$ *components* S **and** *compact* C
and *opeSU*: *openin* (*top_of_set* S) U **and** $C \subseteq U$
obtains K **where** *openin* (*top_of_set* S) K *compact* K $C \subseteq K$ $K \subseteq U$

corollary *Sura_Bura*:
fixes S :: *'a::euclidean_space set*
assumes *locally compact* S $C \in$ *components* S *compact* C
shows $C = \bigcap\ \{K.\ C \subseteq K \wedge$ *compact* $K \wedge$ *openin* (*top_of_set* S) $K\}$
(**is** C = *?rhs*)

6.4.15 Special cases of local connectedness and path connectedness

proposition *locally_path_connected*:
locally path_connected $S \longleftrightarrow$
$(\forall V\ x.$ *openin* (*top_of_set* S) $V \wedge x \in V$
$\longrightarrow (\exists U.$ *openin* (*top_of_set* S) $U \wedge$ *path_connected* $U \wedge x \in U \wedge U \subseteq V))$

proposition *locally_path_connected_open_path_component*:
locally path_connected S $\longleftrightarrow$
($\forall$ *t x*. *openin* (*top_of_set S*) *t* $\wedge$ *x* $\in$ *t*
$\longrightarrow$ *openin* (*top_of_set S*) (*path_component_set t x*))

proposition *locally_connected_im_kleinen*:
locally connected S $\longleftrightarrow$
($\forall$ *v x*. *openin* (*top_of_set S*) *v* $\wedge$ *x* $\in$ *v*
$\longrightarrow$ ($\exists$ *u*. *openin* (*top_of_set S*) *u* $\wedge$
x $\in$ *u* $\wedge$ *u* $\subseteq$ *v* $\wedge$
($\forall$ *y*. *y* $\in$ *u* $\longrightarrow$ ($\exists$ *c*. *connected c* $\wedge$ *c* $\subseteq$ *v* $\wedge$ *x* $\in$ *c* $\wedge$ *y* $\in$ *c*))))
(**is** *?lhs* = *?rhs*)

proposition *locally_path_connected_im_kleinen*:
locally path_connected S $\longleftrightarrow$
($\forall$ *v x*. *openin* (*top_of_set S*) *v* $\wedge$ *x* $\in$ *v*
$\longrightarrow$ ($\exists$ *u*. *openin* (*top_of_set S*) *u* $\wedge$
x $\in$ *u* $\wedge$ *u* $\subseteq$ *v* $\wedge$
($\forall$ *y*. *y* $\in$ *u* $\longrightarrow$ ($\exists$ *p*. *path p* $\wedge$ *path_image p* $\subseteq$ *v* $\wedge$
pathstart p = *x* $\wedge$ *pathfinish p* = *y*))))
(**is** *?lhs* = *?rhs*)

6.4.16 Relations between components and path components

proposition *locally_connected_quotient_image*:
assumes *lcS*: *locally connected S*
and *oo*: $\bigwedge$*T*. *T* $\subseteq$ *f* ‘ *S*
$\Longrightarrow$ *openin* (*top_of_set S*) (*S* $\cap$ *f* −‘ *T*) $\longleftrightarrow$
openin (*top_of_set* (*f* ‘ *S*)) *T*
shows *locally connected* (*f* ‘ *S*)
proposition *locally_path_connected_quotient_image*:
assumes *lcS*: *locally path_connected S*
and *oo*: $\bigwedge$*T*. *T* $\subseteq$ *f* ‘ *S*
$\Longrightarrow$ *openin* (*top_of_set S*) (*S* $\cap$ *f* −‘ *T*) $\longleftrightarrow$ *openin* (*top_of_set* (*f*
‘ *S*)) *T*
shows *locally path_connected* (*f* ‘ *S*)

6.4.17 Existence of isometry between subspaces of same dimension

proposition *isometries_subspaces*:
fixes *S* :: *'a*::*euclidean_space set*
and *T* :: *'b*::*euclidean_space set*
assumes *S*: *subspace S*

and *T*: *subspace T*
and *d*: *dim S* = *dim T*
obtains *f g* **where** *linear f linear g f ‘ S* = *T g ‘ T* = *S*
$\bigwedge x.\ x \in S \Longrightarrow norm(f\ x) = norm\ x$
$\bigwedge x.\ x \in T \Longrightarrow norm(g\ x) = norm\ x$
$\bigwedge x.\ x \in S \Longrightarrow g(f\ x) = x$
$\bigwedge x.\ x \in T \Longrightarrow f(g\ x) = x$

corollary *isometry_subspaces*:
fixes *S* :: *'a::euclidean_space set*
and *T* :: *'b::euclidean_space set*
assumes *S*: *subspace S*
and *T*: *subspace T*
and *d*: *dim S* = *dim T*
obtains *f* **where** *linear f f ‘ S* = *T* $\bigwedge x.\ x \in S \Longrightarrow norm(f\ x) = norm\ x$

corollary *isomorphisms_UNIV_UNIV*:
assumes *DIM('M)* = *DIM('N)*
obtains *f::'M::euclidean_space ⇒'N::euclidean_space* **and** *g*
where *linear f linear g*
$\bigwedge x.\ norm(f\ x) = norm\ x$ $\bigwedge y.\ norm(g\ y) = norm\ y$
$\bigwedge x.\ g\ (f\ x) = x$ $\bigwedge y.\ f(g\ y) = y$

6.4.18 Retracts, in a general sense, preserve (co)homotopic triviality)

locale *Retracts* =
fixes *S h t k*
assumes *conth*: *continuous_on S h*
and *imh*: *h ‘ S* = *t*
and *contk*: *continuous_on t k*
and *imk*: *k* ∈ *t* → *S*
and *idhk*: $\bigwedge y.\ y \in t \Longrightarrow h(k\ y) = y$

begin

6.4.19 Homotopy equivalence

6.4.20 Homotopy equivalence of topological spaces.

definition *homotopy_equivalent_space*
(**infix** *‹homotopy'_equivalent'_space› 50*)
where *X homotopy_equivalent_space Y* ≡
(∃*f g. continuous_map X Y f* ∧
continuous_map Y X g ∧
homotopic_with (λ*x. True*) *X X* (*g* ∘ *f*) *id* ∧
homotopic_with (λ*x. True*) *Y Y* (*f* ∘ *g*) *id*)

6.4.21 Contractible spaces

corollary *contractible_space_euclideanreal*: *contractible_space euclideanreal*

abbreviation *homotopy_eqv* :: *'a::topological_space set* ⇒ *'b::topological_space set* ⇒ *bool*
(**infix** ‹*homotopy'_eqv*› *50*)
where *S homotopy_eqv T* ≡ *top_of_set S homotopy_equivalent_space top_of_set T*

corollary *bounded_path_connected_Compl_real*:
fixes *S* :: *real set*
assumes *bounded S path_connected*(− *S*) **shows** *S* = {}
proposition *path_connected_convex_diff_countable*:
fixes *U* :: *'a::euclidean_space set*
assumes *convex U* ¬ *collinear U countable S*
shows *path_connected*(*U* − *S*)

corollary *connected_convex_diff_countable*:
fixes *U* :: *'a::euclidean_space set*
assumes *convex U* ¬ *collinear U countable S*
shows *connected*(*U* − *S*)

proposition *path_connected_openin_diff_countable*:
fixes *S* :: *'a::euclidean_space set*
assumes *connected S* **and** *ope*: *openin* (*top_of_set* (*affine hull S*)) *S*
and ¬ *collinear S countable T*
shows *path_connected*(*S* − *T*)

corollary *connected_openin_diff_countable*:
fixes *S* :: *'a::euclidean_space set*
assumes *connected S* **and** *ope*: *openin* (*top_of_set* (*affine hull S*)) *S*
and ¬ *collinear S countable T*
shows *connected*(*S* − *T*)

corollary *path_connected_open_diff_countable*:
fixes *S* :: *'a::euclidean_space set*
assumes *2* ≤ *DIM*(*'a*) *open S connected S countable T*
shows *path_connected*(*S* − *T*)

corollary *connected_open_diff_countable*:
fixes *S* :: *'a::euclidean_space set*

assumes $2 \leq DIM('a)$ *open S connected S countable T*
shows *connected*$(S - T)$

6.4.22 Nullhomotopic mappings

proposition *nullhomotopic_from_sphere_extension*:
fixes $f :: 'M::euclidean\_space \Rightarrow 'a::real\_normed\_vector$
shows $(\exists c.\ homotopic\_with\_canon\ (\lambda x.\ True)\ (sphere\ a\ r)\ S\ f\ (\lambda x.\ c)) \longleftrightarrow$
$(\exists g.\ continuous\_on\ (cball\ a\ r)\ g \wedge g\ `\ (cball\ a\ r) \subseteq S \wedge$
$(\forall x \in sphere\ a\ r.\ g\ x = f\ x))$
(**is** *?lhs* = *?rhs*)

end

6.5 Euclidean space and n-spheres, as subtopologies of n-dimensional space

theory *Abstract_Euclidean_Space*
imports *Homotopy Locally*
begin

6.5.1 Euclidean spaces as abstract topologies

6.5.2 n-dimensional spheres

proposition *contractible_space_upper_hemisphere*:
assumes $k \leq n$
shows *contractible_space*$(subtopology\ (nsphere\ n)\ \{x.\ x\ k \geq 0\})$

corollary *contractible_space_lower_hemisphere*:
assumes $k \leq n$
shows *contractible_space*$(subtopology\ (nsphere\ n)\ \{x.\ x\ k \leq 0\})$

proposition *nullhomotopic_nonsurjective_sphere_map*:
assumes f: *continuous_map* $(nsphere\ p)\ (nsphere\ p)\ f$
and *fim*: $f\ `\ (topspace(nsphere\ p)) \neq topspace(nsphere\ p)$
obtains a **where** $homotopic\_with\ (\lambda x.\ True)\ (nsphere\ p)\ (nsphere\ p)\ f\ (\lambda x.\ a)$

end

6.6 Various Forms of Topological Spaces

theory *Abstract_Topological_Spaces*
imports *Lindelof_Spaces Locally Abstract_Euclidean_Space Sum_Topology FSigma*
begin

6.6.1 Connected topological spaces

6.6.2 The notion of "separated between" (complement of "connected between)"

6.6.3 Connected components

6.6.4 Monotone maps (in the general topological sense)

proposition *connected_space_monotone_quotient_map_preimage*:
assumes *f*: *monotone_map X Y f quotient_map X Y f* **and** *connected_space Y*
shows *connected_space X*

6.6.5 Other countability properties

6.6.6 Neigbourhood bases EXTRAS

6.6.7 T_0 spaces and the Kolmogorov quotient

proposition *t0_space_product_topology*:
t0_space (*product_topology X I*) $\longleftrightarrow$ *product_topology X I* = *trivial_topology* $\vee$ ($\forall$ *i* $\in$ *I*. *t0_space* (*X i*))
(**is** *?lhs=?rhs*)

6.6.8 Kolmogorov quotients

6.6.9 Closed diagonals and graphs

6.6.10 KC spaces, those where all compact sets are closed.

proposition *kc_space_prod_topology_left*:
assumes *X*: *kc_space X* **and** *Y*: *Hausdorff_space Y*
shows *kc_space* (*prod_topology X Y*)

6.6.11 Technical results about proper maps, perfect maps, etc

6.6.12 Regular spaces

proposition *regular_space_continuous_proper_map_image*:
assumes *regular_space X* **and** *contf*: *continuous_map X Y f* **and** *pmapf*: *proper_map X Y f*
and *fim*: *f* ‘ (*topspace X*) = *topspace Y*
shows *regular_space Y*

proposition *regular_space_perfect_map_image_eq*:
assumes *Hausdorff_space X* **and** *perf*: *perfect_map X Y f*
shows *regular_space X* ⟷ *regular_space Y* (**is** *?lhs*=*?rhs*)

6.6.13 Locally compact spaces

proposition *quotient_map_prod_right*:
assumes *loc*: *locally_compact_space Z*
and *reg*: *Hausdorff_space Z* ∨ *regular_space Z*
and *f*: *quotient_map X Y f*
shows *quotient_map* (*prod_topology Z X*) (*prod_topology Z Y*) (λ(*x*,*y*). (*x*,*f y*))

6.6.14 Special characterizations of classes of functions into and out of R

6.6.15 Normal spaces

6.6.16 Hereditary topological properties

6.6.17 Limits in a topological space

6.6.18 Quasi-components

6.6.19 Additional quasicomponent and continuum properties like Boundary Bumping

6.6.20 Compactly generated spaces (k-spaces)

end

6.7 Abstract Metric Spaces

theory *Abstract_Metric_Spaces*
imports *Elementary_Metric_Spaces Abstract_Limits Abstract_Topological_Spaces*
begin

6.7.1 Metric topology

6.7.2 Bounded sets

6.7.3 Subspace of a metric space

6.7.4 Abstract type of metric spaces

6.7.5 The discrete metric

6.7.6 Metrizable spaces

6.7.7 Limits at a point in a topological space

6.7.8 Normal spaces and metric spaces

6.7.9 Topological limitin in metric spaces

6.7.10 Cauchy sequences and complete metric spaces

6.7.11 Totally bounded subsets of metric spaces

6.7.12 Compactness in metric spaces

6.7.13 Continuous functions on metric spaces

6.7.14 Completely metrizable spaces

6.7.15 Product metric

6.7.16 More sequential characterizations in a metric space

6.7.17 Three strong notions of continuity for metric spaces

6.7.18 Isometries

6.7.19 "Capped" equivalent bounded metrics and general product metrics

proposition *metrizable_space_product_topology*:
  metrizable_space (*product_topology X I*) $\longleftrightarrow$
    (*product_topology X I*) = *trivial_topology* $\vee$

$countable\ \{i \in I.\ \neg\ (\exists\, a.\ topspace(X\ i) \subseteq \{a\})\} \wedge$
$(\forall\, i \in I.\ metrizable\_space\ (X\ i))$

proposition *completely_metrizable_space_product_topology*:
$completely\_metrizable\_space\ (product\_topology\ X\ I) \longleftrightarrow$
$(product\_topology\ X\ I) = trivial\_topology \vee$
$countable\ \{i \in I.\ \neg\ (\exists\, a.\ topspace(X\ i) \subseteq \{a\})\} \wedge$
$(\forall\, i \in I.\ completely\_metrizable\_space\ (X\ i))$

end

6.8 Infinite sums

theory *Infinite_Sum*
imports
Elementary_Topology
HOL−Library.Extended_Nonnegative_Real
HOL−Library.Complex_Order
begin

6.8.1 Definition and syntax

6.8.2 General properties

6.8.3 Absolute convergence

6.8.4 Extended reals and nats

6.8.5 Real numbers

6.8.6 Complex numbers

class *complete_uniform_space* = *uniform_space* +
assumes *cauchy_filter_convergent'*: *cauchy_filter* (*F* :: *'a filter*) $\Longrightarrow$ *F* $\neq$ *bot* $\Longrightarrow$ *convergent_filter F*

theorem (**in** *uniform_space*) *controlled_sequences_convergent_imp_complete*:
fixes *U* :: *nat* $\Rightarrow$ (*'a* $\times$ *'a*) *set*
assumes *gen*: *countably_generated_filter* (*uniformity* :: (*'a* $\times$ *'a*) *filter*)
assumes *U*: $\bigwedge$*n. eventually* (λ*z. z* $\in$ *U n*) *uniformity*
assumes *conv*: $\bigwedge$(*u* :: *nat* $\Rightarrow$ *'a*). ($\bigwedge$*N m n. N* $\leq$ *m* $\Longrightarrow$ *N* $\leq$ *n* $\Longrightarrow$ (*u m*, *u n*) $\in$ *U N*) $\Longrightarrow$ *convergent u*
shows *class.complete_uniform_space open uniformity*

theorem (**in** *uniform_space*) *controlled_seq_imp_Cauchy_seq*:
fixes *U* :: *nat* $\Rightarrow$ (*'a* $\times$ *'a*) *set*
assumes *U*: $\bigwedge$*P. eventually P uniformity* $\Longrightarrow$ ($\exists$ *n.* $\forall$ *x*$\in$*U n. P x*)
assumes *controlled*: $\bigwedge$*N m n. N* $\leq$ *m* $\Longrightarrow$ *N* $\leq$ *n* $\Longrightarrow$ (*f m*, *f n*) $\in$ *U N*
shows *Cauchy f*

theorem (**in** *uniform_space*) *Cauchy_seq_convergent_imp_complete*:
fixes *U* :: *nat* $\Rightarrow$ (*'a* $\times$ *'a*) *set*
assumes *gen*: *countably_generated_filter* (*uniformity* :: (*'a* $\times$ *'a*) *filter*)
assumes *conv*: $\bigwedge$(*u* :: *nat* $\Rightarrow$ *'a*). *Cauchy u* $\Longrightarrow$ *convergent u*
shows *class.complete_uniform_space open uniformity*

end

6.9 Ordered Euclidean Space

theory *Ordered_Euclidean_Space*
imports
Convex_Euclidean_Space Abstract_Limits
HOL−Library.Product_Order
beginclass *ordered_euclidean_space* = *ord* + *inf* + *sup* + *abs* + *Inf* + *Sup* + *euclidean_space* +

assumes *eucl_le*: $x \leq y \longleftrightarrow (\forall i{\in}Basis.\ x \cdot i \leq y \cdot i)$
assumes *eucl_less_le_not_le*: $x < y \longleftrightarrow x \leq y \wedge \neg\ y \leq x$
assumes *eucl_inf*: $inf\ x\ y = (\sum i{\in}Basis.\ inf\ (x \cdot i)\ (y \cdot i) *_R i)$
assumes *eucl_sup*: $sup\ x\ y = (\sum i{\in}Basis.\ sup\ (x \cdot i)\ (y \cdot i) *_R i)$
assumes *eucl_Inf*: $Inf\ X = (\sum i{\in}Basis.\ (INF\ x{\in}X.\ x \cdot i) *_R i)$
assumes *eucl_Sup*: $Sup\ X = (\sum i{\in}Basis.\ (SUP\ x{\in}X.\ x \cdot i) *_R i)$
assumes *eucl_abs*: $|x| = (\sum i{\in}Basis.\ |x \cdot i| *_R i)$
begin

proposition *compact_attains_Inf_componentwise*:
fixes b::'a::*ordered_euclidean_space*
assumes $b \in Basis$ **assumes** $X \neq \{\}$ *compact X*
obtains x **where** $x \in X$ $x \cdot b = Inf\ X \cdot b$ $\bigwedge y.\ y \in X \Longrightarrow x \cdot b \leq y \cdot b$

proposition
compact_attains_Sup_componentwise:
fixes b::'a::*ordered_euclidean_space*
assumes $b \in Basis$ **assumes** $X \neq \{\}$ *compact X*
obtains x **where** $x \in X$ $x \cdot b = Sup\ X \cdot b$ $\bigwedge y.\ y \in X \Longrightarrow y \cdot b \leq x \cdot b$
proposition
fixes a :: 'a::*ordered_euclidean_space*
shows *cbox_interval*: $cbox\ a\ b = \{a..b\}$
and *interval_cbox*: $\{a..b\} = cbox\ a\ b$
and *eucl_le_atMost*: $\{x.\ \forall i{\in}Basis.\ x \cdot i <= a \cdot i\} = \{..a\}$
and *eucl_le_atLeast*: $\{x.\ \forall i{\in}Basis.\ a \cdot i <= x \cdot i\} = \{a..\}$

instantiation *vec* :: (*ordered_euclidean_space*, *finite*) *ordered_euclidean_space*
begin

definition $inf\ x\ y = (\chi\ i.\ inf\ (x\ \$\ i)\ (y\ \$\ i))$
definition $sup\ x\ y = (\chi\ i.\ sup\ (x\ \$\ i)\ (y\ \$\ i))$
definition $Inf\ X = (\chi\ i.\ (INF\ x{\in}X.\ x\ \$\ i))$
definition $Sup\ X = (\chi\ i.\ (SUP\ x{\in}X.\ x\ \$\ i))$
definition $|x| = (\chi\ i.\ |x\ \$\ i|)$

end

6.10 Arcwise-Connected Sets

theory *Arcwise_Connected*
imports *Path_Connected Ordered_Euclidean_Space HOL−Computational_Algebra.Primes*
begin

6.10.1 The Brouwer reduction theorem

theorem *Brouwer_reduction_theorem_gen*:
fixes S :: 'a::*euclidean_space set*
assumes *closed* S φ S

and φ: $\bigwedge F.\ [\![\bigwedge n.\ closed(F\ n);\ \bigwedge n.\ \varphi(F\ n);\ \bigwedge n.\ F(Suc\ n) \subseteq F\ n]\!] \Longrightarrow \varphi(\bigcap(range\ F))$
obtains T **where** $T \subseteq S$ $closed\ T$ $\varphi\ T$ $\bigwedge U.\ [\![U \subseteq S;\ closed\ U;\ \varphi\ U]\!] \Longrightarrow \neg\ (U \subset T)$

corollary *Brouwer_reduction_theorem*:
fixes S :: *'a::euclidean_space set*
assumes $compact\ S$ $\varphi\ S$ $S \neq \{\}$
and φ: $\bigwedge F.\ [\![\bigwedge n.\ compact(F\ n);\ \bigwedge n.\ F\ n \neq \{\};\ \bigwedge n.\ \varphi(F\ n);\ \bigwedge n.\ F(Suc\ n) \subseteq F\ n]\!] \Longrightarrow \varphi(\bigcap(range\ F))$
obtains T **where** $T \subseteq S$ $compact\ T$ $T \neq \{\}$ $\varphi\ T$
$\bigwedge U.\ [\![U \subseteq S;\ closed\ U;\ U \neq \{\};\ \varphi\ U]\!] \Longrightarrow \neg\ (U \subset T)$

6.10.2 Density of points with dyadic rational coordinates

proposition *closure_dyadic_rationals*:
$closure\ (\bigcup k.\ \bigcup f \in Basis \to \mathbb{Z}.$
$\{\ \sum i :: 'a :: euclidean\_space \in Basis.\ (f\ i\ /\ 2\hat{\ }k) *_R\ i\ \}) = UNIV$

corollary *closure_rational_coordinates*:
$closure\ (\bigcup f \in Basis \to \mathbb{Q}.\ \{\ \sum i :: 'a :: euclidean\_space \in Basis.\ f\ i *_R\ i\ \}) = UNIV$

theorem *homeomorphic_monotone_image_interval*:
fixes f :: *real ⇒ 'a::{real_normed_vector,complete_space}*
assumes *cont_f*: $continuous\_on\ \{0..1\}\ f$
and *conn*: $\bigwedge y.\ connected\ (\{0..1\} \cap f -` \{y\})$
and *f_1not0*: $f\ 1 \neq f\ 0$
shows $(f\ `\ \{0..1\})\ homeomorphic\ \{0..1::real\}$

theorem *path_contains_arc*:
fixes p :: *real ⇒ 'a::{complete_space,real_normed_vector}*
assumes *path p* **and** *a*: $pathstart\ p = a$ **and** *b*: $pathfinish\ p = b$ **and** $a \neq b$
obtains q **where** $arc\ q$ $path\_image\ q \subseteq path\_image\ p$ $pathstart\ q = a$ $pathfinish\ q = b$

corollary *path_connected_arcwise*:
fixes S :: *'a::{complete_space,real_normed_vector} set*
shows $path\_connected\ S \longleftrightarrow$

$(\forall x \in S. \forall y \in S. x \neq y \longrightarrow (\exists g.\ arc\ g \wedge path\_image\ g \subseteq S \wedge pathstart\ g = x \wedge pathfinish\ g = y))$
(**is** *?lhs = ?rhs*)

corollary *arc_connected_trans*:
fixes $g :: real \Rightarrow 'a::\{complete\_space,real\_normed\_vector\}$
assumes $arc\ g\ arc\ h\ pathfinish\ g = pathstart\ h\ pathstart\ g \neq pathfinish\ h$
obtains *i* **where** $arc\ i\ path\_image\ i \subseteq path\_image\ g \cup path\_image\ h$
$pathstart\ i = pathstart\ g\ pathfinish\ i = pathfinish\ h$

6.10.3 Accessibility of frontier points

end

6.11 The Urysohn lemma, its consequences and other advanced material about metric spaces

theory *Urysohn*
imports *Abstract_Topological_Spaces Abstract_Metric_Spaces Infinite_Sum Arcwise_Connected*
begin

6.11.1 Urysohn lemma and Tietze's theorem

proposition *Urysohn_lemma*:
fixes *a b* :: *real*
assumes $normal\_space\ X\ closedin\ X\ S\ closedin\ X\ T\ disjnt\ S\ T\ a \leq b$
obtains *f* **where** $continuous\_map\ X\ (top\_of\_set\ \{a..b\})\ f\ f\ `\ S \subseteq \{a\}\ f\ `\ T \subseteq \{b\}$

theorem *Tietze_extension_closed_real_interval*:
assumes *normal_space X* **and** *closedin X S*
and *contf*: *continuous_map (subtopology X S) euclideanreal f*
and *fim*: $f\ `\ S \subseteq \{a..b\}$ **and** $a \leq b$
obtains *g*
where *continuous_map X euclideanreal g*
$\bigwedge x.\ x \in S \Longrightarrow g\ x = f\ x\ g\ `\ topspace\ X \subseteq \{a..b\}$

theorem *Tietze_extension_realinterval*:
assumes *XS*: *normal_space X closedin X S* **and** *T*: $is\_interval\ T\ T \neq \{\}$
and *contf*: *continuous_map (subtopology X S) euclideanreal f*
and $f\ `\ S \subseteq T$
obtains *g* **where** $continuous\_map\ X\ euclideanreal\ g\ \ g\ `\ topspace\ X \subseteq T\ \ \bigwedge x.\ x \in S \Longrightarrow g\ x = f\ x$

6.11.2 Random metric space stuff

6.11.3 Hereditarily normal spaces

6.11.4 Completely regular spaces

proposition *locally_compact_regular_imp_completely_regular_space*:
assumes *locally_compact_space X Hausdorff_space X* ∨ *regular_space X*
shows *completely_regular_space X*

proposition *completely_regular_space_product_topology*:
completely_regular_space (*product_topology X I*) ⟷
(∃ *i*∈*I*. *X i* = *trivial_topology*) ∨ (∀ *i* ∈ *I*. *completely_regular_space* (*X i*))
(**is** *?lhs* ⟷ *?rhs*)

6.11.5 More generally, the k-ification functor

6.11.6 One-point compactifications and the Alexandroff extension construction

proposition *kc_space_one_point_compactification_gen*:
assumes *compact_space X*
shows *kc_space X* ⟷
openin X (*topspace X* − {*a*}) ∧ (∀ *K*. *compactin X K* ∧ *a*∉*K* ⟶ *closedin X K*) ∧
k_space (*subtopology X* (*topspace X* − {*a*})) ∧ *kc_space* (*subtopology X* (*topspace X* − {*a*}))
(**is** *?lhs* ⟷ *?rhs*)

proposition *istopology_Alexandroff_open*: *istopology* (*Alexandroff_open X*)

proposition *regular_space_one_point_compactification*:
assumes *compact_space* X **and** *ope*: *openin* X (*topspace* $X - \{a\}$)
and §: $\bigwedge K$. ⟦*compactin* (*subtopology* X (*topspace* $X - \{a\}$)) K; *closedin* (*subtopology* X (*topspace* $X - \{a\}$)) K⟧ $\Longrightarrow$ *closedin* X K
shows *regular_space* $X \longleftrightarrow$
regular_space (*subtopology* X (*topspace* $X - \{a\}$)) $\wedge$ *locally_compact_space* (*subtopology* X (*topspace* $X - \{a\}$))
(**is** *?lhs* $\longleftrightarrow$ *?rhs*)

proposition *Hausdorff_space_one_point_compactification_asymmetric_prod*:
assumes *compact_space* X
shows *Hausdorff_space* $X \longleftrightarrow$
kc_space (*prod_topology* X (*subtopology* X (*topspace* $X - \{a\}$))) $\wedge$
k_space (*prod_topology* X (*subtopology* X (*topspace* $X - \{a\}$))) (**is** *?lhs* $\longleftrightarrow$ *?rhs*)

6.11.7 Extending continuous maps "pointwise" in a regular space

6.11.8 Extending Cauchy continuous functions to the closure

6.11.9 Metric space of bounded functions

6.11.10 Metric space of continuous bounded functions

6.11.11 Existence of completion for any metric space M as a subspace of M=>R

6.11.12 Contractions

6.11.13 The Baire Category Theorem

6.11.14 Sierpinski-Hausdorff type results about countable closed unions

6.11.15 The Tychonoff embedding

6.11.16 Urysohn and Tietze analogs for completely regular spaces

6.11.17 Size bounds on connected or path-connected spaces

6.11.18 Lavrentiev extension etc

6.11.19 Embedding in products and hence more about completely metrizable spaces

6.11.20 Theorems from Kuratowski

6.11.21 A perfect set in common cases must have at least the cardinality of the continuum

proposition *Kuratowski_component_number_invariance_aux*:
assumes *compact_space X* **and** *HsX*: *Hausdorff_space X*
and *lcX*: *locally_connected_space X* **and** *hnX*: *hereditarily normal_space X*
and *hom*: (*subtopology X S*) *homeomorphic_space* (*subtopology X T*)
and *leXS*: {..<*n*::*nat*} ≲ *connected_components_of* (*subtopology X* (*topspace X* − *S*))
assumes §: ⋀*S T*.
⟦*closedin X S*; *closedin X T*; (*subtopology X S*) *homeomorphic_space* (*subtopology X T*);
{..<*n*::*nat*} ≲ *connected_components_of* (*subtopology X* (*topspace X* − *S*))⟧
⟹ {..<*n*::*nat*} ≲ *connected_components_of* (*subtopology X* (*topspace X* − *T*))
shows {..<*n*::*nat*} ≲ *connected_components_of* (*subtopology X* (*topspace X* − *T*))

theorem *Kuratowski_component_number_invariance*:
assumes *compact_space X Hausdorff_space X locally_connected_space X hereditarily normal_space X*
shows ((∀ *S T n*.
closedin X S ∧ *closedin X T* ∧
(*subtopology X S*) *homeomorphic_space* (*subtopology X T*)
⟶ (*connected_components_of*
(*subtopology X* (*topspace X* − *S*)) ≈ {..<*n*::*nat*} ⟷
connected_components_of
(*subtopology X* (*topspace X* − *T*)) ≈ {..<*n*::*nat*})) ⟷
(∀ *S T n*.
(*subtopology X S*) *homeomorphic_space* (*subtopology X T*)

$\longrightarrow$ (*connected_components_of*
(*subtopology X* (*topspace X* $-$ *S*)) $\approx$ {..<*n*::*nat*} $\longleftrightarrow$
connected_components_of
(*subtopology X* (*topspace X* $-$ *T*)) $\approx$ {..<*n*::*nat*})))
(**is** *?lhs* = *?rhs*)

end
theory *Sparse_In*
imports *Homotopy*

begin

6.11.22 A set of points sparse in another set

6.11.23 Co-sparseness filter

end
theory *Isolated*
imports *Elementary_Metric_Spaces Sparse_In*

begin

6.11.24 Isolate and discrete

end

6.12 Operator Norm

theory *Operator_Norm*
imports *Complex_Main*
begin

definition
onorm :: (*'a*::*real_normed_vector* $\Rightarrow$ *'b*::*real_normed_vector*) $\Rightarrow$ *real* **where**
onorm f = (*SUP x. norm* (*f x*) / *norm x*)

proposition *onorm_bound*:
assumes *0* $\leq$ *b* **and** $\bigwedge$*x. norm* (*f x*) $\leq$ *b* $*$ *norm x*
shows *onorm f* $\leq$ *b*

end

6.13 Limits on the Extended Real Number Line

theory *Extended_Real_Limits*
imports
  Topology_Euclidean_Space
  HOL−Library.Extended_Real
  HOL−Library.Extended_Nonnegative_Real
  HOL−Library.Indicator_Function
begin

6.13.1 Extended-Real.thy

Continuity of addition

Continuity of multiplication

Continuity of division

6.13.2 Extended-Nonnegative-Real.thy

6.13.3 monoset

6.13.4 Relate extended reals and the indicator function

end

6.14 Radius of Convergence and Summation Tests

theory *Summation_Tests*
imports
  Complex_Main
  HOL−Library.Discrete_Functions
  HOL−Library.Extended_Real
  HOL−Library.Liminf_Limsup
  Extended_Real_Limits
begin

6.14.1 Convergence tests for infinite sums

theorem *root_test_convergence'*:
  fixes $f :: nat \Rightarrow 'a :: banach$
  defines $l \equiv limsup\ (\lambda n.\ ereal\ (root\ n\ (norm\ (f\ n))))$

assumes *l*: $l < 1$
shows *summable f*

theorem *root_test_divergence*:
fixes f :: $nat \Rightarrow 'a$ :: *banach*
defines $l \equiv limsup\ (\lambda n.\ ereal\ (root\ n\ (norm\ (f\ n))))$
assumes *l*: $l > 1$
shows $\neg summable\ f$

theorem *condensation_test*:
assumes *mono*: $\bigwedge m.\ 0 < m \Longrightarrow f\ (Suc\ m) \le f\ m$
assumes *nonneg*: $\bigwedge n.\ f\ n \ge 0$
shows $summable\ f \longleftrightarrow summable\ (\lambda n.\ 2\hat{}n * f\ (2\hat{}n))$

theorem *summable_complex_powr_iff*:
assumes $Re\ s < -1$
shows $summable\ (\lambda n.\ exp\ (of\_real\ (ln\ (of\_nat\ n)) * s))$
theorem *kummers_test_convergence*:
fixes $f\ p$ :: $nat \Rightarrow real$
assumes *pos_f*: $eventually\ (\lambda n.\ f\ n > 0)\ sequentially$
assumes *nonneg_p*: $eventually\ (\lambda n.\ p\ n \ge 0)\ sequentially$
defines $l \equiv liminf\ (\lambda n.\ ereal\ (p\ n * f\ n\ /\ f\ (Suc\ n) - p\ (Suc\ n)))$
assumes *l*: $l > 0$
shows *summable f*

theorem *kummers_test_divergence*:
fixes $f\ p$ :: $nat \Rightarrow real$
assumes *pos_f*: $eventually\ (\lambda n.\ f\ n > 0)\ sequentially$
assumes *pos_p*: $eventually\ (\lambda n.\ p\ n > 0)\ sequentially$
assumes *divergent_p*: $\neg summable\ (\lambda n.\ inverse\ (p\ n))$
defines $l \equiv limsup\ (\lambda n.\ ereal\ (p\ n * f\ n\ /\ f\ (Suc\ n) - p\ (Suc\ n)))$
assumes *l*: $l < 0$
shows $\neg summable\ f$
theorem *ratio_test_convergence*:
fixes f :: $nat \Rightarrow real$
assumes *pos_f*: $eventually\ (\lambda n.\ f\ n > 0)\ sequentially$
defines $l \equiv liminf\ (\lambda n.\ ereal\ (f\ n\ /\ f\ (Suc\ n)))$
assumes *l*: $l > 1$
shows *summable f*

theorem *ratio_test_divergence*:
fixes f :: $nat \Rightarrow real$
assumes *pos_f*: $eventually\ (\lambda n.\ f\ n > 0)\ sequentially$
defines $l \equiv limsup\ (\lambda n.\ ereal\ (f\ n\ /\ f\ (Suc\ n)))$
assumes *l*: $l < 1$
shows $\neg summable\ f$
theorem *raabes_test_convergence*:
fixes f :: $nat \Rightarrow real$

assumes *pos*: *eventually* (λ*n*. *f n* > *0*) *sequentially*
defines *l* ≡ *liminf* (λ*n*. *ereal* (*of_nat n* ∗ (*f n* / *f* (*Suc n*) − *1*)))
assumes *l*: *l* > *1*
shows *summable f*

theorem *raabes_test_divergence*:
fixes *f* :: *nat* ⇒ *real*
assumes *pos*: *eventually* (λ*n*. *f n* > *0*) *sequentially*
defines *l* ≡ *limsup* (λ*n*. *ereal* (*of_nat n* ∗ (*f n* / *f* (*Suc n*) − *1*)))
assumes *l*: *l* < *1*
shows ¬*summable f*

6.14.2 Radius of convergence

definition *conv_radius* :: (*nat* ⇒ *'a* :: *banach*) ⇒ *ereal* **where**
conv_radius f = *inverse* (*limsup* (λ*n*. *ereal* (*root n* (*norm* (*f n*)))))

theorem *abs_summable_in_conv_radius*:
fixes *f* :: *nat* ⇒ *'a* :: {*banach*, *real_normed_div_algebra*}
assumes *ereal* (*norm z*) < *conv_radius f*
shows *summable* (λ*n*. *norm* (*f n* ∗ *z* ^ *n*))

theorem *not_summable_outside_conv_radius*:
fixes *f* :: *nat* ⇒ *'a* :: {*banach*, *real_normed_div_algebra*}
assumes *ereal* (*norm z*) > *conv_radius f*
shows ¬*summable* (λ*n*. *f n* ∗ *z* ^ *n*)

end

6.15 Uniform Limit and Uniform Convergence

theory *Uniform_Limit*
imports *Connected Summation_Tests Infinite_Sum*
begin

6.15.1 Definition

definition *uniformly_on* :: *'a set* ⇒ (*'a* ⇒ *'b*::*metric_space*) ⇒ (*'a* ⇒ *'b*) *filter*
where *uniformly_on S l* = (*INF e*∈{*0* <..}. *principal* {*f*. ∀ *x*∈*S*. *dist* (*f x*) (*l x*) < *e*})

abbreviation
uniform_limit S f l ≡ *filterlim f* (*uniformly_on S l*)

proposition *uniform_limit_iff*:
uniform_limit S f l F ⟷ (∀ *e*>*0*. ∀$_F$ *n in F*. ∀ *x*∈*S*. *dist* (*f n x*) (*l x*) < *e*)

6.15.2 Exchange limits

proposition *swap_uniform_limit*:
assumes *f*: $\forall_F$ *n in F*. (*f n* ⟶ *g n*) (*at x within S*)
assumes *g*: (*g* ⟶ *l*) *F*
assumes *uc*: *uniform_limit S f h F*
assumes ¬*trivial_limit F*
shows (*h* ⟶ *l*) (*at x within S*)

6.15.3 Uniform limit theorem

theorem *uniform_limit_theorem*:
assumes *c*: $\forall_F$ *n in F*. *continuous_on A* (*f n*)
assumes *ul*: *uniform_limit A f l F*
assumes ¬ *trivial_limit F*
shows *continuous_on A l*

6.15.4 Comparison Test

6.15.5 Weierstrass M-Test

proposition *Weierstrass_m_test_ev*:
fixes *f* :: _ ⇒ _ ⇒ _ :: *banach*
assumes *eventually* (λ*n*. ∀ *x*∈*A*. *norm* (*f n x*) ≤ *M n*) *sequentially*
assumes *summable M*
shows *uniform_limit A* (λ*n x*. ∑ *i*<*n*. *f i x*) (λ*x*. *suminf* (λ*i*. *f i x*)) *sequentially*

6.15.6 Power series and uniform convergence

proposition *powser_uniformly_convergent*:
fixes *a* :: *nat* ⇒ ′*a*::{*real_normed_div_algebra*,*banach*}
assumes *r* < *conv_radius a*
shows *uniformly_convergent_on* (*cball* ξ *r*) (λ*n x*. ∑ *i*<*n*. *a i* ∗ (*x* − ξ) ^ *i*)

end

6.16 Bounded Linear Function

theory *Bounded_Linear_Function*
imports
Topology_Euclidean_Space
Operator_Norm
Uniform_Limit

Function_Topology

begin

6.16.1 Type of bounded linear functions

typedef (**overloaded**) ('*a*, '*b*) *blinfun* (‹(‹*notation*=‹*infix* $\Rightarrow_L$››_ $\Rightarrow_L$ /_)› [*22*, *21*] *21*) =
{*f*::'*a*::*real_normed_vector*⇒'*b*::*real_normed_vector*. *bounded_linear f*}
morphisms *blinfun_apply Blinfun*

6.16.2 Type class instantiations

instantiation *blinfun* :: (*real_normed_vector*, *real_normed_vector*) *real_normed_vector*
begin

lift_definition *norm_blinfun* :: '*a* $\Rightarrow_L$ '*b* ⇒ *real* **is** *onorm*
lift_definition *zero_blinfun* :: '*a* $\Rightarrow_L$ '*b* **is** λ*x*. *0*

lift_definition *plus_blinfun* :: '*a* $\Rightarrow_L$ '*b* ⇒ '*a* $\Rightarrow_L$ '*b* ⇒ '*a* $\Rightarrow_L$ '*b*
is λ*f g x*. *f x* + *g x*

lift_definition *scaleR_blinfun*::*real* ⇒ '*a* $\Rightarrow_L$ '*b* ⇒ '*a* $\Rightarrow_L$ '*b* **is** λ*r f x*. *r* $*_R$ *f x*

6.16.3 The strong operator topology on continuous linear operators

definition *strong_operator_topology*::('*a*::*real_normed_vector* $\Rightarrow_L$'*b*::*real_normed_vector*) *topology*
where *strong_operator_topology* = *pullback_topology UNIV blinfun_apply euclidean*

end

6.17 Derivative

theory *Derivative*
imports
Bounded_Linear_Function

Line_Segment
Convex_Euclidean_Space
begin

6.17.1 Derivatives

proposition *has_derivative_within'*:
$(f\ has\_derivative\ f')(at\ x\ within\ s) \longleftrightarrow$
$bounded\_linear\ f' \wedge$
$(\forall e>0.\ \exists d>0.\ \forall x'\in s.\ 0 < norm\ (x' - x) \wedge norm\ (x' - x) < d \longrightarrow$
$norm\ (f\ x' - f\ x - f'(x' - x))\ /\ norm\ (x' - x) < e)$

6.17.2 Differentiability

definition
$differentiable\_on :: ('a::real\_normed\_vector \Rightarrow 'b::real\_normed\_vector) \Rightarrow 'a\ set \Rightarrow bool$
(**infix** ‹*differentiable'_on*› *50*)
where $f\ differentiable\_on\ s \longleftrightarrow (\forall x\in s.\ f\ differentiable\ (at\ x\ within\ s))$

6.17.3 Frechet derivative and Jacobian matrix

proposition *frechet_derivative_works*:
$f\ differentiable\ net \longleftrightarrow (f\ has\_derivative\ (frechet\_derivative\ f\ net))\ net$

6.17.4 Differentiability implies continuity

proposition *differentiable_imp_continuous_within*:
$f\ differentiable\ (at\ x\ within\ s) \Longrightarrow continuous\ (at\ x\ within\ s)\ f$

6.17.5 The chain rule

proposition *diff_chain_within*[*derivative_intros*]:
assumes $(f\ has\_derivative\ f')\ (at\ x\ within\ s)$
and $(g\ has\_derivative\ g')\ (at\ (f\ x)\ within\ (f\ `\ s))$
shows $((g \circ f)\ has\_derivative\ (g' \circ f'))(at\ x\ within\ s)$

6.17.6 Uniqueness of derivative

The general result is a bit messy because we need approachability of the limit point from any direction. But OK for nontrivial intervals etc.

proposition *frechet_derivative_unique_within*:

fixes f :: $'a$::*euclidean_space* ⇒ $'b$::*real_normed_vector*
assumes *1*: (f *has_derivative* f') (*at* x *within* S)
and *2*: (f *has_derivative* f'') (*at* x *within* S)
and S: $\bigwedge i\ e.\ [\![i{\in}Basis;\ e{>}0]\!] \Longrightarrow \exists\, d.\ 0 < |d| \wedge |d| < e \wedge (x + d *_R i) \in S$
shows $f' = f''$

proposition *frechet_derivative_unique_within_closed_interval*:
fixes f::$'a$::*euclidean_space* ⇒ $'b$::*real_normed_vector*
assumes *ab*: $\bigwedge i.\ i{\in}Basis \Longrightarrow a{\cdot}i < b{\cdot}i$
and x: $x \in$ *cbox* a b
and (f *has_derivative* f') (*at* x *within* *cbox* a b)
and (f *has_derivative* f'') (*at* x *within* *cbox* a b)
shows $f' = f''$

6.17.7 Derivatives of local minima and maxima are zero

6.17.8 One-dimensional mean value theorem

6.17.9 More general bound theorems

proposition *differentiable_bound_general*:
fixes f :: *real* ⇒ $'a$::*real_normed_vector*
assumes $a < b$
and *f_cont*: *continuous_on* $\{a..b\}$ f
and *phi_cont*: *continuous_on* $\{a..b\}$ φ
and f': $\bigwedge x.\ a < x \Longrightarrow x < b \Longrightarrow$ (f *has_vector_derivative* f' x) (*at* x)
and *phi'*: $\bigwedge x.\ a < x \Longrightarrow x < b \Longrightarrow$ (φ *has_vector_derivative* φ' x) (*at* x)
and *bnd*: $\bigwedge x.\ a < x \Longrightarrow x < b \Longrightarrow$ *norm* $(f'\ x) \le \varphi'\ x$
shows *norm* $(f\ b - f\ a) \le \varphi\ b - \varphi\ a$

6.17.10 Differentiability of inverse function (most basic form)

proposition *has_derivative_inverse*:
fixes f :: $'a$::*real_normed_vector* ⇒ $'b$::*real_normed_vector*
assumes *compact* S
and $x \in S$
and *fx*: $f\ x \in$ *interior* (f ‘ S)
and *continuous_on* S f
and *gf*: $\bigwedge y.\ y \in S \Longrightarrow g\ (f\ y) = y$
and B: (f *has_derivative* f') (*at* x) *bounded_linear* g' $g' \circ f' = id$
shows (g *has_derivative* g') (*at* (f x))
proposition *has_derivative_locally_injective*:
fixes f :: $'n$::*euclidean_space* ⇒ $'m$::*euclidean_space*
assumes $a \in S$
and *open* S
and *bling*: *bounded_linear* g'
and $g' \circ f'\ a = id$
and *derf*: $\bigwedge x.\ x \in S \Longrightarrow$ (f *has_derivative* f' x) (*at* x)

and $\bigwedge e.\ e > 0 \Longrightarrow \exists\, d{>}0.\ \forall\, x.\ dist\ a\ x < d \longrightarrow onorm\ (\lambda v.\ f'\ x\ v - f'\ a\ v) < e$
obtains r **where** $r > 0$ $ball\ a\ r \subseteq S$ $inj\_on\ f\ (ball\ a\ r)$

6.17.11 Uniformly convergent sequence of derivatives

proposition *has_derivative_sequence*:
fixes $f :: nat \Rightarrow 'a::real\_normed\_vector \Rightarrow 'b::banach$
assumes *convex S*
and *derf*: $\bigwedge n\ x.\ x \in S \Longrightarrow ((f\ n)\ has\_derivative\ (f'\ n\ x))\ (at\ x\ within\ S)$
and *nle*: $\bigwedge e.\ e > 0 \Longrightarrow \forall_F\ n\ in\ sequentially.\ \forall\, x{\in}S.\ \forall\, h.\ norm\ (f'\ n\ x\ h - g'\ x\ h) \le e * norm\ h$
and $x0 \in S$
and *lim*: $((\lambda n.\ f\ n\ x0) \longrightarrow l)\ sequentially$
shows $\exists\, g.\ \forall\, x{\in}S.\ (\lambda n.\ f\ n\ x) \longrightarrow g\ x \wedge (g\ has\_derivative\ g'(x))\ (at\ x\ within\ S)$

6.17.12 Differentiation of a series

proposition *has_derivative_series*:
fixes $f :: nat \Rightarrow 'a::real\_normed\_vector \Rightarrow 'b::banach$
assumes *convex S*
and $\bigwedge n\ x.\ x \in S \Longrightarrow ((f\ n)\ has\_derivative\ (f'\ n\ x))\ (at\ x\ within\ S)$
and $\bigwedge e.\ e{>}0 \Longrightarrow \forall_F\ n\ in\ sequentially.\ \forall\, x{\in}S.\ \forall\, h.\ norm\ (sum\ (\lambda i.\ f'\ i\ x\ h)\ \{..{<}n\} - g'\ x\ h) \le e * norm\ h$
and $x \in S$
and $(\lambda n.\ f\ n\ x)\ sums\ l$
shows $\exists\, g.\ \forall\, x{\in}S.\ (\lambda n.\ f\ n\ x)\ sums\ (g\ x) \wedge (g\ has\_derivative\ g'\ x)\ (at\ x\ within\ S)$

6.17.13 Derivative as a vector

proposition *vector_derivative_works*:
$f\ differentiable\ net \longleftrightarrow (f\ has\_vector\_derivative\ (vector\_derivative\ f\ net))\ net$
(**is** $?l = ?r$)

6.17.14 Field differentiability

definition $field\_differentiable :: ['a \Rightarrow 'a::real\_normed\_field, 'a\ filter] \Rightarrow bool$
(**infixr** ‹(field'_differentiable)› *50*)
where $f\ field\_differentiable\ F \equiv \exists f'.\ (f\ has\_field\_derivative\ f')\ F$

6.17.15 Field derivative

definition *deriv* :: $('a \Rightarrow 'a::real\_normed\_field) \Rightarrow 'a \Rightarrow 'a$ **where**
$deriv\ f\ x \equiv SOME\ D.\ DERIV\ f\ x :> D$

proposition *field_differentiable_derivI*:
$f\ field\_differentiable\ (at\ x) \Longrightarrow (f\ has\_field\_derivative\ deriv\ f\ x)\ (at\ x)$

6.17.16 Relation between convexity and derivative

proposition *convex_on_imp_above_tangent*:
assumes *convex*: $convex\_on\ A\ f$ **and** *connected*: $connected\ A$
assumes *c*: $c \in interior\ A$ **and** $x : x \in A$
assumes *deriv*: $(f\ has\_field\_derivative\ f')\ (at\ c\ within\ A)$
shows $f\ x - f\ c \geq f' * (x - c)$

6.17.17 Partial derivatives

proposition *has_derivative_partialsI*:
fixes $f::'a::real\_normed\_vector \Rightarrow 'b::real\_normed\_vector \Rightarrow 'c::real\_normed\_vector$
assumes *fx*: $((\lambda x.\ f\ x\ y)\ has\_derivative\ fx)\ (at\ x\ within\ X)$
assumes *fy*: $\bigwedge x\ y.\ x \in X \Longrightarrow y \in Y \Longrightarrow ((\lambda y.\ f\ x\ y)\ has\_derivative\ blinfun\_apply\ (fy\ x\ y))\ (at\ y\ within\ Y)$
assumes *fy_cont*[*unfolded continuous_within*]: $continuous\ (at\ (x, y)\ within\ X \times Y)\ (\lambda(x, y).\ fy\ x\ y)$
assumes $y \in Y\ convex\ Y$
shows $((\lambda(x, y).\ f\ x\ y)\ has\_derivative\ (\lambda(tx, ty).\ fx\ tx + fy\ x\ y\ ty))\ (at\ (x, y)\ within\ X \times Y)$

6.17.18 The Inverse Function Theorem

theorem *inverse_function_theorem*:
fixes $f::'a::euclidean\_space \Rightarrow 'a$
and $f'::'a \Rightarrow ('a \Rightarrow_L 'a)$
assumes $open\ U$
and *derf*: $\bigwedge x.\ x \in U \Longrightarrow (f\ has\_derivative\ (blinfun\_apply\ (f'\ x)))\ (at\ x)$
and *contf*: $continuous\_on\ U\ f'$
and $x0 \in U$
and *invf*: $invf\ o_L\ f'\ x0 = id\_blinfun$
obtains $U'\ V\ g\ g'$ **where** $open\ U'\ U' \subseteq U\ x0 \in U'\ open\ V\ f\ x0 \in V\ homeomorphism\ U'\ V\ f\ g$
$\bigwedge y.\ y \in V \Longrightarrow (g\ has\_derivative\ (g'\ y))\ (at\ y)$
$\bigwedge y.\ y \in V \Longrightarrow g'\ y = inv\ (blinfun\_apply\ (f'(g\ y)))$
$\bigwedge y.\ y \in V \Longrightarrow bij\ (blinfun\_apply\ (f'(g\ y)))$

6.17.19 The concept of continuously differentiable

definition *C1_differentiable_on* :: (*real* ⇒ ′*a*::*real_normed_vector*) ⇒ *real set* ⇒ *bool*
(**infix** ‹*C1*′*_differentiable*′*_on*› *50*)
where
f C1_differentiable_on S ⟷
(∃ *D*. (∀ *x* ∈ *S*. (*f has_vector_derivative* (*D x*)) (*at x*)) ∧ *continuous_on S D*)

definition *piecewise_C1_differentiable_on*
(**infixr** ‹*piecewise*′*_C1*′*_differentiable*′*_on*› *50*)
where *f piecewise_C1_differentiable_on i* ≡
continuous_on i f ∧
(∃ *S*. *finite S* ∧ (*f C1_differentiable_on* (*i* − *S*)))

end

6.18 Finite Cartesian Products of Euclidean Spaces

theory *Cartesian_Euclidean_Space*
imports *Derivative*
begin

6.18.1 Closures and interiors of halfspaces

6.18.2 Bounds on components etc. relative to operator norm

6.18.3 Convex Euclidean Space

6.18.4 Arbitrarily good rational approximations

proposition *matrix_rational_approximation*:
fixes *A* :: *real*^′*n*^′*m*
assumes *e* > *0*
obtains *B* **where** ⋀*i j*. *B*\$*i*\$*j* ∈ ℚ *onorm*(λ*x*. (*A* − *B*) ∗*v x*) < *e*

6.18.5 Derivative

definition *jacobian f net* = *matrix*(*frechet_derivative f net*)

proposition *jacobian_works*:
(*f*::(*real*^′*a*) ⇒ (*real*^′*b*)) *differentiable net* ⟷

(*f has_derivative* ($\lambda h.$ (*jacobian f net*) $*v$ *h*)) *net* (**is** *?lhs* = *?rhs*)
proposition *differential_zero_maxmin_cart*:
fixes *f::real^'a ⇒ real^'b*
assumes $0 < e$ ((∀ *y* ∈ *ball x e*. (*f y*)$*k* ≤ (*f x*)$*k*) ∨ (∀ *y*∈*ball x e*. (*f x*)$*k* ≤ (*f y*)$*k*))
f differentiable (*at x*)
shows *jacobian f* (*at x*) $ *k* = *0*

end

6.19 Complex Analysis Basics

theory *Complex_Analysis_Basics*
imports *Derivative HOL−Library.Nonpos_Ints Uncountable_Sets*
begin

6.19.1 Holomorphic functions

definition *holomorphic_on* :: [*complex* ⇒ *complex*, *complex set*] ⇒ *bool*
(**infixl** ‹(*holomorphic'_on*)› *50*)
where *f holomorphic_on s* ≡ ∀ *x*∈*s*. *f field_differentiable* (*at x within s*)

named_theorems *holomorphic_intros structural introduction rules for holomorphic_on*

6.19.2 Analyticity on a set

definition *analytic_on* (**infixl** ‹(*analytic'_on*)› *50*)
where *f analytic_on S* ≡ ∀ *x* ∈ *S*. ∃ε. *0* < ε ∧ *f holomorphic_on* (*ball x* ε)

named_theorems *analytic_intros introduction rules for proving analyticity*

end

6.20 Complex Transcendental Functions

theory *Complex_Transcendental*
imports
Complex_Analysis_Basics Summation_Tests HOL−Library.Periodic_Fun
begin

6.20.1 Möbius transformations

definition *moebius a b c d* ≡ (λ*z*. (*a*∗*z*+*b*) / (*c*∗*z*+*d* :: *'a* :: *field*))

theorem *moebius_inverse*:
assumes *a* ∗ *d* ≠ *b* ∗ *c* *c* ∗ *z* + *d* ≠ *0*

shows $moebius\ d\ (-b)\ (-c)\ a\ (moebius\ a\ b\ c\ d\ z) = z$

6.20.2 Euler and de Moivre formulas

theorem *exp_Euler*: $exp(i * z) = cos(z) + i * sin(z)$

theorem *Euler*: $exp(z) = of\_real(exp(Re\ z)) *$
$(of\_real(cos(Im\ z)) + i * of\_real(sin(Im\ z)))$

6.20.3 The argument of a complex number (HOL Light version)

definition *is_Arg* :: [*complex*,*real*] ⇒ *bool*
where $is\_Arg\ z\ r \equiv z = of\_real(norm\ z) * exp(i * of\_real\ r)$

definition *Arg2pi* :: *complex* ⇒ *real*
where $Arg2pi\ z \equiv if\ z = 0\ then\ 0\ else\ THE\ t.\ 0 \leq t \wedge t < 2*pi \wedge is\_Arg\ z\ t$

6.20.4 The principal branch of the Complex logarithm

instantiation *complex* :: *ln*
begin

definition *ln_complex* :: *complex* ⇒ *complex*
where $ln\_complex \equiv \lambda z.\ THE\ w.\ exp\ w = z\ \&\ -pi < Im(w)\ \&\ Im(w) \leq pi$
theorem *Ln_series*:
fixes *z* :: *complex*
assumes $norm\ z < 1$
shows $(\lambda n.\ (-1)\hat{}Suc\ n\ /\ of\_nat\ n * z\hat{}n)\ sums\ ln\ (1 + z)$ (**is** $(\lambda n.\ ?f\ n * z\hat{}n)\ sums\ \_$)

corollary *norm_Ln_prod_le*:
fixes *f* :: *'a* ⇒ *complex*
assumes $\bigwedge x.\ x \in A \Longrightarrow f\ x \neq 0$
shows $cmod\ (Ln\ (prod\ f\ A)) \leq (\sum x \in A.\ cmod\ (Ln\ (f\ x)))$

6.20.5 The Argument of a Complex Number

lemma *Arg_def*:
shows $Arg\ z = (if\ z = 0\ then\ 0\ else\ Im\ (Ln\ z))$

6.20.6 The Unwinding Number and the Ln product Formula

definition *unwinding* :: *complex* ⇒ *int* **where**
unwinding z ≡ *THE k. of_int k* = (*z* − *Ln*(*exp z*)) / (*of_real*(*2*∗*pi*) ∗ i)

6.20.7 Characterisation of *Im* (*Ln z*) (Wenda Li)

6.20.8 Complex arctangent

definition *Arctan* :: *complex* ⇒ *complex* **where**
Arctan ≡ λ*z*. (i/*2*) ∗ *Ln*((*1* − i∗*z*) / (*1* + i∗*z*))

theorem *Arctan_series*:
assumes *z*: *norm* (*z* :: *complex*) < *1*
defines *g* ≡ λ*n*. *if odd n then* −i∗i^*n* / *n else 0*
defines *h* ≡ λ*z n*. (−*1*)^*n* / *of_nat* (*2*∗*n*+*1*) ∗ (*z*::*complex*)^(*2*∗*n*+*1*)
shows (λ*n*. *g n* ∗ *z*^*n*) *sums Arctan z*
and *h z sums Arctan z*
theorem *ln_series_quadratic*:
assumes *x*: *x* > (*0*::*real*)
shows (λ*n*. (*2*∗((*x* − *1*) / (*x* + *1*)) ^ (*2*∗*n*+*1*) / *of_nat* (*2*∗*n*+*1*))) *sums ln x*

6.20.9 Inverse Sine

definition *Arcsin* :: *complex* ⇒ *complex* **where**
Arcsin ≡ λ*z*. −i ∗ *Ln*(i ∗ *z* + *csqrt*(*1* − z^2))

6.20.10 Inverse Cosine

definition *Arccos* :: *complex* ⇒ *complex* **where**
Arccos ≡ λ*z*. −i ∗ *Ln*(*z* + i ∗ *csqrt*(*1* − z^2))

6.20.11 Roots of unity

theorem *complex_root_unity*:
fixes *j*::*nat*
assumes *n* ≠ *0*
shows *exp*(*2* ∗ *of_real pi* ∗ i ∗ *of_nat j* / *of_nat n*)^*n* = *1*

corollary *bij_betw_roots_unity*:
bij_betw (λ*j*. *exp*(*2* ∗ *of_real pi* ∗ i ∗ *of_nat j* / *of_nat n*))
{..<*n*} {*exp*(*2* ∗ *of_real pi* ∗ i ∗ *of_nat j* / *of_nat n*) | *j*. *j* < *n*}

end

Chapter 7

Measure and Integration Theory

theory *Sigma_Algebra*
imports
  Complex_Main
  HOL−Library.Countable_Set
  HOL−Library.FuncSet
  HOL−Library.Indicator_Function
  HOL−Library.Extended_Nonnegative_Real
  HOL−Library.Disjoint_Sets
begin

7.1 Sigma Algebra

7.1.1 Families of sets

locale *subset_class* =
  fixes Ω :: *'a set* **and** M :: *'a set set*
  assumes *space_closed*: $M \subseteq Pow\ \Omega$
locale *semiring_of_sets* = *subset_class* +
  assumes *empty_sets*[*iff*]: $\{\} \in M$
  assumes *Int*[*intro*]: $\bigwedge a\ b.\ a \in M \Longrightarrow b \in M \Longrightarrow a \cap b \in M$
  assumes *Diff_cover*:
    $\bigwedge a\ b.\ a \in M \Longrightarrow b \in M \Longrightarrow \exists\, C{\subseteq}M.\ finite\ C \wedge disjoint\ C \wedge a - b = \bigcup C$
locale *ring_of_sets* = *semiring_of_sets* +
  assumes *Un* [*intro*]: $\bigwedge a\ b.\ a \in M \Longrightarrow b \in M \Longrightarrow a \cup b \in M$
locale *algebra* = *ring_of_sets* +
  assumes *top* [*iff*]: $\Omega \in M$

proposition *algebra_iff_Un*:
  $algebra\ \Omega\ M \longleftrightarrow$
    $M \subseteq Pow\ \Omega\ \wedge$
    $\{\} \in M\ \wedge$
    $(\forall\, a \in M.\ \Omega - a \in M)\ \wedge$

(∀a ∈ M. ∀ b ∈ M. a ∪ b ∈ M) (**is** _ ⟷ ?Un)

proposition algebra_iff_Int:
algebra Ω M ⟷
M ⊆ Pow Ω & {} ∈ M &
(∀a ∈ M. Ω − a ∈ M) &
(∀a ∈ M. ∀ b ∈ M. a ∩ b ∈ M) (**is** _ ⟷ ?Int)
locale sigma_algebra = algebra +
assumes countable_nat_UN [intro]: ⋀A. range A ⊆ M ⟹ (⋃i::nat. A i) ∈ M

Sigma algebras can naturally be created as the closure of any set of M with regard to the properties just postulated.

inductive_set sigma_sets :: 'a set ⇒ 'a set set ⇒ 'a set set
for sp :: 'a set **and** A :: 'a set set
where
Basic[intro, simp]: a ∈ A ⟹ a ∈ sigma_sets sp A
| Empty: {} ∈ sigma_sets sp A
| Compl: a ∈ sigma_sets sp A ⟹ sp − a ∈ sigma_sets sp A
| Union: (⋀i::nat. a i ∈ sigma_sets sp A) ⟹ (⋃i. a i) ∈ sigma_sets sp A
definition closed_cdi :: 'a set ⇒ 'a set set ⇒ bool **where**
closed_cdi Ω M ⟷
M ⊆ Pow Ω &
(∀s ∈ M. Ω − s ∈ M) &
(∀A. (range A ⊆ M) & (A 0 = {}) & (∀n. A n ⊆ A (Suc n)) ⟶
(⋃i. A i) ∈ M) &
(∀A. (range A ⊆ M) & disjoint_family A ⟶ (⋃i::nat. A i) ∈ M)

locale Dynkin_system = subset_class +
assumes space: Ω ∈ M
and compl[intro!]: ⋀A. A ∈ M ⟹ Ω − A ∈ M
and UN[intro!]: ⋀A. disjoint_family A ⟹ range A ⊆ M
⟹ (⋃i::nat. A i) ∈ M
definition Int_stable :: 'a set set ⇒ bool **where**
Int_stable M ⟷ (∀ a ∈ M. ∀ b ∈ M. a ∩ b ∈ M)
definition Dynkin :: 'a set ⇒ 'a set set ⇒ 'a set set **where**
Dynkin Ω M = (⋂{D. Dynkin_system Ω D ∧ M ⊆ D})

The reason to introduce Dynkin-systems is the following induction rules for σ-algebras generated by a generator closed under intersection.

proposition sigma_sets_induct_disjoint[consumes 3, case_names basic empty compl union]:
assumes Int_stable G
and closed: G ⊆ Pow Ω
and A: A ∈ sigma_sets Ω G
assumes basic: ⋀A. A ∈ G ⟹ P A
and empty: P {}
and compl: ⋀A. A ∈ sigma_sets Ω G ⟹ P A ⟹ P (Ω − A)

and *union*: $\bigwedge A.$ *disjoint_family* $A \Longrightarrow$ *range* $A \subseteq$ *sigma_sets* Ω $G \Longrightarrow (\bigwedge i.$ P $(A$ $i)) \Longrightarrow P$ $(\bigcup i{::}nat.$ A $i)$
shows P A

7.1.2 Measure type

definition *positive* :: *'a set set* ⇒ (*'a set* ⇒ *ennreal*) ⇒ *bool* **where**
positive M $\mu \longleftrightarrow \mu$ $\{\} = 0$

definition *countably_additive* :: *'a set set* ⇒ (*'a set* ⇒ *ennreal*) ⇒ *bool* **where**
countably_additive M f $\longleftrightarrow$
$(\forall A.$ *range* $A \subseteq M \longrightarrow$ *disjoint_family* $A \longrightarrow (\bigcup i.$ A $i) \in M \longrightarrow$
$(\sum i.$ f $(A$ $i)) = f$ $(\bigcup i.$ A $i))$

definition *measure_space* :: *'a set* ⇒ *'a set set* ⇒ (*'a set* ⇒ *ennreal*) ⇒ *bool* **where**
measure_space Ω A $\mu \longleftrightarrow$
sigma_algebra Ω $A \wedge$ *positive* A $\mu \wedge$ *countably_additive* A μ

typedef *'a measure* =
$\{(\Omega{::}'a\ set,\ A,\ \mu).\ (\forall a{\in}{-}A.\ \mu\ a = 0) \wedge$ *measure_space* Ω A μ $\}$

definition *space* :: *'a measure* ⇒ *'a set* **where**
space M = fst (Rep_measure M)

definition *sets* :: *'a measure* ⇒ *'a set set* **where**
sets M = fst (snd (Rep_measure M))

definition *emeasure* :: *'a measure* ⇒ *'a set* ⇒ *ennreal* **where**
emeasure M = snd (snd (Rep_measure M))

definition *measure* :: *'a measure* ⇒ *'a set* ⇒ *real* **where**
measure M A = enn2real (emeasure M A)

definition *measure_of* :: *'a set* ⇒ *'a set set* ⇒ (*'a set* ⇒ *ennreal*) ⇒ *'a measure* **where**
measure_of Ω A μ =
Abs_measure (Ω, *if* $A \subseteq$ *Pow* Ω *then sigma_sets* Ω A *else* $\{\{\}, \Omega\}$,
$\lambda a.$ *if* $a \in$ *sigma_sets* Ω $A \wedge$ *measure_space* Ω (*sigma_sets* Ω A) μ *then* μ a *else* 0)
proposition *emeasure_measure_of*:
assumes M: $M =$ *measure_of* Ω A μ
assumes *ms*: $A \subseteq$ *Pow* Ω *positive* (*sets M*) μ *countably_additive* (*sets M*) μ
assumes X: $X \in$ *sets M*
shows *emeasure M X* $= \mu$ X
definition *measurable* :: *'a measure* ⇒ *'b measure* ⇒ (*'a* ⇒ *'b*) *set*
(**infixr** ‹$\rightarrow_M$› *60*) **where**
measurable A B $= \{f \in$ *space A* $\rightarrow$ *space B*. $\forall y \in$ *sets B*. f −‘ $y \cap$ *space A* $\in$ *sets*

A}
definition *count_space* :: *'a set* ⇒ *'a measure* **where**
count_space Ω = *measure_of* Ω (*Pow* Ω) (λ*A*. *if finite A then of_nat* (*card A*) *else* ∞)

7.1.3 The smallest σ-algebra regarding a function

definition *vimage_algebra* :: *'a set* ⇒ (*'a* ⇒ *'b*) ⇒ *'b measure* ⇒ *'a measure* **where**
vimage_algebra X f M = *sigma X* {*f* −‘ *A* ∩ *X* | *A*. *A* ∈ *sets M*}

end

7.2 Measurability Prover

theory *Measurable*
imports
Sigma_Algebra
HOL−Library.Order_Continuity
begin

method_setup *measurable* = ‹ *Scan.lift* (*Scan.succeed* (*METHOD o Measurable.measurable_tac*)) ›
measurability prover

simproc_setup *measurable* (*A* ∈ *sets M* | *f* ∈ *measurable M N*) =
‹*K Measurable.proc*›

end

7.3 Measure Spaces

theory *Measure_Space*
imports
Measurable HOL−Library.Extended_Nonnegative_Real
begin

7.3.1 μ-null sets

definition *null_sets* :: *'a measure* ⇒ *'a set set* **where**
null_sets M = {*N*∈*sets M*. *emeasure M N* = *0*}

7.3.2 The almost everywhere filter (i.e. quantifier)

definition *ae_filter* :: *'a measure* ⇒ *'a filter* **where**
ae_filter M = (*INF N*∈*null_sets M*. *principal* (*space M* − *N*))

7.3.3 σ-finite Measures

locale *sigma_finite_measure* =
fixes M :: *'a measure*
assumes *sigma_finite_countable*:
$\exists A::'a\ set\ set.\ countable\ A \wedge A \subseteq sets\ M \wedge (\bigcup A) = space\ M \wedge (\forall a{\in}A.\ emeasure\ M\ a \neq \infty)$

7.3.4 Measure space induced by distribution of $(\rightarrow_M)$-functions

definition *distr* :: *'a measure* ⇒ *'b measure* ⇒ (*'a* ⇒ *'b*) ⇒ *'b measure* **where**
$distr\ M\ N\ f =$
$measure\_of\ (space\ N)\ (sets\ N)\ (\lambda A.\ emeasure\ M\ (f\ -`\ A \cap space\ M))$

proposition *distr_distr*:
$g \in measurable\ N\ L \Longrightarrow f \in measurable\ M\ N \Longrightarrow distr\ (distr\ M\ N\ f)\ L\ g = distr\ M\ L\ (g \circ f)$

7.3.5 Set of measurable sets with finite measure

definition *fmeasurable* :: *'a measure* ⇒ *'a set set* **where**
$fmeasurable\ M = \{A{\in}sets\ M.\ emeasure\ M\ A < \infty\}$

7.3.6 Measure spaces with $emeasure\ M\ (space\ M) < \infty$

locale *finite_measure* = *sigma_finite_measure* M **for** M +
assumes *finite_emeasure_space*: $emeasure\ M\ (space\ M) \neq top$

7.3.7 Scaling a measure

definition *scale_measure* :: *ennreal* ⇒ *'a measure* ⇒ *'a measure* **where**
$scale\_measure\ r\ M = measure\_of\ (space\ M)\ (sets\ M)\ (\lambda A.\ r * emeasure\ M\ A)$

7.3.8 Complete lattice structure on measures

proposition *unsigned_Hahn_decomposition*:
assumes [*simp*]: $sets\ N = sets\ M$ **and** [*measurable*]: $A \in sets\ M$
and [*simp*]: $emeasure\ M\ A \neq top\ emeasure\ N\ A \neq top$
shows $\exists Y{\in}sets\ M.\ Y \subseteq A \wedge (\forall X{\in}sets\ M.\ X \subseteq Y \longrightarrow N\ X \leq M\ X) \wedge (\forall X{\in}sets\ M.\ X \subseteq A \longrightarrow X \cap Y = \{\} \longrightarrow M\ X \leq N\ X)$

Define a lexicographical order on *measure*, in the order space, sets and measure. The parts of the lexicographical order are point-wise ordered.

instantiation *measure* :: (*type*) *order_bot*
begin

definition *less_measure* :: *'a measure* ⇒ *'a measure* ⇒ *bool* **where**
less_measure M N ⟷ (*M* ≤ *N* ∧ ¬ *N* ≤ *M*)

definition *bot_measure* :: *'a measure* **where**
bot_measure = *sigma* {} {}

proposition *le_measure*: *sets M* = *sets N* ⟹ *M* ≤ *N* ⟷ (∀ *A*∈*sets M*. *emeasure M A* ≤ *emeasure N A*)

definition *sup_measure'* :: *'a measure* ⇒ *'a measure* ⇒ *'a measure* **where**
sup_measure' A B =
measure_of (*space A*) (*sets A*)
(λ*X*. *SUP Y*∈*sets A*. *emeasure A* (*X* ∩ *Y*) + *emeasure B* (*X* ∩ − *Y*))

definition *sup_lexord* :: *'a* ⇒ *'a* ⇒ (*'a* ⇒ *'b*::*order*) ⇒ *'a* ⇒ *'a* ⇒ *'a* **where**
sup_lexord A B k s c =
(*if k A* = *k B then c else*
if ¬ *k A* ≤ *k B* ∧ ¬ *k B* ≤ *k A then s else*
if k B ≤ *k A then A else B*)

instantiation *measure* :: (*type*) *semilattice_sup*
begin

definition *sup_measure* :: *'a measure* ⇒ *'a measure* ⇒ *'a measure* **where**
sup_measure A B =
sup_lexord A B space (*sigma* (*space A* ∪ *space B*) {})
(*sup_lexord A B sets* (*sigma* (*space A*) (*sets A* ∪ *sets B*)) (*sup_measure' A B*))

definition
Sup_lexord :: (*'a* ⇒ *'b*::*complete_lattice*) ⇒ (*'a set* ⇒ *'a*) ⇒ (*'a set* ⇒ *'a*) ⇒ *'a set* ⇒ *'a*
where
Sup_lexord k c s A =
(*let U* = (*SUP a*∈*A*. *k a*)
in if ∃ *a*∈*A*. *k a* = *U then c* {*a*∈*A*. *k a* = *U*} *else s A*)

instantiation *measure* :: (*type*) *complete_lattice*
begin

definition *Sup_measure'* :: *'a measure set* ⇒ *'a measure* **where**
Sup_measure' M =
measure_of (⋃ *a*∈*M*. *space a*) (⋃ *a*∈*M*. *sets a*)
(λ*X*. (*SUP P*∈{*P*. *finite P* ∧ *P* ⊆ *M* }. *sup_measure.F id P X*))

definition *Sup_measure* :: *'a measure set* ⇒ *'a measure* **where**
Sup_measure =
  Sup_lexord space
    (*Sup_lexord sets Sup_measure'*
      (λU. *sigma* ($\bigcup u \in U$. *space u*) ($\bigcup u \in U$. *sets u*)))
    (λU. *sigma* ($\bigcup u \in U$. *space u*) {})

definition *Inf_measure* :: *'a measure set* ⇒ *'a measure* **where**
  Inf_measure A = *Sup* {*x*. ∀ *a*∈*A*. *x* ≤ *a*}

definition *inf_measure* :: *'a measure* ⇒ *'a measure* ⇒ *'a measure* **where**
  inf_measure a b = *Inf* {*a*, *b*}

definition *top_measure* :: *'a measure* **where**
  top_measure = *Inf* {}

end

7.4 Borel Space

theory *Borel_Space*
imports
  Measurable Derivative Ordered_Euclidean_Space Extended_Real_Limits
begin

proposition *open_prod_generated*: *open* = *generate_topology* {*A* × *B* | *A B*. *open A* ∧ *open B*}

proposition *mono_on_imp_deriv_nonneg*:
  assumes *mono*: *mono_on A f* **and** *deriv*: (*f has_real_derivative D*) (*at x*)
  assumes *x* ∈ *interior A*
  shows *D* ≥ *0*

proposition *mono_on_ctble_discont*:
  fixes *f* :: *real* ⇒ *real*
  fixes *A* :: *real set*
  assumes *mono_on A f*
  shows *countable* {*a*∈*A*. ¬ *continuous* (*at a within A*) *f*}

7.4.1 Generic Borel spaces

definition (**in** *topological_space*) *borel* :: *'a measure* **where**
  borel = *sigma UNIV* {*S*. *open S*}

theorem *second_countable_borel_measurable*:
  fixes *X* :: *'a::second_countable_topology set set*
  assumes *eq*: *open = generate_topology X*
  shows *borel = sigma UNIV X*

proposition *borel_eq_countable_basis*:
  fixes *B::'a::topological_space set set*
  assumes *countable B*
  assumes *topological_basis B*
  shows *borel = sigma UNIV B*

7.4.2 Borel spaces on order topologies

7.4.3 Borel spaces on topological monoids

7.4.4 Borel spaces on Euclidean spaces

7.4.5 Borel measurable operators

lemma *borel_measurable_complex_iff*:
  $f \in$ *borel_measurable M* $\longleftrightarrow$
    $(\lambda x.$ *Re* $(f\ x)) \in$ *borel_measurable M* $\wedge$ $(\lambda x.$ *Im* $(f\ x)) \in$ *borel_measurable M*
(**is** *?lhs* $\longleftrightarrow$ *?rhs*)

7.4.6 Borel space on the extended reals

theorem *borel_measurable_ereal_iff_real*:
  fixes *f* :: *'a* $\Rightarrow$ *ereal*
  shows $f \in$ *borel_measurable M* $\longleftrightarrow$
    $((\lambda x.$ *real_of_ereal* $(f\ x)) \in$ *borel_measurable M* $\wedge$ f -` $\{\infty\} \cap$ *space M* $\in$ *sets M* $\wedge$ $f$ -` $\{-\infty\} \cap$ *space M* $\in$ *sets M*)

7.4.7 Borel space on the extended non-negative reals

definition [*simp*]: *is_borel f M* $\longleftrightarrow$ $f \in$ *borel_measurable M*

7.4.8 LIMSEQ is borel measurable

proposition *measurable_limit* [*measurable*]:
  fixes *f::nat* $\Rightarrow$ *'a* $\Rightarrow$ *'b::first_countable_topology*
  assumes [*measurable*]: $\bigwedge n$*::nat. f n* $\in$ *borel_measurable M*
  shows *Measurable.pred M* $(\lambda x.\ (\lambda n.\ f\ n\ x) \longlongrightarrow c)$

end

7.5 Lebesgue Integration for Nonnegative Functions

theory *Nonnegative_Lebesgue_Integration*
 imports *Measure_Space Borel_Space*
begin

7.5.1 Simple function

definition *simple_function M g* $\longleftrightarrow$
 finite $(g$ ‘ *space* $M) \wedge$
 $(\forall x \in g$ ‘ *space* $M.\ g -$‘ $\{x\} \cap$ *space* $M \in$ *sets* $M)$

lemma *borel_measurable_implies_simple_function_sequence*:
 fixes $u :: 'a \Rightarrow$ *ennreal*
 assumes *u*[*measurable*]: $u \in$ *borel_measurable* M
 shows $\exists f.$ *incseq* $f \wedge (\forall i. (\forall x. f\ i\ x <$ *top*$) \wedge$ *simple_function* $M\ (f\ i)) \wedge u =$ (*SUP* $i.\ f\ i$)

lemma *simple_function_induct*
 [*consumes 1*, *case_names cong set mult add*, *induct set*: *simple_function*]:
 fixes $u :: 'a \Rightarrow$ *ennreal*
 assumes *u*: *simple_function M u*
 assumes *cong*: $\bigwedge f\ g.$ *simple_function* $M\ f \Longrightarrow$ *simple_function* $M\ g \Longrightarrow$ (*AE* x *in* $M.\ f\ x = g\ x) \Longrightarrow P\ f \Longrightarrow P\ g$
 assumes *set*: $\bigwedge A.\ A \in$ *sets* $M \Longrightarrow P$ (*indicator* A)
 assumes *mult*: $\bigwedge u\ c.\ P\ u \Longrightarrow P\ (\lambda x.\ c * u\ x)$
 assumes *add*: $\bigwedge u\ v.\ P\ u \Longrightarrow P\ v \Longrightarrow P\ (\lambda x.\ v\ x + u\ x)$
 shows $P\ u$

lemma *borel_measurable_induct*
 [*consumes 1*, *case_names cong set mult add seq*, *induct set*: *borel_measurable*]:
 fixes $u :: 'a \Rightarrow$ *ennreal*
 assumes *u*: $u \in$ *borel_measurable* M
 assumes *cong*: $\bigwedge f\ g.\ f \in$ *borel_measurable* $M \Longrightarrow g \in$ *borel_measurable* $M \Longrightarrow (\bigwedge x.\ x \in$ *space* $M \Longrightarrow f\ x = g\ x) \Longrightarrow P\ g \Longrightarrow P\ f$
 assumes *set*: $\bigwedge A.\ A \in$ *sets* $M \Longrightarrow P$ (*indicator* A)
 assumes *mult'*: $\bigwedge u\ c.\ c <$ *top* $\Longrightarrow u \in$ *borel_measurable* $M \Longrightarrow (\bigwedge x.\ x \in$ *space* $M \Longrightarrow u\ x <$ *top*$) \Longrightarrow P\ u \Longrightarrow P\ (\lambda x.\ c * u\ x)$
 assumes *add*: $\bigwedge u\ v.\ u \in$ *borel_measurable* $M \Longrightarrow (\bigwedge x.\ x \in$ *space* $M \Longrightarrow u\ x <$ *top*$) \Longrightarrow P\ u \Longrightarrow v \in$ *borel_measurable* $M \Longrightarrow (\bigwedge x.\ x \in$ *space* $M \Longrightarrow v\ x <$ *top*$) \Longrightarrow (\bigwedge x.\ x \in$ *space* $M \Longrightarrow u\ x = 0 \vee v\ x = 0) \Longrightarrow P\ v \Longrightarrow P\ (\lambda x.\ v\ x + u\ x)$
 assumes *seq*: $\bigwedge U.\ (\bigwedge i.\ U\ i \in$ *borel_measurable* $M) \Longrightarrow (\bigwedge i\ x.\ x \in$ *space* $M \Longrightarrow U\ i\ x <$ *top*$) \Longrightarrow (\bigwedge i.\ P\ (U\ i)) \Longrightarrow$ *incseq* $U \Longrightarrow u =$ (*SUP* $i.\ U\ i) \Longrightarrow P$ (*SUP* $i.\ U\ i$)

shows *P u*

7.5.2 Simple integral

definition *simple_integral* :: *'a measure* ⇒ (*'a* ⇒ *ennreal*) ⇒ *ennreal* (‹$integral^S$›)
where
$integral^S$ *M f* = ($\sum$ *x* ∈ *f* ‘ *space M*. *x* ∗ *emeasure M* (*f* −‘ {*x*} ∩ *space M*))

7.5.3 Integral on nonnegative functions

definition *nn_integral* :: *'a measure* ⇒ (*'a* ⇒ *ennreal*) ⇒ *ennreal* (‹$integral^N$›)
where
$integral^N$ *M f* = (*SUP g* ∈ {*g*. *simple_function M g* ∧ *g* ≤ *f*}. $integral^S$ *M g*)

theorem *nn_integral_monotone_convergence_SUP_AE*:
assumes *f*: ⋀*i*. *AE x in M*. *f i x* ≤ *f* (*Suc i*) *x* ⋀*i*. *f i* ∈ *borel_measurable M*
shows ($\int^+$ *x*. (*SUP i*. *f i x*) ∂*M*) = (*SUP i*. $integral^N$ *M* (*f i*))

theorem *nn_integral_suminf*:
assumes *f*: ⋀*i*. *f i* ∈ *borel_measurable M*
shows ($\int^+$ *x*. ($\sum$ *i*. *f i x*) ∂*M*) = ($\sum$ *i*. $integral^N$ *M* (*f i*))

theorem *nn_integral_Markov_inequality*:
assumes *u*: (λ*x*. *u x* ∗ *indicator A x*) ∈ *borel_measurable M* **and** *A* ∈ *sets M*
shows (*emeasure M*) ({*x*∈*A*. *1* ≤ *c* ∗ *u x*}) ≤ *c* ∗ ($\int^+$ *x*. *u x* ∗ *indicator A x* ∂*M*)
(**is** (*emeasure M*) *?A* ≤ _ ∗ *?PI*)

theorem *nn_integral_monotone_convergence_INF_AE*:
fixes *f* :: *nat* ⇒ *'a* ⇒ *ennreal*
assumes *f*: ⋀*i*. *AE x in M*. *f* (*Suc i*) *x* ≤ *f i x*
and [*measurable*]: ⋀*i*. *f i* ∈ *borel_measurable M*
and *fin*: ($\int^+$ *x*. *f i x* ∂*M*) < ∞
shows ($\int^+$ *x*. (*INF i*. *f i x*) ∂*M*) = (*INF i*. $integral^N$ *M* (*f i*))

theorem *nn_integral_liminf*:
fixes *u* :: *nat* ⇒ *'a* ⇒ *ennreal*
assumes *u*: ⋀*i*. *u i* ∈ *borel_measurable M*
shows ($\int^+$ *x*. *liminf* (λ*n*. *u n x*) ∂*M*) ≤ *liminf* (λ*n*. $integral^N$ *M* (*u n*))

theorem *nn_integral_limsup*:
fixes *u* :: *nat* ⇒ *'a* ⇒ *ennreal*
assumes [*measurable*]: ⋀*i*. *u i* ∈ *borel_measurable M w* ∈ *borel_measurable M*
assumes *bounds*: ⋀*i*. *AE x in M*. *u i x* ≤ *w x* **and** *w*: ($\int^+$*x*. *w x* ∂*M*) < ∞
shows *limsup* (λ*n*. $integral^N$ *M* (*u n*)) ≤ ($\int^+$ *x*. *limsup* (λ*n*. *u n x*) ∂*M*)

theorem *nn_integral_dominated_convergence*:
assumes [*measurable*]:

$\bigwedge i.\ u\ i \in borel\_measurable\ M\ u' \in borel\_measurable\ M\ w \in borel\_measurable\ M$
and *bound*: $\bigwedge j.\ AE\ x\ in\ M.\ u\ j\ x \leq w\ x$
and *w*: $(\int^+ x.\ w\ x\ \partial M) < \infty$
and *u'*: $AE\ x\ in\ M.\ (\lambda i.\ u\ i\ x) \longrightarrow u'\ x$
shows $(\lambda i.\ (\int^+ x.\ u\ i\ x\ \partial M)) \longrightarrow (\int^+ x.\ u'\ x\ \partial M)$

theorem *nn_integral_lfp*:
assumes *sets*[*simp*]: $\bigwedge s.\ sets\ (M\ s) = sets\ N$
assumes *f*: *sup_continuous f*
assumes *g*: *sup_continuous g*
assumes *meas*: $\bigwedge F.\ F \in borel\_measurable\ N \Longrightarrow f\ F \in borel\_measurable\ N$
assumes *step*: $\bigwedge F\ s.\ F \in borel\_measurable\ N \Longrightarrow integral^N\ (M\ s)\ (f\ F) = g\ (\lambda s.\ integral^N\ (M\ s)\ F)\ s$
shows $(\int^+ \omega.\ lfp\ f\ \omega\ \partial M\ s) = lfp\ g\ s$

theorem *nn_integral_gfp*:
assumes *sets*[*simp*]: $\bigwedge s.\ sets\ (M\ s) = sets\ N$
assumes *f*: *inf_continuous f* **and** *g*: *inf_continuous g*
assumes *meas*: $\bigwedge F.\ F \in borel\_measurable\ N \Longrightarrow f\ F \in borel\_measurable\ N$
assumes *bound*: $\bigwedge F\ s.\ F \in borel\_measurable\ N \Longrightarrow (\int^+ x.\ f\ F\ x\ \partial M\ s) < \infty$
assumes *non_zero*: $\bigwedge s.\ emeasure\ (M\ s)\ (space\ (M\ s)) \neq 0$
assumes *step*: $\bigwedge F\ s.\ F \in borel\_measurable\ N \Longrightarrow integral^N\ (M\ s)\ (f\ F) = g\ (\lambda s.\ integral^N\ (M\ s)\ F)\ s$
shows $(\int^+ \omega.\ gfp\ f\ \omega\ \partial M\ s) = gfp\ g\ s$

7.5.4 Integral under concrete measures

definition *density* :: *'a measure* ⇒ (*'a* ⇒ *ennreal*) ⇒ *'a measure* **where**
$density\ M\ f = measure\_of\ (space\ M)\ (sets\ M)\ (\lambda A.\ \int^+ x.\ f\ x * indicator\ A\ x\ \partial M)$

lemma *nn_integral_density*:
assumes *f*: $f \in borel\_measurable\ M$
assumes *g*: $g \in borel\_measurable\ M$
shows $integral^N\ (density\ M\ f)\ g = (\int^+ x.\ f\ x * g\ x\ \partial M)$
definition *point_measure* :: *'a set* ⇒ (*'a* ⇒ *ennreal*) ⇒ *'a measure* **where**
$point\_measure\ A\ f = density\ (count\_space\ A)\ f$
definition $uniform\_measure\ M\ A = density\ M\ (\lambda x.\ indicator\ A\ x\ /\ emeasure\ M\ A)$
definition $uniform\_count\_measure\ A = point\_measure\ A\ (\lambda x.\ 1\ /\ card\ A)$

end

7.6 Binary Product Measure

theory *Binary_Product_Measure*

imports *Nonnegative_Lebesgue_Integration*
begin

7.6.1 Binary products

definition *pair_measure* (**infixr** ‹$\bigotimes_M$› *80*) **where**
$A \bigotimes_M B = measure\_of\ (space\ A \times space\ B)$
$\{a \times b \mid a\ b.\ a \in sets\ A \wedge b \in sets\ B\}$
$(\lambda X.\ \int^+ x.\ (\int^+ y.\ indicator\ X\ (x,y)\ \partial B)\ \partial A)$

proposition (**in** *sigma_finite_measure*) *emeasure_pair_measure_Times*:
assumes A: $A \in sets\ N$ **and** B: $B \in sets\ M$
shows $emeasure\ (N \bigotimes_M M)\ (A \times B) = emeasure\ N\ A * emeasure\ M\ B$

7.6.2 Binary products of σ-finite emeasure spaces

proposition (**in** *pair_sigma_finite*) *sigma_finite_up_in_pair_measure_generator*:
defines $E \equiv \{A \times B \mid A\ B.\ A \in sets\ M1 \wedge B \in sets\ M2\}$
shows $\exists F{::}nat \Rightarrow ('a \times 'b)\ set.\ range\ F \subseteq E \wedge incseq\ F \wedge (\bigcup i.\ F\ i) = space\ M1 \times space\ M2 \wedge$
$(\forall i.\ emeasure\ (M1 \bigotimes_M M2)\ (F\ i) \neq \infty)$

7.6.3 Fubinis theorem

proposition (**in** *pair_sigma_finite*) *nn_integral_snd*:
assumes $f[measurable]$: $f \in borel\_measurable\ (M1 \bigotimes_M M2)$
shows $(\int^+ y.\ (\int^+ x.\ f\ (x,\ y)\ \partial M1)\ \partial M2) = integral^N\ (M1 \bigotimes_M M2)\ f$

theorem (**in** *pair_sigma_finite*) *Fubini*:
assumes f: $f \in borel\_measurable\ (M1 \bigotimes_M M2)$
shows $(\int^+ y.\ (\int^+ x.\ f\ (x,\ y)\ \partial M1)\ \partial M2) = (\int^+ x.\ (\int^+ y.\ f\ (x,\ y)\ \partial M2)\ \partial M1)$

theorem (**in** *pair_sigma_finite*) *Fubini'*:
assumes f: $case\_prod\ f \in borel\_measurable\ (M1 \bigotimes_M M2)$
shows $(\int^+ y.\ (\int^+ x.\ f\ x\ y\ \partial M1)\ \partial M2) = (\int^+ x.\ (\int^+ y.\ f\ x\ y\ \partial M2)\ \partial M1)$

7.6.4 Products on counting spaces, densities and distributions

proposition *sigma_prod*:
assumes X_cover: $\exists E \subseteq A.\ countable\ E \wedge X = \bigcup E$ **and** A: $A \subseteq Pow\ X$
assumes Y_cover: $\exists E \subseteq B.\ countable\ E \wedge Y = \bigcup E$ **and** B: $B \subseteq Pow\ Y$

shows *sigma X A* $\bigotimes_M$ *sigma Y B* = *sigma* (*X* × *Y*) {*a* × *b* | *a b*. *a* ∈ *A* ∧ *b* ∈ *B*}
(**is** *?P* = *?S*)

proposition *sets_pair_eq*:
assumes *Ea*: *Ea* ⊆ *Pow* (*space A*) *sets A* = *sigma_sets* (*space A*) *Ea*
and *Ca*: *countable Ca Ca* ⊆ *Ea* ⋃ *Ca* = *space A*
and *Eb*: *Eb* ⊆ *Pow* (*space B*) *sets B* = *sigma_sets* (*space B*) *Eb*
and *Cb*: *countable Cb Cb* ⊆ *Eb* ⋃ *Cb* = *space B*
shows *sets* (*A* $\bigotimes_M$ *B*) = *sets* (*sigma* (*space A* × *space B*) { *a* × *b* | *a b*. *a* ∈ *Ea* ∧ *b* ∈ *Eb* })
(**is** _ = *sets* (*sigma* *?Ω* *?E*))

proposition *borel_prod*:
(*borel* $\bigotimes_M$ *borel*) = (*borel* :: (*'a::second_countable_topology* × *'b::second_countable_topology*) *measure*)
(**is** *?P* = *?B*)

proposition *pair_measure_count_space*:
assumes *A*: *finite A* **and** *B*: *finite B*
shows *count_space A* $\bigotimes_M$ *count_space B* = *count_space* (*A* × *B*) (**is** *?P* = *?C*)

theorem *pair_measure_density*:
assumes *f*: *f* ∈ *borel_measurable M1*
assumes *g*: *g* ∈ *borel_measurable M2*
assumes *sigma_finite_measure M2 sigma_finite_measure* (*density M2 g*)
shows *density M1 f* $\bigotimes_M$ *density M2 g* = *density* (*M1* $\bigotimes_M$ *M2*) (λ(*x*,*y*). *f x* ∗ *g y*) (**is** *?L* = *?R*)

proposition *nn_integral_fst_count_space*:
($\int^+$ *x*. $\int^+$ *y*. *f* (*x*, *y*) ∂*count_space UNIV* ∂*count_space UNIV*) = $integral^N$ (*count_space UNIV*) *f*
(**is** *?lhs* = *?rhs*)

proposition *nn_integral_snd_count_space*:
($\int^+$ *y*. $\int^+$ *x*. *f* (*x*, *y*) ∂*count_space UNIV* ∂*count_space UNIV*) = $integral^N$ (*count_space UNIV*) *f*
(**is** *?lhs* = *?rhs*)

7.6.5 Product of Borel spaces

theorem *borel_Times*:
fixes *A* :: *'a::topological_space set* **and** *B* :: *'b::topological_space set*
assumes *A*: *A* ∈ *sets borel* **and** *B*: *B* ∈ *sets borel*
shows *A* × *B* ∈ *sets borel*

end

7.7 Finite Product Measure

theory *Finite_Product_Measure*
imports *Binary_Product_Measure Function_Topology*
begin

7.7.1 Finite product spaces

definition *prod_emb* **where**
prod_emb I M K X = (λ*x*. *restrict x K*) −' *X* ∩ (Π_E *i*∈*I*. *space* (*M i*))

definition *PiM* :: *'i set* ⇒ (*'i* ⇒ *'a measure*) ⇒ (*'i* ⇒ *'a*) *measure* **where**
PiM I M = *extend_measure* (Π_E *i*∈*I*. *space* (*M i*))
{(*J*, *X*). (*J* ≠ {} ∨ *I* = {}) ∧ *finite J* ∧ *J* ⊆ *I* ∧ *X* ∈ (Π *j*∈*J*. *sets* (*M j*))}
(λ(*J*, *X*). *prod_emb I M J* (Π_E *j*∈*J*. *X j*))
(λ(*J*, *X*). ∏*j*∈*J* ∪ {*i*∈*I*. *emeasure* (*M i*) (*space* (*M i*)) ≠ *1*}. *if j* ∈ *J then emeasure* (*M j*) (*X j*) *else emeasure* (*M j*) (*space* (*M j*)))

definition *prod_algebra* :: *'i set* ⇒ (*'i* ⇒ *'a measure*) ⇒ (*'i* ⇒ *'a*) *set set* **where**
prod_algebra I M = (λ(*J*, *X*). *prod_emb I M J* (Π_E *j*∈*J*. *X j*)) '
{(*J*, *X*). (*J* ≠ {} ∨ *I* = {}) ∧ *finite J* ∧ *J* ⊆ *I* ∧ *X* ∈ (Π *j*∈*J*. *sets* (*M j*))}

proposition *prod_algebra_mono*:
assumes *space*: ⋀*i*. *i* ∈ *I* ⟹ *space* (*E i*) = *space* (*F i*)
assumes *sets*: ⋀*i*. *i* ∈ *I* ⟹ *sets* (*E i*) ⊆ *sets* (*F i*)
shows *prod_algebra I E* ⊆ *prod_algebra I F*
proposition *prod_algebra_cong*:
assumes *I* = *J* **and** (⋀*i*. *i* ∈ *I* ⟹ *sets* (*M i*) = *sets* (*N i*))
shows *prod_algebra I M* = *prod_algebra J N*

proposition *sets_PiM_single*: *sets* (*PiM I M*) =
sigma_sets (Π_E *i*∈*I*. *space* (*M i*)) {{*f*∈Π_E *i*∈*I*. *space* (*M i*). *f i* ∈ *A*} | *i A*. *i* ∈ *I* ∧ *A* ∈ *sets* (*M i*)}
(**is** _ = *sigma_sets* *?Ω ?R*)

proposition *sets_PiM_sigma*:
assumes *Ω_cover*: ⋀*i*. *i* ∈ *I* ⟹ ∃*S*⊆*E i*. *countable S* ∧ Ω *i* = ⋃*S*
assumes *E*: ⋀*i*. *i* ∈ *I* ⟹ *E i* ⊆ *Pow* (Ω *i*)
assumes *J*: ⋀*j*. *j* ∈ *J* ⟹ *finite j* ⋃*J* = *I*
defines *P* ≡ {{*f*∈(Π_E *i*∈*I*. Ω *i*). ∀*i*∈*j*. *f i* ∈ *A i*} | *A j*. *j* ∈ *J* ∧ *A* ∈ *Pi j E*}
shows *sets* (Π_M *i*∈*I*. *sigma* (Ω *i*) (*E i*)) = *sets* (*sigma* (Π_E *i*∈*I*. Ω *i*) *P*)

proposition *measurable_PiM*:
assumes *space*: *f* ∈ *space N* → (Π_E *i*∈*I*. *space* (*M i*))
assumes *sets*: ⋀*X J*. *J* ≠ {} ∨ *I* = {} ⟹ *finite J* ⟹ *J* ⊆ *I* ⟹ (⋀*i*. *i* ∈ *J*

$\Longrightarrow X\ i \in sets\ (M\ i)) \Longrightarrow$
  $f -` prod\_emb\ I\ M\ J\ (Pi_E\ J\ X) \cap space\ N \in sets\ N$
shows $f \in measurable\ N\ (PiM\ I\ M)$

proposition *measurable_fun_upd*:
assumes I: $I = J \cup \{i\}$
assumes $f[measurable]$: $f \in measurable\ N\ (PiM\ J\ M)$
assumes $h[measurable]$: $h \in measurable\ N\ (M\ i)$
shows $(\lambda x.\ (f\ x)\ (i := h\ x)) \in measurable\ N\ (PiM\ I\ M)$

proposition *measure_eqI_PiM_finite*:
assumes $[simp]$: $finite\ I\ sets\ P = PiM\ I\ M\ sets\ Q = PiM\ I\ M$
assumes eq: $\bigwedge A.\ (\bigwedge i.\ i \in I \Longrightarrow A\ i \in sets\ (M\ i)) \Longrightarrow P\ (Pi_E\ I\ A) = Q\ (Pi_E\ I\ A)$
assumes A: $range\ A \subseteq prod\_algebra\ I\ M\ (\bigcup i.\ A\ i) = space\ (PiM\ I\ M)\ \bigwedge i::nat.\ P\ (A\ i) \neq \infty$
shows $P = Q$

proposition *measure_eqI_PiM_infinite*:
assumes $[simp]$: $sets\ P = PiM\ I\ M\ sets\ Q = PiM\ I\ M$
assumes eq: $\bigwedge A\ J.\ finite\ J \Longrightarrow J \subseteq I \Longrightarrow (\bigwedge i.\ i \in J \Longrightarrow A\ i \in sets\ (M\ i)) \Longrightarrow$
  $P\ (prod\_emb\ I\ M\ J\ (Pi_E\ J\ A)) = Q\ (prod\_emb\ I\ M\ J\ (Pi_E\ J\ A))$
assumes A: $finite\_measure\ P$
shows $P = Q$

proposition (**in** *finite_product_sigma_finite*) *sigma_finite_pairs*:
$\exists F::'i \Rightarrow nat \Rightarrow 'a\ set.$
  $(\forall i \in I.\ range\ (F\ i) \subseteq sets\ (M\ i)) \wedge$
  $(\forall k.\ \forall i \in I.\ emeasure\ (M\ i)\ (F\ i\ k) \neq \infty) \wedge incseq\ (\lambda k.\ \Pi_E\ i \in I.\ F\ i\ k) \wedge$
  $(\bigcup k.\ \Pi_E\ i \in I.\ F\ i\ k) = space\ (PiM\ I\ M)$

lemma (**in** *product_sigma_finite*) *distr_merge*:
assumes $IJ[simp]$: $I \cap J = \{\}$ **and** fin: $finite\ I\ finite\ J$
shows $distr\ (Pi_M\ I\ M \bigotimes_M Pi_M\ J\ M)\ (Pi_M\ (I \cup J)\ M)\ (merge\ I\ J) = Pi_M\ (I \cup J)\ M$
  (**is** *?D = ?P*)

proposition (**in** *product_sigma_finite*) *product_nn_integral_fold*:
assumes IJ: $I \cap J = \{\}\ finite\ I\ finite\ J$
and $f[measurable]$: $f \in borel\_measurable\ (Pi_M\ (I \cup J)\ M)$
shows $integral^N\ (Pi_M\ (I \cup J)\ M)\ f = (\int^+ x.\ (\int^+ y.\ f\ (merge\ I\ J\ (x, y))\ \partial(Pi_M\ J\ M))\ \partial(Pi_M\ I\ M))$
  (**is** *?lhs = ?rhs*)

proposition (**in** *product_sigma_finite*) *product_nn_integral_insert*:
assumes $I[simp]$: $finite\ I\ i \notin I$
  and f: $f \in borel\_measurable\ (Pi_M\ (insert\ i\ I)\ M)$
shows $integral^N\ (Pi_M\ (insert\ i\ I)\ M)\ f = (\int^+ x.\ (\int^+ y.\ f\ (x(i := y))\ \partial(M\ i))$

$\partial(Pi_M\ I\ M))$

proposition (**in** *product_sigma_finite*) *product_nn_integral_pair*:
assumes [*measurable*]: $case\_prod\ f \in borel\_measurable\ (M\ x \bigotimes_M\ M\ y)$
assumes *xy*: $x \neq y$
shows $(\int^+ \sigma.\ f\ (\sigma\ x)\ (\sigma\ y)\ \partial PiM\ \{x,\ y\}\ M) = (\int^+ z.\ f\ (fst\ z)\ (snd\ z)\ \partial(M\ x \bigotimes_M\ M\ y))$

7.7.2 Measurability

proposition *sets_PiM_equal_borel*:
$sets\ (Pi_M\ UNIV\ (\lambda i::('a::countable).\ borel::('b::second\_countable\_topology\ measure))) = sets\ borel$

end

7.8 Caratheodory Extension Theorem

theory *Caratheodory*
imports *Measure_Space*
begin

7.8.1 Characterizations of Measures

definition *outer_measure_space* **where**
$outer\_measure\_space\ M\ f \longleftrightarrow positive\ M\ f \wedge increasing\ M\ f \wedge countably\_subadditive\ M\ f$

Lambda Systems

definition *lambda_system* :: $'a\ set \Rightarrow 'a\ set\ set \Rightarrow ('a\ set \Rightarrow ennreal) \Rightarrow 'a\ set\ set$
where
$lambda\_system\ \Omega\ M\ f = \{l \in M.\ \forall x \in M.\ f\ (l \cap x) + f\ ((\Omega - l) \cap x) = f\ x\}$

proposition (**in** *sigma_algebra*) *lambda_system_caratheodory*:
assumes *oms*: $outer\_measure\_space\ M\ f$
and *A*: $range\ A \subseteq lambda\_system\ \Omega\ M\ f$
and *disj*: $disjoint\_family\ A$
shows $(\bigcup i.\ A\ i) \in lambda\_system\ \Omega\ M\ f \wedge (\sum i.\ f\ (A\ i)) = f\ (\bigcup i.\ A\ i)$

proposition (**in** *sigma_algebra*) *caratheodory_lemma*:
assumes *oms*: $outer\_measure\_space\ M\ f$
defines $L \equiv lambda\_system\ \Omega\ M\ f$
shows $measure\_space\ \Omega\ L\ f$

definition *outer_measure* :: *'a set set* ⇒ (*'a set* ⇒ *ennreal*) ⇒ *'a set* ⇒ *ennreal* **where**
outer_measure M f X =
(*INF A*∈{*A*. *range A* ⊆ *M* ∧ *disjoint_family A* ∧ *X* ⊆ (⋃ *i*. *A i*)}. ∑ *i*. *f* (*A i*))

7.8.2 Caratheodory's theorem

theorem (**in** *ring_of_sets*) *caratheodory'*:
assumes *posf*: *positive M f* **and** *ca*: *countably_additive M f*
shows ∃ μ :: *'a set* ⇒ *ennreal*. (∀ *s* ∈ *M*. μ *s* = *f s*) ∧ *measure_space* Ω (*sigma_sets* Ω *M*) μ

7.8.3 Volumes

definition *volume* :: *'a set set* ⇒ (*'a set* ⇒ *ennreal*) ⇒ *bool* **where**
volume M f ⟷
(*f* {} = *0*) ∧ (∀ *a*∈*M*. *0* ≤ *f a*) ∧
(∀ *C*⊆*M*. *disjoint C* ⟶ *finite C* ⟶ ⋃ *C* ∈ *M* ⟶ *f* (⋃ *C*) = (∑ *c*∈*C*. *f c*))

proposition *volume_finite_additive*:
assumes *volume M f*
assumes *A*: ⋀*i*. *i* ∈ *I* ⟹ *A i* ∈ *M disjoint_family_on A I finite I* ⋃(*A* ‘ *I*) ∈ *M*
shows *f* (⋃(*A* ‘ *I*)) = (∑ *i*∈*I*. *f* (*A i*))

proposition (**in** *semiring_of_sets*) *extend_volume*:
assumes *volume M* μ
shows ∃ μ'. *volume generated_ring* μ' ∧ (∀ *a*∈*M*. μ' *a* = μ *a*)

Caratheodory on semirings

theorem (**in** *semiring_of_sets*) *caratheodory*:
assumes *pos*: *positive M* μ **and** *ca*: *countably_additive M* μ
shows ∃ μ' :: *'a set* ⇒ *ennreal*. (∀ *s* ∈ *M*. μ' *s* = μ *s*) ∧ *measure_space* Ω (*sigma_sets* Ω *M*) μ'

proposition *extend_measure_caratheodory_pair*:
fixes *G* :: *'i* ⇒ *'j* ⇒ *'a set*
assumes *M*: *M* = *extend_measure* Ω {(*a*, *b*). *P a b*} (λ(*a*, *b*). *G a b*) (λ(*a*, *b*). μ *a b*)
assumes *P i j*
assumes *semiring*: *semiring_of_sets* Ω {*G a b* | *a b*. *P a b*}
assumes *empty*: ⋀*i j*. *P i j* ⟹ *G i j* = {} ⟹ μ *i j* = *0*
assumes *inj*: ⋀*i j k l*. *P i j* ⟹ *P k l* ⟹ *G i j* = *G k l* ⟹ μ *i j* = μ *k l*
assumes *nonneg*: ⋀*i j*. *P i j* ⟹ *0* ≤ μ *i j*
assumes *add*: ⋀*A*::*nat* ⇒ *'i*. ⋀*B*::*nat* ⇒ *'j*. ⋀*j k*.

$(\bigwedge n.\ P\ (A\ n)\ (B\ n)) \Longrightarrow P\ j\ k \Longrightarrow disjoint\_family\ (\lambda n.\ G\ (A\ n)\ (B\ n)) \Longrightarrow$
$(\bigcup i.\ G\ (A\ i)\ (B\ i)) = G\ j\ k \Longrightarrow (\sum n.\ \mu\ (A\ n)\ (B\ n)) = \mu\ j\ k$
shows *emeasure M (G i j) = μ i j*

end

7.9 Bochner Integration for Vector-Valued Functions

theory *Bochner_Integration*
imports *Finite_Product_Measure*
beginproposition *borel_measurable_implies_sequence_metric*:
fixes *f* :: *'a ⇒ 'b* :: {*metric_space*, *second_countable_topology*}
assumes [*measurable*]: $f \in borel\_measurable\ M$
shows $\exists F.\ (\forall i.\ simple\_function\ M\ (F\ i)) \wedge (\forall x{\in}space\ M.\ (\lambda i.\ F\ i\ x) \longrightarrow f\ x) \wedge$
$(\forall i.\ \forall x{\in}space\ M.\ dist\ (F\ i\ x)\ z \leq 2 * dist\ (f\ x)\ z)$

definition *simple_bochner_integral* :: *'a measure ⇒ ('a ⇒ 'b::real_vector) ⇒ 'b*
where
$simple\_bochner\_integral\ M\ f = (\sum y{\in}f`space\ M.\ measure\ M\ \{x{\in}space\ M.\ f\ x = y\} *_R\ y)$

proposition *simple_bochner_integral_partition*:
assumes *f*: *simple_bochner_integrable M f* **and** *g*: *simple_function M g*
assumes *sub*: $\bigwedge x\ y.\ x \in space\ M \Longrightarrow y \in space\ M \Longrightarrow g\ x = g\ y \Longrightarrow f\ x = f\ y$
assumes *v*: $\bigwedge x.\ x \in space\ M \Longrightarrow f\ x = v\ (g\ x)$
shows $simple\_bochner\_integral\ M\ f = (\sum y{\in}g\ `\ space\ M.\ measure\ M\ \{x{\in}space\ M.\ g\ x = y\} *_R\ v\ y)$
(**is** _ = *?r*)

proposition *has_bochner_integral_implies_finite_norm*:
$has\_bochner\_integral\ M\ f\ x \Longrightarrow (\int^+ x.\ norm\ (f\ x)\ \partial M) < \infty$

proposition *has_bochner_integral_norm_bound*:
assumes *i*: *has_bochner_integral M f x*
shows $norm\ x \leq (\int^+ x.\ norm\ (f\ x)\ \partial M)$

definition *lebesgue_integral* (‹$integral^L$›) **where**
$integral^L\ M\ f = (if\ \exists x.\ has\_bochner\_integral\ M\ f\ x\ then\ THE\ x.\ has\_bochner\_integral\ M\ f\ x\ else\ 0)$

proposition *nn_integral_dominated_convergence_norm*:
fixes *u'* :: _ ⇒ _::{*real_normed_vector*, *second_countable_topology*}
assumes [*measurable*]:
$\bigwedge i.\ u\ i \in borel\_measurable\ M\ u' \in borel\_measurable\ M\ w \in borel\_measurable\ M$
and *bound*: $\bigwedge j.\ AE\ x\ in\ M.\ norm\ (u\ j\ x) \leq w\ x$

and *w*: $(\int^+ x.\ w\ x\ \partial M) < \infty$
and *u'*: *AE x in M*. $(\lambda i.\ u\ i\ x) \longrightarrow u'\ x$
shows $(\lambda i.\ (\int^+ x.\ norm\ (u'\ x - u\ i\ x)\ \partial M)) \longrightarrow 0$

proposition *integrableI_bounded*:
fixes $f :: 'a \Rightarrow 'b::\{banach, second\_countable\_topology\}$
assumes *f*[*measurable*]: $f \in borel\_measurable\ M$ **and** *fin*: $(\int^+ x.\ norm\ (f\ x)\ \partial M) < \infty$
shows *integrable M f*

proposition *nn_integral_eq_integral*:
assumes *f*: *integrable M f*
assumes *nonneg*: *AE x in M*. $0 \le f\ x$
shows $(\int^+ x.\ f\ x\ \partial M) = integral^L\ M\ f$

proposition *integral_norm_bound* [*simp*]:
fixes $f :: \_ \Rightarrow 'a :: \{banach, second\_countable\_topology\}$
shows $norm\ (integral^L\ M\ f) \le (\int x.\ norm\ (f\ x)\ \partial M)$

proposition *integral_abs_bound* [*simp*]:
fixes $f :: 'a \Rightarrow real$ **shows** $abs\ (\int x.\ f\ x\ \partial M) \le (\int x.\ |f\ x|\ \partial M)$

proposition *integrable_induct*[*consumes 1*, *case_names base add lim*, *induct pred*: *integrable*]:
fixes $f :: 'a \Rightarrow 'b::\{banach, second\_countable\_topology\}$
assumes *integrable M f*
assumes *base*: $\bigwedge A\ c.\ A \in sets\ M \Longrightarrow emeasure\ M\ A < \infty \Longrightarrow P\ (\lambda x.\ indicator\ A\ x *_R c)$
assumes *add*: $\bigwedge f\ g.\ integrable\ M\ f \Longrightarrow P\ f \Longrightarrow integrable\ M\ g \Longrightarrow P\ g \Longrightarrow P\ (\lambda x.\ f\ x + g\ x)$
assumes *lim*: $\bigwedge f\ s.\ (\bigwedge i.\ integrable\ M\ (s\ i)) \Longrightarrow (\bigwedge i.\ P\ (s\ i)) \Longrightarrow$
$(\bigwedge x.\ x \in space\ M \Longrightarrow (\lambda i.\ s\ i\ x) \longrightarrow f\ x) \Longrightarrow$
$(\bigwedge i\ x.\ x \in space\ M \Longrightarrow norm\ (s\ i\ x) \le 2 * norm\ (f\ x)) \Longrightarrow integrable\ M\ f \Longrightarrow P\ f$
shows *P f*

theorem *integral_Markov_inequality*:
assumes [*measurable*]: *integrable M u* **and** *AE x in M*. $0 \le u\ x\ 0 < (c::real)$
shows $(emeasure\ M)\ \{x \in space\ M.\ u\ x \ge c\} \le (1/c) * (\int x.\ u\ x\ \partial M)$

theorem *integral_Markov_inequality_measure*:
assumes [*measurable*]: *integrable M u* **and** $A \in sets\ M$ **and** *AE x in M*. $0 \le u\ x\ 0 < (c::real)$
shows $measure\ M\ \{x \in space\ M.\ u\ x \ge c\} \le (\int x.\ u\ x\ \partial M) / c$

theorem (**in** *finite_measure*) *second_moment_method*:
assumes [*measurable*]: $f \in M \rightarrow_M borel$
assumes $integrable\ M\ (\lambda x.\ f\ x\ \hat{}\ 2)$

defines $\mu \equiv$ *lebesgue_integral M f*
assumes $a > 0$
shows *measure M* $\{x{\in}space\ M.\ |f\,x| \geq a\} \leq$ *lebesgue_integral M* $(\lambda x.\ f\,x\ \hat{}\ 2)$ / a^2
proof −
have *integrable*: *integrable M f*
using *assms* **by** (*blast dest*: *square_integrable_imp_integrable*)
have $\{x{\in}space\ M.\ |f\,x| \geq a\} = \{x{\in}space\ M.\ f\,x\ \hat{}\ 2 \geq a^2\}$
using ‹$a > 0$› **by** (*simp flip*: *abs_le_square_iff*)
hence *measure M* $\{x{\in}space\ M.\ |f\,x| \geq a\} =$ *measure M* $\{x{\in}space\ M.\ f\,x\ \hat{}\ 2 \geq a^2\}$
by *simp*
also have ... $\leq$ *lebesgue_integral M* $(\lambda x.\ f\,x\ \hat{}\ 2)$ / a^2
using *assms* **by** (*intro integral_Markov_inequality_measure*) *auto*
finally show *?thesis* **.**
qed

proposition *tendsto_L1_int*:
fixes u :: _ ⇒ _ ⇒ $'b$::{*banach*, *second_countable_topology*}
assumes [*measurable*]: $\bigwedge n.$ *integrable M* $(u\ n)$ *integrable M f*
and $((\lambda n.\ (\int^+ x.\ norm(u\ n\ x - f\,x)\ \partial M)) \longrightarrow 0)\ F$
shows $((\lambda n.\ (\int x.\ u\ n\ x\ \partial M)) \longrightarrow (\int x.\ f\,x\ \partial M))\ F$
proposition *tendsto_L1_AE_subseq*:
fixes u :: *nat* ⇒ $'a$ ⇒ $'b$::{*banach*, *second_countable_topology*}
assumes [*measurable*]: $\bigwedge n.$ *integrable M* $(u\ n)$
and $(\lambda n.\ (\int x.\ norm(u\ n\ x)\ \partial M)) \longrightarrow 0$
shows $\exists\, r$::*nat*⇒*nat*. *strict_mono r* ∧ (*AE x in M*. $(\lambda n.\ u\ (r\ n)\ x) \longrightarrow 0)$

7.9.1 Restricted measure spaces

7.9.2 Measure spaces with an associated density

7.9.3 Distributions

7.9.4 Lebesgue integration on *count_space*

7.9.5 Point measure

proposition *integrable_point_measure_finite*:
fixes g :: $'a$ ⇒ $'b$::{*banach*, *second_countable_topology*} **and** f :: $'a$ ⇒ *real*
assumes *finite A*
shows *integrable* (*point_measure A f*) g

7.9.6 Lebesgue integration on *null_measure*

7.9.7 Legacy lemmas for the real-valued Lebesgue integral

theorem *real_lebesgue_integral_def*:

assumes $f[measurable]$: *integrable* $M\ f$
shows $integral^L\ M\ f = enn2real\ (\int^+ x.\ f\ x\ \partial M) - enn2real\ (\int^+ x.\ ennreal\ (-\ f\ x)\ \partial M)$

theorem *real_integrable_def*:
$integrable\ M\ f \longleftrightarrow f \in borel\_measurable\ M \wedge$
$(\int^+ x.\ ennreal\ (f\ x)\ \partial M) \neq \infty \wedge (\int^+ x.\ ennreal\ (-\ f\ x)\ \partial M) \neq \infty$

7.9.8 Product measure

proposition (**in** *sigma_finite_measure*) *borel_measurable_lebesgue_integral*[*measurable* (*raw*)]:
fixes $f :: \_ \Rightarrow \_ \Rightarrow \_::\{banach, second\_countable\_topology\}$
assumes $f[measurable]$: $case\_prod\ f \in borel\_measurable\ (N \bigotimes_M M)$
shows $(\lambda x.\ \int y.\ f\ x\ y\ \partial M) \in borel\_measurable\ N$

theorem (**in** *pair_sigma_finite*) *Fubini_integrable*:
fixes $f :: \_ \Rightarrow \_::\{banach, second\_countable\_topology\}$
assumes $f[measurable]$: $f \in borel\_measurable\ (M1 \bigotimes_M M2)$
and *integ1*: $integrable\ M1\ (\lambda x.\ \int y.\ norm\ (f\ (x, y))\ \partial M2)$
and *integ2*: $AE\ x\ in\ M1.\ integrable\ M2\ (\lambda y.\ f\ (x, y))$
shows $integrable\ (M1 \bigotimes_M M2)\ f$

proposition (**in** *pair_sigma_finite*) *integral_fst'*:
fixes $f :: \_ \Rightarrow \_::\{banach, second\_countable\_topology\}$
assumes f: $integrable\ (M1 \bigotimes_M M2)\ f$
shows $(\int x.\ (\int y.\ f\ (x, y)\ \partial M2)\ \partial M1) = integral^L\ (M1 \bigotimes_M M2)\ f$

proposition (**in** *pair_sigma_finite*) *Fubini_integral*:
fixes $f :: \_ \Rightarrow \_ \Rightarrow \_ :: \{banach, second\_countable\_topology\}$
assumes f: $integrable\ (M1 \bigotimes_M M2)\ (case\_prod\ f)$
shows $(\int y.\ (\int x.\ f\ x\ y\ \partial M1)\ \partial M2) = (\int x.\ (\int y.\ f\ x\ y\ \partial M2)\ \partial M1)$

end

7.10 Complete Measures

theory *Complete_Measure*
imports *Bochner_Integration*
begin

locale *complete_measure* =
fixes $M :: 'a\ measure$
assumes *complete*: $\bigwedge A\ B.\ B \subseteq A \Longrightarrow A \in null\_sets\ M \Longrightarrow B \in sets\ M$

definition
$split\_completion\ M\ A\ p = (if\ A \in sets\ M\ then\ p = (A, \{\})\ else$

$\exists N'.\ A = fst\ p \cup snd\ p \wedge fst\ p \cap snd\ p = \{\} \wedge fst\ p \in sets\ M \wedge snd\ p \subseteq N' \wedge N' \in null\_sets\ M)$

definition
main_part M A = fst (Eps (split_completion M A))

definition
null_part M A = snd (Eps (split_completion M A))

definition *completion* :: *'a measure* ⇒ *'a measure* **where**
$completion\ M = measure\_of\ (space\ M)\ \{\ S \cup N\ |S\ N\ N'.\ S \in sets\ M \wedge N' \in null\_sets\ M \wedge N \subseteq N'\ \}$
$(emeasure\ M \circ main\_part\ M)$

lemma *sets_completion*:
$sets\ (completion\ M) = \{\ S \cup N\ |S\ N\ N'.\ S \in sets\ M \wedge N' \in null\_sets\ M \wedge N \subseteq N'\ \}$

lemma *measurable_completion*: $f \in M \rightarrow_M N \Longrightarrow f \in completion\ M \rightarrow_M N$

lemma *split_completion*:
assumes $A \in sets\ (completion\ M)$
shows *split_completion M A (main_part M A, null_part M A)*

lemma *emeasure_completion*[*simp*]:
assumes *S*: $S \in sets\ (completion\ M)$
shows *emeasure (completion M) S = emeasure M (main_part M S)*

lemma *completion_ex_borel_measurable*:
fixes *g* :: *'a* ⇒ *ennreal*
assumes *g*: $g \in borel\_measurable\ (completion\ M)$
shows $\exists g' \in borel\_measurable\ M.\ (AE\ x\ in\ M.\ g\ x = g'\ x)$

locale *semifinite_measure* =
fixes *M* :: *'a measure*
assumes *semifinite*:
$\bigwedge A.\ A \in sets\ M \Longrightarrow emeasure\ M\ A = \infty \Longrightarrow \exists B \in sets\ M.\ B \subseteq A \wedge emeasure\ M\ B < \infty$

locale *locally_determined_measure* = *semifinite_measure* +
assumes *locally_determined*:
$\bigwedge A.\ A \subseteq space\ M \Longrightarrow (\bigwedge B.\ B \in sets\ M \Longrightarrow emeasure\ M\ B < \infty \Longrightarrow A \cap B \in sets\ M) \Longrightarrow A \in sets\ M$

locale *cld_measure* =
complete_measure M + *locally_determined_measure M* **for** *M* :: *'a measure*

definition *outer_measure_of* :: *'a measure* ⇒ *'a set* ⇒ *ennreal*
where $outer\_measure\_of\ M\ A = (INF\ B \in \{B \in sets\ M.\ A \subseteq B\}.\ emeasure\ M$

B)

definition *measurable_envelope* :: *'a measure* ⇒ *'a set* ⇒ *'a set* ⇒ *bool*
where *measurable_envelope M A E* ⟷
(*A* ⊆ *E* ∧ *E* ∈ *sets M* ∧ (∀ *F*∈*sets M*. *emeasure M* (*F* ∩ *E*) = *outer_measure_of M* (*F* ∩ *A*)))

lemma *measurable_envelope_eq2*:
assumes *A* ⊆ *E E* ∈ *sets M emeasure M E* < ∞
shows *measurable_envelope M A E* ⟷ (*emeasure M E* = *outer_measure_of M A*)

proposition (**in** *complete_measure*) *fmeasurable_inner_outer*:
S ∈ *fmeasurable M* ⟷
(∀ *e*>*0*. ∃ *T*∈*fmeasurable M*. ∃ *U*∈*fmeasurable M*. *T* ⊆ *S* ∧ *S* ⊆ *U* ∧ |*measure M T* − *measure M U*| < *e*)
(**is** _ ⟷ *?approx*)

end

7.11 Regularity of Measures

theory *Regularity*
imports *Measure_Space Borel_Space*
begin

theorem
fixes *M*::*'a*::{*second_countable_topology*, *complete_space*} *measure*
assumes *sb*: *sets M* = *sets borel*
assumes *emeasure M* (*space M*) ≠ ∞
assumes *B* ∈ *sets borel*
shows *inner_regular*: *emeasure M B* =
(*SUP K* ∈ {*K*. *K* ⊆ *B* ∧ *compact K*}. *emeasure M K*) (**is** *?inner B*)
and *outer_regular*: *emeasure M B* =
(*INF U* ∈ {*U*. *B* ⊆ *U* ∧ *open U*}. *emeasure M U*) (**is** *?outer B*)

end

7.12 Lebesgue Measure

theory *Lebesgue_Measure*
imports
Finite_Product_Measure
Caratheodory
Complete_Measure
Summation_Tests
Regularity
begin

7.12.1 Measures defined by monotonous functions

definition *interval_measure* :: (*real* ⇒ *real*) ⇒ *real measure* **where**
interval_measure F =
extend_measure UNIV {(*a*, *b*). *a* ≤ *b*} (λ(*a*, *b*). {*a*<..*b*}) (λ(*a*, *b*). *ennreal* (*F b* − *F a*))

lemma *emeasure_interval_measure_Ioc*:
assumes *a* ≤ *b*
assumes *mono_F*: ⋀*x y*. *x* ≤ *y* ⟹ *F x* ≤ *F y*
assumes *right_cont_F* : ⋀*a*. *continuous* (*at_right a*) *F*
shows *emeasure* (*interval_measure F*) {*a*<..*b*} = *F b* − *F a*

lemma *sets_interval_measure* [*simp*, *measurable_cong*]:
sets (*interval_measure F*) = *sets borel*

lemma *sigma_finite_interval_measure*:
assumes *mono_F*: ⋀*x y*. *x* ≤ *y* ⟹ *F x* ≤ *F y*
assumes *right_cont_F* : ⋀*a*. *continuous* (*at_right a*) *F*
shows *sigma_finite_measure* (*interval_measure F*)

7.12.2 Lebesgue-Borel measure

definition *lborel* :: (′*a* :: *euclidean_space*) *measure* **where**
lborel = *distr* (Π_M *b*∈*Basis*. *interval_measure* (λ*x*. *x*)) *borel* (λ*f*. ∑*b*∈*Basis*. *f b* $*_R$ *b*)

abbreviation *lebesgue* :: ′*a*::*euclidean_space measure*
where *lebesgue* ≡ *completion lborel*

abbreviation *lebesgue_on* :: ′*a set* ⇒ ′*a*::*euclidean_space measure*
where *lebesgue_on* Ω ≡ *restrict_space* (*completion lborel*) Ω

7.12.3 Borel measurability

lemma *emeasure_lborel_cbox*[*simp*]:
assumes [*simp*]: ⋀*b*. *b* ∈ *Basis* ⟹ *l* · *b* ≤ *u* · *b*
shows *emeasure lborel* (*cbox l u*) = (∏*b*∈*Basis*. (*u* − *l*) · *b*)

7.12.4 Affine transformation on the Lebesgue-Borel

lemma *lborel_eqI*:
fixes *M* :: ′*a*::*euclidean_space measure*

assumes *emeasure_eq*: $\bigwedge l\ u.\ (\bigwedge b.\ b \in Basis \Longrightarrow l \cdot b \leq u \cdot b) \Longrightarrow emeasure\ M\ (box\ l\ u) = (\prod b{\in}Basis.\ (u - l) \cdot b)$
assumes *sets_eq*: $sets\ M = sets\ borel$
shows $lborel = M$

lemma *lborel_affine_euclidean*:
fixes $c :: 'a::euclidean\_space \Rightarrow real$ **and** t
defines $T\ x \equiv t + (\sum j{\in}Basis.\ (c\ j * (x \cdot j)) *_R j)$
assumes c: $\bigwedge j.\ j \in Basis \Longrightarrow c\ j \neq 0$
shows $lborel = density\ (distr\ lborel\ borel\ T)\ (\lambda\_.\ (\prod j{\in}Basis.\ |c\ j|))$ (**is** $\_ = ?D$)

lemma *lborel_integral_real_affine*:
fixes $f :: real \Rightarrow 'a :: \{banach,\ second\_countable\_topology\}$ **and** $c :: real$
assumes c: $c \neq 0$ **shows** $(\int x.\ f\ x\ \partial\ lborel) = |c| *_R (\int x.\ f\ (t + c * x)\ \partial lborel)$

corollary *lebesgue_real_affine*:
$c \neq 0 \Longrightarrow lebesgue = density\ (distr\ lebesgue\ lebesgue\ (\lambda x.\ t + c * x))\ (\lambda\_.\ ennreal\ (abs\ c))$

lemma *lborel_prod*:
$lborel \bigotimes_M lborel = (lborel :: ('a::euclidean\_space \times 'b::euclidean\_space)\ measure)$

7.12.5 Lebesgue measurable sets

abbreviation $lmeasurable :: 'a::euclidean\_space\ set\ set$
where
$lmeasurable \equiv fmeasurable\ lebesgue$

lemma *lmeasurable_iff_integrable*:
$S \in lmeasurable \longleftrightarrow integrable\ lebesgue\ (indicator\ S :: 'a::euclidean\_space \Rightarrow real)$

7.12.6 A nice lemma for negligibility proofs

proposition *starlike_negligible_bounded_gmeasurable*:
fixes $S :: 'a :: euclidean\_space\ set$
assumes S: $S \in sets\ lebesgue$ **and** $bounded\ S$
and *eq1*: $\bigwedge c\ x.\ [\![(c *_R x) \in S;\ 0 \leq c;\ x \in S]\!] \Longrightarrow c = 1$
shows $S \in null\_sets\ lebesgue$

corollary *starlike_negligible_compact*:
$compact\ S \Longrightarrow (\bigwedge c\ x.\ [\![(c *_R x) \in S;\ 0 \leq c;\ x \in S]\!] \Longrightarrow c = 1) \Longrightarrow S \in null\_sets\ lebesgue$

proposition *outer_regular_lborel_le*:

assumes *B*[*measurable*]: $B \in$ *sets borel* **and** $0 < (e::real)$
obtains *U* **where** *open U* $B \subseteq U$ **and** *emeasure lborel* $(U - B) \leq e$

lemma *outer_regular_lborel*:
assumes *B*: $B \in$ *sets borel* **and** $0 < (e::real)$
obtains *U* **where** *open U* $B \subseteq U$ *emeasure lborel* $(U - B) < e$

7.12.7 *F_sigma* and *G_delta* sets.

inductive *fsigma* :: *'a::topological_space set* ⇒ *bool* **where**
$(\bigwedge n::nat.\ closed\ (F\ n)) \Longrightarrow fsigma\ (\bigcup (F\ `\ UNIV))$

inductive *gdelta* :: *'a::topological_space set* ⇒ *bool* **where**
$(\bigwedge n::nat.\ open\ (F\ n)) \Longrightarrow gdelta\ (\bigcap (F\ `\ UNIV))$

end

7.13 Tagged Divisions for Henstock-Kurzweil Integration

theory *Tagged_Division*
imports *Topology_Euclidean_Space*
begin

7.13.1 Some useful lemmas about intervals

7.13.2 Bounds on intervals where they exist

definition *interval_upperbound* :: (*'a::euclidean_space*) *set* ⇒ *'a*
where *interval_upperbound s* = $(\sum i \in Basis.\ (SUP\ x \in s.\ x \cdot i) *_R\ i)$

definition *interval_lowerbound* :: (*'a::euclidean_space*) *set* ⇒ *'a*
where *interval_lowerbound s* = $(\sum i \in Basis.\ (INF\ x \in s.\ x \cdot i) *_R\ i)$

7.13.3 The notion of a gauge — simply an open set containing the point

definition *gauge* $\gamma \longleftrightarrow (\forall x.\ x \in \gamma\ x \wedge open\ (\gamma\ x))$

7.13.4 Attempt a systematic general set of "offset" results for components

7.13.5 Divisions

definition *division_of* (**infixl** ‹*division'_of*› *40*)

where
s division_of i $\longleftrightarrow$
finite s $\wedge$
$(\forall K \in s.\ K \subseteq i \wedge K \neq \{\} \wedge (\exists a\ b.\ K = cbox\ a\ b)) \wedge$
$(\forall K1 \in s.\ \forall K2 \in s.\ K1 \neq K2 \longrightarrow interior(K1) \cap interior(K2) = \{\}) \wedge$
$(\bigcup s = i)$

proposition *partial_division_extend_interval*:
assumes *p division_of* $(\bigcup p)$ $(\bigcup p) \subseteq$ *cbox a b*
obtains *q* **where** $p \subseteq q$ *q division_of cbox a* (*b*::′*a*::*euclidean_space*)

proposition *division_union_intervals_exists*:
assumes *cbox a b* $\neq \{\}$
obtains *p* **where** (*insert* (*cbox a b*) *p*) *division_of* (*cbox a b* $\cup$ *cbox c d*)

7.13.6 Tagged (partial) divisions

definition *tagged_partial_division_of* (**infixr** ‹*tagged′_partial′_division′_of*› *40*)
where *s tagged_partial_division_of i* $\longleftrightarrow$
finite s $\wedge$
$(\forall x\ K.\ (x, K) \in s \longrightarrow x \in K \wedge K \subseteq i \wedge (\exists a\ b.\ K = cbox\ a\ b)) \wedge$
$(\forall x1\ K1\ x2\ K2.\ (x1, K1) \in s \wedge (x2, K2) \in s \wedge (x1, K1) \neq (x2, K2) \longrightarrow$
$interior\ K1 \cap interior\ K2 = \{\})$

definition *tagged_division_of* (**infixr** ‹*tagged′_division′_of*› *40*)
where *s tagged_division_of i* $\longleftrightarrow$ *s tagged_partial_division_of i* $\wedge$ $(\bigcup\{K.\ \exists x.\ (x,K) \in s\} = i)$

7.13.7 Functions closed on boxes: morphisms from boxes to monoids

Using additivity of lifted function to encode definedness. **definition** *lift_option* :: (′*a* ⇒ ′*b* ⇒ ′*c*) ⇒ ′*a option* ⇒ ′*b option* ⇒ ′*c option*
where
lift_option f a′ b′ = *Option.bind a′* (λ*a*. *Option.bind b′* (λ*b*. *Some* (*f a b*)))

lemma *comm_monoid_lift_option*:
assumes *comm_monoid f z*
shows *comm_monoid* (*lift_option f*) (*Some z*)

Misc

Division points **definition** *division_points* (*k*::(′*a*::*euclidean_space*) *set*) *d* =

{(*j*,*x*). *j* ∈ *Basis* ∧ (*interval_lowerbound k*)•*j* < *x* ∧ *x* < (*interval_upperbound k*)•*j* ∧
(∃ *i*∈*d*. (*interval_lowerbound i*)•*j* = *x* ∨ (*interval_upperbound i*)•*j* = *x*)}

Operative
proposition *tagged_division*:
assumes *d tagged_division_of* (*cbox a b*)
shows *F* (λ(_, *l*). *g l*) *d* = *g* (*cbox a b*)

7.13.8 Special case of additivity we need for the FTC

7.13.9 Fine-ness of a partition w.r.t. a gauge

definition *fine* (**infixr** ‹*fine*› *46*)
where *d fine s* ⟷ (∀ (*x*,*k*) ∈ *s*. *k* ⊆ *d x*)

7.13.10 Some basic combining lemmas

7.13.11 General bisection principle for intervals; might be useful elsewhere

7.13.12 Cousin's lemma

7.13.13 A technical lemma about "refinement" of division

Covering lemma

proposition *covering_lemma*:
assumes *S* ⊆ *cbox a b box a b* ≠ {} *gauge g*
obtains 𝒟 **where**
countable 𝒟 ⋃𝒟 ⊆ *cbox a b*
⋀*K*. *K* ∈ 𝒟 ⟹ *interior K* ≠ {} ∧ (∃ *c d*. *K* = *cbox c d*)
pairwise (λ*A B*. *interior A* ∩ *interior B* = {}) 𝒟
⋀*K*. *K* ∈ 𝒟 ⟹ ∃ *x* ∈ *S* ∩ *K*. *K* ⊆ *g x*
⋀*u v*. *cbox u v* ∈ 𝒟 ⟹ ∃ *n*. ∀ *i* ∈ *Basis*. *v* · *i* − *u* · *i* = (*b* · *i* − *a* · *i*) / *2^n*
S ⊆ ⋃𝒟

7.13.14 Division filter

definition *division_filter* :: ′*a*::*euclidean_space set* ⇒ (′*a* × ′*a set*) *set filter*
where *division_filter s* = (*INF g*∈{*g*. *gauge g*}. *principal* {*p*. *p tagged_division_of s* ∧ *g fine p*})

proposition *eventually_division_filter*:
(∀$_F$ *p in division_filter s*. *P p*) ⟷
(∃ *g*. *gauge g* ∧ (∀ *p*. *p tagged_division_of s* ∧ *g fine p* ⟶ *P p*))

end

7.14 Henstock-Kurzweil Gauge Integration in Many Dimensions

theory *Henstock_Kurzweil_Integration*
imports
Lebesgue_Measure Tagged_Division
begin

7.14.1 Content (length, area, volume...) of an interval

7.14.2 Gauge integral

7.14.3 Basic theorems about integrals

corollary *integral_mult_left* [*simp*]:
fixes *c*:: *'a*::{*real_normed_algebra*,*division_ring*}
shows *integral S* (*λx. f x* ∗ *c*) = *integral S f* ∗ *c*

corollary *integral_mult_right* [*simp*]:
fixes *c*:: *'a*::{*real_normed_field*}
shows *integral S* (*λx. c* ∗ *f x*) = *c* ∗ *integral S f*

corollary *integral_divide* [*simp*]:
fixes *z* :: *'a*::*real_normed_field*
shows *integral S* (*λx. f x* / *z*) = *integral S* (*λx. f x*) / *z*

7.14.4 Cauchy-type criterion for integrability

proposition *integrable_Cauchy*:
fixes *f* :: *'n*::*euclidean_space* ⇒ *'a*::{*real_normed_vector*,*complete_space*}
shows *f integrable_on cbox a b* ⟷
(∀ *e*>*0*. ∃γ. *gauge* γ ∧
(∀𝒟*1* 𝒟*2*. 𝒟*1 tagged_division_of* (*cbox a b*) ∧ γ *fine* 𝒟*1* ∧
𝒟*2 tagged_division_of* (*cbox a b*) ∧ γ *fine* 𝒟*2* ⟶
norm ((∑(*x*,*K*)∈𝒟*1*. *content K* $*_R$ *f x*) − (∑(*x*,*K*)∈𝒟*2*. *content K* $*_R$
f x)) < *e*))
(**is** *?l* = (∀ *e*>*0*. ∃γ. *?P e* γ))

7.14.5 Additivity of integral on abutting intervals

proposition *has_integral_split*:
fixes *f* :: *'a*::*euclidean_space* ⇒ *'b*::*real_normed_vector*

assumes *fi*: (*f has_integral i*) (*cbox a b* ∩ {*x*. *x*•*k* ≤ *c*})
and *fj*: (*f has_integral j*) (*cbox a b* ∩ {*x*. *x*•*k* ≥ *c*})
and *k*: *k* ∈ *Basis*
shows (*f has_integral* (*i* + *j*)) (*cbox a b*)

7.14.6 A sort of converse, integrability on subintervals

7.14.7 Bounds on the norm of Riemann sums and the integral itself

corollary *integrable_bound*:
fixes *f* :: *'a::euclidean_space* ⇒ *'b::real_normed_vector*
assumes *0* ≤ *B*
and *f integrable_on* (*cbox a b*)
and ⋀*x*. *x*∈*cbox a b* ⟹ *norm* (*f x*) ≤ *B*
shows *norm* (*integral* (*cbox a b*) *f*) ≤ *B* ∗ *content* (*cbox a b*)

7.14.8 Similar theorems about relationship among components

7.14.9 Uniform limit of integrable functions is integrable

7.14.10 Negligible sets

proposition *negligible_standard_hyperplane*[*intro*]:
fixes *k* :: *'a::euclidean_space*
assumes *k*: *k* ∈ *Basis*
shows *negligible* {*x*. *x*•*k* = *c*}

corollary *negligible_standard_hyperplane_cart*:
fixes *k* :: *'a::finite*
shows *negligible* {*x*. *x*$*k* = (*0::real*)}

proposition *has_integral_negligible*:
fixes *f* :: *'b::euclidean_space* ⇒ *'a::real_normed_vector*
assumes *negs*: *negligible S*
and ⋀*x*. *x* ∈ (*T* − *S*) ⟹ *f x* = *0*
shows (*f has_integral 0*) *T*

7.14.11 Some other trivialities about negligible sets

7.14.12 Finite case of the spike theorem is quite commonly needed

corollary *has_integral_bound_real*:
  fixes f :: *real* ⇒ *'b::real_normed_vector*
  assumes $0 \leq B$ *finite* S
    and (f *has_integral* i) $\{a..b\}$
    and $\bigwedge x.\ x \in \{a..b\} - S \Longrightarrow$ *norm* $(f\ x) \leq B$
  shows *norm* $i \leq B *$ *content* $\{a..b\}$

7.14.13 In particular, the boundary of an interval is negligible

7.14.14 Integrability of continuous functions

7.14.15 Specialization of additivity to one dimension

7.14.16 A useful lemma allowing us to factor out the content size

7.14.17 Fundamental theorem of calculus

theorem *fundamental_theorem_of_calculus*:
  fixes f :: *real* ⇒ *'a::banach*
  assumes $a \leq b$
    and *vecd*: $\bigwedge x.\ x \in \{a..b\} \Longrightarrow$ (f *has_vector_derivative* f' x) (*at* x *within* $\{a..b\}$)
  shows (f' *has_integral* $(f\ b - f\ a)$) $\{a..b\}$

7.14.18 Taylor series expansion

7.14.19 Only need trivial subintervals if the interval itself is trivial

proposition *division_of_nontrivial*:
  fixes $\mathcal{D}$ :: *'a::euclidean_space set set*
  assumes *sdiv*: $\mathcal{D}$ *division_of* (*cbox* a b)
    and *cont0*: *content* (*cbox* a b) $\neq 0$
  shows $\{k.\ k \in \mathcal{D} \wedge$ *content* $k \neq 0\}$ *division_of* (*cbox* a b)

7.14.20 Integrability on subintervals

7.14.21 Combining adjacent intervals in 1 dimension

7.14.22 Reduce integrability to "local" integrability

7.14.23 Second FTC or existence of antiderivative

7.14.24 Combined fundamental theorem of calculus

7.14.25 General "twiddling" for interval-to-interval function image

7.14.26 Special case of a basic affine transformation

7.14.27 Special case of stretching coordinate axes separately

7.14.28 even more special cases

7.14.29 Stronger form of FCT; quite a tedious proof

theorem *fundamental_theorem_of_calculus_interior*:
  fixes f :: *real* ⇒ *'a::real_normed_vector*
  assumes $a \leq b$
    and *contf*: *continuous_on* $\{a..b\}$ f
    and *derf*: $\bigwedge x.\ x \in \{a <..< b\} \Longrightarrow$ (f *has_vector_derivative* f' x) (*at* x)
  shows (f' *has_integral* ($f\ b - f\ a$)) $\{a..b\}$

7.14.30 Stronger form with finite number of exceptional points

corollary *fundamental_theorem_of_calculus_strong*:
  fixes f :: *real* ⇒ *'a::banach*
  assumes *finite* S
    and $a \leq b$
    and *vec*: $\bigwedge x.\ x \in \{a..b\} - S \Longrightarrow$ (f *has_vector_derivative* $f'(x)$) (*at* x)
    and *continuous_on* $\{a..b\}$ f
  shows (f' *has_integral* ($f\ b - f\ a$)) $\{a..b\}$

proposition *indefinite_integral_continuous_left*:
  fixes f:: *real* ⇒ *'a::banach*
  assumes *intf*: f *integrable_on* $\{a..b\}$ **and** $a < c$ $c \leq b$ $e > 0$
  obtains d **where** $d > 0$
    and $\forall t.\ c - d < t \wedge t \leq c \longrightarrow$ *norm* (*integral* $\{a..c\}$ f − *integral* $\{a..t\}$ f) $<$
e

theorem *integral_has_vector_derivative'*:

fixes f :: *real* ⇒ *'b::banach*
assumes *continuous_on* {*a..b*} *f*
and $x \in \{a..b\}$
shows ((λ*u*. *integral* {*u..b*} *f*) *has_vector_derivative* − *f x*) (*at x within* {*a..b*})

7.14.31 This doesn't directly involve integration, but that gives an easy proof

7.14.32 Generalize a bit to any convex set

7.14.33 Integrating characteristic function of an interval

corollary *has_integral_restrict_UNIV*:
fixes f :: *'n::euclidean_space* ⇒ *'a::banach*
shows ((λ*x*. *if* $x \in s$ *then f x else 0*) *has_integral i*) *UNIV* ⟷ (*f has_integral i*) *s*

7.14.34 Integrals on set differences

corollary *integral_spike_set*:
fixes f :: *'n::euclidean_space* ⇒ *'a::banach*
assumes *negligible* {$x \in S - T$. *f x* ≠ *0*} *negligible* {$x \in T - S$. *f x* ≠ *0*}
shows *integral S f* = *integral T f*

7.14.35 More lemmas that are useful later

7.14.36 Continuity of the integral (for a 1-dimensional interval)

7.14.37 A straddling criterion for integrability

7.14.38 Adding integrals over several sets

7.14.39 Also tagged divisions

7.14.40 Henstock's lemma

7.14.41 Monotone convergence (bounded interval first)

7.14.42 differentiation under the integral sign

7.14.43 Exchange uniform limit and integral

7.14.44 Integration by parts

7.14.45 Integration by substitution

7.14.46 Compute a double integral using iterated integrals and switching the order of integration

theorem *integral_swap_continuous*:
  fixes *f* :: [*'a*::*euclidean_space*, *'b*::*euclidean_space*] ⇒ *'c*::*banach*
  assumes *continuous_on* (*cbox* (*a*,*c*) (*b*,*d*)) (λ(*x*,*y*). *f x y*)
    shows *integral* (*cbox a b*) (λ*x*. *integral* (*cbox c d*) (*f x*)) =
           integral (*cbox c d*) (λ*y*. *integral* (*cbox a b*) (λ*x*. *f x y*))

7.14.47 Definite integrals for exponential and power function

end

Chapter 8

Kronecker's Theorem with Applications

theory *Kronecker_Approximation_Theorem*

imports *Complex_Transcendental Henstock_Kurzweil_Integration*
HOL−Real_Asymp.Real_Asymp

begin

8.1 Dirichlet's Approximation Theorem

theorem *Dirichlet_approx_simult*:
fixes ϑ :: *nat* ⇒ *real* **and** *N n* :: *nat*
assumes $N > 0$
obtains *q p* **where** $0<q$ $q \leq int\ (N\hat{\ }n)$
and $\bigwedge i.\ i<n \Longrightarrow |of\_int\ q * \vartheta\ i - of\_int(p\ i)| < 1/N$
corollary *Dirichlet_approx*:
fixes ϑ:: *real* **and** *N*:: *nat*
assumes $N > 0$
obtains *h k* **where** $0 < k$ $k \leq int\ N$ $|of\_int\ k * \vartheta - of\_int\ h| < 1/N$
corollary *Dirichlet_approx_coprime*:
fixes ϑ:: *real* **and** *N*:: *nat*
assumes $N > 0$
obtains *h k* **where** *coprime h k* $0 < k$ $k \leq int\ N$ $|of\_int\ k * \vartheta - of\_int\ h| < 1/N$
theorem *infinite_approx_set*:
assumes *infinite* (*approx_set* ϑ)
shows $\exists\, h\ k.\ (h,k) \in approx\_set\ \vartheta \wedge k > K$
theorem *rational_iff_finite_approx_set*:
shows $\vartheta \in \mathbb{Q} \longleftrightarrow finite\ (approx\_set\ \vartheta)$

8.2 Kronecker's Approximation Theorem: the One-dimensional Case

theorem *Kronecker_approx_1_explicit*:
  fixes ϑ :: *real*
  assumes $\vartheta \notin \mathbb{Q}$ **and** α: $0 \le \alpha$ $\alpha \le 1$ **and** $\varepsilon > 0$
  obtains k **where** $k>0$ $|frac(real\ k * \vartheta) - \alpha| < \varepsilon$
corollary *Kronecker_approx_1*:
  fixes ϑ :: *real*
  assumes $\vartheta \notin \mathbb{Q}$
  shows *closure* (*range* ($\lambda n.$ *frac* (*real* $n * \vartheta$))) = {*0..1*} (**is** *?C* = _)
corollary *sequence_of_fractional_parts_is_dense*:
  fixes ϑ :: *real*
  assumes $\vartheta \notin \mathbb{Q}$ $\varepsilon > 0$
  obtains h k **where** $k > 0$ $|of\_int\ k * \vartheta - of\_int\ h - \alpha| < \varepsilon$

8.3 Extension of Kronecker's Theorem to Simultaneous Approximation

8.3.1 Towards Lemma 1

8.3.2 Towards Lemma 2

8.3.3 Towards lemma 3

8.3.4 And finally Kroncker's theorem itself

theorem *Kronecker_thm_1*:
  fixes α ϑ:: *nat* ⇒ *real* **and** n:: *nat*
  assumes *indp*: *module.independent* ($\lambda r.$ (∗) (*real_of_int* r)) (ϑ ‘ {..<n})
    and *inj*ϑ: *inj_on* ϑ {..<n} **and** $\varepsilon > 0$
  obtains t h **where** $\bigwedge i.\ i < n \Longrightarrow |t * \vartheta\ i - of\_int\ (h\ i) - \alpha\ i| < \varepsilon$
corollary *Kronecker_thm_2*:
  fixes α ϑ :: *nat* ⇒ *real* **and** n :: *nat*
  assumes *indp*: *module.independent* ($\lambda r\ x.$ *of_int* $r * x$) (ϑ ‘ {..n})
    and *inj*ϑ: *inj_on* ϑ {..n} **and** [*simp*]: $\vartheta\ n = 1$ **and** $\varepsilon > 0$
  obtains k m **where** $\bigwedge i.\ i < n \Longrightarrow |of\_int\ k * \vartheta\ i - of\_int\ (m\ i) - \alpha\ i| < \varepsilon$

end

8.4 Bernstein-Weierstrass and Stone-Weierstrass

theory *Weierstrass_Theorems*
imports *Uniform_Limit Path_Connected Derivative*
begin

8.4.1 Bernstein polynomials

definition *Bernstein* :: [*nat*,*nat*,*real*] ⇒ *real* **where**
Bernstein n k x ≡ *of_nat* (*n choose k*) ∗ *x* ^ *k* ∗ (*1* − *x*) ^ (*n* − *k*)

8.4.2 Explicit Bernstein version of the 1D Weierstrass approximation theorem

theorem *Bernstein_Weierstrass*:
fixes *f* :: *real* ⇒ *real*
assumes *contf*: *continuous_on* {*0*..*1*} *f* **and** *e*: *0* < *e*
shows ∃ *N*. ∀ *n x*. *N* ≤ *n* ∧ *x* ∈ {*0*..*1*}
−→ |*f x* − (∑ *k*≤*n*. *f*(*k*/*n*) ∗ *Bernstein n k x*)| < *e*

8.4.3 General Stone-Weierstrass theorem

definition *normf* :: (′*a*::*t2_space* ⇒ *real*) ⇒ *real*
where *normf f* ≡ *SUP x*∈*S*. |*f x*|
proposition (**in** *function_ring_on*) *Stone_Weierstrass_basic*:
assumes *f*: *continuous_on S f* **and** *e*: *e* > *0*
shows ∃ *g* ∈ *R*. ∀ *x*∈*S*. |*f x* − *g x*| < *e*

theorem (**in** *function_ring_on*) *Stone_Weierstrass*:
assumes *f*: *continuous_on S f*
shows ∃ *F*∈*UNIV* → *R*. *LIM n sequentially*. *F n* :> *uniformly_on S f*
corollary *Stone_Weierstrass_HOL*:
fixes *R* :: (′*a*::*t2_space* ⇒ *real*) *set* **and** *S* :: ′*a set*
assumes *compact S* ⋀*c*. *P*(λ*x*. *c*::*real*)
⋀*f*. *P f* =⇒ *continuous_on S f*
⋀*f g*. *P*(*f*) ∧ *P*(*g*) =⇒ *P*(λ*x*. *f x* + *g x*) ⋀*f g*. *P*(*f*) ∧ *P*(*g*) =⇒ *P*(λ*x*. *f x* ∗ *g x*)
⋀*x y*. *x* ∈ *S* ∧ *y* ∈ *S* ∧ *x* ≠ *y* =⇒ ∃ *f*. *P*(*f*) ∧ *f x* ≠ *f y*
continuous_on S f
0 < *e*
shows ∃ *g*. *P*(*g*) ∧ (∀ *x* ∈ *S*. |*f x* − *g x*| < *e*)

8.4.4 Polynomial functions

definition *polynomial_function* :: (*'a*::*real_normed_vector* ⇒ *'b*::*real_normed_vector*) ⇒ *bool*
where
polynomial_function p ≡ (∀ *f*. *bounded_linear f* ⟶ *real_polynomial_function* (*f o p*))

8.4.5 Stone-Weierstrass theorem for polynomial functions

theorem *Stone_Weierstrass_polynomial_function*:
fixes *f* :: *'a*::*euclidean_space* ⇒ *'b*::*euclidean_space*
assumes *S*: *compact S*
and *f*: *continuous_on S f*
and *e*: *0* < *e*
shows ∃ *g*. *polynomial_function g* ∧ (∀ *x* ∈ *S*. *norm*(*f x* − *g x*) < *e*)

proposition *Stone_Weierstrass_uniform_limit*:
fixes *f* :: *'a*::*euclidean_space* ⇒ *'b*::*euclidean_space*
assumes *S*: *compact S*
and *f*: *continuous_on S f*
obtains *g* **where** *uniform_limit S g f sequentially* ⋀*n*. *polynomial_function* (*g n*)

8.4.6 Polynomial functions as paths

proposition *connected_open_polynomial_connected*:
fixes *S* :: *'a*::*euclidean_space set*
assumes *S*: *open S connected S*
and *x* ∈ *S y* ∈ *S*
shows ∃ *g*. *polynomial_function g* ∧ *path_image g* ⊆ *S* ∧ *pathstart g* = *x* ∧ *pathfinish g* = *y*

theorem *Stone_Weierstrass_polynomial_function_subspace*:
fixes *f* :: *'a*::*euclidean_space* ⇒ *'b*::*euclidean_space*
assumes *compact S*
and *contf*: *continuous_on S f*
and *0* < *e*
and *subspace T f* ' *S* ⊆ *T*
obtains *g* **where** *polynomial_function g g* ' *S* ⊆ *T*
⋀*x*. *x* ∈ *S* ⟹ *norm*(*f x* − *g x*) < *e*

end

8.5 Radon-Nikodým Derivative

theory *Radon_Nikodym*
imports *Bochner_Integration*
begin

definition *diff_measure* :: *'a measure* ⇒ *'a measure* ⇒ *'a measure*
where
diff_measure M N = *measure_of* (*space M*) (*sets M*) (λ*A*. *emeasure M A* − *emeasure N A*)
proposition (**in** *sigma_finite_measure*) *obtain_positive_integrable_function*:
obtains *f*::*'a* ⇒ *real* **where**
f ∈ *borel_measurable M*
⋀*x*. *f x* > *0*
⋀*x*. *f x* ≤ *1*
integrable M f

8.5.1 Absolutely continuous

definition *absolutely_continuous* :: *'a measure* ⇒ *'a measure* ⇒ *bool* **where**
absolutely_continuous M N ⟷ *null_sets M* ⊆ *null_sets N*

8.5.2 Existence of the Radon-Nikodym derivative

proposition
(**in** *finite_measure*) *Radon_Nikodym_finite_measure*:
assumes *finite_measure N* **and** *sets_eq*[*simp*]: *sets N* = *sets M*
assumes *absolutely_continuous M N*
shows ∃*f* ∈ *borel_measurable M*. *density M f* = *N*

proposition (**in** *finite_measure*) *Radon_Nikodym_finite_measure_infinite*:
assumes *absolutely_continuous M N* **and** *sets_eq*: *sets N* = *sets M*
shows ∃*f*∈*borel_measurable M*. *density M f* = *N*

theorem (**in** *sigma_finite_measure*) *Radon_Nikodym*:
assumes *ac*: *absolutely_continuous M N* **assumes** *sets_eq*: *sets N* = *sets M*
shows ∃*f* ∈ *borel_measurable M*. *density M f* = *N*

8.5.3 Uniqueness of densities

proposition (**in** *sigma_finite_measure*) *density_unique*:
assumes *f*: *f* ∈ *borel_measurable M*
assumes *f'*: *f'* ∈ *borel_measurable M*
assumes *density_eq*: *density M f* = *density M f'*
shows *AE x in M*. *f x* = *f' x*

8.5.4 Radon-Nikodym derivative

definition *RN_deriv* :: *'a measure* ⇒ *'a measure* ⇒ *'a* ⇒ *ennreal* **where**
RN_deriv M N =
(*if* ∃*f*. *f* ∈ *borel_measurable M* ∧ *density M f* = *N*
then SOME f. *f* ∈ *borel_measurable M* ∧ *density M f* = *N*
else (λ_. *0*))

proposition (**in** *sigma_finite_measure*) *real_RN_deriv*:
assumes *finite_measure N*
assumes *ac*: *absolutely_continuous M N sets N* = *sets M*
obtains *D* **where** *D* ∈ *borel_measurable M*
and *AE x in M*. *RN_deriv M N x* = *ennreal* (*D x*)
and *AE x in N*. *0* < *D x*
and ⋀*x*. *0* ≤ *D x*

end

Chapter 9

Integrals over a Set

theory *Set_Integral*
  imports *Radon_Nikodym*
begin

9.1 Notation

definition *set_borel_measurable M A f* $\equiv$ (λx. *indicator A x* $*_R$ *f x*) $\in$ *borel_measurable M*

definition *set_integrable M A f* $\equiv$ *integrable M* (λx. *indicator A x* $*_R$ *f x*)

definition *set_lebesgue_integral M A f* $\equiv$ *lebesgue_integral M* (λx. *indicator A x* $*_R$ *f x*)

9.2 Basic properties

proposition *set_borel_measurable_subset*:
  fixes *f* :: _ $\Rightarrow$ _ :: {*banach*, *second_countable_topology*}
  assumes [*measurable*]: *set_borel_measurable M A f B* $\in$ *sets M* **and** $B \subseteq A$
  shows *set_borel_measurable M B f*

9.3 Complex integrals

9.4 NN Set Integrals

proposition *nn_integral_disjoint_family*:
  assumes [*measurable*]: $f \in$ *borel_measurable M* $\bigwedge(n{::}nat).\ B\ n \in$ *sets M*
    and *disjoint_family B*
  shows $(\int^+ x \in (\bigcup n.\ B\ n).\ f\ x\ \partial M) = (\sum n.\ (\int^+ x \in B\ n.\ f\ x\ \partial M))$

9.5 Scheffé's lemma

proposition *Scheffe_lemma1*:
  assumes $\bigwedge n.$ *integrable M (F n) integrable M f*
    $AE\ x\ in\ M.\ (\lambda n.\ F\ n\ x) \longlonglongrightarrow f\ x$
    $limsup\ (\lambda n.\ \int^+ x.\ norm(F\ n\ x)\ \partial M) \leq (\int^+ x.\ norm(f\ x)\ \partial M)$
  shows $(\lambda n.\ \int^+ x.\ norm(F\ n\ x - f\ x)\ \partial M) \longlonglongrightarrow 0$

proposition *Scheffe_lemma2*:
  fixes $F{::}nat \Rightarrow {'a} \Rightarrow {'b}{::}\{banach,\ second\_countable\_topology\}$
  assumes $\bigwedge\ n{::}nat.\ F\ n \in$ *borel_measurable M integrable M f*
    $AE\ x\ in\ M.\ (\lambda n.\ F\ n\ x) \longlonglongrightarrow f\ x$
    $\bigwedge n.\ (\int^+ x.\ norm(F\ n\ x)\ \partial M) \leq (\int^+ x.\ norm(f\ x)\ \partial M)$
  shows $(\lambda n.\ \int^+ x.\ norm(F\ n\ x - f\ x)\ \partial M) \longlonglongrightarrow 0$

9.6 Convergence of integrals over an interval

proposition *tendsto_set_lebesgue_integral_at_top*:
  fixes $f :: real \Rightarrow {'a}{::}\{banach,\ second\_countable\_topology\}$
  assumes *sets*: $\bigwedge b.\ b \geq a \Longrightarrow \{a..b\} \in$ *sets M*
    and *int*: *set_integrable M* $\{a..\}$ *f*
  shows $((\lambda b.$ *set_lebesgue_integral M* $\{a..b\}\ f) \longrightarrow$ *set_lebesgue_integral M* $\{a..\}\ f)$ *at_top*

proposition *tendsto_set_lebesgue_integral_at_bot*:
  fixes $f :: real \Rightarrow {'a}{::}\{banach,\ second\_countable\_topology\}$
  assumes *sets*: $\bigwedge a.\ a \leq b \Longrightarrow \{a..b\} \in$ *sets M*
    and *int*: *set_integrable M* $\{..b\}$ *f*
  shows $((\lambda a.$ *set_lebesgue_integral M* $\{a..b\}\ f) \longrightarrow$ *set_lebesgue_integral M* $\{..b\}\ f)$ *at_bot*

theorem *integral_Markov_inequality'*:
  fixes $u :: {'a} \Rightarrow real$
  assumes [*measurable*]: *set_integrable M A u* **and** $A \in$ *sets M*
  assumes $AE\ x\ in\ M.\ x \in A \longrightarrow u\ x \geq 0$ **and** $0 < (c{::}real)$

shows *emeasure M* $\{x{\in}A.\ u\ x \geq c\} \leq (1/c{::}real) * (\int x{\in}A.\ u\ x\ \partial M)$

theorem *integral_Markov_inequality'_measure*:
assumes [*measurable*]: *set_integrable M A u* **and** $A \in$ *sets M*
and *AE x in M*. $x \in A \longrightarrow 0 \leq u\ x\ 0 < (c{::}real)$
shows *measure M* $\{x{\in}A.\ u\ x \geq c\} \leq (\int x{\in}A.\ u\ x\ \partial M) / c$

theorem (**in** *finite_measure*) *Chernoff_ineq_ge*:
assumes *s*: $s > 0$
assumes *integrable*: *set_integrable M A* $(\lambda x.\ exp\ (s * f\ x))$ **and** $A \in$ *sets M*
shows *measure M* $\{x{\in}A.\ f\ x \geq a\} \leq exp\ (-s * a) * (\int x{\in}A.\ exp\ (s * f\ x)\ \partial M)$
proof –
have $\{x{\in}A.\ f\ x \geq a\} = \{x{\in}A.\ exp\ (s * f\ x) \geq exp\ (s * a)\}$
using *s* **by** *auto*
also have *measure M* ... $\leq$ *set_lebesgue_integral M A* $(\lambda x.\ exp\ (s * f\ x))$ / *exp* $(s * a)$
by (*intro integral_Markov_inequality'_measure assms*) *auto*
finally show *?thesis*
by (*simp add*: *exp_minus field_simps*)
qed

theorem (**in** *finite_measure*) *Chernoff_ineq_le*:
assumes *s*: $s > 0$
assumes *integrable*: *set_integrable M A* $(\lambda x.\ exp\ (-s * f\ x))$ **and** $A \in$ *sets M*
shows *measure M* $\{x{\in}A.\ f\ x \leq a\} \leq exp\ (s * a) * (\int x{\in}A.\ exp\ (-s * f\ x)\ \partial M)$
proof –
have $\{x{\in}A.\ f\ x \leq a\} = \{x{\in}A.\ exp\ (-s * f\ x) \geq exp\ (-s * a)\}$
using *s* **by** *auto*
also have *measure M* ... $\leq$ *set_lebesgue_integral M A* $(\lambda x.\ exp\ (-s * f\ x))$ / *exp* $(-s * a)$
by (*intro integral_Markov_inequality'_measure assms*) *auto*
finally show *?thesis*
by (*simp add*: *exp_minus field_simps*)
qed

9.7 Integrable Simple Functions

lemma *integrable_simple_function_induct*[*consumes 2*, *case_names cong indicator add*, *induct set*: *simple_function*]:
fixes $f :: 'a \Rightarrow 'b ::$ {*second_countable_topology*, *banach*}
assumes *f*: *simple_function M f emeasure M* $\{y \in space\ M.\ f\ y \neq 0\} \neq \infty$
assumes *cong*: $\bigwedge f\ g.$ *simple_function M f* $\Longrightarrow$ *emeasure M* $\{y \in space\ M.\ f\ y \neq 0\} \neq \infty$
$\Longrightarrow$ *simple_function M g* $\Longrightarrow$ *emeasure M* $\{y \in space\ M.\ g\ y \neq 0\} \neq \infty$
$\Longrightarrow (\bigwedge x.\ x \in space\ M \Longrightarrow f\ x = g\ x) \Longrightarrow P\ f \Longrightarrow P\ g$
assumes *indicator*: $\bigwedge A\ y.\ A \in sets\ M \Longrightarrow emeasure\ M\ A < \infty \Longrightarrow P\ (\lambda x.\ indicator\ A\ x *_R\ y)$
assumes *add*: $\bigwedge f\ g.$ *simple_function M f* $\Longrightarrow$ *emeasure M* $\{y \in space\ M.\ f\ y \neq$

0} $\neq \infty \Longrightarrow$
simple_function M g $\Longrightarrow$ *emeasure M* {$y \in$ *space M*. *g y* $\neq$ *0*} $\neq \infty \Longrightarrow$
($\bigwedge z.\ z \in$ *space M* $\Longrightarrow$ *norm* (*f z* + *g z*) = *norm* (*f z*) + *norm* (*g z*)) $\Longrightarrow$
P f $\Longrightarrow$ *P g* $\Longrightarrow$ *P* ($\lambda x.$ *f x* + *g x*)
shows *P f*

lemma *integrable_simple_function_induct_nn*[*consumes 3*, *case_names cong indicator add*, *induct set*: *simple_function*]:
fixes *f* :: *'a* $\Rightarrow$ *'b* :: {*second_countable_topology*, *banach*, *linorder_topology*, *ordered_real_vector*}
assumes *f*: *simple_function M f emeasure M* {$y \in$ *space M*. *f y* $\neq$ *0*} $\neq \infty$ $\bigwedge x.$ $x \in$ *space M* $\longrightarrow$ *f x* $\geq$ *0*
assumes *cong*: $\bigwedge$*f g*. *simple_function M f* $\Longrightarrow$ *emeasure M* {$y \in$ *space M*. *f y* $\neq$ *0*} $\neq \infty \Longrightarrow$ ($\bigwedge x.\ x \in$ *space M* $\Longrightarrow$ *f x* $\geq$ *0*) $\Longrightarrow$ *simple_function M g* $\Longrightarrow$ *emeasure M* {$y \in$ *space M*. *g y* $\neq$ *0*} $\neq \infty \Longrightarrow$ ($\bigwedge x.\ x \in$ *space M* $\Longrightarrow$ *g x* $\geq$ *0*) $\Longrightarrow$ ($\bigwedge x.\ x \in$ *space M* $\Longrightarrow$ *f x* = *g x*) $\Longrightarrow$ *P f* $\Longrightarrow$ *P g*
assumes *indicator*: $\bigwedge$*A y*. *y* $\geq$ *0* $\Longrightarrow$ *A* $\in$ *sets M* $\Longrightarrow$ *emeasure M A* $< \infty \Longrightarrow$ *P* ($\lambda x.$ *indicator A x* $*_R$ *y*)
assumes *add*: $\bigwedge$*f g*. ($\bigwedge x.\ x \in$ *space M* $\Longrightarrow$ *f x* $\geq$ *0*) $\Longrightarrow$ *simple_function M f* $\Longrightarrow$ *emeasure M* {$y \in$ *space M*. *f y* $\neq$ *0*} $\neq \infty \Longrightarrow$
($\bigwedge x.\ x \in$ *space M* $\Longrightarrow$ *g x* $\geq$ *0*) $\Longrightarrow$ *simple_function M g* $\Longrightarrow$ *emeasure M* {$y \in$ *space M*. *g y* $\neq$ *0*} $\neq \infty \Longrightarrow$
($\bigwedge z.\ z \in$ *space M* $\Longrightarrow$ *norm* (*f z* + *g z*) = *norm* (*f z*) + *norm* (*g z*)) $\Longrightarrow$
P f $\Longrightarrow$ *P g* $\Longrightarrow$ *P* ($\lambda x.$ *f x* + *g x*)
shows *P f*

9.7.1 Totally Ordered Banach Spaces

9.7.2 Auxiliary Lemmas for Set Integrals

9.7.3 Integrability and Measurability of the Diameter

9.7.4 Averaging Theorem

corollary *integral_nonneg_eq_0_iff_AE_banach*:
fixes *f* :: *'a* $\Rightarrow$ *'b* :: {*second_countable_topology*, *banach*, *linorder_topology*, *ordered_real_vector*}
assumes *f*[*measurable*]: *integrable M f* **and** *nonneg*: *AE x in M*. *0* $\leq$ *f x*
shows *integral*^{L} *M f* = *0* $\longleftrightarrow$ (*AE x in M*. *f x* = *0*)

corollary *integral_eq_mono_AE_eq_AE*:
fixes *f g* :: *'a* $\Rightarrow$ *'b* :: {*second_countable_topology*, *banach*, *linorder_topology*, *ordered_real_vector*}
assumes *integrable M f integrable M g integral*^{L} *M f* = *integral*^{L} *M g AE x in M*. *f x* $\leq$ *g x*
shows *AE x in M*. *f x* = *g x*

end

9.8 Homeomorphism Theorems

theory *Homeomorphism*
imports *Homotopy*
begin

9.8.1 Homeomorphism of all convex compact sets with nonempty interior

proposition
fixes *S* :: *'a::euclidean_space set*
assumes *compact S* **and** *0*: *0* ∈ *rel_interior S*
and *star*: ⋀*x*. *x* ∈ *S* ⟹ *open_segment 0 x* ⊆ *rel_interior S*
shows *starlike_compact_projective1_0*:
S − *rel_interior S homeomorphic sphere 0 1* ∩ *affine hull S*
(**is** *?SMINUS homeomorphic ?SPHER*)
and *starlike_compact_projective2_0*:
S homeomorphic cball 0 1 ∩ *affine hull S*
(**is** *S homeomorphic ?CBALL*)

corollary
fixes *S* :: *'a::euclidean_space set*
assumes *compact S* **and** *a*: *a* ∈ *rel_interior S*
and *star*: ⋀*x*. *x* ∈ *S* ⟹ *open_segment a x* ⊆ *rel_interior S*
shows *starlike_compact_projective1*:
S − *rel_interior S homeomorphic sphere a 1* ∩ *affine hull S*
and *starlike_compact_projective2*:
S homeomorphic cball a 1 ∩ *affine hull S*

corollary *starlike_compact_projective_special*:
assumes *compact S*
and *cb01*: *cball* (*0::'a::euclidean_space*) *1* ⊆ *S*
and *scale*: ⋀*x u*. ⟦*x* ∈ *S*; *0* ≤ *u*; *u* < *1*⟧ ⟹ *u* $*_R$ *x* ∈ *S* − *frontier S*
shows *S homeomorphic* (*cball* (*0::'a::euclidean_space*) *1*)

9.8.2 Homeomorphisms between punctured spheres and affine sets

theorem *homeomorphic_punctured_affine_sphere_affine*:
fixes *a* :: *'a* :: *euclidean_space*
assumes *0* < *r b* ∈ *sphere a r affine T a* ∈ *T b* ∈ *T affine p*
and *aff*: *aff_dim T* = *aff_dim p* + *1*
shows (*sphere a r* ∩ *T*) − {*b*} *homeomorphic p*

corollary *homeomorphic_punctured_sphere_affine*:
fixes *a* :: *'a* :: *euclidean_space*
assumes *0* < *r* **and** *b*: *b* ∈ *sphere a r*
and *affine T* **and** *affS*: *aff_dim T* + *1* = *DIM*(*'a*)
shows (*sphere a r* − {*b*}) *homeomorphic T*

corollary *homeomorphic_punctured_sphere_hyperplane*:
fixes *a* :: *'a* :: *euclidean_space*
assumes *0* < *r* **and** *b*: *b* ∈ *sphere a r*
and *c* ≠ *0*
shows (*sphere a r* − {*b*}) *homeomorphic* {*x*::*'a*. *c* • *x* = *d*}

proposition *homeomorphic_punctured_sphere_affine_gen*:
fixes *a* :: *'a* :: *euclidean_space*
assumes *convex S bounded S* **and** *a*: *a* ∈ *rel_frontier S*
and *affine T* **and** *affS*: *aff_dim S* = *aff_dim T* + *1*
shows *rel_frontier S* − {*a*} *homeomorphic T*
proposition *homeomorphic_closedin_convex*:
fixes *S* :: *'m*::*euclidean_space set*
assumes *aff_dim S* < *DIM*(*'n*)
obtains *U* **and** *T* :: *'n*::*euclidean_space set*
where *convex U U* ≠ {} *closedin* (*top_of_set U*) *T*
S homeomorphic T

9.8.3 Locally compact sets in an open set

proposition *locally_compact_homeomorphic_closed*:
fixes *S* :: *'a*::*euclidean_space set*
assumes *locally compact S* **and** *dimlt*: *DIM*(*'a*) < *DIM*(*'b*)
obtains *T* :: *'b*::*euclidean_space set* **where** *closed T S homeomorphic T*

proposition *homeomorphic_convex_compact_cball*:
fixes *e* :: *real*
and *S* :: *'a*::*euclidean_space set*
assumes *S*: *convex S compact S interior S* ≠ {} **and** *e* > *0*
shows *S homeomorphic* (*cball* (*b*::*'a*) *e*)

corollary *homeomorphic_convex_compact*:
fixes *S* :: *'a*::*euclidean_space set*
and *T* :: *'a set*
assumes *convex S compact S interior S* ≠ {}
and *convex T compact T interior T* ≠ {}
shows *S homeomorphic T*

9.8.4 Covering spaces and lifting results for them

definition *covering_space*
  :: $'a$::*topological_space set* ⇒ ($'a$ ⇒ $'b$) ⇒ $'b$::*topological_space set* ⇒ *bool*
 where
 covering_space c p S ≡
  continuous_on c p ∧ *p ‘ c* = *S* ∧
  (∀ *x* ∈ *S*. ∃ *T*. *x* ∈ *T* ∧ *openin* (*top_of_set S*) *T* ∧
     (∃ *v*. ⋃*v* = *c* ∩ *p* −‘ *T* ∧
      (∀ *u* ∈ *v*. *openin* (*top_of_set c*) *u*) ∧
      pairwise disjnt v ∧
      (∀ *u* ∈ *v*. ∃ *q*. *homeomorphism u T p q*)))

proposition *covering_space_open_map*:
 fixes *S* :: $'a$ :: *metric_space set* **and** *T* :: $'b$ :: *metric_space set*
 assumes *p*: *covering_space c p S* **and** *T*: *openin* (*top_of_set c*) *T*
  shows *openin* (*top_of_set S*) (*p ‘ T*)

proposition *covering_space_lift_unique*:
 fixes *f* :: $'a$::*topological_space* ⇒ $'b$::*topological_space*
 fixes *g1* :: $'a$ ⇒ $'c$::*real_normed_vector*
 assumes *covering_space c p S*
   g1 a = *g2 a*
   continuous_on T f *f* ∈ *T* → *S*
   continuous_on T g1 *g1* ∈ *T* → *c* ⋀*x*. *x* ∈ *T* ⟹ *f x* = *p*(*g1 x*)
   continuous_on T g2 *g2* ∈ *T* → *c* ⋀*x*. *x* ∈ *T* ⟹ *f x* = *p*(*g2 x*)
   connected T *a* ∈ *T* *x* ∈ *T*
 shows *g1 x* = *g2 x*

proposition *covering_space_locally_eq*:
 fixes *p* :: $'a$::*real_normed_vector* ⇒ $'b$::*real_normed_vector*
 assumes *cov*: *covering_space C p S*
  and *pim*: ⋀*T*. ⟦*T* ⊆ *C*; *φ T*⟧ ⟹ *ψ*(*p ‘ T*)
  and *qim*: ⋀*q U*. ⟦*U* ⊆ *S*; *continuous_on U q*; *ψ U*⟧ ⟹ *φ*(*q ‘ U*)
  shows *locally ψ S* ⟷ *locally φ C*
   (**is** *?lhs* = *?rhs*)

proposition *covering_space_lift_homotopy*:
 fixes *p* :: $'a$::*real_normed_vector* ⇒ $'b$::*real_normed_vector*
  and *h* :: *real* × $'c$::*real_normed_vector* ⇒ $'b$
 assumes *cov*: *covering_space C p S*
  and *conth*: *continuous_on* ({*0..1*} × *U*) *h*
  and *him*: *h* ∈ ({*0..1*} × *U*) → *S*

and *heq*: $\bigwedge y.\ y \in U \Longrightarrow h\ (0,y) = p(f\ y)$
and *contf*: *continuous_on U f* **and** *fim*: $f \in U \to C$
obtains *k* **where** *continuous_on* $(\{0..1\} \times U)$ *k*
$k \in (\{0..1\} \times U) \to C$
$\bigwedge y.\ y \in U \Longrightarrow k(0,\ y) = f\ y$
$\bigwedge z.\ z \in \{0..1\} \times U \Longrightarrow h\ z = p(k\ z)$

corollary *covering_space_lift_homotopy_alt*:
fixes *p* :: *'a::real_normed_vector* ⇒ *'b::real_normed_vector*
and *h* :: *'c::real_normed_vector* × *real* ⇒ *'b*
assumes *cov*: *covering_space C p S*
and *conth*: *continuous_on* $(U \times \{0..1\})$ *h*
and *him*: $h \in (U \times \{0..1\}) \to S$
and *heq*: $\bigwedge y.\ y \in U \Longrightarrow h\ (y,0) = p(f\ y)$
and *contf*: *continuous_on U f* **and** *fim*: $f \in U \to C$
obtains *k* **where** *continuous_on* $(U \times \{0..1\})$ *k*
$k \in (U \times \{0..1\}) \to C$
$\bigwedge y.\ y \in U \Longrightarrow k(y,\ 0) = f\ y$
$\bigwedge z.\ z \in U \times \{0..1\} \Longrightarrow h\ z = p(k\ z)$

corollary *covering_space_lift_homotopic_function*:
fixes *p* :: *'a::real_normed_vector* ⇒ *'b::real_normed_vector* **and** *g*:: *'c::real_normed_vector* ⇒ *'a*
assumes *cov*: *covering_space C p S*
and *contg*: *continuous_on U g*
and *gim*: $g \in U \to C$
and *pgeq*: $\bigwedge y.\ y \in U \Longrightarrow p(g\ y) = f\ y$
and *hom*: *homotopic_with_canon* (λ*x. True*) *U S f f'*
obtains *g'* **where** *continuous_on U g' image g' U* ⊆ *C* $\bigwedge y.\ y \in U \Longrightarrow p(g'\ y) = f'\ y$

corollary *covering_space_lift_inessential_function*:
fixes *p* :: *'a::real_normed_vector* ⇒ *'b::real_normed_vector* **and** *U* :: *'c::real_normed_vector set*
assumes *cov*: *covering_space C p S*
and *hom*: *homotopic_with_canon* (λ*x. True*) *U S f* (λ*x. a*)
obtains *g* **where** *continuous_on U g g ‘ U* ⊆ *C* $\bigwedge y.\ y \in U \Longrightarrow p(g\ y) = f\ y$

9.8.5 Lifting of general functions to covering space

proposition *covering_space_lift_path_strong*:
fixes *p* :: *'a::real_normed_vector* ⇒ *'b::real_normed_vector*
and *f* :: *'c::real_normed_vector* ⇒ *'b*
assumes *cov*: *covering_space C p S* **and** $a \in C$
and *path g* **and** *pag*: *path_image g* ⊆ *S* **and** *pas*: *pathstart g = p a*
obtains *h* **where** *path h path_image h* ⊆ *C pathstart h = a*
and $\bigwedge t.\ t \in \{0..1\} \Longrightarrow p(h\ t) = g\ t$

corollary *covering_space_lift_path*:
fixes p :: $'a$::*real_normed_vector* ⇒ $'b$::*real_normed_vector*
assumes *cov*: *covering_space C p S* **and** *path g* **and** *pig*: *path_image g* ⊆ *S*
obtains *h* **where** *path h path_image h* ⊆ *C* ⋀*t. t* ∈ {*0..1*} ⟹ *p*(*h t*) = *g t*

proposition *covering_space_lift_homotopic_paths*:
fixes p :: $'a$::*real_normed_vector* ⇒ $'b$::*real_normed_vector*
assumes *cov*: *covering_space C p S*
and *path g1* **and** *pig1*: *path_image g1* ⊆ *S*
and *path g2* **and** *pig2*: *path_image g2* ⊆ *S*
and *hom*: *homotopic_paths S g1 g2*
and *path h1* **and** *pih1*: *path_image h1* ⊆ *C* **and** *ph1*: ⋀*t. t* ∈ {*0..1*} ⟹ *p*(*h1 t*) = *g1 t*
and *path h2* **and** *pih2*: *path_image h2* ⊆ *C* **and** *ph2*: ⋀*t. t* ∈ {*0..1*} ⟹ *p*(*h2 t*) = *g2 t*
and *h1h2*: *pathstart h1* = *pathstart h2*
shows *homotopic_paths C h1 h2*

corollary *covering_space_monodromy*:
fixes p :: $'a$::*real_normed_vector* ⇒ $'b$::*real_normed_vector*
assumes *cov*: *covering_space C p S*
and *path g1* **and** *pig1*: *path_image g1* ⊆ *S*
and *path g2* **and** *pig2*: *path_image g2* ⊆ *S*
and *hom*: *homotopic_paths S g1 g2*
and *path h1* **and** *pih1*: *path_image h1* ⊆ *C* **and** *ph1*: ⋀*t. t* ∈ {*0..1*} ⟹ *p*(*h1 t*) = *g1 t*
and *path h2* **and** *pih2*: *path_image h2* ⊆ *C* **and** *ph2*: ⋀*t. t* ∈ {*0..1*} ⟹ *p*(*h2 t*) = *g2 t*
and *h1h2*: *pathstart h1* = *pathstart h2*
shows *pathfinish h1* = *pathfinish h2*

corollary *covering_space_lift_homotopic_path*:
fixes p :: $'a$::*real_normed_vector* ⇒ $'b$::*real_normed_vector*
assumes *cov*: *covering_space C p S*
and *hom*: *homotopic_paths S f f'*
and *path g* **and** *pig*: *path_image g* ⊆ *C*
and *a*: *pathstart g* = *a* **and** *b*: *pathfinish g* = *b*
and *pgeq*: ⋀*t. t* ∈ {*0..1*} ⟹ *p*(*g t*) = *f t*
obtains *g'* **where** *path g' path_image g'* ⊆ *C*
pathstart g' = *a pathfinish g'* = *b* ⋀*t. t* ∈ {*0..1*} ⟹ *p*(*g' t*) = *f' t*

proposition *covering_space_lift_general*:
fixes p :: $'a$::*real_normed_vector* ⇒ $'b$::*real_normed_vector*
and f :: $'c$::*real_normed_vector* ⇒ $'b$
assumes *cov*: *covering_space C p S* **and** *a* ∈ *C z* ∈ *U*

and *U*: *path_connected U locally path_connected U*
and *contf*: *continuous_on U f* **and** *fim*: *f* ∈ *U* → *S*
and *feq*: *f z* = *p a*
and *hom*: ⋀*r*. ⟦*path r*; *path_image r* ⊆ *U*; *pathstart r* = *z*; *pathfinish r* = *z*⟧
⟹ ∃ *q*. *path q* ∧ *path_image q* ⊆ *C* ∧
pathstart q = *a* ∧ *pathfinish q* = *a* ∧
homotopic_paths S (*f* ∘ *r*) (*p* ∘ *q*)
obtains *g* **where** *continuous_on U g g* ∈ *U* → *C g z* = *a* ⋀*y*. *y* ∈ *U* ⟹ *p*(*g y*) = *f y*

corollary *covering_space_lift_stronger*:
fixes *p* :: *'a*::*real_normed_vector* ⇒ *'b*::*real_normed_vector*
and *f* :: *'c*::*real_normed_vector* ⇒ *'b*
assumes *cov*: *covering_space C p S a* ∈ *C z* ∈ *U*
and *U*: *path_connected U locally path_connected U*
and *contf*: *continuous_on U f* **and** *fim*: *f* ∈ *U* → *S*
and *feq*: *f z* = *p a*
and *hom*: ⋀*r*. ⟦*path r*; *path_image r* ⊆ *U*; *pathstart r* = *z*; *pathfinish r* = *z*⟧
⟹ ∃ *b*. *homotopic_paths S* (*f* ∘ *r*) (*linepath b b*)
obtains *g* **where** *continuous_on U g g* ∈ *U* → *C g z* = *a* ⋀*y*. *y* ∈ *U* ⟹ *p*(*g y*) = *f y*

corollary *covering_space_lift_strong*:
fixes *p* :: *'a*::*real_normed_vector* ⇒ *'b*::*real_normed_vector*
and *f* :: *'c*::*real_normed_vector* ⇒ *'b*
assumes *cov*: *covering_space C p S a* ∈ *C z* ∈ *U*
and *scU*: *simply_connected U* **and** *lpcU*: *locally path_connected U*
and *contf*: *continuous_on U f* **and** *fim*: *f* ∈ *U* → *S*
and *feq*: *f z* = *p a*
obtains *g* **where** *continuous_on U g g* ∈ *U* → *C g z* = *a* ⋀*y*. *y* ∈ *U* ⟹ *p*(*g y*) = *f y*

corollary *covering_space_lift*:
fixes *p* :: *'a*::*real_normed_vector* ⇒ *'b*::*real_normed_vector*
and *f* :: *'c*::*real_normed_vector* ⇒ *'b*
assumes *cov*: *covering_space C p S*
and *U*: *simply_connected U locally path_connected U*
and *contf*: *continuous_on U f* **and** *fim*: *f* ∈ *U* → *S*
obtains *g* **where** *continuous_on U g g* ∈ *U* → *C* ⋀*y*. *y* ∈ *U* ⟹ *p*(*g y*) = *f y*

end

theory *Equivalence_Lebesgue_Henstock_Integration*
imports
Lebesgue_Measure
Henstock_Kurzweil_Integration
Complete_Measure
Set_Integral

Homeomorphism
Cartesian_Euclidean_Space
begin

9.8.6 Equivalence Lebesgue integral on *lborel* and HK-integral

9.8.7 Absolute integrability (this is the same as Lebesgue integrability)

9.8.8 Applications to Negligibility

corollary *eventually_ae_filter_negligible*:
$eventually\ P\ (ae\_filter\ lebesgue) \longleftrightarrow (\exists N.\ negligible\ N \wedge \{x.\ \neg\ P\ x\} \subseteq N)$

proposition *negligible_convex_frontier*:
fixes $S :: 'N :: euclidean\_space\ set$
assumes *convex S*
shows *negligible*(*frontier S*)

corollary *negligible_sphere*: *negligible* (*sphere a e*)

proposition *open_not_negligible*:
assumes *open S* $S \neq \{\}$
shows $\neg$ *negligible S*

9.8.9 Negligibility of image under non-injective linear map

9.8.10 Negligibility of a Lipschitz image of a negligible set

proposition *negligible_locally_Lipschitz_image*:
fixes $f :: 'M::euclidean\_space \Rightarrow 'N::euclidean\_space$
assumes *MleN*: $DIM('M) \leq DIM('N)$ *negligible S*
and *lips*: $\bigwedge x.\ x \in S$
$\Longrightarrow \exists T\ B.\ open\ T \wedge x \in T \wedge$
$(\forall y \in S \cap T.\ norm(f\ y - f\ x) \leq B * norm(y - x))$
shows *negligible* (f ‘ S)

corollary *negligible_differentiable_image_negligible*:
fixes $f :: 'M::euclidean\_space \Rightarrow 'N::euclidean\_space$
assumes *MleN*: $DIM('M) \leq DIM('N)$ *negligible S*
and *diff_f*: *f differentiable_on S*
shows *negligible* (f ‘ S)

corollary *negligible_differentiable_image_lowdim*:
fixes $f :: 'M::euclidean\_space \Rightarrow 'N::euclidean\_space$
assumes *MlessN*: $DIM('M) < DIM('N)$ **and** *diff_f*: *f differentiable_on S*
shows *negligible* $(f ` S)$

9.8.11 Measurability of countable unions and intersections of various kinds.

9.8.12 Negligibility is a local property

9.8.13 Integral bounds

proposition *bounded_variation_absolutely_integrable_interval*:
fixes $f :: 'n::euclidean\_space \Rightarrow 'm::euclidean\_space$
assumes *f*: *f integrable_on cbox a b*
and $*$: $\bigwedge d.\ d\ division\_of\ (cbox\ a\ b) \Longrightarrow sum\ (\lambda K.\ norm(integral\ K\ f))\ d \le B$
shows *f absolutely_integrable_on cbox a b*

9.8.14 Outer and inner approximation of measurable sets by well-behaved sets.

proposition *measurable_outer_intervals_bounded*:
assumes $S \in lmeasurable\ S \subseteq cbox\ a\ b\ e > 0$
obtains $\mathcal{D}$
where *countable* $\mathcal{D}$
$\bigwedge K.\ K \in \mathcal{D} \Longrightarrow K \subseteq cbox\ a\ b \wedge K \ne \{\} \wedge (\exists c\ d.\ K = cbox\ c\ d)$
$pairwise\ (\lambda A\ B.\ interior\ A \cap interior\ B = \{\})\ \mathcal{D}$
$\bigwedge u\ v.\ cbox\ u\ v \in \mathcal{D} \Longrightarrow \exists n.\ \forall i \in Basis.\ v \cdot i - u \cdot i = (b \cdot i - a \cdot i)/2\hat{}n$
$\bigwedge K.\ [\![K \in \mathcal{D};\ box\ a\ b \ne \{\}]\!] \Longrightarrow interior\ K \ne \{\}$
$S \subseteq \bigcup\mathcal{D}\ \bigcup\mathcal{D} \in lmeasurable\ measure\ lebesgue\ (\bigcup\mathcal{D}) \le measure\ lebesgue\ S + e$

9.8.15 Transformation of measure by linear maps

proposition *measure_linear_sufficient*:
fixes $f :: 'n::euclidean\_space \Rightarrow 'n$
assumes *linear f* **and** *S*: $S \in lmeasurable$
and *im*: $\bigwedge a\ b.\ measure\ lebesgue\ (f ` (cbox\ a\ b)) = m * measure\ lebesgue\ (cbox\ a\ b)$
shows $f ` S \in lmeasurable \wedge m * measure\ lebesgue\ S = measure\ lebesgue\ (f ` S)$

9.8.16 Lemmas about absolute integrability

corollary *absolutely_integrable_on_const* [*simp*]:
fixes *c* :: *'a::euclidean_space*
assumes *S* ∈ *lmeasurable*
shows (λ*x*. *c*) *absolutely_integrable_on S*

9.8.17 Componentwise

proposition *absolutely_integrable_componentwise_iff*:
shows *f absolutely_integrable_on A* ⟷ (∀ *b*∈*Basis*. (λ*x*. *f x* · *b*) *absolutely_integrable_on A*)

corollary *absolutely_integrable_max_1*:
fixes *f* :: *'n::euclidean_space* ⇒ *real*
assumes *f absolutely_integrable_on S g absolutely_integrable_on S*
shows (λ*x*. *max* (*f x*) (*g x*)) *absolutely_integrable_on S*

corollary *absolutely_integrable_min_1*:
fixes *f* :: *'n::euclidean_space* ⇒ *real*
assumes *f absolutely_integrable_on S g absolutely_integrable_on S*
shows (λ*x*. *min* (*f x*) (*g x*)) *absolutely_integrable_on S*

9.8.18 Dominated convergence

proposition *integral_countable_UN*:
fixes *f* :: *real^'m* ⇒ *real^'n*
assumes *f*: *f absolutely_integrable_on* (⋃(*range s*))
and *s*: ⋀*m*. *s m* ∈ *sets lebesgue*
shows ⋀*n*. *f absolutely_integrable_on* (⋃*m*≤*n*. *s m*)
and (λ*n*. *integral* (⋃*m*≤*n*. *s m*) *f*) ⟶ *integral* (⋃(*s* ‘ *UNIV*)) *f* (**is** *?F* ⟶ *?I*)

9.8.19 Fundamental Theorem of Calculus for the Lebesgue integral

9.8.20 Integration by parts

9.8.21 A non-negative continuous function whose integral is zero must be zero

corollary *integral_cbox_eq_0_iff*:
fixes f :: *'a::euclidean_space* ⇒ *real*
assumes *continuous_on* (*cbox a b*) f **and** *box a b* ≠ {}
and $\bigwedge x.\ x \in$ *cbox a b* $\Longrightarrow f\ x \geq 0$
shows *integral* (*cbox a b*) f = *0* ⟷ (∀ x ∈ *cbox a b*. $f\ x$ = *0*) (**is** *?lhs* = *?rhs*)

9.8.22 Various common equivalent forms of function measurability

9.8.23 Lebesgue sets and continuous images

proposition *lebesgue_regular_inner*:
assumes S ∈ *sets lebesgue*
obtains K C **where** *negligible* K ⋀*n::nat*. *compact*(C n) S = (⋃ n. C n) ∪ K

9.8.24 Affine lemmas

lemma *lebesgue_integral_real_affine*:
fixes f :: *real* ⇒ *'a* :: *euclidean_space* **and** c :: *real*
assumes c: $c \neq 0$ **shows** (∫ x. f x ∂ *lebesgue*) = |c| $*_R$ (∫ x. $f(t + c * x)$ ∂*lebesgue*)

9.8.25 More results on integrability

proposition *measurable_bounded_by_integrable_imp_integrable*:
fixes f :: *'a::euclidean_space* ⇒ *'b::euclidean_space*
assumes f: f ∈ *borel_measurable* (*lebesgue_on* S) **and** g: g *integrable_on* S
and *normf*: ⋀x. $x \in S \Longrightarrow$ *norm*(f x) ≤ g x **and** S: S ∈ *sets lebesgue*
shows f *integrable_on* S

9.8.26 Relation between Borel measurability and integrability.

proposition *negligible_differentiable_vimage*:
  fixes f :: $'a \Rightarrow$ $'a$::*euclidean_space*
  assumes *negligible T*
    and f': $\bigwedge x.\ x \in S \Longrightarrow$ *inj*(f' x)
    and *derf*: $\bigwedge x.\ x \in S \Longrightarrow$ (f *has_derivative* f' x) (*at* x *within* S)
  shows *negligible* $\{x \in S.\ f\ x \in T\}$
proposition *has_derivative_inverse_within*:
  fixes f :: $'a$::*real_normed_vector* $\Rightarrow$ $'b$::*euclidean_space*
  assumes *der_f*: (f *has_derivative* f') (*at* a *within* S)
    and *cont_g*: *continuous* (*at* (f a) *within* f ‘ S) g
    and $a \in S$ *linear* g' **and** *id*: $g' \circ f' = id$
    and *gf*: $\bigwedge x.\ x \in S \Longrightarrow g(f\ x) = x$
  shows (g *has_derivative* g') (*at* (f a) *within* f ‘ S)

end

9.9 Harmonic Numbers

theory *Harmonic_Numbers*
imports
  Complex_Transcendental
  Summation_Tests
begin

9.9.1 The Harmonic numbers

definition *harm* :: *nat* $\Rightarrow$ $'a$ :: *real_normed_field* **where**
  harm n = ($\sum k$=*1*..n. *inverse* (*of_nat* k))

theorem *not_convergent_harm*: $\neg$*convergent* (*harm* :: *nat* $\Rightarrow$ $'a$ :: *real_normed_field*)

9.9.2 The Euler-Mascheroni constant

lemma *euler_mascheroni_LIMSEQ*:
  ($\lambda n.$ *harm* n − *ln* (*of_nat* n) :: *real*) $\longrightarrow$ *euler_mascheroni*

theorem *alternating_harmonic_series_sums*: ($\lambda k.$ (−*1*)^k / *real_of_nat* (*Suc* k)) *sums ln 2*

end

9.10 The Gamma Function

theory *Gamma_Function*
imports
Equivalence_Lebesgue_Henstock_Integration
Summation_Tests
Harmonic_Numbers
HOL−Library.Nonpos_Ints
HOL−Library.Periodic_Fun
begin

9.10.1 The Euler form and the logarithmic Gamma function

definition *Gamma_series* :: (*'a* :: {*banach*,*real_normed_field*}) ⇒ *nat* ⇒ *'a* **where**
Gamma_series z n = *fact n* ∗ *exp* (*z* ∗ *of_real* (*ln* (*of_nat n*))) / *pochhammer z* (*n*+*1*)
definition *ln_Gamma_series* :: (*'a* :: {*banach*,*real_normed_field*,*ln*}) ⇒ *nat* ⇒ *'a* **where**
ln_Gamma_series z n = *z* ∗ *ln* (*of_nat n*) − *ln z* − (∑ *k*=*1*..*n*. *ln* (*z* / *of_nat k* + *1*))

theorem *ln_Gamma_complex_LIMSEQ*: (*z* :: *complex*) $\notin \mathbb{Z}_{\leq 0}$ ⟹ *ln_Gamma_series z* ⟶ *ln_Gamma z*

9.10.2 The Polygamma functions

definition *Polygamma* :: *nat* ⇒ (*'a* :: {*real_normed_field*,*banach*}) ⇒ *'a* **where**
Polygamma n z = (*if n* = *0 then*
(∑ *k*. *inverse* (*of_nat* (*Suc k*)) − *inverse* (*z* + *of_nat k*)) − *euler_mascheroni else*
(−*1*)^*Suc n* ∗ *fact n* ∗ (∑ *k*. *inverse* ((*z* + *of_nat k*)^*Suc n*)))

abbreviation *Digamma* :: (*'a* :: {*real_normed_field*,*banach*}) ⇒ *'a* **where**
Digamma ≡ *Polygamma 0*

theorem *Digamma_LIMSEQ*:
fixes *z* :: *'a* :: {*banach*,*real_normed_field*}
assumes *z*: *z* ≠ *0*
shows (λ*m*. *of_real* (*ln* (*real m*)) − (∑ *n*<*m*. *inverse* (*z* + *of_nat n*))) ⟶ *Digamma z*

theorem *Polygamma_LIMSEQ*:
fixes *z* :: *'a* :: {*banach*,*real_normed_field*}
assumes *z* ≠ *0* **and** *n* > *0*

shows ($\lambda k.$ *inverse* ((*z* + *of_nat k*)^*Suc n*)) *sums* ((−*1*) ^ *Suc n* ∗ *Polygamma n z* / *fact n*)

theorem *has_field_derivative_ln_Gamma_complex* [*derivative_intros*]:
fixes *z* :: *complex*
assumes *z*: $z \notin \mathbb{R}_{\leq 0}$
shows (*ln_Gamma has_field_derivative Digamma z*) (*at z*)

theorem *Polygamma_plus1*:
assumes $z \neq 0$
shows *Polygamma n* (*z* + *1*) = *Polygamma n z* + (−*1*)^*n* ∗ *fact n* / (*z* ^ *Suc n*)

theorem *Digamma_of_nat*:
Digamma (*of_nat* (*Suc n*) :: *'a* :: {*real_normed_field*,*banach*}) = *harm n* − *euler_mascheroni*

theorem *has_field_derivative_Polygamma* [*derivative_intros*]:
fixes *z* :: *'a* :: {*real_normed_field*,*euclidean_space*}
assumes *z*: $z \notin \mathbb{Z}_{\leq 0}$
shows (*Polygamma n has_field_derivative Polygamma* (*Suc n*) *z*) (*at z within A*)

9.10.3 Basic properties

theorem *Gamma_series_LIMSEQ* [*tendsto_intros*]:
Gamma_series z ⟶ *Gamma z*

theorem *Gamma_plus1*: $z \notin \mathbb{Z}_{\leq 0} \Longrightarrow$ *Gamma* (*z* + *1*) = *z* ∗ *Gamma z*

theorem *pochhammer_Gamma*: $z \notin \mathbb{Z}_{\leq 0} \Longrightarrow$ *pochhammer z n* = *Gamma* (*z* + *of_nat n*) / *Gamma z*

theorem *Gamma_fact*: *Gamma* (*1* + *of_nat n*) = *fact n*

9.10.4 Differentiability

theorem *has_field_derivative_Gamma* [*derivative_intros*]:
$z \notin \mathbb{Z}_{\leq 0} \Longrightarrow$ (*Gamma has_field_derivative Gamma z* ∗ *Digamma z*) (*at z within A*)

theorem *log_convex_Gamma_real*: *convex_on* {*0*<..} (*ln* ∘ *Gamma* :: *real* ⇒ *real*)

9.10.5 The uniqueness of the real Gamma function

theorem *Gamma_pos_real_unique*:
assumes *x*: $x > 0$
shows $G\ x = Gamma\ x$

9.10.6 The Beta function

theorem *Beta_plus1_plus1*:
assumes $x \notin \mathbb{Z}_{\leq 0}$ $y \notin \mathbb{Z}_{\leq 0}$
shows $Beta\ (x + 1)\ y + Beta\ x\ (y + 1) = Beta\ x\ y$

theorem *Beta_plus1_left*:
assumes $x \notin \mathbb{Z}_{\leq 0}$
shows $(x + y) * Beta\ (x + 1)\ y = x * Beta\ x\ y$

theorem *Beta_plus1_right*:
assumes $y \notin \mathbb{Z}_{\leq 0}$
shows $(x + y) * Beta\ x\ (y + 1) = y * Beta\ x\ y$

9.10.7 Legendre duplication theorem

theorem *Gamma_legendre_duplication*:
fixes *z* :: *complex*
assumes $z \notin \mathbb{Z}_{\leq 0}$ $z + 1/2 \notin \mathbb{Z}_{\leq 0}$
shows $Gamma\ z * Gamma\ (z + 1/2) =$
$exp\ ((1 - 2*z) * of\_real\ (ln\ 2)) * of\_real\ (sqrt\ pi) * Gamma\ (2*z)$

9.10.8 Alternative definitions

theorem *Gamma_series_euler'*:
assumes *z*: $(z :: 'a :: Gamma) \notin \mathbb{Z}_{\leq 0}$
shows $(\lambda n.\ Gamma\_series\_euler'\ z\ n) \longrightarrow Gamma\ z$

theorem *Gamma_Weierstrass_complex*: *Gamma_series_Weierstrass z* $\longlongrightarrow$ *Gamma* (*z* :: *complex*)

theorem *gbinomial_Gamma*:
assumes $z + 1 \notin \mathbb{Z}_{\leq 0}$
shows (*z gchoose n*) = *Gamma* (*z* + *1*) / (*fact n* ∗ *Gamma* (*z* − *of_nat n* + *1*))

theorem *Gamma_integral_complex*:
assumes *z*: *Re z* > *0*
shows ((λ*t*. *of_real t powr* (*z* − *1*) / *of_real* (*exp t*)) *has_integral Gamma z*) {*0*..}

theorem *has_integral_Beta_real*:
assumes *a*: *a* > *0* **and** *b*: *b* > (*0* :: *real*)
shows ((λ*t*. *t powr* (*a* − *1*) ∗ (*1* − *t*) *powr* (*b* − *1*)) *has_integral Beta a b*) {*0*..*1*}

9.10.9 The Weierstraß product formula for the sine

theorem *sin_product_formula_complex*:
fixes *z* :: *complex*
shows (λ*n*. *of_real pi* ∗ *z* ∗ (∏*k*=*1*..*n*. *1* − *z*^*2* / *of_nat k*^*2*)) $\longrightarrow$ *sin* (*of_real pi* ∗ *z*)

theorem *wallis*: (λ*n*. ∏*k*=*1*..*n*. (*4*∗*real k*^*2*) / (*4*∗*real k*^*2* − *1*)) $\longrightarrow$ *pi* / *2*

9.10.10 The Solution to the Basel problem

theorem *inverse_squares_sums*: (λ*n*. *1* / $(n + 1)^2$) *sums* (pi^2 / *6*)

end

theory *Interval_Integral*
imports *Equivalence_Lebesgue_Henstock_Integration*
begin

9.10.11 Approximating a (possibly infinite) interval

proposition *einterval_Icc_approximation*:
fixes *a b* :: *ereal*
assumes *a* < *b*
obtains *u l* :: *nat* ⇒ *real* **where**
einterval a b = (⋃*i*. {*l i* .. *u i*})
incseq u decseq l ⋀*i*. *l i* < *u i* ⋀*i*. *a* < *l i* ⋀*i*. *u i* < *b*

l ⟶ *a u* ⟶ *b*

definition *interval_lebesgue_integral* :: *real measure* ⇒ *ereal* ⇒ *ereal* ⇒ (*real* ⇒ *'a*) ⇒ *'a*::{*banach*, *second_countable_topology*} **where**
interval_lebesgue_integral M a b f =
(*if a* ≤ *b then* (*LINT x*:*einterval a b*|*M*. *f x*) *else* − (*LINT x*:*einterval b a*|*M*. *f x*))

definition *interval_lebesgue_integrable* :: *real measure* ⇒ *ereal* ⇒ *ereal* ⇒ (*real* ⇒ *'a*::{*banach*, *second_countable_topology*}) ⇒ *bool* **where**
interval_lebesgue_integrable M a b f =
(*if a* ≤ *b then set_integrable M* (*einterval a b*) *f else set_integrable M* (*einterval b a*) *f*)

9.10.12 Basic properties of integration over an interval

proposition *interval_integrable_to_infinity_eq*: (*interval_lebesgue_integrable M a* ∞ *f*) =
(*set_integrable M* {*a*<..} *f*)

9.10.13 Basic properties of integration over an interval wrt lebesgue measure

9.10.14 General limit approximation arguments

proposition *interval_integral_Icc_approx_nonneg*:
fixes *a b* :: *ereal*
assumes *a* < *b*
fixes *u l* :: *nat* ⇒ *real*
assumes *approx*: *einterval a b* = (⋃*i*. {*l i* .. *u i*})
incseq u decseq l ⋀*i*. *l i* < *u i* ⋀*i*. *a* < *l i* ⋀*i*. *u i* < *b*
l ⟶ *a u* ⟶ *b*
fixes *f* :: *real* ⇒ *real*
assumes *f_integrable*: ⋀*i*. *set_integrable lborel* {*l i*..*u i*} *f*
assumes *f_nonneg*: *AE x in lborel*. *a* < *ereal x* ⟶ *ereal x* < *b* ⟶ *0* ≤ *f x*
assumes *f_measurable*: *set_borel_measurable lborel* (*einterval a b*) *f*
assumes *lbint_lim*: (λ*i*. *LBINT x*=*l i*.. *u i*. *f x*) ⟶ *C*
shows
set_integrable lborel (*einterval a b*) *f*
(*LBINT x*=*a*..*b*. *f x*) = *C*

proposition *interval_integral_Icc_approx_integrable*:

fixes $u\ l :: nat \Rightarrow real$ **and** $a\ b :: ereal$
fixes $f :: real \Rightarrow 'a::\{banach, second\_countable\_topology\}$
assumes $a < b$
assumes $approx: einterval\ a\ b = (\bigcup i.\ \{l\ i\ ..\ u\ i\})$
$incseq\ u\ decseq\ l\ \bigwedge i.\ l\ i < u\ i\ \bigwedge i.\ a < l\ i\ \bigwedge i.\ u\ i < b$
$l \longrightarrow a\ u \longrightarrow b$
assumes $f\_integrable: set\_integrable\ lborel\ (einterval\ a\ b)\ f$
shows $(\lambda i.\ LBINT\ x{=}l\ i..\ u\ i.\ f\ x) \longrightarrow (LBINT\ x{=}a..b.\ f\ x)$

9.10.15 A slightly stronger Fundamental Theorem of Calculus

theorem *interval_integral_FTC_integrable*:
fixes $f\ F :: real \Rightarrow 'a::euclidean\_space$ **and** $a\ b :: ereal$
assumes $a < b$
assumes $F: \bigwedge x.\ a < ereal\ x \Longrightarrow ereal\ x < b \Longrightarrow (F\ has\_vector\_derivative\ f\ x)$ $(at\ x)$
assumes $f: \bigwedge x.\ a < ereal\ x \Longrightarrow ereal\ x < b \Longrightarrow isCont\ f\ x$
assumes $f\_integrable: set\_integrable\ lborel\ (einterval\ a\ b)\ f$
assumes $A: ((F \circ real\_of\_ereal) \longrightarrow A)\ (at\_right\ a)$
assumes $B: ((F \circ real\_of\_ereal) \longrightarrow B)\ (at\_left\ b)$
shows $(LBINT\ x{=}a..b.\ f\ x) = B - A$

theorem *interval_integral_FTC2*:
fixes $a\ b\ c :: real$ **and** $f :: real \Rightarrow 'a::euclidean\_space$
assumes $a \le c\ c \le b$
and $contf: continuous\_on\ \{a..b\}\ f$
fixes $x :: real$
assumes $a \le x$ **and** $x \le b$
shows $((\lambda u.\ LBINT\ y{=}c..u.\ f\ y)\ has\_vector\_derivative\ (f\ x))\ (at\ x\ within\ \{a..b\})$

proposition *einterval_antiderivative*:
fixes $a\ b :: ereal$ **and** $f :: real \Rightarrow 'a::euclidean\_space$
assumes $a < b$ **and** $contf: \bigwedge x :: real.\ a < x \Longrightarrow x < b \Longrightarrow isCont\ f\ x$
shows $\exists F.\ \forall x :: real.\ a < x \longrightarrow x < b \longrightarrow (F\ has\_vector\_derivative\ f\ x)\ (at\ x)$

9.10.16 The substitution theorem

theorem *interval_integral_substitution_finite*:
fixes $a\ b :: real$ **and** $f :: real \Rightarrow 'a::euclidean\_space$
assumes $a \le b$
and $derivg: \bigwedge x.\ a \le x \Longrightarrow x \le b \Longrightarrow (g\ has\_real\_derivative\ (g'\ x))\ (at\ x\ within$ $\{a..b\})$

and *contf* : *continuous_on* (*g* ‘ {*a*..*b*}) *f*
and *contg'*: *continuous_on* {*a*..*b*} *g'*
shows (*LBINT x*=*a*..*b*. *g' x* $*_R$ *f* (*g x*)) = (*LBINT y*=*g a*..*g b*. *f y*)

theorem *interval_integral_substitution_integrable*:
fixes *f* :: *real* ⇒ *'a*::*euclidean_space* **and** *a b u v* :: *ereal*
assumes *a* < *b*
and *deriv_g*: ⋀*x*. *a* < *ereal x* ⟹ *ereal x* < *b* ⟹ *DERIV g x* :> *g' x*
and *contf*: ⋀*x*. *a* < *ereal x* ⟹ *ereal x* < *b* ⟹ *isCont f* (*g x*)
and *contg'*: ⋀*x*. *a* < *ereal x* ⟹ *ereal x* < *b* ⟹ *isCont g' x*
and *g'_nonneg*: ⋀*x*. *a* ≤ *ereal x* ⟹ *ereal x* ≤ *b* ⟹ *0* ≤ *g' x*
and *A*: ((*ereal* ∘ *g* ∘ *real_of_ereal*) ⟶ *A*) (*at_right a*)
and *B*: ((*ereal* ∘ *g* ∘ *real_of_ereal*) ⟶ *B*) (*at_left b*)
and *integrable*: *set_integrable lborel* (*einterval a b*) (λ*x*. *g' x* $*_R$ *f* (*g x*))
and *integrable2*: *set_integrable lborel* (*einterval A B*) (λ*x*. *f x*)
shows (*LBINT x*=*A*..*B*. *f x*) = (*LBINT x*=*a*..*b*. *g' x* $*_R$ *f* (*g x*))

theorem *interval_integral_substitution_nonneg*:
fixes *f g g'*:: *real* ⇒ *real* **and** *a b u v* :: *ereal*
assumes *a* < *b*
and *deriv_g*: ⋀*x*. *a* < *ereal x* ⟹ *ereal x* < *b* ⟹ *DERIV g x* :> *g' x*
and *contf*: ⋀*x*. *a* < *ereal x* ⟹ *ereal x* < *b* ⟹ *isCont f* (*g x*)
and *contg'*: ⋀*x*. *a* < *ereal x* ⟹ *ereal x* < *b* ⟹ *isCont g' x*
and *f_nonneg*: ⋀*x*. *a* < *ereal x* ⟹ *ereal x* < *b* ⟹ *0* ≤ *f* (*g x*)
and *g'_nonneg*: ⋀*x*. *a* ≤ *ereal x* ⟹ *ereal x* ≤ *b* ⟹ *0* ≤ *g' x*
and *A*: ((*ereal* ∘ *g* ∘ *real_of_ereal*) ⟶ *A*) (*at_right a*)
and *B*: ((*ereal* ∘ *g* ∘ *real_of_ereal*) ⟶ *B*) (*at_left b*)
and *integrable_fg*: *set_integrable lborel* (*einterval a b*) (λ*x*. *f* (*g x*) ∗ *g' x*)
shows
set_integrable lborel (*einterval A B*) *f*
(*LBINT x*=*A*..*B*. *f x*) = (*LBINT x*=*a*..*b*. (*f* (*g x*) ∗ *g' x*))

proposition *interval_integral_norm*:
fixes *f* :: *real* ⇒ *'a* :: {*banach*, *second_countable_topology*}
shows *interval_lebesgue_integrable lborel a b f* ⟹ *a* ≤ *b* ⟹
norm (*LBINT t*=*a*..*b*. *f t*) ≤ *LBINT t*=*a*..*b*. *norm* (*f t*)

proposition *interval_integral_norm2*:
interval_lebesgue_integrable lborel a b f ⟹
norm (*LBINT t*=*a*..*b*. *f t*) ≤ |*LBINT t*=*a*..*b*. *norm* (*f t*)|

end

9.11 Integration by Substition for the Lebesgue Integral

theory *Lebesgue_Integral_Substitution*
imports *Interval_Integral*
begin

theorem *nn_integral_substitution*:
fixes f :: *real* ⇒ *real*
assumes *Mf*[*measurable*]: *set_borel_measurable borel* {*g a..g b*} *f*
assumes *derivg*: $\bigwedge x.\ x \in \{a..b\} \Longrightarrow$ (*g has_real_derivative g′ x*) (*at x*)
assumes *contg′*: *continuous_on* {*a..b*} *g′*
assumes *derivg_nonneg*: $\bigwedge x.\ x \in \{a..b\} \Longrightarrow g'\ x \geq 0$
assumes $a \leq b$
shows ($\int^{+}x.\ f\ x * indicator\ \{g\ a..g\ b\}\ x\ \partial lborel$) =
($\int^{+}x.\ f\ (g\ x) * g'\ x * indicator\ \{a..b\}\ x\ \partial lborel$)

theorem *integral_substitution*:
assumes *integrable*: *set_integrable lborel* {*g a..g b*} *f*
assumes *derivg*: $\bigwedge x.\ x \in \{a..b\} \Longrightarrow$ (*g has_real_derivative g′ x*) (*at x*)
assumes *contg′*: *continuous_on* {*a..b*} *g′*
assumes *derivg_nonneg*: $\bigwedge x.\ x \in \{a..b\} \Longrightarrow g'\ x \geq 0$
assumes $a \leq b$
shows *set_integrable lborel* {*a..b*} ($\lambda x.\ f\ (g\ x) * g'\ x$)
and (*LBINT x. f x* ∗ *indicator* {*g a..g b*} *x*) = (*LBINT x. f* (*g x*) ∗ *g′ x* ∗ *indicator* {*a..b*} *x*)

theorem *interval_integral_substitution*:
assumes *integrable*: *set_integrable lborel* {*g a..g b*} *f*
assumes *derivg*: $\bigwedge x.\ x \in \{a..b\} \Longrightarrow$ (*g has_real_derivative g′ x*) (*at x*)
assumes *contg′*: *continuous_on* {*a..b*} *g′*
assumes *derivg_nonneg*: $\bigwedge x.\ x \in \{a..b\} \Longrightarrow g'\ x \geq 0$
assumes $a \leq b$
shows *set_integrable lborel* {*a..b*} ($\lambda x.\ f\ (g\ x) * g'\ x$)
and (*LBINT x=g a..g b. f x*) = (*LBINT x=a..b. f* (*g x*) ∗ *g′ x*)

end

9.12 The Volume of an n-Dimensional Ball

theory *Ball_Volume*
imports *Gamma_Function Lebesgue_Integral_Substitution*
begindefinition *unit_ball_vol* :: *real* ⇒ *real* **where**
unit_ball_vol n = *pi powr* (*n* / *2*) / *Gamma* (*n* / *2* + *1*)

corollary *content_ball*:
content (*ball c r*) = *unit_ball_vol* (*DIM*(′*a*)) ∗ *r* ^ *DIM*(′*a*)

end

9.13 Integral Test for Summability

theory *Integral_Test*
imports *Henstock_Kurzweil_Integration*
beginlocale *antimono_fun_sum_integral_diff* =
  fixes f :: *real* ⇒ *real*
  assumes *dec*: $\bigwedge x\ y.\ x \geq 0 \Longrightarrow x \leq y \Longrightarrow f\ x \geq f\ y$
  assumes *nonneg*: $\bigwedge x.\ x \geq 0 \Longrightarrow f\ x \geq 0$
  assumes *cont*: *continuous_on* {*0*..} *f*
begin

theorem *integral_test*:
  summable (λ*n*. *f* (*of_nat n*)) ⟷ *convergent* (λ*n*. *integral* {*0*..*of_nat n*} *f*)

end

9.14 Continuity of the indefinite integral; improper integral theorem

theory *Improper_Integral*
  imports *Equivalence_Lebesgue_Henstock_Integration*
begin

9.14.1 Equiintegrability

definition *equiintegrable_on* (**infixr** ‹*equiintegrable′_on*› *46*)
  where *F equiintegrable_on I* ≡
    (∀ *f* ∈ *F*. *f integrable_on I*) ∧
    (∀ *e* > *0*. ∃γ. *gauge* γ ∧
      (∀ *f* 𝒟. *f* ∈ *F* ∧ 𝒟 *tagged_division_of I* ∧ γ *fine* 𝒟
        ⟶ *norm* ((∑ (*x*,*K*) ∈ 𝒟. *content K* $*_R$ *f x*) − *integral I f*)
< *e*))

corollary *equiintegrable_sum_real*:
  fixes *F* :: (*real* ⇒ ′*b*::*euclidean_space*) *set*
  assumes *F equiintegrable_on* {*a*..*b*}
  shows (⋃ *I* ∈ *Collect finite*. ⋃ *c* ∈ {*c*. (∀ *i* ∈ *I*. *c i* ≥ *0*) ∧ *sum c I* = *1*}.
    ⋃ *f* ∈ *I* → *F*. {(λ*x*. *sum* (λ*i*. *c i* $*_R$ *f i x*) *I*)})
    equiintegrable_on {*a*..*b*}
theorem *equiintegrable_limit*:
  fixes *g* :: ′*a* :: *euclidean_space* ⇒ ′*b* :: *banach*

assumes *feq*: *range f equiintegrable_on cbox a b*
and *to_g*: $\bigwedge x.\ x \in cbox\ a\ b \Longrightarrow (\lambda n.\ f\ n\ x) \longrightarrow g\ x$
shows g *integrable_on cbox a b* $\wedge$ $(\lambda n.\ integral\ (cbox\ a\ b)\ (f\ n)) \longrightarrow integral\ (cbox\ a\ b)\ g$

9.14.2 Subinterval restrictions for equiintegrable families

proposition *sum_content_area_over_thin_division*:
assumes *div*: $\mathcal{D}$ *division_of S* **and** *S*: $S \subseteq cbox\ a\ b$ **and** *i*: $i \in Basis$
and $a \cdot i \le c\ c \le b \cdot i$
and *nonmt*: $\bigwedge K.\ K \in \mathcal{D} \Longrightarrow K \cap \{x.\ x \cdot i = c\} \neq \{\}$
shows $(b \cdot i - a \cdot i) * (\sum K{\in}\mathcal{D}.\ content\ K\ /\ (interval\_upperbound\ K \cdot i - interval\_lowerbound\ K \cdot i))$
$\le 2 * content(cbox\ a\ b)$

proposition *bounded_equiintegral_over_thin_tagged_partial_division*:
fixes $f :: 'a::euclidean\_space \Rightarrow 'b::euclidean\_space$
assumes *F*: *F equiintegrable_on cbox a b* **and** *f*: $f \in F$ **and** $0 < \varepsilon$
and *norm_f*: $\bigwedge h\ x.\ [\![h \in F;\ x \in cbox\ a\ b]\!] \Longrightarrow norm(h\ x) \le norm(f\ x)$
obtains γ **where** *gauge* γ
$\bigwedge c\ i\ S\ h.\ [\![c \in cbox\ a\ b;\ i \in Basis;\ S\ tagged\_partial\_division\_of\ cbox\ a\ b;$
$\gamma\ fine\ S;\ h \in F;\ \bigwedge x\ K.\ (x,K) \in S \Longrightarrow (K \cap \{x.\ x \cdot i = c \cdot i\} \neq \{\})]\!]$
$\Longrightarrow (\sum (x,K) \in S.\ norm\ (integral\ K\ h)) < \varepsilon$

proposition *equiintegrable_halfspace_restrictions_le*:
fixes $f :: 'a::euclidean\_space \Rightarrow 'b::euclidean\_space$
assumes *F*: *F equiintegrable_on cbox a b* **and** *f*: $f \in F$
and *norm_f*: $\bigwedge h\ x.\ [\![h \in F;\ x \in cbox\ a\ b]\!] \Longrightarrow norm(h\ x) \le norm(f\ x)$
shows $(\bigcup i \in Basis.\ \bigcup c.\ \bigcup h \in F.\ \{(\lambda x.\ if\ x \cdot i \le c\ then\ h\ x\ else\ 0)\})$
equiintegrable_on cbox a b

corollary *equiintegrable_halfspace_restrictions_ge*:
fixes $f :: 'a::euclidean\_space \Rightarrow 'b::euclidean\_space$
assumes *F*: *F equiintegrable_on cbox a b* **and** *f*: $f \in F$
and *norm_f*: $\bigwedge h\ x.\ [\![h \in F;\ x \in cbox\ a\ b]\!] \Longrightarrow norm(h\ x) \le norm(f\ x)$
shows $(\bigcup i \in Basis.\ \bigcup c.\ \bigcup h \in F.\ \{(\lambda x.\ if\ x \cdot i \ge c\ then\ h\ x\ else\ 0)\})$
equiintegrable_on cbox a b

corollary *equiintegrable_halfspace_restrictions_lt*:
fixes f :: $'a$::*euclidean_space* ⇒ $'b$::*euclidean_space*
assumes F: F *equiintegrable_on cbox a b* **and** f: $f \in F$
and *norm_f*: $\bigwedge h\ x.\ [\![h \in F;\ x \in cbox\ a\ b]\!] \Longrightarrow norm(h\ x) \le norm(f\ x)$
shows $(\bigcup i \in Basis.\ \bigcup c.\ \bigcup h \in F.\ \{(\lambda x.\ if\ x \cdot i < c\ then\ h\ x\ else\ 0)\})$ *equiintegrable_on cbox a b*
(**is** *?G equiintegrable_on cbox a b*)

corollary *equiintegrable_halfspace_restrictions_gt*:
fixes f :: $'a$::*euclidean_space* ⇒ $'b$::*euclidean_space*
assumes F: F *equiintegrable_on cbox a b* **and** f: $f \in F$
and *norm_f*: $\bigwedge h\ x.\ [\![h \in F;\ x \in cbox\ a\ b]\!] \Longrightarrow norm(h\ x) \le norm(f\ x)$
shows $(\bigcup i \in Basis.\ \bigcup c.\ \bigcup h \in F.\ \{(\lambda x.\ if\ x \cdot i > c\ then\ h\ x\ else\ 0)\})$ *equiintegrable_on cbox a b*
(**is** *?G equiintegrable_on cbox a b*)

proposition *equiintegrable_closed_interval_restrictions*:
fixes f :: $'a$::*euclidean_space* ⇒ $'b$::*euclidean_space*
assumes f: f *integrable_on cbox a b*
shows $(\bigcup c\ d.\ \{(\lambda x.\ if\ x \in cbox\ c\ d\ then\ f\ x\ else\ 0)\})$ *equiintegrable_on cbox a b*

9.14.3 Continuity of the indefinite integral

proposition *indefinite_integral_continuous*:
fixes f :: $'a$ :: *euclidean_space* ⇒ $'b$ :: *euclidean_space*
assumes *int_f*: f *integrable_on cbox a b*
and c: $c \in cbox\ a\ b$ **and** d: $d \in cbox\ a\ b$ $0 < \varepsilon$
obtains δ **where** $0 < \delta$
$\bigwedge c'\ d'.\ [\![c' \in cbox\ a\ b;\ d' \in cbox\ a\ b;\ norm(c' - c) \le \delta;\ norm(d' - d) \le \delta]\!]$
$\Longrightarrow norm(integral(cbox\ c'\ d')\ f - integral(cbox\ c\ d)\ f) < \varepsilon$

corollary *indefinite_integral_uniformly_continuous*:
fixes f :: $'a$ :: *euclidean_space* ⇒ $'b$ :: *euclidean_space*
assumes f *integrable_on cbox a b*
shows *uniformly_continuous_on* $(cbox\ (Pair\ a\ a)\ (Pair\ b\ b))\ (\lambda y.\ integral\ (cbox\ (fst\ y)\ (snd\ y))\ f)$

corollary *bounded_integrals_over_subintervals*:
fixes f :: $'a$ :: *euclidean_space* ⇒ $'b$ :: *euclidean_space*
assumes f *integrable_on cbox a b*
shows *bounded* $\{integral\ (cbox\ c\ d)\ f\ |c\ d.\ cbox\ c\ d \subseteq cbox\ a\ b\}$
theorem *absolutely_integrable_improper*:
fixes f :: $'M$::*euclidean_space* ⇒ $'N$::*euclidean_space*
assumes *int_f*: $\bigwedge c\ d.\ cbox\ c\ d \subseteq box\ a\ b \Longrightarrow f$ *integrable_on cbox c d*
and *bo*: *bounded* $\{integral\ (cbox\ c\ d)\ f\ |c\ d.\ cbox\ c\ d \subseteq box\ a\ b\}$
and *absi*: $\bigwedge i.\ i \in Basis$

$\Longrightarrow \exists g.\ g$ *absolutely_integrable_on cbox a b* $\wedge$
$((\forall x \in cbox\ a\ b.\ f\ x \cdot i \leq g\ x) \vee (\forall x \in cbox\ a\ b.\ f\ x \cdot i \geq g\ x))$
shows *f absolutely_integrable_on cbox a b*

9.14.4 Second mean value theorem and corollaries

theorem *second_mean_value_theorem_full*:
fixes f :: *real* $\Rightarrow$ *real*
assumes f: f *integrable_on* $\{a..b\}$ **and** $a \leq b$
and g: $\bigwedge x\ y.\ [\![a \leq x;\ x \leq y;\ y \leq b]\!] \Longrightarrow g\ x \leq g\ y$
obtains c **where** $c \in \{a..b\}$
and $((\lambda x.\ g\ x * f\ x)$ *has_integral* $(g\ a *$ *integral* $\{a..c\}\ f + g\ b *$ *integral* $\{c..b\}$
$f))\ \{a..b\}$

corollary *second_mean_value_theorem*:
fixes f :: *real* $\Rightarrow$ *real*
assumes f: f *integrable_on* $\{a..b\}$ **and** $a \leq b$
and g: $\bigwedge x\ y.\ [\![a \leq x;\ x \leq y;\ y \leq b]\!] \Longrightarrow g\ x \leq g\ y$
obtains c **where** $c \in \{a..b\}$
integral $\{a..b\}\ (\lambda x.\ g\ x * f\ x) = g\ a *$ *integral* $\{a..c\}\ f + g\ b *$ *integral*
$\{c..b\}\ f$

end

9.15 Continuous Extensions of Functions

theory *Continuous_Extension*
imports *Starlike*
begin

9.15.1 Partitions of unity subordinate to locally finite open coverings

proposition *subordinate_partition_of_unity*:
fixes S :: *'a::metric_space set*
assumes $S \subseteq \bigcup \mathcal{C}$ **and** *opC*: $\bigwedge T.\ T \in \mathcal{C} \Longrightarrow$ *open* T
and *fin*: $\bigwedge x.\ x \in S \Longrightarrow \exists V.$ *open* $V \wedge x \in V \wedge$ *finite* $\{U \in \mathcal{C}.\ U \cap V \neq \{\}\}$
obtains F :: $['a\ set, 'a] \Rightarrow$ *real*
where $\bigwedge U.\ U \in \mathcal{C} \Longrightarrow$ *continuous_on* $S\ (F\ U) \wedge (\forall x \in S.\ 0 \leq F\ U\ x)$
and $\bigwedge x\ U.\ [\![U \in \mathcal{C};\ x \in S;\ x \notin U]\!] \Longrightarrow F\ U\ x = 0$
and $\bigwedge x.\ x \in S \Longrightarrow$ *supp_sum* $(\lambda W.\ F\ W\ x)\ \mathcal{C} = 1$
and $\bigwedge x.\ x \in S \Longrightarrow \exists V.$ *open* $V \wedge x \in V \wedge$ *finite* $\{U \in \mathcal{C}.\ \exists x{\in}V.\ F\ U\ x \neq$
$0\}$

9.15.2 Urysohn's Lemma for Euclidean Spaces

proposition *Urysohn_local_strong*:
assumes *US*: *closedin* (*top_of_set U*) *S*
and *UT*: *closedin* (*top_of_set U*) *T*
and $S \cap T = \{\}$ $a \neq b$
obtains *f* :: *'a::euclidean_space* ⇒ *'b::euclidean_space*
where *continuous_on U f*
$\bigwedge x.\ x \in U \Longrightarrow f\ x \in closed\_segment\ a\ b$
$\bigwedge x.\ x \in U \Longrightarrow (f\ x = a \longleftrightarrow x \in S)$
$\bigwedge x.\ x \in U \Longrightarrow (f\ x = b \longleftrightarrow x \in T)$

proposition *Urysohn*:
assumes *US*: *closed S*
and *UT*: *closed T*
and $S \cap T = \{\}$
obtains *f* :: *'a::euclidean_space* ⇒ *'b::euclidean_space*
where *continuous_on UNIV f*
$\bigwedge x.\ f\ x \in closed\_segment\ a\ b$
$\bigwedge x.\ x \in S \Longrightarrow f\ x = a$
$\bigwedge x.\ x \in T \Longrightarrow f\ x = b$

9.15.3 Dugundji's Extension Theorem and Tietze Variants

theorem *Dugundji*:
fixes *f* :: *'a::{metric_space,second_countable_topology}* ⇒ *'b::real_inner*
assumes *convex C* $C \neq \{\}$
and *cloin*: *closedin* (*top_of_set U*) *S*
and *contf*: *continuous_on S f* **and** f ‘ $S \subseteq C$
obtains *g* **where** *continuous_on U g* g ‘ $U \subseteq C$
$\bigwedge x.\ x \in S \Longrightarrow g\ x = f\ x$

corollary *Tietze*:
fixes *f* :: *'a::{metric_space,second_countable_topology}* ⇒ *'b::real_inner*
assumes *continuous_on S f*
and *closedin* (*top_of_set U*) *S*
and $0 \leq B$
and $\bigwedge x.\ x \in S \Longrightarrow norm(f\ x) \leq B$
obtains *g* **where** *continuous_on U g* $\bigwedge x.\ x \in S \Longrightarrow g\ x = f\ x$
$\bigwedge x.\ x \in U \Longrightarrow norm(g\ x) \leq B$

end

9.16 Equivalence Between Classical Borel Measurability and HOL Light's

theory *Equivalence_Measurable_On_Borel*
imports *Equivalence_Lebesgue_Henstock_Integration Improper_Integral Continuous_Extension*
begin

9.16.1 Austin's Lemma

9.16.2 A differentiability-like property of the indefinite integral.

proposition *integrable_ccontinuous_explicit*:
fixes $f :: 'a::euclidean\_space \Rightarrow 'b::euclidean\_space$
assumes $\bigwedge a\ b::'a.\ f\ integrable\_on\ cbox\ a\ b$
obtains N **where**
negligible N
$\bigwedge x\ e.\ [\![x \notin N;\ 0 < e]\!] \Longrightarrow$
$\exists d>0.\ \forall h.\ 0 < h \wedge h < d \longrightarrow$
$norm(integral\ (cbox\ x\ (x + h *_R One))\ f\ /_R\ h\ \hat{}\ DIM('a) - f\ x) < e$

9.16.3 HOL Light measurability

proposition *integrable_subintervals_imp_measurable*:
fixes $f :: 'a::euclidean\_space \Rightarrow 'b::euclidean\_space$
assumes $\bigwedge a\ b.\ f\ integrable\_on\ cbox\ a\ b$
shows *f measurable_on UNIV*

9.16.4 Composing continuous and measurable functions; a few variants

proposition *indicator_measurable_on*:
assumes $S \in sets\ lebesgue$
shows *indicat_real S measurable_on UNIV*

lemma *simple_function_induct_real*
[*consumes 1*, *case_names cong set mult add*, *induct set*: *simple_function*]:
fixes $u :: 'a \Rightarrow real$
assumes *u*: *simple_function M u*

assumes *cong*: $\bigwedge f\ g.\ simple\_function\ M\ f \Longrightarrow simple\_function\ M\ g \Longrightarrow (AE\ x\ in\ M.\ f\ x = g\ x) \Longrightarrow P\ f \Longrightarrow P\ g$
assumes *set*: $\bigwedge A.\ A \in sets\ M \Longrightarrow P\ (indicator\ A)$
assumes *mult*: $\bigwedge u\ c.\ P\ u \Longrightarrow P\ (\lambda x.\ c * u\ x)$
assumes *add*: $\bigwedge u\ v.\ P\ u \Longrightarrow P\ v \Longrightarrow P\ (\lambda x.\ u\ x + v\ x)$
and *nn*: $\bigwedge x.\ u\ x \geq 0$
shows $P\ u$

proposition *simple_function_measurable_on_UNIV*:
fixes $f :: 'a::euclidean\_space \Rightarrow real$
assumes *f*: *simple_function lebesgue f* **and** *nn*: $\bigwedge x.\ f\ x \geq 0$
shows *f measurable_on UNIV*

corollary *simple_function_measurable_on*:
fixes $f :: 'a::euclidean\_space \Rightarrow real$
assumes *f*: *simple_function lebesgue f* **and** *nn*: $\bigwedge x.\ f\ x \geq 0$ **and** *S*: $S \in sets\ lebesgue$
shows *f measurable_on S*

proposition *measurable_on_componentwise_UNIV*:
$f\ measurable\_on\ UNIV \longleftrightarrow (\forall\, i{\in}Basis.\ (\lambda x.\ (f\ x \cdot i) *_R i)\ measurable\_on\ UNIV)$
(**is** *?lhs = ?rhs*)

corollary *measurable_on_componentwise*:
$f\ measurable\_on\ S \longleftrightarrow (\forall\, i{\in}Basis.\ (\lambda x.\ (f\ x \cdot i) *_R i)\ measurable\_on\ S)$

lemma *borel_measurable_implies_simple_function_sequence_real*:
fixes $u :: 'a \Rightarrow real$
assumes *u*[*measurable*]: $u \in borel\_measurable\ M$ **and** *nn*: $\bigwedge x.\ u\ x \geq 0$
shows $\exists f.\ incseq\ f \wedge (\forall\, i.\ simple\_function\ M\ (f\ i)) \wedge (\forall\, x.\ bdd\_above\ (range\ (\lambda i.\ f\ i\ x))) \wedge$
$(\forall\, i\ x.\ 0 \leq f\ i\ x) \wedge u = (SUP\ i.\ f\ i)$

proposition *homeomorphic_box_UNIV*:
fixes $a\ b :: 'a::euclidean\_space$
assumes $box\ a\ b \neq \{\}$
shows $box\ a\ b\ homeomorphic\ (UNIV::'a\ set)$

proposition *measurable_on_imp_borel_measurable_lebesgue_UNIV*:
fixes $f :: 'a::euclidean\_space \Rightarrow 'b::euclidean\_space$
assumes *f measurable_on UNIV*
shows $f \in borel\_measurable\ lebesgue$

corollary *measurable_on_imp_borel_measurable_lebesgue*:
fixes f :: $'a$::*euclidean_space* ⇒ $'b$::*euclidean_space*
assumes f *measurable_on* S **and** S: $S \in$ *sets lebesgue*
shows $f \in$ *borel_measurable* (*lebesgue_on* S)

proposition *measurable_on_limit*:
fixes f :: *nat* ⇒ $'a$::*euclidean_space* ⇒ $'b$::*euclidean_space*
assumes f: $\bigwedge n.\ f\ n$ *measurable_on* S **and** N: *negligible* N
and *lim*: $\bigwedge x.\ x \in S - N \Longrightarrow (\lambda n.\ f\ n\ x) \longrightarrow g\ x$
shows g *measurable_on* S

proposition *lebesgue_measurable_imp_measurable_on*:
fixes f :: $'a$::*euclidean_space* ⇒ $'b$::*euclidean_space*
assumes f: $f \in$ *borel_measurable lebesgue* **and** S: $S \in$ *sets lebesgue*
shows f *measurable_on* S

proposition *measurable_on_iff_borel_measurable*:
fixes f :: $'a$::*euclidean_space* ⇒ $'b$::*euclidean_space*
assumes $S \in$ *sets lebesgue*
shows f *measurable_on* $S \longleftrightarrow f \in$ *borel_measurable* (*lebesgue_on* S) (**is** *?lhs* = *?rhs*)

9.16.5 Monotonic functions are Lebesgue integrable

9.16.6 Measurability on generalisations of the binary product

end

9.17 Embedding Measure Spaces with a Function

theory *Embed_Measure*
imports *Binary_Product_Measure*
begindefinition *embed_measure* :: $'a$ *measure* ⇒ ($'a$ ⇒ $'b$) ⇒ $'b$ *measure* **where**
embed_measure $M\ f$ = *measure_of* (f ‘ *space* M) {f ‘ A |A. $A \in$ *sets* M}
(λA. *emeasure* M (f −‘ $A \cap$ *space* M))

end

9.18 Brouwer's Fixed Point Theorem

theory *Brouwer_Fixpoint*
imports *Homeomorphism Derivative*
begin

9.18.1 Retractions

9.18.2 Kuhn Simplices

9.18.3 Brouwer's fixed point theorem

theorem *brouwer*:
fixes *f* :: *'a::euclidean_space* ⇒ *'a*
assumes *S*: *compact S convex S S* ≠ {}
and *contf*: *continuous_on S f*
and *fim*: *f* ∈ *S* → *S*
obtains *x* **where** *x* ∈ *S* **and** *f x* = *x*

9.18.4 Applications

corollary *no_retraction_cball*:
fixes *a* :: *'a::euclidean_space*
assumes *e* > *0*
shows ¬ (*frontier* (*cball a e*) *retract_of* (*cball a e*))

corollary *contractible_sphere*:
fixes *a* :: *'a::euclidean_space*
shows *contractible*(*sphere a r*) ⟷ *r* ≤ *0*

corollary *connected_sphere_eq*:
fixes *a* :: *'a* :: *euclidean_space*
shows *connected*(*sphere a r*) ⟷ *2* ≤ *DIM*(*'a*) ∨ *r* ≤ *0*
(**is** *?lhs* = *?rhs*)

corollary *path_connected_sphere_eq*:
fixes *a* :: *'a* :: *euclidean_space*
shows *path_connected*(*sphere a r*) ⟷ *2* ≤ *DIM*(*'a*) ∨ *r* ≤ *0*
(**is** *?lhs* = *?rhs*)

proposition *frontier_subset_retraction*:
fixes *S* :: *'a::euclidean_space set*
assumes *bounded S* **and** *fros*: *frontier S* ⊆ *T*
and *contf*: *continuous_on* (*closure S*) *f*
and *fim*: *f* ∈ *S* → *T*
and *fid*: ⋀*x*. *x* ∈ *T* ⟹ *f x* = *x*

shows $S \subseteq T$

corollary *rel_frontier_retract_of_punctured_affine_hull*:
fixes S :: *'a::euclidean_space set*
assumes *bounded S convex S* $a \in$ *rel_interior S*
shows *rel_frontier S retract_of* (*affine hull S* $-$ $\{a\}$)

corollary *rel_boundary_retract_of_punctured_affine_hull*:
fixes S :: *'a::euclidean_space set*
assumes *compact S convex S* $a \in$ *rel_interior S*
shows (S $-$ *rel_interior S*) *retract_of* (*affine hull S* $-$ $\{a\}$)
theorem *has_derivative_inverse_on*:
fixes f :: *'n::euclidean_space* $\Rightarrow$ *'n*
assumes *open S*
and $\bigwedge x.\ x \in S \Longrightarrow$ (*f has_derivative* $f'(x)$) (*at x*)
and $\bigwedge x.\ x \in S \Longrightarrow g\ (f\ x) = x$
and $f'\ x \circ g'\ x = id$
and $x \in S$
shows (*g has_derivative* $g'(x)$) (*at* (*f x*))

end

9.19 Fashoda Meet Theorem

theory *Fashoda_Theorem*
imports *Brouwer_Fixpoint Path_Connected Cartesian_Euclidean_Space*
begin

9.19.1 Bijections between intervals

definition *interval_bij* :: *'a* $\times$ *'a* $\Rightarrow$ *'a* $\times$ *'a* $\Rightarrow$ *'a* $\Rightarrow$ *'a::euclidean_space*
where *interval_bij* =
($\lambda(a,\ b)\ (u,\ v)\ x.\ (\sum i{\in}Basis.\ (u{\cdot}i + (x{\cdot}i - a{\cdot}i) / (b{\cdot}i - a{\cdot}i) * (v{\cdot}i - u{\cdot}i))$
$*_R\ i))$

9.19.2 Fashoda meet theorem

proposition *fashoda_unit*:
fixes *f g* :: *real* $\Rightarrow$ *real^2*
assumes *f* ‘ $\{-1\ ..\ 1\} \subseteq$ *cbox* $(-1)\ 1$
and *g* ‘ $\{-1\ ..\ 1\} \subseteq$ *cbox* $(-1)\ 1$
and *continuous_on* $\{-1\ ..\ 1\}$ *f*
and *continuous_on* $\{-1\ ..\ 1\}$ *g*
and $f\ (-\ 1)\$1 = -\ 1$
and $f\ 1\$1 = 1\ g\ (-\ 1)\ \$2 = -1$
and $g\ 1\ \$2 = 1$

shows $\exists\, s\in\{-1\ ..\ 1\}.\ \exists\, t\in\{-1\ ..\ 1\}.\ f\ s = g\ t$

proposition *fashoda_unit_path*:
fixes *f g* :: *real* ⇒ *real*^*2*
assumes *path f*
and *path g*
and *path_image f* ⊆ *cbox* (−*1*) *1*
and *path_image g* ⊆ *cbox* (−*1*) *1*
and (*pathstart f*)$*1* = −*1*
and (*pathfinish f*)$*1* = *1*
and (*pathstart g*)$*2* = −*1*
and (*pathfinish g*)$*2* = *1*
obtains *z* **where** *z* ∈ *path_image f* **and** *z* ∈ *path_image g*

theorem *fashoda*:
fixes *b* :: *real*^*2*
assumes *path f*
and *path g*
and *path_image f* ⊆ *cbox a b*
and *path_image g* ⊆ *cbox a b*
and (*pathstart f*)$*1* = *a*$*1*
and (*pathfinish f*)$*1* = *b*$*1*
and (*pathstart g*)$*2* = *a*$*2*
and (*pathfinish g*)$*2* = *b*$*2*
obtains *z* **where** *z* ∈ *path_image f* **and** *z* ∈ *path_image g*

9.19.3 Useful Fashoda corollary pointed out to me by Tom Hales

corollary *fashoda_interlace*:
fixes *a* :: *real*^*2*
assumes *path f*
and *path g*
and *paf*: *path_image f* ⊆ *cbox a b*
and *pag*: *path_image g* ⊆ *cbox a b*
and (*pathstart f*)$*2* = *a*$*2*
and (*pathfinish f*)$*2* = *a*$*2*
and (*pathstart g*)$*2* = *a*$*2*
and (*pathfinish g*)$*2* = *a*$*2*
and (*pathstart f*)$*1* < (*pathstart g*)$*1*
and (*pathstart g*)$*1* < (*pathfinish f*)$*1*
and (*pathfinish f*)$*1* < (*pathfinish g*)$*1*
obtains *z* **where** *z* ∈ *path_image f* **and** *z* ∈ *path_image g*

end

9.20 Vector Cross Products in 3 Dimensions

theory *Cross3*
  imports *Determinants Cartesian_Euclidean_Space*
begin

definition *cross3* :: [*real^3*, *real^3*] ⇒ *real^3* (**infixr** ‹×› *80*)
  where $a \times b \equiv$
    vector [*a$2* ∗ *b$3* − *a$3* ∗ *b$2*,
        a$3 ∗ *b$1* − *a$1* ∗ *b$3*,
        a$1 ∗ *b$2* − *a$2* ∗ *b$1*]

9.20.1 Basic lemmas

proposition *Jacobi*: $x \times (y \times z) + y \times (z \times x) + z \times (x \times y) = 0$ **for** *x*::*real^3*

proposition *Lagrange*: $x \times (y \times z) = (x \cdot z) *_R y - (x \cdot y) *_R z$

proposition *cross_triple*: $(x \times y) \cdot z = (y \times z) \cdot x$

proposition *dot_cross*: $(w \times x) \cdot (y \times z) = (w \cdot y) * (x \cdot z) - (w \cdot z) * (x \cdot y)$

proposition *norm_cross*: $(norm\ (x \times y))^2 = (norm\ x)^2 * (norm\ y)^2 - (x \cdot y)^2$

9.20.2 Preservation by rotation, or other orthogonal transformation up to sign

9.20.3 Continuity

end

9.21 Bounded Continuous Functions

theory *Bounded_Continuous_Function*
  imports
    Topology_Euclidean_Space
    Uniform_Limit
begin

9.21.1 Definition

definition *bcontfun* = {*f*. *continuous_on UNIV f* ∧ *bounded* (*range f*)}

instantiation *bcontfun* :: (*topological_space*, *metric_space*) *metric_space*

begin

lift_definition *dist_bcontfun* :: $'a \Rightarrow_C 'b \Rightarrow 'a \Rightarrow_C 'b \Rightarrow real$
is $\lambda f\ g.\ (SUP\ x.\ dist\ (f\ x)\ (g\ x))$

9.21.2 Complete Space

instance *bcontfun* :: (*metric_space*, *complete_space*) *complete_space*

end

9.22 Infinite Products

theory *Infinite_Products*
imports *Topology_Euclidean_Space Complex_Transcendental*
begin

9.22.1 Definitions and basic properties

definition *raw_has_prod* :: [*nat* ⇒ *'a*::{*t2_space*, *comm_semiring_1*}, *nat*, *'a*] ⇒ *bool*
where *raw_has_prod f M p* ≡ $(\lambda n.\ \prod i{\le}n.\ f\ (i{+}M)) \longrightarrow p \wedge p \neq 0$

definition
has_prod :: (*nat* ⇒ *'a*::{*t2_space*, *comm_semiring_1*}) ⇒ *'a* ⇒ *bool* (**infixr** ‹*has'_prod*› *80*)
where *f has_prod p* ≡ *raw_has_prod f 0 p* ∨ (∃ *i q*. *p* = *0* ∧ *f i* = *0* ∧ *raw_has_prod f* (*Suc i*) *q*)

definition *convergent_prod* :: (*nat* ⇒ *'a* :: {*t2_space*,*comm_semiring_1*}) ⇒ *bool*
where
convergent_prod f ≡ ∃ *M p*. *raw_has_prod f M p*

definition *prodinf* :: (*nat* ⇒ *'a*::{*t2_space*, *comm_semiring_1*}) ⇒ *'a*
(**binder** ‹∏› *10*)
where *prodinf f* = (*THE p*. *f has_prod p*)

9.22.2 Absolutely convergent products

definition *abs_convergent_prod* :: (*nat* ⇒ _) ⇒ *bool* **where**
abs_convergent_prod f ⟷ *convergent_prod* $(\lambda i.\ 1 + norm\ (f\ i - 1))$

lemma *convergent_prod_iff_convergent*:
fixes *f* :: *nat* ⇒ *'a* :: {*topological_semigroup_mult*,*t2_space*,*idom*}
assumes $\bigwedge i.\ f\ i \neq 0$
shows *convergent_prod f* ⟷ *convergent* $(\lambda n.\ \prod i{\le}n.\ f\ i) \wedge lim\ (\lambda n.\ \prod i{\le}n.\ f\ i) \neq 0$

theorem *abs_convergent_prod_conv_summable*:
fixes f :: *nat* ⇒ *'a* :: *real_normed_div_algebra*
shows *abs_convergent_prod* $f \longleftrightarrow$ *summable* ($\lambda i.$ *norm* ($f\ i - 1$))

9.22.3 More elementary properties

theorem *abs_convergent_prod_imp_convergent_prod*:
fixes f :: *nat* ⇒ *'a* :: {*real_normed_div_algebra*,*complete_space*,*comm_ring_1*}
assumes *abs_convergent_prod* f
shows *convergent_prod* f

corollary *convergent_prod_offset_0*:
fixes f :: *nat* ⇒ *'a* :: {*idom*,*topological_semigroup_mult*,*t2_space*}
assumes *convergent_prod* f $\bigwedge i.\ f\ i \neq 0$
shows $\exists p.$ *raw_has_prod* f 0 p

theorem *has_prod_iff*: f *has_prod* $x \longleftrightarrow$ *convergent_prod* $f \wedge$ *prodinf* $f = x$

9.22.4 Exponentials and logarithms

theorem *convergent_prod_iff_summable_real*:
fixes a :: *nat* ⇒ *real*
assumes $\bigwedge n.\ a\ n > 0$
shows *convergent_prod* ($\lambda k.\ 1 + a\ k$) $\longleftrightarrow$ *summable* a (**is** *?lhs* = *?rhs*)

theorem *Ln_prodinf_complex*:
fixes z :: *nat* ⇒ *complex*
assumes z: $\bigwedge j.\ z\ j \neq 0$ **and** ξ: $\xi \neq 0$
shows (($\lambda n.\ \prod j\leq n.\ z\ j$) $\longrightarrow \xi$) $\longleftrightarrow$ ($\exists k.$ ($\lambda n.$ ($\sum j\leq n.$ *Ln* ($z\ j$))) $\longrightarrow$ *Ln* ξ + *of_int* k ∗ (*of_real*(2∗*pi*) ∗ i)) (**is** *?lhs* = *?rhs*)
proposition *convergent_prod_iff_summable_complex*:
fixes z :: *nat* ⇒ *complex*
assumes $\bigwedge k.\ z\ k \neq 0$
shows *convergent_prod* ($\lambda k.\ z\ k$) $\longleftrightarrow$ *summable* ($\lambda k.$ *Ln* ($z\ k$)) (**is** *?lhs* = *?rhs*)
proposition *summable_imp_convergent_prod_complex*:
fixes z :: *nat* ⇒ *complex*
assumes z: *summable* ($\lambda k.$ *norm* ($z\ k$)) **and** *non0*: $\bigwedge k.\ z\ k \neq -1$
shows *convergent_prod* ($\lambda k.\ 1 + z\ k$)

corollary *summable_imp_convergent_prod_real*:
fixes z :: *nat* ⇒ *real*
assumes z: *summable* ($\lambda k.\ |z\ k|$) **and** *non0*: $\bigwedge k.\ z\ k \neq -1$
shows *convergent_prod* ($\lambda k.\ 1 + z\ k$)

end

9.23 Sums over Infinite Sets

theory *Infinite_Set_Sum*
imports *Set_Integral Infinite_Sum*
begin

definition *abs_summable_on* ::
('a ⇒ 'b :: {banach, second_countable_topology}) ⇒ 'a set ⇒ bool
(**infix** ‹abs'_summable'_on› 50)
where
f abs_summable_on A ⟷ integrable (count_space A) f

definition *infsetsum* ::
('a ⇒ 'b :: {banach, second_countable_topology}) ⇒ 'a set ⇒ 'b
where
infsetsum f A = lebesgue_integral (count_space A) f

theorem *infsetsum_reindex*:
assumes inj_on g A
shows infsetsum f (g ‘ A) = infsetsum (λx. f (g x)) A

theorem *infsetsum_Sigma*:
fixes A :: 'a set **and** B :: 'a ⇒ 'b set
assumes [simp]: countable A **and** ⋀i. countable (B i)
assumes summable: f abs_summable_on (Sigma A B)
shows infsetsum f (Sigma A B) = infsetsum (λx. infsetsum (λy. f (x, y)) (B x)) A

theorem *abs_summable_on_Sigma_iff*:
assumes [simp]: countable A **and** ⋀x. x ∈ A ⟹ countable (B x)
shows f abs_summable_on Sigma A B ⟷
(∀x∈A. (λy. f (x, y)) abs_summable_on B x) ∧
((λx. infsetsum (λy. norm (f (x, y))) (B x)) abs_summable_on A)

theorem *infsetsum_prod_PiE*:
fixes f :: 'a ⇒ 'b ⇒ 'c :: {real_normed_field,banach,second_countable_topology}
assumes finite: finite A **and** countable: ⋀x. x ∈ A ⟹ countable (B x)
assumes summable: ⋀x. x ∈ A ⟹ f x abs_summable_on B x
shows infsetsum (λg. ∏x∈A. f x (g x)) (PiE A B) = (∏x∈A. infsetsum (f x) (B x))

end

9.24 Faces, Extreme Points, Polytopes, Polyhedra etc

theory *Polytope*
imports *Cartesian_Euclidean_Space Path_Connected*
begin

9.24.1 Faces of a (usually convex) set

definition *face_of* :: $[{'a}::real\_vector\ set, {'a}\ set] \Rightarrow bool$ (**infixr** ‹(face'_of)› *50*)
where
$T\ face\_of\ S \longleftrightarrow$
$T \subseteq S \wedge convex\ T \wedge$
$(\forall a \in S. \forall b \in S. \forall x \in T. x \in open\_segment\ a\ b \longrightarrow a \in T \wedge b \in T)$

proposition *face_of_imp_eq_affine_Int*:
fixes S :: $'a::euclidean\_space\ set$
assumes S: $convex\ S$ **and** T: $T\ face\_of\ S$
shows $T = (affine\ hull\ T) \cap S$

proposition *face_of_conic*:
assumes $conic\ S\ f\ face\_of\ S$
shows $conic\ f$

proposition *face_of_convex_hulls*:
assumes S: $finite\ S\ T \subseteq S$ **and** $disj$: $affine\ hull\ T \cap convex\ hull\ (S - T) = \{\}$
shows $(convex\ hull\ T)\ face\_of\ (convex\ hull\ S)$

proposition *face_of_convex_hull_insert*:
assumes $finite\ S\ a \notin affine\ hull\ S$ **and** T: $T\ face\_of\ convex\ hull\ S$
shows $T\ face\_of\ convex\ hull\ insert\ a\ S$

proposition *face_of_affine_trivial*:
assumes $affine\ S\ T\ face\_of\ S$
shows $T = \{\} \vee T = S$

proposition *Inter_faces_finite_altbound*:
fixes T :: $'a::euclidean\_space\ set\ set$
assumes $cfaI$: $\bigwedge c.\ c \in T \Longrightarrow c\ face\_of\ S$
shows $\exists F'.\ finite\ F' \wedge F' \subseteq T \wedge card\ F' \leq DIM('a) + 2 \wedge \bigcap F' = \bigcap T$

proposition *face_of_Times*:
assumes F *face_of* S **and** F' *face_of* S'
shows $(F \times F')$ *face_of* $(S \times S')$

corollary *face_of_Times_decomp*:
fixes S :: *'a::euclidean_space set* **and** S' :: *'b::euclidean_space set*
shows C *face_of* $(S \times S') \longleftrightarrow (\exists F\ F'.\ F$ *face_of* $S \land F'$ *face_of* $S' \land C = F \times F')$
(**is** *?lhs* = *?rhs*)

9.24.2 Exposed faces

definition *exposed_face_of* :: ['a::euclidean_space set, 'a set] ⇒ bool
(**infixr** ‹(exposed'_face'_of)› 50)
where T *exposed_face_of* $S \longleftrightarrow$
T *face_of* $S \land (\exists a\ b.\ S \subseteq \{x.\ a \cdot x \le b\} \land T = S \cap \{x.\ a \cdot x = b\})$

proposition *exposed_face_of_Int*:
assumes T *exposed_face_of* S
and U *exposed_face_of* S
shows $(T \cap U)$ *exposed_face_of* S

proposition *exposed_face_of_Inter*:
fixes P :: *'a::euclidean_space set set*
assumes $P \ne \{\}$
and $\bigwedge T.\ T \in P \Longrightarrow T$ *exposed_face_of* S
shows $\bigcap P$ *exposed_face_of* S

proposition *exposed_face_of_sums*:
assumes *convex* S **and** *convex* T
and F *exposed_face_of* $\{x + y \mid x\ y.\ x \in S \land y \in T\}$
(**is** F *exposed_face_of* *?ST*)
obtains $k\ l$
where k *exposed_face_of* S l *exposed_face_of* T
$F = \{x + y \mid x\ y.\ x \in k \land y \in l\}$

proposition *exposed_face_of_parallel*:
T *exposed_face_of* $S \longleftrightarrow$
T *face_of* $S \land$
$(\exists a\ b.\ S \subseteq \{x.\ a \cdot x \le b\} \land T = S \cap \{x.\ a \cdot x = b\} \land$
$(T \ne \{\} \longrightarrow T \ne S \longrightarrow a \ne 0) \land$
$(T \ne S \longrightarrow (\forall w \in$ *affine hull* $S.\ (w + a) \in$ *affine hull* $S)))$
(**is** *?lhs* = *?rhs*)

9.24.3 Extreme points of a set: its singleton faces

definition *extreme_point_of* :: ['a::real_vector, 'a set] ⇒ bool
(**infixr** ‹(extreme'_point'_of)› 50)

where x *extreme_point_of* $S \longleftrightarrow$
$x \in S \land (\forall a \in S.\ \forall b \in S.\ x \notin$ *open_segment* $a\ b)$

proposition *extreme_points_of_convex_hull*:
$\{x.\ x$ *extreme_point_of* (*convex hull* $S)\} \subseteq S$

9.24.4 Facets

definition *facet_of* :: ['a::*euclidean_space set*, 'a *set*] ⇒ *bool*
(**infixr** ‹(*facet'_of*)› *50*)
where F *facet_of* $S \longleftrightarrow F$ *face_of* $S \land F \neq \{\} \land$ *aff_dim* $F =$ *aff_dim* $S - 1$

9.24.5 Edges: faces of affine dimension 1

definition *edge_of* :: ['a::*euclidean_space set*, 'a *set*] ⇒ *bool* (**infixr** ‹(*edge'_of*)› *50*)
where e *edge_of* $S \longleftrightarrow e$ *face_of* $S \land$ *aff_dim* $e = 1$

9.24.6 Existence of extreme points

proposition *different_norm_3_collinear_points*:
fixes a :: 'a::*euclidean_space*
assumes $x \in$ *open_segment* $a\ b$ *norm*$(a) =$ *norm*(b) *norm*$(x) =$ *norm*(b)
shows *False*

proposition *extreme_point_exists_convex*:
fixes S :: 'a::*euclidean_space set*
assumes *compact* S *convex* S $S \neq \{\}$
obtains x **where** x *extreme_point_of* S

9.24.7 Krein-Milman, the weaker form

proposition *Krein_Milman*:
fixes S :: 'a::*euclidean_space set*
assumes *compact* S *convex* S
shows $S =$ *closure*(*convex hull* $\{x.\ x$ *extreme_point_of* $S\})$

theorem *Krein_Milman_Minkowski*:
fixes S :: 'a::*euclidean_space set*
assumes *compact* S *convex* S
shows $S =$ *convex hull* $\{x.\ x$ *extreme_point_of* $S\}$

9.24.8 Applying it to convex hulls of explicitly indicated finite sets

corollary *Krein_Milman_polytope*:
fixes *S* :: *'a*::*euclidean_space set*
shows
finite S
$\Longrightarrow$ *convex hull S* =
convex hull {*x. x extreme_point_of* (*convex hull S*)}

proposition *face_of_convex_hull_insert_eq*:
fixes *a* :: *'a* :: *euclidean_space*
assumes *finite S* **and** *a*: *a* ∉ *affine hull S*
shows (*F face_of* (*convex hull* (*insert a S*)) ⟷
F face_of (*convex hull S*) ∨
(∃ *F'*. *F' face_of* (*convex hull S*) ∧ *F* = *convex hull* (*insert a F'*)))
(**is** *F face_of ?CAS* ⟷ _)

proposition *face_of_convex_hull_affine_independent*:
fixes *S* :: *'a*::*euclidean_space set*
assumes ¬ *affine_dependent S*
shows (*T face_of* (*convex hull S*) ⟷ (∃ *c*. *c* ⊆ *S* ∧ *T* = *convex hull c*))
(**is** *?lhs* = *?rhs*)

proposition *Krein_Milman_frontier*:
fixes *S* :: *'a*::*euclidean_space set*
assumes *convex S compact S*
shows *S* = *convex hull* (*frontier S*)
(**is** *?lhs* = *?rhs*)

9.24.9 Polytopes

definition *polytope* **where**
polytope S ≡ ∃ *v*. *finite v* ∧ *S* = *convex hull v*

proposition *face_of_polytope_insert2*:
fixes *a* :: *'a* :: *euclidean_space*
assumes *polytope S a* ∉ *affine hull S F face_of S*
shows *convex hull* (*insert a F*) *face_of convex hull* (*insert a S*)

9.24.10 Polyhedra

definition *polyhedron* **where**
polyhedron S ≡
∃ *F*. *finite F* ∧

$S = \bigcap F \wedge$
$(\forall h \in F.\ \exists a\ b.\ a \neq 0 \wedge h = \{x.\ a \cdot x \leq b\})$

9.24.11 Canonical polyhedron representation making facial structure explicit

proposition *polyhedron_Int_affine*:
fixes S :: $'a$ :: *euclidean_space set*
shows *polyhedron* $S \longleftrightarrow$
$(\exists F.\ finite\ F \wedge S = (affine\ hull\ S) \cap \bigcap F \wedge$
$(\forall h \in F.\ \exists a\ b.\ a \neq 0 \wedge h = \{x.\ a \cdot x \leq b\}))$

proposition *rel_interior_polyhedron_explicit*:
assumes *finite F*
and *seq*: $S = affine\ hull\ S \cap \bigcap F$
and *faceq*: $\bigwedge h.\ h \in F \Longrightarrow a\ h \neq 0 \wedge h = \{x.\ a\ h \cdot x \leq b\ h\}$
and *psub*: $\bigwedge F'.\ F' \subset F \Longrightarrow S \subset affine\ hull\ S \cap \bigcap F'$
shows $rel\_interior\ S = \{x \in S.\ \forall h \in F.\ a\ h \cdot x < b\ h\}$

proposition *polyhedron_Int_affine_parallel_minimal*:
fixes S :: $'a$ :: *euclidean_space set*
shows *polyhedron* $S \longleftrightarrow$
$(\exists F.\ finite\ F \wedge$
$S = (affine\ hull\ S) \cap (\bigcap F) \wedge$
$(\forall h \in F.\ \exists a\ b.\ a \neq 0 \wedge h = \{x.\ a \cdot x \leq b\} \wedge$
$(\forall x \in affine\ hull\ S.\ (x + a) \in affine\ hull\ S)) \wedge$
$(\forall F'.\ F' \subset F \longrightarrow S \subset (affine\ hull\ S) \cap (\bigcap F')))$
(**is** *?lhs* = *?rhs*)

proposition *facet_of_polyhedron_explicit*:
assumes *finite F*
and *seq*: $S = affine\ hull\ S \cap \bigcap F$
and *faceq*: $\bigwedge h.\ h \in F \Longrightarrow a\ h \neq 0 \wedge h = \{x.\ a\ h \cdot x \leq b\ h\}$
and *psub*: $\bigwedge F'.\ F' \subset F \Longrightarrow S \subset affine\ hull\ S \cap \bigcap F'$
shows $C\ facet\_of\ S \longleftrightarrow (\exists h.\ h \in F \wedge C = S \cap \{x.\ a\ h \cdot x = b\ h\})$

proposition *face_of_polyhedron_explicit*:
fixes S :: $'a$ :: *euclidean_space set*
assumes *finite F*
and *seq*: $S = affine\ hull\ S \cap \bigcap F$
and *faceq*: $\bigwedge h.\ h \in F \Longrightarrow a\ h \neq 0 \wedge h = \{x.\ a\ h \cdot x \leq b\ h\}$
and *psub*: $\bigwedge F'.\ F' \subset F \Longrightarrow S \subset affine\ hull\ S \cap \bigcap F'$
and *C*: $C\ face\_of\ S$ **and** $C \neq \{\}$ $C \neq S$
shows $C = \bigcap \{S \cap \{x.\ a\ h \cdot x = b\ h\} \mid h.\ h \in F \wedge C \subseteq S \cap \{x.\ a\ h \cdot x = b$

h}}

9.24.12 More general corollaries from the explicit representation

corollary *facet_of_polyhedron*:
assumes *polyhedron S* **and** *C facet_of S*
obtains *a b* **where** $a \neq 0$ $S \subseteq \{x.\ a \cdot x \leq b\}$ $C = S \cap \{x.\ a \cdot x = b\}$

corollary *face_of_polyhedron*:
assumes *polyhedron S* **and** *C face_of S* **and** $C \neq \{\}$ **and** $C \neq S$
shows $C = \bigcap\{F.\ F\ \mathit{facet\_of}\ S \wedge C \subseteq F\}$

proposition *rel_interior_of_polyhedron*:
fixes *S* :: *'a* :: *euclidean_space set*
assumes *polyhedron S*
shows *rel_interior S* $= S - \bigcup\{F.\ F\ \mathit{facet\_of}\ S\}$
proposition *polyhedron_eq_finite_exposed_faces*:
fixes *S* :: *'a* :: *euclidean_space set*
shows *polyhedron S* $\longleftrightarrow$ *closed S* $\wedge$ *convex S* $\wedge$ *finite* $\{F.\ F\ \mathit{exposed\_face\_of}\ S\}$
(**is** *?lhs* = *?rhs*)

corollary *polyhedron_eq_finite_faces*:
fixes *S* :: *'a* :: *euclidean_space set*
shows *polyhedron S* $\longleftrightarrow$ *closed S* $\wedge$ *convex S* $\wedge$ *finite* $\{F.\ F\ \mathit{face\_of}\ S\}$
(**is** *?lhs* = *?rhs*)

9.24.13 Relation between polytopes and polyhedra

proposition *polytope_eq_bounded_polyhedron*:
fixes *S* :: *'a* :: *euclidean_space set*
shows *polytope S* $\longleftrightarrow$ *polyhedron S* $\wedge$ *bounded S*
(**is** *?lhs* = *?rhs*)

9.24.14 Relative and absolute frontier of a polytope

proposition *frontier_of_convex_hull*:
fixes *S* :: *'a::euclidean_space set*
assumes *card S* = *Suc* (*DIM*(*'a*))
shows *frontier*(*convex hull S*) $= \bigcup$ $\{\mathit{convex\ hull}\ (S - \{a\})\ |\ a.\ a \in S\}$

9.24.15 Special case of a triangle

proposition *frontier_of_triangle*:
  fixes $a :: 'a::euclidean\_space$
  assumes $DIM('a) = 2$
  shows $frontier(convex\ hull\ \{a,b,c\}) = closed\_segment\ a\ b \cup closed\_segment\ b\ c \cup closed\_segment\ c\ a$
    (**is** $?lhs = ?rhs$)

corollary *inside_of_triangle*:
  fixes $a :: 'a::euclidean\_space$
  assumes $DIM('a) = 2$
  shows $inside\ (closed\_segment\ a\ b \cup closed\_segment\ b\ c \cup closed\_segment\ c\ a) = interior(convex\ hull\ \{a,b,c\})$

corollary *interior_of_triangle*:
  fixes $a :: 'a::euclidean\_space$
  assumes $DIM('a) = 2$
  shows $interior(convex\ hull\ \{a,b,c\}) = convex\ hull\ \{a,b,c\} - (closed\_segment\ a\ b \cup closed\_segment\ b\ c \cup closed\_segment\ c\ a)$

9.24.16 Subdividing a cell complex

proposition *cell_complex_subdivision_exists*:
 fixes $\mathcal{F} :: 'a::euclidean\_space\ set\ set$
 assumes $0 < e$ $finite\ \mathcal{F}$
  and *poly*: $\bigwedge X.\ X \in \mathcal{F} \Longrightarrow polytope\ X$
  and *aff*: $\bigwedge X.\ X \in \mathcal{F} \Longrightarrow aff\_dim\ X \le d$
  and *face*: $\bigwedge X\ Y.\ [\![X \in \mathcal{F};\ Y \in \mathcal{F}]\!] \Longrightarrow X \cap Y\ face\_of\ X$
 obtains $\mathcal{F}'$ **where** $finite\ \mathcal{F}'$ $\bigcup\mathcal{F}' = \bigcup\mathcal{F}$ $\bigwedge X.\ X \in \mathcal{F}' \Longrightarrow diameter\ X < e$
   $\bigwedge X.\ X \in \mathcal{F}' \Longrightarrow polytope\ X$ $\bigwedge X.\ X \in \mathcal{F}' \Longrightarrow aff\_dim\ X \le d$
   $\bigwedge X\ Y.\ [\![X \in \mathcal{F}';\ Y \in \mathcal{F}']\!] \Longrightarrow X \cap Y\ face\_of\ X$
   $\bigwedge C.\ C \in \mathcal{F}' \Longrightarrow \exists D.\ D \in \mathcal{F} \wedge C \subseteq D$
   $\bigwedge C\ x.\ C \in \mathcal{F} \wedge x \in C \Longrightarrow \exists D.\ D \in \mathcal{F}' \wedge x \in D \wedge D \subseteq C$

9.24.17 Simplexes

definition $simplex :: int \Rightarrow 'a::euclidean\_space\ set \Rightarrow bool$ (**infix** ‹*simplex*› *50*)
 where $n\ simplex\ S \equiv \exists C.\ \neg\ affine\_dependent\ C \wedge int(card\ C) = n + 1 \wedge S = convex\ hull\ C$

9.24.18 Simplicial complexes and triangulations

definition *simplicial_complex* **where**
simplicial_complex $\mathcal{C} \equiv$
finite $\mathcal{C} \wedge$
$(\forall S \in \mathcal{C}.\ \exists n.\ n$ *simplex* $S) \wedge$
$(\forall F\ S.\ S \in \mathcal{C} \wedge F$ *face_of* $S \longrightarrow F \in \mathcal{C}) \wedge$
$(\forall S\ S'.\ S \in \mathcal{C} \wedge S' \in \mathcal{C} \longrightarrow (S \cap S')$ *face_of* $S)$

definition *triangulation* **where**
triangulation $\mathcal{T} \equiv$
finite $\mathcal{T} \wedge$
$(\forall T \in \mathcal{T}.\ \exists n.\ n$ *simplex* $T) \wedge$
$(\forall T\ T'.\ T \in \mathcal{T} \wedge T' \in \mathcal{T} \longrightarrow (T \cap T')$ *face_of* $T)$

9.24.19 Refining a cell complex to a simplicial complex

proposition *convex_hull_insert_Int_eq*:
fixes z :: *'a* :: *euclidean_space*
assumes z: $z \in$ *rel_interior* S
and T: $T \subseteq$ *rel_frontier* S
and U: $U \subseteq$ *rel_frontier* S
and *convex* S *convex* T *convex* U
shows *convex hull* (*insert* z T) $\cap$ *convex hull* (*insert* z U) = *convex hull* (*insert* z $(T \cap U)$)
(**is** *?lhs* = *?rhs*)

proposition *simplicial_subdivision_of_cell_complex*:
assumes *finite* $\mathcal{M}$
and *poly*: $\bigwedge C.\ C \in \mathcal{M} \Longrightarrow$ *polytope* C
and *face*: $\bigwedge C1\ C2.\ [\![C1 \in \mathcal{M};\ C2 \in \mathcal{M}]\!] \Longrightarrow C1 \cap C2$ *face_of* $C1$
obtains $\mathcal{T}$ **where** *simplicial_complex* $\mathcal{T}$
$\bigcup \mathcal{T} = \bigcup \mathcal{M}$
$\bigwedge C.\ C \in \mathcal{M} \Longrightarrow \exists F.$ *finite* $F \wedge F \subseteq \mathcal{T} \wedge C = \bigcup F$
$\bigwedge K.\ K \in \mathcal{T} \Longrightarrow \exists C.\ C \in \mathcal{M} \wedge K \subseteq C$

corollary *fine_simplicial_subdivision_of_cell_complex*:
assumes $0 < e$ *finite* $\mathcal{M}$
and *poly*: $\bigwedge C.\ C \in \mathcal{M} \Longrightarrow$ *polytope* C
and *face*: $\bigwedge C1\ C2.\ [\![C1 \in \mathcal{M};\ C2 \in \mathcal{M}]\!] \Longrightarrow C1 \cap C2$ *face_of* $C1$
obtains $\mathcal{T}$ **where** *simplicial_complex* $\mathcal{T}$
$\bigwedge K.\ K \in \mathcal{T} \Longrightarrow$ *diameter* $K < e$
$\bigcup \mathcal{T} = \bigcup \mathcal{M}$
$\bigwedge C.\ C \in \mathcal{M} \Longrightarrow \exists F.$ *finite* $F \wedge F \subseteq \mathcal{T} \wedge C = \bigcup F$
$\bigwedge K.\ K \in \mathcal{T} \Longrightarrow \exists C.\ C \in \mathcal{M} \wedge K \subseteq C$

9.24.20 Some results on cell division with full-dimensional cells only

proposition *fine_triangular_subdivision_of_cell_complex*:
assumes $0 < e$ *finite* $\mathcal{M}$
and *poly*: $\bigwedge C.\ C \in \mathcal{M} \Longrightarrow$ *polytope* C
and *aff*: $\bigwedge C.\ C \in \mathcal{M} \Longrightarrow$ *aff_dim* $C = d$
and *face*: $\bigwedge C1\ C2.\ [\![C1 \in \mathcal{M};\ C2 \in \mathcal{M}]\!] \Longrightarrow C1 \cap C2$ *face_of* $C1$
obtains $\mathcal{T}$ **where** *triangulation* $\mathcal{T}$ $\bigwedge k.\ k \in \mathcal{T} \Longrightarrow$ *diameter* $k < e$
$\bigwedge k.\ k \in \mathcal{T} \Longrightarrow$ *aff_dim* $k = d$ $\bigcup \mathcal{T} = \bigcup \mathcal{M}$
$\bigwedge C.\ C \in \mathcal{M} \Longrightarrow \exists f.$ *finite* $f \wedge f \subseteq \mathcal{T} \wedge C = \bigcup f$
$\bigwedge k.\ k \in \mathcal{T} \Longrightarrow \exists C.\ C \in \mathcal{M} \wedge k \subseteq C$

9.25 Finitely generated cone is polyhedral, and hence closed

proposition *polyhedron_convex_cone_hull*:
fixes S :: *'a::euclidean_space set*
assumes *finite* S
shows *polyhedron*(*convex_cone hull* S)

end

9.26 Absolute Retracts, Absolute Neighbourhood Retracts and Euclidean Neighbourhood Retracts

theory *Retracts*
imports
Brouwer_Fixpoint
Continuous_Extension
begindefinition *AR* :: *'a::topological_space set* ⇒ *bool* **where**
AR S ≡ ∀ *U*. ∀ *S'*::(*'a* ∗ *real*) *set*.
S homeomorphic S' ∧ *closedin* (*top_of_set U*) *S'* ⟶ *S' retract_of U*

definition *ANR* :: *'a::topological_space set* ⇒ *bool* **where**
ANR S ≡ ∀ *U*. ∀ *S'*::(*'a* ∗ *real*) *set*.
S homeomorphic S' ∧ *closedin* (*top_of_set U*) *S'*
⟶ (∃ *T*. *openin* (*top_of_set U*) *T* ∧ *S' retract_of T*)

definition *ENR* :: *'a::topological_space set* ⇒ *bool* **where**
ENR S ≡ ∃ *U*. *open U* ∧ *S retract_of U*

corollary *ANR_imp_absolute_neighbourhood_retract*:
fixes S :: $'a$::*euclidean_space set* **and** S' :: $'b$::*euclidean_space set*
assumes *ANR S S homeomorphic* S'
and *clo*: *closedin* (*top_of_set U*) S'
obtains V **where** *openin* (*top_of_set U*) V S' *retract_of V*

corollary *ANR_imp_absolute_neighbourhood_retract_UNIV*:
fixes S :: $'a$::*euclidean_space set* **and** S' :: $'b$::*euclidean_space set*
assumes *ANR S* **and** *hom*: *S homeomorphic* S' **and** *clo*: *closed* S'
obtains V **where** *open V* S' *retract_of V*

corollary *neighbourhood_extension_into_ANR*:
fixes f :: $'a$::*euclidean_space* $\Rightarrow$ $'b$::*euclidean_space*
assumes *contf*: *continuous_on S f* **and** *fim*: $f \in S \to T$ **and** *ANR T closed S*
obtains V g **where** $S \subseteq V$ *open V continuous_on V g*
$g \in V \to T$ $\bigwedge x.\ x \in S \Longrightarrow g\ x = f\ x$

9.26.1 Analogous properties of ENRs

corollary *ENR_imp_absolute_neighbourhood_retract_UNIV*:
fixes S :: $'a$::*euclidean_space set* **and** S' :: $'b$::*euclidean_space set*
assumes *ENR S S homeomorphic* S'
obtains T' **where** *open* T' S' *retract_of* T'

corollary *AR_closed_Un*:
fixes S :: $'a$::*euclidean_space set*
shows $[\![$*closed S*; *closed T*; *AR S*; *AR T*; *AR* $(S \cap T)]\!] \Longrightarrow$ *AR* $(S \cup T)$

corollary *ANR_closed_Un*:
fixes S :: $'a$::*euclidean_space set*
shows $[\![$*closed S*; *closed T*; *ANR S*; *ANR T*; *ANR* $(S \cap T)]\!] \Longrightarrow$ *ANR* $(S \cup T)$

9.26.2 More advanced properties of ANRs and ENRs

9.26.3 Original ANR material, now for ENRs

9.26.4 Finally, spheres are ANRs and ENRs

9.26.5 Spheres are connected, etc

9.26.6 Borsuk homotopy extension theorem

theorem *Borsuk_homotopy_extension_homotopic*:
fixes f :: *'a::euclidean_space* ⇒ *'b::euclidean_space*
assumes *cloTS*: *closedin* (*top_of_set* T) S
and *anr*: (*ANR* S ∧ *ANR* T) ∨ *ANR* U
and *contf*: *continuous_on* T f
and $f \in T \to U$
and *homotopic_with_canon* (λ*x*. *True*) S U f g
obtains g' **where** *homotopic_with_canon* (λ*x*. *True*) T U f g'
continuous_on T g' *image* g' $T \subseteq U$
$\bigwedge x.\ x \in S \Longrightarrow g'\ x = g\ x$

9.26.7 More extension theorems

9.26.8 The complement of a set and path-connectedness

theorem *connected_complement_homeomorphic_convex_compact*:
fixes S :: *'a::euclidean_space set* **and** T :: *'b::euclidean_space set*
assumes *hom*: S *homeomorphic* T **and** T: *convex* T *compact* T **and** *2*: $2 \leq$ *DIM*(*'a*)
shows *connected*($-$ S)

corollary *path_connected_complement_homeomorphic_convex_compact*:
fixes S :: *'a::euclidean_space set* **and** T :: *'b::euclidean_space set*
assumes *hom*: S *homeomorphic* T *convex* T *compact* T $2 \leq$ *DIM*(*'a*)
shows *path_connected*($-$ S)

end

9.27 Extending Continous Maps, Invariance of Domain, etc

theory *Further_Topology*
imports *Weierstrass_Theorems Polytope Complex_Transcendental Equivalence_Lebesgue_Henstock_Integration Retracts*
begin

9.27.1 A map from a sphere to a higher dimensional sphere is nullhomotopic

proposition *inessential_spheremap_lowdim_gen*:
fixes f :: $'M$::*euclidean_space* ⇒ $'a$::*euclidean_space*
assumes *convex S bounded S convex T bounded T*
and *affST*: *aff_dim S* < *aff_dim T*
and *contf*: *continuous_on* (*rel_frontier S*) *f*
and *fim*: $f \in$ (*rel_frontier S*) → *rel_frontier T*
obtains *c* **where** *homotopic_with_canon* (λ*z*. *True*) (*rel_frontier S*) (*rel_frontier T*) *f* (λ*x*. *c*)

9.27.2 Some technical lemmas about extending maps from cell complexes

theorem *extend_map_cell_complex_to_sphere*:
assumes *finite* $\mathcal{F}$ **and** *S*: $S \subseteq \bigcup\mathcal{F}$ *closed S* **and** *T*: *convex T bounded T*
and *poly*: $\bigwedge X.\ X \in \mathcal{F} \Longrightarrow$ *polytope X*
and *aff*: $\bigwedge X.\ X \in \mathcal{F} \Longrightarrow$ *aff_dim X* < *aff_dim T*
and *face*: $\bigwedge X\ Y.\ [\![X \in \mathcal{F};\ Y \in \mathcal{F}]\!] \Longrightarrow (X \cap Y)$ *face_of X*
and *contf*: *continuous_on S f* **and** *fim*: $f \in S \rightarrow$ *rel_frontier T*
obtains *g* **where** *continuous_on* $(\bigcup\mathcal{F})$ *g*
$g \in (\bigcup\mathcal{F}) \rightarrow$ *rel_frontier T* $\bigwedge x.\ x \in S \Longrightarrow g\ x = f\ x$

theorem *extend_map_cell_complex_to_sphere_cofinite*:
assumes *finite* $\mathcal{F}$ **and** *S*: $S \subseteq \bigcup\mathcal{F}$ *closed S* **and** *T*: *convex T bounded T*
and *poly*: $\bigwedge X.\ X \in \mathcal{F} \Longrightarrow$ *polytope X*
and *aff*: $\bigwedge X.\ X \in \mathcal{F} \Longrightarrow$ *aff_dim X* ≤ *aff_dim T*
and *face*: $\bigwedge X\ Y.\ [\![X \in \mathcal{F};\ Y \in \mathcal{F}]\!] \Longrightarrow (X \cap Y)$ *face_of X*
and *contf*: *continuous_on S f* **and** *fim*: $f \in S \rightarrow$ *rel_frontier T*
obtains *C g* **where** *finite C disjnt C S continuous_on* $(\bigcup\mathcal{F} - C)$ *g*
$g \in (\bigcup\mathcal{F} - C) \rightarrow$ *rel_frontier T* $\bigwedge x.\ x \in S \Longrightarrow g\ x = f\ x$

9.27.3 Special cases and corollaries involving spheres

proposition *extend_map_affine_to_sphere_cofinite_simple*:
fixes f :: $'a$::*euclidean_space* ⇒ $'b$::*euclidean_space*
assumes *compact S convex U bounded U*
and *aff*: *aff_dim T* ≤ *aff_dim U*
and $S \subseteq T$ **and** *contf*: *continuous_on S f*
and *fim*: $f \in S \rightarrow$ *rel_frontier U*

obtains K g **where** *finite* K $K \subseteq T$ *disjnt* K S *continuous_on* $(T - K)$ g
$g \in (T - K) \to$ *rel_frontier* U
$\bigwedge x.\ x \in S \Longrightarrow g\ x = f\ x$

9.27.4 Extending maps to spheres

proposition *extend_map_affine_to_sphere_cofinite_gen*:
fixes f :: *'a::euclidean_space* ⇒ *'b::euclidean_space*
assumes *SUT*: *compact* S *convex* U *bounded* U *affine* T $S \subseteq T$
and *aff*: *aff_dim* $T \leq$ *aff_dim* U
and *contf*: *continuous_on* S f
and *fim*: $f \in S \to$ *rel_frontier* U
and *dis*: $\bigwedge C.\ [\![C \in components(T - S);\ bounded\ C]\!] \Longrightarrow C \cap L \neq \{\}$
obtains K g **where** *finite* K $K \subseteq L$ $K \subseteq T$ *disjnt* K S *continuous_on* $(T - K)$ g
$g \in (T - K) \to$ *rel_frontier* U
$\bigwedge x.\ x \in S \Longrightarrow g\ x = f\ x$

corollary *extend_map_affine_to_sphere_cofinite*:
fixes f :: *'a::euclidean_space* ⇒ *'b::euclidean_space*
assumes *SUT*: *compact* S *affine* T $S \subseteq T$
and *aff*: *aff_dim* $T \leq$ *DIM('b)* **and** $0 \leq r$
and *contf*: *continuous_on* S f
and *fim*: $f \in S \to$ *sphere* a r
and *dis*: $\bigwedge C.\ [\![C \in components(T - S);\ bounded\ C]\!] \Longrightarrow C \cap L \neq \{\}$
obtains K g **where** *finite* K $K \subseteq L$ $K \subseteq T$ *disjnt* K S *continuous_on* $(T - K)$ g
$g \in (T - K) \to$ *sphere* a r $\bigwedge x.\ x \in S \Longrightarrow g\ x = f\ x$

corollary *extend_map_UNIV_to_sphere_cofinite*:
fixes f :: *'a::euclidean_space* ⇒ *'b::euclidean_space*
assumes *DIM('a)* $\leq$ *DIM('b)* **and** $0 \leq r$
and *compact* S
and *continuous_on* S f
and $f \in S \to$ *sphere* a r
and $\bigwedge C.\ [\![C \in components(-\ S);\ bounded\ C]\!] \Longrightarrow C \cap L \neq \{\}$
obtains K g **where** *finite* K $K \subseteq L$ *disjnt* K S *continuous_on* $(-\ K)$ g
$g \in (-\ K) \to$ *sphere* a r $\bigwedge x.\ x \in S \Longrightarrow g\ x = f\ x$

corollary *extend_map_UNIV_to_sphere_no_bounded_component*:
fixes f :: *'a::euclidean_space* ⇒ *'b::euclidean_space*
assumes *aff*: *DIM('a)* $\leq$ *DIM('b)* **and** $0 \leq r$
and *SUT*: *compact* S
and *contf*: *continuous_on* S f

and *fim*: $f \in S \to sphere\ a\ r$
and *dis*: $\bigwedge C.\ C \in components(-\ S) \Longrightarrow \neg\ bounded\ C$
obtains g **where** $continuous\_on\ UNIV\ g\ g \in UNIV \to sphere\ a\ r\ \bigwedge x.\ x \in S \Longrightarrow g\ x = f\ x$

theorem *Borsuk_separation_theorem_gen*:
fixes $S :: 'a::euclidean\_space\ set$
assumes *compact S*
shows $(\forall c \in components(-\ S).\ \neg bounded\ c) \longleftrightarrow$
$(\forall f.\ continuous\_on\ S\ f \wedge f \in S \to sphere\ (0::'a)\ 1$
$\longrightarrow (\exists c.\ homotopic\_with\_canon\ (\lambda x.\ True)\ S\ (sphere\ 0\ 1)\ f\ (\lambda x.\ c)))$
(**is** *?lhs* = *?rhs*)

corollary *Borsuk_separation_theorem*:
fixes $S :: 'a::euclidean\_space\ set$
assumes *compact S* **and** *2*: $2 \le DIM('a)$
shows $connected(-\ S) \longleftrightarrow$
$(\forall f.\ continuous\_on\ S\ f \wedge f \in S \to sphere\ (0::'a)\ 1$
$\longrightarrow (\exists c.\ homotopic\_with\_canon\ (\lambda x.\ True)\ S\ (sphere\ 0\ 1)\ f\ (\lambda x.\ c)))$
(**is** *?lhs* = *?rhs*)

proposition *Jordan_Brouwer_separation*:
fixes $S :: 'a::euclidean\_space\ set$ **and** $a::'a$
assumes *hom*: *S homeomorphic sphere a r* **and** $0 < r$
shows $\neg\ connected(-\ S)$

proposition *Jordan_Brouwer_frontier*:
fixes $S :: 'a::euclidean\_space\ set$ **and** $a::'a$
assumes *S*: *S homeomorphic sphere a r* **and** *T*: $T \in components(-\ S)$ **and** *2*: $2 \le DIM('a)$
shows *frontier T = S*

proposition *Jordan_Brouwer_nonseparation*:
fixes $S :: 'a::euclidean\_space\ set$ **and** $a::'a$
assumes *S*: *S homeomorphic sphere a r* **and** $T \subset S$ **and** *2*: $2 \le DIM('a)$
shows $connected(-\ T)$

9.27.5 Invariance of domain and corollaries

theorem *invariance_of_domain*:
fixes $f :: 'a \Rightarrow 'a::euclidean\_space$
assumes *continuous_on S f open S inj_on f S*

shows *open*(*f* ‘ *S*)

corollary *invariance_of_domain_subspaces*:
fixes *f* :: *'a::euclidean_space* ⇒ *'b::euclidean_space*
assumes *ope*: *openin* (*top_of_set U*) *S*
and *subspace U subspace V* **and** *VU*: *dim V* ≤ *dim U*
and *contf*: *continuous_on S f* **and** *fim*: *f* ∈ *S* → *V*
and *injf*: *inj_on f S*
shows *openin* (*top_of_set V*) (*f* ‘ *S*)

corollary *invariance_of_dimension_subspaces*:
fixes *f* :: *'a::euclidean_space* ⇒ *'b::euclidean_space*
assumes *ope*: *openin* (*top_of_set U*) *S*
and *subspace U subspace V*
and *contf*: *continuous_on S f* **and** *fim*: *f* ∈ *S* → *V*
and *injf*: *inj_on f S* **and** *S* ≠ {}
shows *dim U* ≤ *dim V*

corollary *invariance_of_domain_affine_sets*:
fixes *f* :: *'a::euclidean_space* ⇒ *'b::euclidean_space*
assumes *ope*: *openin* (*top_of_set U*) *S*
and *aff*: *affine U affine V aff_dim V* ≤ *aff_dim U*
and *contf*: *continuous_on S f* **and** *fim*: *f* ∈ *S* → *V*
and *injf*: *inj_on f S*
shows *openin* (*top_of_set V*) (*f* ‘ *S*)

corollary *invariance_of_dimension_affine_sets*:
fixes *f* :: *'a::euclidean_space* ⇒ *'b::euclidean_space*
assumes *ope*: *openin* (*top_of_set U*) *S*
and *aff*: *affine U affine V*
and *contf*: *continuous_on S f* **and** *fim*: *f* ∈ *S* → *V*
and *injf*: *inj_on f S* **and** *S* ≠ {}
shows *aff_dim U* ≤ *aff_dim V*

corollary *invariance_of_dimension*:
fixes *f* :: *'a::euclidean_space* ⇒ *'b::euclidean_space*
assumes *contf*: *continuous_on S f* **and** *open S*
and *injf*: *inj_on f S* **and** *S* ≠ {}
shows *DIM*(*'a*) ≤ *DIM*(*'b*)

corollary *continuous_injective_image_subspace_dim_le*:
fixes *f* :: *'a::euclidean_space* ⇒ *'b::euclidean_space*
assumes *subspace S subspace T*
and *contf*: *continuous_on S f* **and** *fim*: *f* ∈ *S* → *T*
and *injf*: *inj_on f S*
shows *dim S* ≤ *dim T*

corollary *invariance_of_domain_homeomorphic*:
fixes f :: $'a$::*euclidean_space* ⇒ $'b$::*euclidean_space*
assumes *open S continuous_on S f DIM('b) ≤ DIM('a) inj_on f S*
shows *S homeomorphic (f ‘ S)*

proposition *homeomorphic_interiors*:
fixes *S* :: $'a$::*euclidean_space set* **and** *T* :: $'b$::*euclidean_space set*
assumes *S homeomorphic T interior S = {} ⟷ interior T = {}*
shows *(interior S) homeomorphic (interior T)*

proposition *uniformly_continuous_homeomorphism_UNIV_trivial*:
fixes f :: $'a$::*euclidean_space* ⇒ $'a$
assumes *contf*: *uniformly_continuous_on S f* **and** *hom*: *homeomorphism S UNIV f g*
shows *S = UNIV*

9.27.6 Formulation of loop homotopy in terms of maps out of type complex

proposition *simply_connected_eq_homotopic_circlemaps*:
fixes *S* :: $'a$::*real_normed_vector set*
shows *simply_connected S ⟷*
(∀ *f g*::*complex* ⇒ $'a$.
continuous_on (sphere 0 1) f ∧ f ∈ (sphere 0 1) → S ∧
continuous_on (sphere 0 1) g ∧ g ∈ (sphere 0 1) → S
⟶ *homotopic_with_canon (λh. True) (sphere 0 1) S f g*)

proposition *simply_connected_eq_contractible_circlemap*:
fixes *S* :: $'a$::*real_normed_vector set*
shows *simply_connected S ⟷*
path_connected S ∧
(∀ *f*::*complex* ⇒ $'a$.
continuous_on (sphere 0 1) f ∧ f ‘(sphere 0 1) ⊆ S
⟶ (∃ *a. homotopic_with_canon (λh. True) (sphere 0 1) S f (λx. a)*)))

corollary *homotopy_eqv_simple_connectedness*:
fixes *S* :: $'a$::*real_normed_vector set* **and** *T* :: $'b$::*real_normed_vector set*
shows *S homotopy_eqv T ⟹ simply_connected S ⟷ simply_connected T*

9.27.7 Homeomorphism of simple closed curves to circles

proposition *homeomorphic_simple_path_image_circle*:
fixes a :: *complex* **and** γ :: *real* $\Rightarrow$ *'a::t2_space*
assumes *simple_path* γ **and** *loop*: *pathfinish* γ = *pathstart* γ **and** $0 < r$
shows (*path_image* γ) *homeomorphic sphere a r*

9.27.8 Dimension-based conditions for various homeomorphisms

9.27.9 more invariance of domain

proposition *invariance_of_domain_sphere_affine_set_gen*:
fixes f :: *'a::euclidean_space* $\Rightarrow$ *'b::euclidean_space*
assumes *contf*: *continuous_on S f* **and** *injf*: *inj_on f S* **and** *fim*: $f \in S \to T$
and U: *bounded U convex U*
and *affine T* **and** *affTU*: *aff_dim T* < *aff_dim U*
and *ope*: *openin* (*top_of_set* (*rel_frontier U*)) S
shows *openin* (*top_of_set T*) (*f ‘ S*)

proposition *simply_connected_punctured_convex*:
fixes a :: *'a::euclidean_space*
assumes *convex S* **and** *3*: $3 \leq$ *aff_dim S*
shows *simply_connected*($S - \{a\}$)

corollary *simply_connected_punctured_universe*:
fixes a :: *'a::euclidean_space*
assumes $3 \leq$ *DIM*(*'a*)
shows *simply_connected*($- \{a\}$)

9.27.10 The power, squaring and exponential functions as covering maps

proposition *covering_space_power_punctured_plane*:
assumes $0 < n$
shows *covering_space* $(- \{0\})$ (λz::*complex*. z^n) $(- \{0\})$

corollary *covering_space_square_punctured_plane*:
covering_space $(- \{0\})$ (λz::*complex*. z^2) $(- \{0\})$

proposition *covering_space_exp_punctured_plane*:
covering_space UNIV (λz::*complex*. *exp z*) $(- \{0\})$

9.27.11 Hence the Borsukian results about mappings into circles

corollary *inessential_imp_continuous_logarithm_circle*:
fixes f :: $'a$::*real_normed_vector* $\Rightarrow$ *complex*
assumes *homotopic_with_canon* (λh. *True*) S (*sphere 0 1*) f (λt. a)
obtains g **where** *continuous_on* S g **and** $\bigwedge x.\ x \in S \Longrightarrow f\ x = exp(g\ x)$

proposition *homotopic_with_sphere_times*:
fixes f :: $'a$::*real_normed_vector* $\Rightarrow$ *complex*
assumes *hom*: *homotopic_with_canon* (λx. *True*) S (*sphere 0 1*) f g **and** *conth*: *continuous_on* S h
and *hin*: $\bigwedge x.\ x \in S \Longrightarrow h\ x \in$ *sphere 0 1*
shows *homotopic_with_canon* (λx. *True*) S (*sphere 0 1*) ($\lambda x.\ f\ x * h\ x$) ($\lambda x.\ g\ x * h\ x$)

proposition *homotopic_circlemaps_divide*:
fixes f :: $'a$::*real_normed_vector* $\Rightarrow$ *complex*
shows *homotopic_with_canon* (λx. *True*) S (*sphere 0 1*) f g $\longleftrightarrow$
continuous_on S f $\wedge$ $f \in S \to$ *sphere 0 1* $\wedge$
continuous_on S g $\wedge$ $g \in S \to$ *sphere 0 1* $\wedge$
($\exists c$. *homotopic_with_canon* (λx. *True*) S (*sphere 0 1*) ($\lambda x.\ f\ x\ /\ g\ x$) ($\lambda x.\ c$))

9.27.12 Upper and lower hemicontinuous functions

proposition *upper_lower_hemicontinuous_explicit*:
fixes T :: ($'b$::{*real_normed_vector*,*heine_borel*}) *set*
assumes *fST*: $\bigwedge x.\ x \in S \Longrightarrow f\ x \subseteq T$
and *ope*: $\bigwedge U$. *openin* (*top_of_set* T) U
$\Longrightarrow$ *openin* (*top_of_set* S) $\{x \in S.\ f\ x \subseteq U\}$
and *clo*: $\bigwedge U$. *closedin* (*top_of_set* T) U
$\Longrightarrow$ *closedin* (*top_of_set* S) $\{x \in S.\ f\ x \subseteq U\}$
and $x \in S$ $0 < e$ **and** *bofx*: *bounded*($f\ x$) **and** *fx_ne*: $f\ x \neq \{\}$
obtains d **where** $0 < d$
$\bigwedge x'.\ [\![x' \in S;\ dist\ x\ x' < d]\!]$
$\Longrightarrow (\forall y \in f\ x.\ \exists y'.\ y' \in f\ x' \wedge dist\ y\ y' < e) \wedge$
$(\forall y' \in f\ x'.\ \exists y.\ y \in f\ x \wedge dist\ y'\ y < e)$

9.27.13 Complex logs exist on various "well-behaved" sets

9.27.14 Another simple case where sphere maps are nullhomotopic

9.27.15 Holomorphic logarithms and square roots

9.27.16 The "Borsukian" property of sets

definition *Borsukian* **where**
$Borsukian\ S \equiv$
$\forall f.\ continuous\_on\ S\ f \land f \in S \to (-\ \{0::complex\})$
$\longrightarrow (\exists a.\ homotopic\_with\_canon\ (\lambda h.\ True)\ S\ (-\ \{0\})\ f\ (\lambda x.\ a))$

proposition *Borsukian_sphere*:
fixes $a :: 'a::euclidean\_space$
shows $3 \le DIM('a) \Longrightarrow Borsukian\ (sphere\ a\ r)$

proposition *Borsukian_open_Un*:
fixes $S :: 'a::real\_normed\_vector\ set$
assumes $opeS$: $openin\ (top\_of\_set\ (S \cup T))\ S$
and $opeT$: $openin\ (top\_of\_set\ (S \cup T))\ T$
and BS: $Borsukian\ S$ **and** BT: $Borsukian\ T$ **and** ST: $connected(S \cap T)$
shows $Borsukian(S \cup T)$

proposition *closed_irreducible_separator*:
fixes $a :: 'a::real\_normed\_vector$
assumes $closed\ S$ **and** ab: $\neg\ connected\_component\ (-\ S)\ a\ b$
obtains T **where** $T \subseteq S\ closed\ T\ T \neq \{\}\ \neg\ connected\_component\ (-\ T)\ a\ b$
$\bigwedge U.\ U \subset T \Longrightarrow connected\_component\ (-\ U)\ a\ b$

9.27.17 Unicoherence (closed)

definition *unicoherent* **where**
$unicoherent\ U \equiv$
$\forall S\ T.\ connected\ S \land connected\ T \land S \cup T = U \land$
$closedin\ (top\_of\_set\ U)\ S \land closedin\ (top\_of\_set\ U)\ T$
$\longrightarrow connected\ (S \cap T)$

proposition *homeomorphic_unicoherent*:
assumes ST: $S\ homeomorphic\ T$ **and** S: $unicoherent\ S$
shows $unicoherent\ T$

corollary *contractible_imp_unicoherent*:
fixes *U* :: *'a::euclidean_space set*
assumes *contractible U* **shows** *unicoherent U*

corollary *convex_imp_unicoherent*:
fixes *U* :: *'a::euclidean_space set*
assumes *convex U* **shows** *unicoherent U*
corollary *unicoherent_UNIV*: *unicoherent* (*UNIV* :: *'a* :: *euclidean_space set*)

9.27.18 Several common variants of unicoherence

9.27.19 Some separation results

proposition *separation_by_component_open*:
fixes *S* :: *'a* :: *euclidean_space set*
assumes *open S* **and** *non*: ¬ *connected*(− *S*)
obtains *C* **where** *C* ∈ *components S* ¬ *connected*(− *C*)

proposition *inessential_eq_extensible*:
fixes *f* :: *'a::euclidean_space* ⇒ *complex*
assumes *closed S*
shows (∃ *a. homotopic_with_canon* (λ*h. True*) *S* (−{*0*}) *f* (λ*t. a*)) ⟷
(∃ *g. continuous_on UNIV g* ∧ (∀ *x* ∈ *S. g x* = *f x*) ∧ (∀ *x. g x* ≠ *0*))
(**is** *?lhs* = *?rhs*)

proposition *Janiszewski_dual*:
fixes *S* :: *complex set*
assumes *compact S compact T connected S connected T connected*(− (*S* ∪ *T*))
shows *connected*(*S* ∩ *T*)

end

9.28 The Jordan Curve Theorem and Applications

theory *Jordan_Curve*
imports *Arcwise_Connected Further_Topology*
begin

9.28.1 Janiszewski's theorem

theorem *Janiszewski*:

fixes *a b* :: *complex*
assumes *compact S closed T* **and** *conST*: *connected* (*S* ∩ *T*)
and *ccS*: *connected_component* (− *S*) *a b* **and** *ccT*: *connected_component* (− *T*) *a b*
shows *connected_component* (− (*S* ∪ *T*)) *a b*

9.28.2 The Jordan Curve theorem

corollary *Jordan_inside_outside*:
fixes *c* :: *real* ⇒ *complex*
assumes *simple_path c pathfinish c* = *pathstart c*
shows *inside*(*path_image c*) ≠ {} ∧
open(*inside*(*path_image c*)) ∧
connected(*inside*(*path_image c*)) ∧
outside(*path_image c*) ≠ {} ∧
open(*outside*(*path_image c*)) ∧
connected(*outside*(*path_image c*)) ∧
bounded(*inside*(*path_image c*)) ∧
¬ *bounded*(*outside*(*path_image c*)) ∧
inside(*path_image c*) ∩ *outside*(*path_image c*) = {} ∧
inside(*path_image c*) ∪ *outside*(*path_image c*) =
− *path_image c* ∧
frontier(*inside*(*path_image c*)) = *path_image c* ∧
frontier(*outside*(*path_image c*)) = *path_image c*

theorem *split_inside_simple_closed_curve*:
fixes *c* :: *real* ⇒ *complex*
assumes *simple_path c1* **and** *c1*: *pathstart c1* = *a pathfinish c1* = *b*
and *simple_path c2* **and** *c2*: *pathstart c2* = *a pathfinish c2* = *b*
and *simple_path c* **and** *c*: *pathstart c* = *a pathfinish c* = *b*
and *a* ≠ *b*
and *c1c2*: *path_image c1* ∩ *path_image c2* = {*a*,*b*}
and *c1c*: *path_image c1* ∩ *path_image c* = {*a*,*b*}
and *c2c*: *path_image c2* ∩ *path_image c* = {*a*,*b*}
and *ne_12*: *path_image c* ∩ *inside*(*path_image c1* ∪ *path_image c2*) ≠ {}
obtains *inside*(*path_image c1* ∪ *path_image c*) ∩ *inside*(*path_image c2* ∪ *path_image c*) = {}
inside(*path_image c1* ∪ *path_image c*) ∪ *inside*(*path_image c2* ∪ *path_image c*) ∪
(*path_image c* − {*a*,*b*}) = *inside*(*path_image c1* ∪ *path_image c2*)

end

9.29 Polynomial Functions: Extremal Behaviour and Root Counts

theory *Poly_Roots*
imports *Complex_Main*
begin

9.29.1 Basics about polynomial functions: extremal behaviour and root counts

proposition *polyfun_extremal_lemma*:
fixes $c :: nat \Rightarrow 'a::real\_normed\_div\_algebra$
assumes $e > 0$
shows $\exists M. \forall z. M \le norm\ z \longrightarrow norm(\sum i{\le}n.\ c\ i * z\hat{\ }i) \le e * norm(z)\ \hat{\ }\ Suc\ n$

proposition *polyfun_extremal*:
fixes $c :: nat \Rightarrow 'a::real\_normed\_div\_algebra$
assumes $\exists k.\ k \ne 0 \wedge k \le n \wedge c\ k \ne 0$
shows $eventually\ (\lambda z.\ norm(\sum i{\le}n.\ c\ i * z\hat{\ }i) \ge B)\ at\_infinity$

proposition *polyfun_rootbound*:
fixes $c :: nat \Rightarrow 'a::\{comm\_ring,real\_normed\_div\_algebra\}$
assumes $\exists k.\ k \le n \wedge c\ k \ne 0$
shows $finite\ \{z.\ (\sum i{\le}n.\ c\ i * z\hat{\ }i) = 0\} \wedge card\ \{z.\ (\sum i{\le}n.\ c\ i * z\hat{\ }i) = 0\} \le n$

corollary
fixes $c :: nat \Rightarrow 'a::\{comm\_ring,real\_normed\_div\_algebra\}$
assumes $\exists k.\ k \le n \wedge c\ k \ne 0$
shows *polyfun_rootbound_finite*: $finite\ \{z.\ (\sum i{\le}n.\ c\ i * z\hat{\ }i) = 0\}$
and *polyfun_rootbound_card*: $card\ \{z.\ (\sum i{\le}n.\ c\ i * z\hat{\ }i) = 0\} \le n$

proposition *polyfun_finite_roots*:
fixes $c :: nat \Rightarrow 'a::\{comm\_ring,real\_normed\_div\_algebra\}$
shows $finite\ \{z.\ (\sum i{\le}n.\ c\ i * z\hat{\ }i) = 0\} \longleftrightarrow (\exists k.\ k \le n \wedge c\ k \ne 0)$

theorem *polyfun_eq_const*:
fixes $c :: nat \Rightarrow 'a::\{comm\_ring,real\_normed\_div\_algebra\}$
shows $(\forall z.\ (\sum i{\le}n.\ c\ i * z\hat{\ }i) = k) \longleftrightarrow c\ 0 = k \wedge (\forall k.\ k \ne 0 \wedge k \le n \longrightarrow c\ k = 0)$

end

9.30 Generalised Binomial Theorem

theory *Generalised_Binomial_Theorem*

imports
Complex_Main
Complex_Transcendental
Summation_Tests
begin

theorem *gen_binomial_complex*:
fixes z :: *complex*
assumes *norm* $z < 1$
shows $(\lambda n.\ (a\ gchoose\ n) * z\hat{\ }n)$ *sums* $(1 + z)$ *powr* a

end

9.31 Vitali Covering Theorem and an Application to Negligibility

theory *Vitali_Covering_Theorem*
imports
HOL−Combinatorics.Permutations
Equivalence_Lebesgue_Henstock_Integration
begin

9.31.1 Vitali covering theorem

theorem *Vitali_covering_theorem_cballs*:
fixes $a :: 'a \Rightarrow 'n::euclidean\_space$
assumes r: $\bigwedge i.\ i \in K \Longrightarrow 0 < r\ i$
and S: $\bigwedge x\ d.\ [\![x \in S;\ 0 < d]\!]$
$\Longrightarrow \exists i.\ i \in K \wedge x \in cball\ (a\ i)\ (r\ i) \wedge r\ i < d$
obtains C **where** *countable* C $C \subseteq K$
pairwise $(\lambda i\ j.\ disjnt\ (cball\ (a\ i)\ (r\ i))\ (cball\ (a\ j)\ (r\ j)))\ C$
negligible$(S - (\bigcup i \in C.\ cball\ (a\ i)\ (r\ i)))$

theorem *Vitali_covering_theorem_balls*:
fixes $a :: 'a \Rightarrow 'b::euclidean\_space$
assumes S: $\bigwedge x\ d.\ [\![x \in S;\ 0 < d]\!] \Longrightarrow \exists i.\ i \in K \wedge x \in ball\ (a\ i)\ (r\ i) \wedge r\ i < d$
obtains C **where** *countable* C $C \subseteq K$
pairwise $(\lambda i\ j.\ disjnt\ (ball\ (a\ i)\ (r\ i))\ (ball\ (a\ j)\ (r\ j)))\ C$
negligible$(S - (\bigcup i \in C.\ ball\ (a\ i)\ (r\ i)))$

proposition *negligible_eq_zero_density*:
negligible $S \longleftrightarrow$

$(\forall x \in S.\ \forall r>0.\ \forall e>0.\ \exists d.\ 0 < d \wedge d \leq r \wedge$
$(\exists U.\ S \cap ball\ x\ d \subseteq U \wedge U \in lmeasurable \wedge measure\ lebesgue\ U < e * measure\ lebesgue\ (ball\ x\ d)))$

end

9.32 Change of Variables Theorems

theory *Change_Of_Vars*
imports *Vitali_Covering_Theorem Determinants*

begin

9.32.1 Measurable Shear and Stretch

proposition
fixes *a* :: *real^'n*
assumes $m \neq n$ **and** *ab_ne*: $cbox\ a\ b \neq \{\}$ **and** *an*: $0 \leq a\$n$
shows *measurable_shear_interval*: $(\lambda x.\ \chi\ i.\ if\ i = m\ then\ x\$m + x\$n\ else\ x\$i)$ ‘ $(cbox\ a\ b) \in lmeasurable$
(**is** *?f* ‘ _ ∈ _)
and *measure_shear_interval*: $measure\ lebesgue\ ((\lambda x.\ \chi\ i.\ if\ i = m\ then\ x\$m + x\$n\ else\ x\$i)$ ‘ $cbox\ a\ b)$
$= measure\ lebesgue\ (cbox\ a\ b)$ (**is** *?Q*)

proposition
fixes *S* :: *(real^'n) set*
assumes $S \in lmeasurable$
shows *measurable_stretch*: $((\lambda x.\ \chi\ k.\ m\ k * x\$k)$ ‘ $S) \in lmeasurable$ (**is** *?f* ‘ *S* ∈ _)
and *measure_stretch*: $measure\ lebesgue\ ((\lambda x.\ \chi\ k.\ m\ k * x\$k)$ ‘ $S) = |prod\ m\ UNIV| * measure\ lebesgue\ S$
(**is** *?MEQ*)

proposition
fixes *f* :: *real^'n::{finite,wellorder}* ⇒ *real^'n::_*
assumes *linear f* $S \in lmeasurable$
shows *measurable_linear_image*: $(f$ ‘ $S) \in lmeasurable$
and *measure_linear_image*: $measure\ lebesgue\ (f$ ‘ $S) = |det\ (matrix\ f)| * measure\ lebesgue\ S$ (**is** *?Q f S*)

proposition *measure_semicontinuous_with_hausdist_explicit*:
assumes *bounded S* **and** *neg*: *negligible(frontier S)* **and** $e > 0$
obtains *d* **where** $d > 0$
$\bigwedge T.\ [\![T \in lmeasurable;\ \bigwedge y.\ y \in T \Longrightarrow \exists x.\ x \in S \wedge dist\ x\ y < d]\!]$
$\Longrightarrow measure\ lebesgue\ T < measure\ lebesgue\ S + e$

proposition
fixes f :: $real$^$'n$::{$finite$,$wellorder$} ⇒ $real$^$'n$::_
assumes S: S ∈ $lmeasurable$
and $deriv$: ⋀x. x ∈ S ⟹ (f $has\_derivative$ f' x) (at x $within$ S)
and int: (λx. |det ($matrix$ (f' x))|) $integrable\_on$ S
and $bounded$: ⋀x. x ∈ S ⟹ |det ($matrix$ (f' x))| ≤ B
shows $measurable\_bounded\_differentiable\_image$:
f ‘ S ∈ $lmeasurable$
and $measure\_bounded\_differentiable\_image$:
$measure$ $lebesgue$ (f ‘ S) ≤ B ∗ $measure$ $lebesgue$ S (**is** ?M)

theorem
fixes f :: $real$^$'n$::{$finite$,$wellorder$} ⇒ $real$^$'n$::_
assumes S: S ∈ $sets$ $lebesgue$
and $deriv$: ⋀x. x ∈ S ⟹ (f $has\_derivative$ f' x) (at x $within$ S)
and int: (λx. |det ($matrix$ (f' x))|) $integrable\_on$ S
shows $measurable\_differentiable\_image$: f ‘ S ∈ $lmeasurable$
and $measure\_differentiable\_image$:
$measure$ $lebesgue$ (f ‘ S) ≤ $integral$ S (λx. |det ($matrix$ (f' x))|) (**is** ?M)

9.32.2 Borel measurable Jacobian determinant

proposition $borel\_measurable\_partial\_derivatives$:
fixes f :: $real$^$'m$::{$finite$,$wellorder$} ⇒ $real$^$'n$
assumes S: S ∈ $sets$ $lebesgue$
and f: ⋀x. x ∈ S ⟹ (f $has\_derivative$ f' x) (at x $within$ S)
shows (λx. ($matrix$(f' x)\$$m$\$$n$)) ∈ $borel\_measurable$ ($lebesgue\_on$ S)

theorem $borel\_measurable\_det\_Jacobian$:
fixes f :: $real$^$'n$::{$finite$,$wellorder$} ⇒ $real$^$'n$::_
assumes S: S ∈ $sets$ $lebesgue$ **and** f: ⋀x. x ∈ S ⟹ (f $has\_derivative$ f' x) (at x $within$ S)
shows (λx. det($matrix$(f' x))) ∈ $borel\_measurable$ ($lebesgue\_on$ S)
theorem $borel\_measurable\_lebesgue\_on\_preimage\_borel$:
fixes f :: $'a$::$euclidean\_space$ ⇒ $'b$::$euclidean\_space$
assumes S ∈ $sets$ $lebesgue$
shows f ∈ $borel\_measurable$ ($lebesgue\_on$ S) ⟷
(∀ T. T ∈ $sets$ $borel$ ⟶ {x ∈ S. f x ∈ T} ∈ $sets$ $lebesgue$)

9.32.3 Simplest case of Sard's theorem (we don't need continuity of derivative)

theorem *baby_Sard*:
fixes f :: *real*^′*m*::{*finite*,*wellorder*} ⇒ *real*^′*n*::{*finite*,*wellorder*}
assumes *mlen*: $CARD('m) \leq CARD('n)$
and *der*: $\bigwedge x.\ x \in S \Longrightarrow (f$ *has_derivative* $f'\ x)$ (*at* x *within* S)
and *rank*: $\bigwedge x.\ x \in S \Longrightarrow rank(matrix(f'\ x)) < CARD('n)$
shows *negligible*(f ‘ S)

9.32.4 A one-way version of change-of-variables not assuming injectivity.

proposition *absolutely_integrable_on_image*:
fixes f :: *real*^′*m*::{*finite*,*wellorder*} ⇒ *real*^′*n* **and** g :: *real*^′*m*::_ ⇒ *real*^′*m*::_
assumes *der_g*: $\bigwedge x.\ x \in S \Longrightarrow (g$ *has_derivative* $g'\ x)$ (*at* x *within* S)
and *intS*: ($\lambda x.\ |det\ (matrix\ (g'\ x))| *_R f(g\ x)$) *absolutely_integrable_on* S
shows f *absolutely_integrable_on* (g ‘ S)

proposition *integral_on_image_ubound*:
fixes f :: *real*^′*n*::{*finite*,*wellorder*} ⇒ *real* **and** g :: *real*^′*n*::_ ⇒ *real*^′*n*::_
assumes $\bigwedge x.\ x \in S \Longrightarrow 0 \leq f(g\ x)$
and $\bigwedge x.\ x \in S \Longrightarrow (g$ *has_derivative* $g'\ x)$ (*at* x *within* S)
and ($\lambda x.\ |det\ (matrix\ (g'\ x))| * f(g\ x)$) *integrable_on* S
shows *integral* (g ‘ S) $f \leq$ *integral* S ($\lambda x.\ |det\ (matrix\ (g'\ x))| * f(g\ x)$)

9.32.5 Change-of-variables theorem

theorem *has_absolute_integral_change_of_variables_invertible*:
fixes f :: *real*^′*m*::{*finite*,*wellorder*} ⇒ *real*^′*n* **and** g :: *real*^′*m*::_ ⇒ *real*^′*m*::_
assumes *der_g*: $\bigwedge x.\ x \in S \Longrightarrow (g$ *has_derivative* $g'\ x)$ (*at* x *within* S)
and *hg*: $\bigwedge x.\ x \in S \Longrightarrow h(g\ x) = x$
and *conth*: *continuous_on* (g ‘ S) h
shows ($\lambda x.\ |det\ (matrix\ (g'\ x))| *_R f(g\ x)$) *absolutely_integrable_on* $S \wedge$ *integral* S ($\lambda x.\ |det\ (matrix\ (g'\ x))| *_R f(g\ x)) = b \longleftrightarrow$
f *absolutely_integrable_on* (g ‘ S) $\wedge$ *integral* (g ‘ S) $f = b$

(**is** ?lhs = ?rhs)

theorem *has_absolute_integral_change_of_variables_compact*:
fixes f :: real^'m::{finite,wellorder} ⇒ real^'n **and** g :: real^'m::_ ⇒ real^'m::_
assumes *compact S*
and *der_g*: $\bigwedge x.\ x \in S \Longrightarrow$ (g has_derivative g' x) (at x within S)
and *inj*: inj_on g S
shows ((λx. |det (matrix (g' x))| $*_R$ f(g x)) absolutely_integrable_on S ∧
integral S (λx. |det (matrix (g' x))| $*_R$ f(g x)) = b)
⟷ f absolutely_integrable_on (g ‘ S) ∧ integral (g ‘ S) f = b)

theorem *has_absolute_integral_change_of_variables*:
fixes f :: real^'m::{finite,wellorder} ⇒ real^'n **and** g :: real^'m::_ ⇒ real^'m::_
assumes *S*: S ∈ sets lebesgue
and *der_g*: $\bigwedge x.\ x \in S \Longrightarrow$ (g has_derivative g' x) (at x within S)
and *inj*: inj_on g S
shows (λx. |det (matrix (g' x))| $*_R$ f(g x)) absolutely_integrable_on S ∧
integral S (λx. |det (matrix (g' x))| $*_R$ f(g x)) = b
⟷ f absolutely_integrable_on (g ‘ S) ∧ integral (g ‘ S) f = b

corollary *absolutely_integrable_change_of_variables*:
fixes f :: real^'m::{finite,wellorder} ⇒ real^'n **and** g :: real^'m::_ ⇒ real^'m::_
assumes S ∈ sets lebesgue
and $\bigwedge x.\ x \in S \Longrightarrow$ (g has_derivative g' x) (at x within S)
and inj_on g S
shows f absolutely_integrable_on (g ‘ S)
⟷ (λx. |det (matrix (g' x))| $*_R$ f(g x)) absolutely_integrable_on S

corollary *integral_change_of_variables*:
fixes f :: real^'m::{finite,wellorder} ⇒ real^'n **and** g :: real^'m::_ ⇒ real^'m::_
assumes *S*: S ∈ sets lebesgue
and *der_g*: $\bigwedge x.\ x \in S \Longrightarrow$ (g has_derivative g' x) (at x within S)
and *inj*: inj_on g S
and *disj*: (f absolutely_integrable_on (g ‘ S) ∨
(λx. |det (matrix (g' x))| $*_R$ f(g x)) absolutely_integrable_on S)
shows integral (g ‘ S) f = integral S (λx. |det (matrix (g' x))| $*_R$ f(g x))

corollary *absolutely_integrable_change_of_variables_1*:
fixes f :: real ⇒ real^'n::{finite,wellorder} **and** g :: real ⇒ real
assumes *S*: S ∈ sets lebesgue
and *der_g*: $\bigwedge x.\ x \in S \Longrightarrow$ (g has_vector_derivative g' x) (at x within S)
and *inj*: inj_on g S
shows (f absolutely_integrable_on g ‘ S ⟷

($\lambda x.\ |g'\ x| *_R f(g\ x)$) *absolutely_integrable_on S*)

9.32.6 Change of variables for integrals: special case of linear function

9.32.7 Change of variable for measure

end

9.33 Lipschitz Continuity

theory *Lipschitz*
  imports
    Derivative Abstract_Metric_Spaces
begin

definition *lipschitz_on*
  where *lipschitz_on C U f* $\longleftrightarrow$ (*0* $\leq$ *C* $\wedge$ ($\forall x \in U.\ \forall y\in U.$ *dist* (*f x*) (*f y*) $\leq$ *C* $*$ *dist x y*))
notation
  lipschitz_on (‹(‹*open_block notation=*‹*postfix lipschitz_on*››_−*lipschitz'_on*)› [*1000*])
proposition *lipschitz_on_uniformly_continuous*:
  assumes *L−lipschitz_on X f*
  shows *uniformly_continuous_on X f*

proposition *lipschitz_on_continuous_on*:
  continuous_on X f **if** *L−lipschitz_on X f*
proposition *bounded_derivative_imp_lipschitz*:
  assumes $\bigwedge x.\ x \in X \Longrightarrow$ (*f has_derivative f' x*) (*at x within X*)
  assumes *convex*: *convex X*
  assumes $\bigwedge x.\ x \in X \Longrightarrow$ *onorm* (*f' x*) $\leq$ *C* *0* $\leq$ *C*
  shows *C−lipschitz_on X f*

9.33.1 Local Lipschitz continuity

proposition *lipschitz_on_closed_Union*:
  assumes $\bigwedge i.\ i \in I \Longrightarrow$ *lipschitz_on M* (*U i*) *f*
    $\bigwedge i.\ i \in I \Longrightarrow$ *closed* (*U i*)
    finite I
    M $\geq$ *0*
    {*u*..(*v*::*real*)} $\subseteq$ ($\bigcup i\in I.$ *U i*)
  shows *lipschitz_on M* {*u*..*v*} *f*

9.33.2 Local Lipschitz continuity (uniform for a family of functions)

definition *local_lipschitz*::
 $'a$::*metric_space set* ⇒ $'b$::*metric_space set* ⇒ ($'a$ ⇒ $'b$ ⇒ $'c$::*metric_space*) ⇒ *bool*
 where
 local_lipschitz T X f ≡ ∀ *x* ∈ *X*. ∀ *t* ∈ *T*.
  ∃ *u*>*0*. ∃ *L*. ∀ *t* ∈ *cball t u* ∩ *T*. *L*−*lipschitz_on* (*cball x u* ∩ *X*) (*f t*)

proposition *c1_implies_local_lipschitz*:
 fixes *T*::*real set* **and** *X*::$'a$::{*banach*,*heine_borel*} *set*
  and *f*::*real* ⇒ $'a$ ⇒ $'a$
 assumes *f'*: ⋀*t x*. *t* ∈ *T* ⟹ *x* ∈ *X* ⟹ (*f t has_derivative blinfun_apply* (*f'* (*t*, *x*))) (*at x*)
 assumes *cont_f'*: *continuous_on* (*T* × *X*) *f'*
 assumes *open T*
 assumes *open X*
 shows *local_lipschitz T X f*

end
theory
 Multivariate_Analysis
imports
 Ordered_Euclidean_Space
 Determinants
 Cross3
 Lipschitz
 Starlike
beginend

9.34 Volume of a Simplex

theory *Simplex_Content*
imports *Change_Of_Vars*
begin

theorem *content_std_simplex*:
 measure lborel (*convex hull* (*insert 0 Basis* :: $'a$ :: *euclidean_space set*)) =
  1 / *fact DIM*($'a$)

proposition *measure_lebesgue_linear_transformation*:
 fixes *A* :: (*real* ^ $'n$ :: {*finite*, *wellorder*}) *set*
 fixes *f* :: _ ⇒ *real* ^ $'n$ :: {*finite*, *wellorder*}
 assumes *bounded A A* ∈ *sets lebesgue linear f*
 shows *measure lebesgue* (*f* ‘ *A*) = |*det* (*matrix f*)| ∗ *measure lebesgue A*

theorem *content_simplex*:

fixes *X* :: (*real* ^ *'n* :: {*finite*, *wellorder*}) *set* **and** *f* :: *'n* :: _ ⇒ *real* ^(*'n* :: _)
assumes *finite X card X* = *Suc CARD*(*'n*) **and** *x0*: *x0* ∈ *X* **and** *bij*: *bij_betw f UNIV* (*X* − {*x0*})
defines *M* ≡ (χ *i*. χ *j*. *f j* $ *i* − *x0* $ *i*)
shows *content* (*convex hull X*) = |*det M*| / *fact* (*CARD*(*'n*))

theorem *content_triangle*:
fixes *A B C* :: *real* ^ *2*
shows *content* (*convex hull* {*A*, *B*, *C*}) =
|(*C* $ *1* − *A* $ *1*) ∗ (*B* $ *2* − *A* $ *2*) − (*B* $ *1* − *A* $ *1*) ∗ (*C* $ *2* − *A* $ *2*)| / *2*

theorem *heron*:
fixes *A B C* :: *real* ^ *2*
defines *a* ≡ *dist B C* **and** *b* ≡ *dist A C* **and** *c* ≡ *dist A B*
defines *s* ≡ (*a* + *b* + *c*) / *2*
shows *content* (*convex hull* {*A*, *B*, *C*}) = *sqrt* (*s* ∗ (*s* − *a*) ∗ (*s* − *b*) ∗ (*s* − *c*))

end

9.35 Convergence of Formal Power Series

theory *FPS_Convergence*
imports
Generalised_Binomial_Theorem
HOL−Computational_Algebra.Formal_Power_Series
begin

9.35.1 Basic properties of convergent power series

definition *fps_conv_radius* :: *'a* :: {*banach*, *real_normed_div_algebra*} *fps* ⇒ *ereal* **where**
fps_conv_radius f = *conv_radius* (*fps_nth f*)

definition *eval_fps* :: *'a* :: {*banach*, *real_normed_div_algebra*} *fps* ⇒ *'a* ⇒ *'a* **where**
eval_fps f z = (∑ *n*. *fps_nth f n* ∗ *z* ^ *n*)

theorem *sums_eval_fps*:
fixes *f* :: *'a* :: {*banach*, *real_normed_div_algebra*} *fps*
assumes *norm z* < *fps_conv_radius f*
shows (λ*n*. *fps_nth f n* ∗ *z* ^ *n*) *sums eval_fps f z*

9.35.2 Evaluating power series

theorem *eval_fps_deriv*:
assumes *norm z* < *fps_conv_radius f*

shows *eval_fps* (*fps_deriv f*) *z* = *deriv* (*eval_fps f*) *z*

theorem *fps_nth_conv_deriv*:
fixes *f* :: *complex fps*
assumes *fps_conv_radius f* > *0*
shows *fps_nth f n* = (*deriv* ^^ *n*) (*eval_fps f*) *0* / *fact n*

theorem *eval_fps_eqD*:
fixes *f g* :: *complex fps*
assumes *fps_conv_radius f* > *0* *fps_conv_radius g* > *0*
assumes *eventually* (λ*z*. *eval_fps f z* = *eval_fps g z*) (*nhds 0*)
shows *f* = *g*

9.35.3 Power series expansions of analytic functions

definition
has_fps_expansion :: (*'a* :: {*banach*,*real_normed_div_algebra*} ⇒ *'a*) ⇒ *'a fps* ⇒ *bool*
(**infixl** ‹*has'_fps'_expansion*› *60*)
where (*f has_fps_expansion F*) ⟷
fps_conv_radius F > *0* ∧ *eventually* (λ*z*. *eval_fps F z* = *f z*) (*nhds 0*)

end
theory *Smooth_Paths*
imports *Retracts*
begin

9.35.4 Piecewise differentiability of paths

9.35.5 Valid paths, and their start and finish

definition *valid_path* :: (*real* ⇒ *'a* :: *real_normed_vector*) ⇒ *bool*
where *valid_path f* ≡ *f piecewise_C1_differentiable_on* {*0..1*::*real*}

end

9.36 Metrics on product spaces

theory *Function_Metric*
imports
Function_Topology
Elementary_Metric_Spaces
begininstantiation *fun* :: (*countable*, *metric_space*) *metric_space*
begin

definition *dist_fun_def*:

dist x y = ($\sum$ *n*. (*1/2*)^*n* ∗ *min* (*dist* (*x* (*from_nat n*)) (*y* (*from_nat n*))) *1*)

definition *uniformity_fun_def*:
(*uniformity*::((′*a* ⇒ ′*b*) × (′*a* ⇒ ′*b*)) *filter*) = (*INF e*∈{*0*<..}. *principal* {(*x*, *y*). *dist* (*x*::(′*a*⇒′*b*)) *y* < *e*})
end
theory *Analysis*
imports

Convex
Determinants

FSigma
Sum_Topology
Abstract_Topological_Spaces
Abstract_Metric_Spaces
Urysohn
Connected
Abstract_Limits
Isolated
Sparse_In

Elementary_Normed_Spaces
Norm_Arith

Convex_Euclidean_Space
Operator_Norm

Line_Segment
Derivative
Cartesian_Euclidean_Space
Kronecker_Approximation_Theorem
Weierstrass_Theorems

Ball_Volume
Integral_Test
Improper_Integral
Equivalence_Measurable_On_Borel
Lebesgue_Integral_Substitution
Embed_Measure
Complete_Measure
Radon_Nikodym
Fashoda_Theorem
Cross3
Homeomorphism
Bounded_Continuous_Function
Abstract_Topology
Product_Topology
Lindelof_Spaces

Infinite_Products
Infinite_Sum
Infinite_Set_Sum
Polytope
Jordan_Curve
Poly_Roots
Generalised_Binomial_Theorem
Gamma_Function
Change_Of_Vars
Multivariate_Analysis
Simplex_Content
FPS_Convergence
Smooth_Paths
Abstract_Euclidean_Space
Function_Metric
begin

end

Bibliography

[1]