Cohomology of Banach Algebras
Fields Mini-Course

Michael C. White

Newcastle University

15 - 16 May, 2014

List of Topics

1.1 Summary

List of Topics

1. Basic Definitions and Notation

1.1 Summary

List of Topics

1. Basic Definitions and Notation
2. Amenability and Weak Amenability

1.1 Summary

List of Topics

1. Basic Definitions and Notation
2. Amenability and Weak Amenability
3. Derivations

1.1 Summary

List of Topics

1. Basic Definitions and Notation
2. Amenability and Weak Amenability
3. Derivations
4. Injective Modules

1.1 Summary

List of Topics

1. Basic Definitions and Notation
2. Amenability and Weak Amenability
3. Derivations
4. Injective Modules
5. Submodules and Quotient Modules

1.1 Summary

List of Topics

1. Basic Definitions and Notation
2. Amenability and Weak Amenability
3. Derivations
4. Injective Modules
5. Submodules and Quotient Modules
6. Higher Cohomology

List of Topics

1. Basic Definitions and Notation
2. Amenability and Weak Amenability
3. Derivations
4. Injective Modules
5. Submodules and Quotient Modules
6. Higher Cohomology
7. Resolutions

1.1 Summary

List of Topics

1. Basic Definitions and Notation
2. Amenability and Weak Amenability
3. Derivations
4. Injective Modules
5. Submodules and Quotient Modules
6. Higher Cohomology
7. Resolutions
8. Bifunctorial cohomology: $\mathtt{Ext}$

1.1 Summary

List of Topics

1. Basic Definitions and Notation
2. Amenability and Weak Amenability
3. Derivations
4. Injective Modules
5. Submodules and Quotient Modules
6. Higher Cohomology
7. Resolutions
8. Bifunctorial cohomology: Ext
9. Cyclic cohomology

2.1 Standing assumptions

Cavets aka Excuses

2.1 Standing assumptions

Cavets aka Excuses

- All Banach algebras will have a unit, or bounded approximate identity

Cavets aka Excuses

- All Banach algebras will have a unit, or bounded approximate identity
- All modules will be Banach modules

2.1 Standing assumptions

Cavets aka Excuses

- All Banach algebras will have a unit, or bounded approximate identity
- All modules will be Banach modules
- In general a unit should be added before using the definitions

2.1 Standing assumptions

Cavets aka Excuses

- All Banach algebras will have a unit, or bounded approximate identity
- All modules will be Banach modules
- In general a unit should be added before using the definitions
- All maps will be continuous and linear

2.1 Standing assumptions

Cavets aka Excuses

- All Banach algebras will have a unit, or bounded approximate identity
- All modules will be Banach modules
- In general a unit should be added before using the definitions
- All maps will be continuous and linear
- Algebras will act unitally on modules, or the b.a.i. will act as such

2.1 Standing assumptions

Cavets aka Excuses

- All Banach algebras will have a unit, or bounded approximate identity
- All modules will be Banach modules
- In general a unit should be added before using the definitions
- All maps will be continuous and linear
- Algebras will act unitally on modules, or the b.a.i. will act as such
- Sub- and quotient modules usually need to be (weakly) complemented as Banach spaces

2.1 Standing assumptions

Cavets aka Excuses

- All Banach algebras will have a unit, or bounded approximate identity
- All modules will be Banach modules
- In general a unit should be added before using the definitions
- All maps will be continuous and linear
- Algebras will act unitally on modules, or the b.a.i. will act as such
- Sub- and quotient modules usually need to be (weakly) complemented as Banach spaces
- For more details see:

2.1 Standing assumptions

Cavets aka Excuses

- All Banach algebras will have a unit, or bounded approximate identity
- All modules will be Banach modules
- In general a unit should be added before using the definitions
- All maps will be continuous and linear
- Algebras will act unitally on modules, or the b.a.i. will act as such
- Sub- and quotient modules usually need to be (weakly) complemented as Banach spaces
- For more details see:
  - `Helemskii` for detailed treatment topological algebras

2.1 Standing assumptions

Cavets aka Excuses

- All Banach algebras will have a unit, or bounded approximate identity
- All modules will be Banach modules
- In general a unit should be added before using the definitions
- All maps will be continuous and linear
- Algebras will act unitally on modules, or the b.a.i. will act as such
- Sub- and quotient modules usually need to be (weakly) complemented as Banach spaces
- For more details see:
  - `Helemskii` for detailed treatment topological algebras
  - `Weibel` for the algebraic background

2.1 Standing assumptions

Cavets aka Excuses

- All Banach algebras will have a unit, or bounded approximate identity
- All modules will be Banach modules
- In general a unit should be added before using the definitions
- All maps will be continuous and linear
- Algebras will act unitally on modules, or the b.a.i. will act as such
- Sub- and quotient modules usually need to be (weakly) complemented as Banach spaces
- For more details see:
  - `Helemskii` for detailed treatment topological algebras
  - `Weibel` for the algebraic background
  - `Loday` for detailed treatment of cyclic cohomology

2.1 Standing assumptions

Cavets aka Excuses

- All Banach algebras will have a unit, or bounded approximate identity
- All modules will be Banach modules
- In general a unit should be added before using the definitions
- All maps will be continuous and linear
- Algebras will act unitally on modules, or the b.a.i. will act as such
- Sub- and quotient modules usually need to be (weakly) complemented as Banach spaces
- For more details see:
  - `Helemskii` for detailed treatment topological algebras
  - `Weibel` for the algebraic background
  - `Loday` for detailed treatment of cyclic cohomology

2.1 Standing assumptions

Cavets aka Excuses

- All Banach algebras will have a unit, or bounded approximate identity
- All modules will be Banach modules
- In general a unit should be added before using the definitions
- All maps will be continuous and linear
- Algebras will act unitally on modules, or the b.a.i. will act as such
- Sub- and quotient modules usually need to be (weakly) complemented as Banach spaces
- For more details see:
  - `Helemskii` for detailed treatment topological algebras
  - `Weibel` for the algebraic background
  - `Loday` for detailed treatment of cyclic cohomology

Definitions and Notation

2.2 Definitions and Notation

Definitions and Notation

- Derivations $\mathcal{Z}^1(A, Y) = \{D : A \to Y, s.t.\ D(ab) = aD(b) + D(a)b\}$

Definitions and Notation

- Derivations $\mathcal{Z}^1(A, Y) = \{D : A \to Y, s.t.\ D(ab) = aD(b) + D(a)b\}$
- Inner derivations $\mathcal{B}^1(A, Y) = \{\delta_y(a) = ay - ya\}$

2.2 Definitions and Notation

Definitions and Notation

- Derivations $\mathcal{Z}^1(A, Y) = \{D : A \to Y, s.t. \ D(ab) = aD(b) + D(a)b\}$
- Inner derivations $\mathcal{B}^1(A, Y) = \{\delta_y(a) = ay - ya\}$
- Commutative bimodules, where $ay = ya$ satisfy $D(a^n) = a^{n-1}D(a)$

2.2 Definitions and Notation

Definitions and Notation

- Derivations $\mathcal{Z}^1(A, Y) = \{D : A \to Y, s.t.\ D(ab) = aD(b) + D(a)b\}$
- Inner derivations $\mathcal{B}^1(A, Y) = \{\delta_y(a) = ay - ya\}$
- Commutative bimodules, where $ay = ya$ satisfy $D(a^n) = a^{n-1}D(a)$
- $\mathcal{H}^1(A, Y) := \mathcal{Z}^1(A, Y)/\mathcal{B}^1(A, Y)$

2.2 Definitions and Notation

Definitions and Notation

- Derivations $\mathcal{Z}^1(A, Y) = \{D : A \to Y, s.t.\ D(ab) = aD(b) + D(a)b\}$
- Inner derivations $\mathcal{B}^1(A, Y) = \{\delta_y(a) = ay - ya\}$
- Commutative bimodules, where $ay = ya$ satisfy $D(a^n) = a^{n-1}D(a)$
- $\mathcal{H}^1(A, Y) := \mathcal{Z}^1(A, Y)/\mathcal{B}^1(A, Y)$
- Note $\mathcal{H}^1(A, Y)$ a Banach space iff $\mathcal{B}^1(A, Y)$ closed [Ex],

2.2 Definitions and Notation

Definitions and Notation

- Derivations $\mathcal{Z}^1(A, Y) = \{D : A \to Y, s.t.\ D(ab) = aD(b) + D(a)b\}$
- Inner derivations $\mathcal{B}^1(A, Y) = \{\delta_y(a) = ay - ya\}$
- Commutative bimodules, where $ay = ya$ satisfy $D(a^n) = a^{n-1}D(a)$
- $\mathcal{H}^1(A, Y) := \mathcal{Z}^1(A, Y)/\mathcal{B}^1(A, Y)$
- Note $\mathcal{H}^1(A, Y)$ a Banach space iff $\mathcal{B}^1(A, Y)$ closed [Ex],

Definitions and Notation

- Derivations $\mathcal{Z}^1(A, Y) = \{D : A \to Y, s.t.\ D(ab) = aD(b) + D(a)b\}$
- Inner derivations $\mathcal{B}^1(A, Y) = \{\delta_y(a) = ay - ya\}$
- Commutative bimodules, where $ay = ya$ satisfy $D(a^n) = a^{n-1}D(a)$
- $\mathcal{H}^1(A, Y) := \mathcal{Z}^1(A, Y)/\mathcal{B}^1(A, Y)$
- Note $\mathcal{H}^1(A, Y)$ a Banach space iff $\mathcal{B}^1(A, Y)$ closed [Ex], $\mathcal{Z}^1(A, Y)$ is always closed

2.2 Definitions and Notation

Definitions and Notation

- Derivations $\mathcal{Z}^1(A, Y) = \{D : A \to Y, s.t.\ D(ab) = aD(b) + D(a)b\}$
- Inner derivations $\mathcal{B}^1(A, Y) = \{\delta_y(a) = ay - ya\}$
- Commutative bimodules, where $ay = ya$ satisfy $D(a^n) = a^{n-1}D(a)$
- $\mathcal{H}^1(A, Y) := \mathcal{Z}^1(A, Y)/\mathcal{B}^1(A, Y)$
- Note $\mathcal{H}^1(A, Y)$ a Banach space iff $\mathcal{B}^1(A, Y)$ closed [Ex], $\mathcal{Z}^1(A, Y)$ is always closed
- For semigroup algebras this is related to *reversal depth*

2.2 Definitions and Notation

Definitions and Notation

- Derivations $\mathcal{Z}^1(A, Y) = \{D : A \to Y, s.t. \; D(ab) = aD(b) + D(a)b\}$
- Inner derivations $\mathcal{B}^1(A, Y) = \{\delta_y(a) = ay - ya\}$
- Commutative bimodules, where $ay = ya$ satisfy $D(a^n) = a^{n-1}D(a)$
- $\mathcal{H}^1(A, Y) := \mathcal{Z}^1(A, Y)/\mathcal{B}^1(A, Y)$
- Note $\mathcal{H}^1(A, Y)$ a Banach space iff $\mathcal{B}^1(A, Y)$ closed [Ex], $\mathcal{Z}^1(A, Y)$ is always closed
- For semigroup algebras this is related to *reversal depth*

Definitions and Notation

- Derivations $\mathcal{Z}^1(A, Y) = \{D : A \to Y, s.t.\ D(ab) = aD(b) + D(a)b\}$
- Inner derivations $\mathcal{B}^1(A, Y) = \{\delta_y(a) = ay - ya\}$
- Commutative bimodules, where $ay = ya$ satisfy $D(a^n) = a^{n-1}D(a)$
- $\mathcal{H}^1(A, Y) := \mathcal{Z}^1(A, Y)/\mathcal{B}^1(A, Y)$
- Note $\mathcal{H}^1(A, Y)$ a Banach space iff $\mathcal{B}^1(A, Y)$ closed [Ex], $\mathcal{Z}^1(A, Y)$ is always closed
- For semigroup algebras this is related to *reversal depth*

Amenability and Cohomology

2.3 Connections with Amenability

Amenability and Cohomology

- A is *amenable* iff $\mathcal{H}^1(A, X') = 0$, for example $L^1(G)$, for amenable G

2.3 Connections with Amenability

Amenability and Cohomology

- A is *amenable* iff $\mathcal{H}^1(A, X') = 0$, for example $L^1(G)$, for amenable G
- A is *weakly amenable* if $\mathcal{H}^1(A, A') = 0$, for example all $L^1(G)$

2.3 Connections with Amenability

Amenability and Cohomology

- A is *amenable* iff $\mathcal{H}^1(A, X') = 0$, for example $L^1(G)$, for amenable G
- A is *weakly amenable* if $\mathcal{H}^1(A, A') = 0$, for example all $L^1(G)$
- Why do we consider derivations into the dual module?

2.3 Connections with Amenability

Amenability and Cohomology

- A is *amenable* iff $\mathcal{H}^1(A, X') = 0$, for example $L^1(G)$, for amenable G
- A is *weakly amenable* if $\mathcal{H}^1(A, A') = 0$, for example all $L^1(G)$
- Why do we consider derivations into the dual module?
- The Singer-Wermer Theorem says that semi-simple algebras have no non-zero derivations $D : A \to A$,

Amenability and Cohomology

- A is *amenable* iff $\mathcal{H}^1(A, X') = 0$, for example $L^1(G)$, for amenable G
- A is *weakly amenable* if $\mathcal{H}^1(A, A') = 0$, for example all $L^1(G)$
- Why do we consider derivations into the dual module?
- The Singer-Wermer Theorem says that semi-simple algebras have no non-zero derivations $D : A \to A$,
- It is *Functoral*, $\theta : A \to B$, gives $\theta^* : \mathcal{H}\mathcal{H}^1(B) \to \mathcal{H}\mathcal{H}^1(A)$ [Ex]

2.3 Connections with Amenability

Amenability and Cohomology

- A is *amenable* iff $\mathcal{H}^1(A, X') = 0$, for example $L^1(G)$, for amenable G
- A is *weakly amenable* if $\mathcal{H}^1(A, A') = 0$, for example all $L^1(G)$
- Why do we consider derivations into the dual module?
- The Singer-Wermer Theorem says that semi-simple algebras have no non-zero derivations $D : A \to A$,
- It is *Functoral*, $\theta : A \to B$, gives $\theta^* : \mathcal{HH}^1(B) \to \mathcal{HH}^1(A)$ [Ex]
- *Simplicially trivial* algebras have $\mathcal{HH}^n(A) = \mathcal{H}^n(A, A') = 0$, e.g. semilattice algebras

2.3 Connections with Amenability

Amenability and Cohomology

- A is *amenable* iff $\mathcal{H}^1(A, X') = 0$, for example $L^1(G)$, for amenable G
- A is *weakly amenable* if $\mathcal{H}^1(A, A') = 0$, for example all $L^1(G)$
- Why do we consider derivations into the dual module?
- The Singer-Wermer Theorem says that semi-simple algebras have no non-zero derivations $D : A \to A$,
- It is *Functoral*, $\theta : A \to B$, gives $\theta^* : \mathcal{HH}^1(B) \to \mathcal{HH}^1(A)$ [Ex]
- *Simplicially trivial* algebras have $\mathcal{HH}^n(A) = \mathcal{H}^n(A, A') = 0$, e.g. semilattice algebras
- *Cyclically amenable* algebras have $\mathcal{HC}^1(A) = 0$, e.g. $\ell^1(\mathbf{Z}_+)$ [Ex]

2.3 Connections with Amenability

Amenability and Cohomology

- A is *amenable* iff $\mathcal{H}^1(A, X') = 0$, for example $L^1(G)$, for amenable G
- A is *weakly amenable* if $\mathcal{H}^1(A, A') = 0$, for example all $L^1(G)$
- Why do we consider derivations into the dual module?
- The Singer-Wermer Theorem says that semi-simple algebras have no non-zero derivations $D : A \to A$,
- It is *Functoral*, $\theta : A \to B$, gives $\theta^* : \mathcal{HH}^1(B) \to \mathcal{HH}^1(A)$ [Ex]
- *Simplicially trivial* algebras have $\mathcal{HH}^n(A) = \mathcal{H}^n(A, A') = 0$, e.g. semilattice algebras
- *Cyclically amenable* algebras have $\mathcal{HC}^1(A) = 0$, e.g. $\ell^1(\mathbf{Z}_+)$ [Ex]
- *Cyclic derivations* also satisfy $D(a)(b) = -D(b)(a)$,

2.3 Connections with Amenability

Amenability and Cohomology

- A is *amenable* iff $\mathcal{H}^1(A, X') = 0$, for example $L^1(G)$, for amenable G
- A is *weakly amenable* if $\mathcal{H}^1(A, A') = 0$, for example all $L^1(G)$
- Why do we consider derivations into the dual module?
- The Singer-Wermer Theorem says that semi-simple algebras have no non-zero derivations $D : A \to A$,
- It is *Functoral*, $\theta : A \to B$, gives $\theta^* : \mathcal{HH}^1(B) \to \mathcal{HH}^1(A)$ [Ex]
- *Simplicially trivial* algebras have $\mathcal{HH}^n(A) = \mathcal{H}^n(A, A') = 0$, e.g. semilattice algebras
- *Cyclically amenable* algebras have $\mathcal{HC}^1(A) = 0$, e.g. $\ell^1(\mathbf{Z}_+)$ [Ex]
- *Cyclic derivations* also satisfy $D(a)(b) = -D(b)(a)$,

2.3 Connections with Amenability

Amenability and Cohomology

- A is *amenable* iff $\mathcal{H}^1(A, X') = 0$, for example $L^1(G)$, for amenable G
- A is *weakly amenable* if $\mathcal{H}^1(A, A') = 0$, for example all $L^1(G)$
- Why do we consider derivations into the dual module?
- The Singer-Wermer Theorem says that semi-simple algebras have no non-zero derivations $D : A \to A$,
- It is *Functoral*, $\theta : A \to B$, gives $\theta^* : \mathcal{HH}^1(B) \to \mathcal{HH}^1(A)$ [Ex]
- *Simplicially trivial* algebras have $\mathcal{HH}^n(A) = \mathcal{H}^n(A, A') = 0$, e.g. semilattice algebras
- *Cyclically amenable* algebras have $\mathcal{HC}^1(A) = 0$, e.g. $\ell^1(\mathbf{Z}_+)$ [Ex]
- *Cyclic derivations* also satisfy $D(a)(b) = -D(b)(a)$,

2.4 Core Facts and Conventions

Multi-linear Miscellany

2.4 Core Facts and Conventions

Multi-linear Miscellany

- A will be a Banach algebra

2.4 Core Facts and Conventions

Multi-linear Miscellany

- A will be a Banach algebra
- Y will be a Banach space which is a left, right or bi-module over A

2.4 Core Facts and Conventions

Multi-linear Miscellany

- A will be a Banach algebra
- Y will be a Banach space which is a left, right or bi-module over A
- All linear maps will be bounded

Multi-linear Miscellany

- A will be a Banach algebra
- Y will be a Banach space which is a left, right or bi-module over A
- All linear maps will be bounded
- X' is the dual module of X in the usual way

2.4 Core Facts and Conventions

Multi-linear Miscellany

- A will be a Banach algebra
- Y will be a Banach space which is a left, right or bi-module over A
- All linear maps will be bounded
- X' is the dual module of X in the usual way
- $E\hat{\otimes}F$ is the *projective tensor product* of Banach spaces,

2.4 Core Facts and Conventions

Multi-linear Miscellany

- A will be a Banach algebra
- Y will be a Banach space which is a left, right or bi-module over A
- All linear maps will be bounded
- X' is the dual module of X in the usual way
- $E\hat{\otimes}F$ is the *projective tensor product* of Banach spaces,
- $(E\hat{\otimes}F)' \cong BL(E,F;\mathbf{C}) \cong L(E\hat{\otimes}F;\mathbf{C}) \cong L(E,F') \cong L(F,E')$

2.4 Core Facts and Conventions

Multi-linear Miscellany

- A will be a Banach algebra
- Y will be a Banach space which is a left, right or bi-module over A
- All linear maps will be bounded
- X' is the dual module of X in the usual way
- $E\hat{\otimes}F$ is the *projective tensor product* of Banach spaces,
- $(E\hat{\otimes}F)' \cong BL(E, F; \mathbf{C}) \cong L(E\hat{\otimes}F; \mathbf{C}) \cong L(E, F') \cong L(F, E')$
- $E\hat{\otimes}F$, and the above, sometimes inherit a bimodule structure

2.4 Core Facts and Conventions

Multi-linear Miscellany

- A will be a Banach algebra
- Y will be a Banach space which is a left, right or bi-module over A
- All linear maps will be bounded
- X' is the dual module of X in the usual way
- $E\hat{\otimes}F$ is the *projective tensor product* of Banach spaces,
- $(E\hat{\otimes}F)' \cong BL(E, F; \mathbf{C}) \cong L(E\hat{\otimes}F; \mathbf{C}) \cong L(E, F') \cong L(F, E')$
- $E\hat{\otimes}F$, and the above, sometimes inherit a bimodule structure
- e.g., A is a left and also a right module, and $A\hat{\otimes}A$ is a bimodule

2.4 Core Facts and Conventions

Multi-linear Miscellany

- A will be a Banach algebra
- Y will be a Banach space which is a left, right or bi-module over A
- All linear maps will be bounded
- X' is the dual module of X in the usual way
- $E\hat{\otimes}F$ is the *projective tensor product* of Banach spaces,
- $(E\hat{\otimes}F)' \cong BL(E,F;\mathbf{C}) \cong L(E\hat{\otimes}F;\mathbf{C}) \cong L(E,F') \cong L(F,E')$
- $E\hat{\otimes}F$, and the above, sometimes inherit a bimodule structure
- e.g., A is a left and also a right module, and $A\hat{\otimes}A$ is a bimodule
- Also $L(E\hat{\otimes}F;G) \cong BL(E,F;G) \cong L(E,L(F,G))$,

2.4 Core Facts and Conventions

Multi-linear Miscellany

- A will be a Banach algebra
- Y will be a Banach space which is a left, right or bi-module over A
- All linear maps will be bounded
- X' is the dual module of X in the usual way
- $E\hat{\otimes}F$ is the *projective tensor product* of Banach spaces,
- $(E\hat{\otimes}F)' \cong BL(E,F;\mathbf{C}) \cong L(E\hat{\otimes}F;\mathbf{C}) \cong L(E,F') \cong L(F,E')$
- $E\hat{\otimes}F$, and the above, sometimes inherit a bimodule structure
- e.g., A is a left and also a right module, and $A\hat{\otimes}A$ is a bimodule
- Also $L(E\hat{\otimes}F;G) \cong BL(E,F;G) \cong L(E,L(F,G))$,
- Where L and BL denote spaces of (bounded) linear and bilinear maps

2.4 Core Facts and Conventions

Multi-linear Miscellany

- A will be a Banach algebra
- Y will be a Banach space which is a left, right or bi-module over A
- All linear maps will be bounded
- X' is the dual module of X in the usual way
- $E\hat{\otimes}F$ is the *projective tensor product* of Banach spaces,
- $(E\hat{\otimes}F)' \cong BL(E, F; \mathbf{C}) \cong L(E\hat{\otimes}F; \mathbf{C}) \cong L(E, F') \cong L(F, E')$
- $E\hat{\otimes}F$, and the above, sometimes inherit a bimodule structure
- e.g., A is a left and also a right module, and $A\hat{\otimes}A$ is a bimodule
- Also $L(E\hat{\otimes}F; G) \cong BL(E, F; G) \cong L(E, L(F, G))$,
- Where L and BL denote spaces of (bounded) linear and bilinear maps
- ${}_Ah(X, Y)$ is the space of left A-module maps, i.e. $T(ax) = aT(x)$

Multi-linear Miscellany

- A will be a Banach algebra
- Y will be a Banach space which is a left, right or bi-module over A
- All linear maps will be bounded
- X' is the dual module of X in the usual way
- $E\hat{\otimes}F$ is the *projective tensor product* of Banach spaces,
- $(E\hat{\otimes}F)' \cong BL(E, F; \mathbf{C}) \cong L(E\hat{\otimes}F; \mathbf{C}) \cong L(E, F') \cong L(F, E')$
- $E\hat{\otimes}F$, and the above, sometimes inherit a bimodule structure
- e.g., A is a left and also a right module, and $A\hat{\otimes}A$ is a bimodule
- Also $L(E\hat{\otimes}F; G) \cong BL(E, F; G) \cong L(E, L(F, G))$,
- Where L and BL denote spaces of (bounded) linear and bilinear maps
- ${}_Ah(X, Y)$ is the space of left A-module maps, i.e. $T(ax) = aT(x)$
- $h_A(E, F)$ and ${}_Ah_A(M, N)$ denote the right and bimodule morphisms

2.4 Core Facts and Conventions

Multi-linear Miscellany

- A will be a Banach algebra
- Y will be a Banach space which is a left, right or bi-module over A
- All linear maps will be bounded
- X' is the dual module of X in the usual way
- $E\hat{\otimes}F$ is the *projective tensor product* of Banach spaces,
- $(E\hat{\otimes}F)' \cong BL(E, F; \mathbf{C}) \cong L(E\hat{\otimes}F; \mathbf{C}) \cong L(E, F') \cong L(F, E')$
- $E\hat{\otimes}F$, and the above, sometimes inherit a bimodule structure
- e.g., A is a left and also a right module, and $A\hat{\otimes}A$ is a bimodule
- Also $L(E\hat{\otimes}F; G) \cong BL(E, F; G) \cong L(E, L(F, G))$,
- Where L and BL denote spaces of (bounded) linear and bilinear maps
- ${}_Ah(X, Y)$ is the space of left A-module maps, i.e. $T(ax) = aT(x)$
- $h_A(E, F)$ and ${}_Ah_A(M, N)$ denote the right and bimodule morphisms

2.4 Core Facts and Conventions

Multi-linear Miscellany

- A will be a Banach algebra
- Y will be a Banach space which is a left, right or bi-module over A
- All linear maps will be bounded
- X' is the dual module of X in the usual way
- $E\hat{\otimes}F$ is the *projective tensor product* of Banach spaces,
- $(E\hat{\otimes}F)' \cong BL(E, F; \mathbf{C}) \cong L(E\hat{\otimes}F; \mathbf{C}) \cong L(E, F') \cong L(F, E')$
- $E\hat{\otimes}F$, and the above, sometimes inherit a bimodule structure
- e.g., A is a left and also a right module, and $A\hat{\otimes}A$ is a bimodule
- Also $L(E\hat{\otimes}F; G) \cong BL(E, F; G) \cong L(E, L(F, G))$,
- Where L and BL denote spaces of (bounded) linear and bilinear maps
- ${}_Ah(X, Y)$ is the space of left A-module maps, i.e. $T(ax) = aT(x)$
- $h_A(E, F)$ and ${}_Ah_A(M, N)$ denote the right and bimodule morphisms

2.5 Cohomology 101: Derivations

- A a Banach algebra

2.5 Cohomology 101: Derivations

- A a Banach algebra
- Y a Banach A-bimodule

2.5 Cohomology 101: Derivations

- A a Banach algebra
- Y a Banach A-bimodule
- D is a *Derivation* if $D : A \to Y$ and it satisfies the 1-*cocycle equation* $(\delta D) = 0$, where

$$(\delta D)(a, b) := +a \cdot D(b) - D(ab) + D(a) \cdot b.$$

2.5 Cohomology 101: Derivations

- A a Banach algebra
- Y a Banach A-bimodule
- D is a *Derivation* if $D : A \to Y$ and it satisfies the 1-*cocycle equation* $(\delta D) = 0$, where

$$(\delta D)(a, b) := +a \cdot D(b) - D(ab) + D(a) \cdot b.$$

2.5 Cohomology 101: Derivations

- A a Banach algebra
- Y a Banach A-bimodule
- D is a *Derivation* if $D : A \to Y$ and it satisfies the 1-*cocycle equation* $(\delta D) = 0$, where

$$(\delta D)(a, b) := +a \cdot D(b) - D(ab) + D(a) \cdot b.$$

We write $D \in \mathcal{Z}^1(A; Y)$.

2.5 Cohomology 101: Derivations

- A a Banach algebra
- Y a Banach A-bimodule
- D is a *Derivation* if $D : A \to Y$ and it satisfies the 1-*cocycle equation* $(\delta D) = 0$, where

$$(\delta D)(a, b) := +a \cdot D(b) - D(ab) + D(a) \cdot b.$$

  We write $D \in \mathcal{Z}^1(A; Y)$.
- An *inner derivation* is map given by a 1-*coboundary* of an element of Y

$$\delta y := (a \mapsto a \cdot y - y \cdot a)$$

  these are always derivations.

2.5 Cohomology 101: Derivations

- A a Banach algebra
- Y a Banach A-bimodule
- D is a *Derivation* if $D : A \to Y$ and it satisfies the 1-*cocycle equation* $(\delta D) = 0$, where

$$(\delta D)(a, b) := +a \cdot D(b) - D(ab) + D(a) \cdot b.$$

We write $D \in \mathcal{Z}^1(A; Y)$.

- An *inner derivation* is map given by a 1-*coboundary* of an element of Y

$$\delta y := (a \mapsto a \cdot y - y \cdot a)$$

these are always derivations.

- A a Banach algebra
- Y a Banach A-bimodule
- D is a *Derivation* if $D : A \to Y$ and it satisfies the 1-*cocycle equation* $(\delta D) = 0$, where

$$(\delta D)(a, b) := +a \cdot D(b) - D(ab) + D(a) \cdot b.$$

We write $D \in \mathcal{Z}^1(A; Y)$.

- An *inner derivation* is map given by a 1-*coboundary* of an element of Y

$$\delta y := (a \mapsto a \cdot y - y \cdot a)$$

these are always derivations. We write $\delta y \in \mathcal{B}^1(A; Y)$.

2.5 Cohomology 101: Derivations

- A a Banach algebra
- Y a Banach A-bimodule
- D is a *Derivation* if $D : A \to Y$ and it satisfies the 1-*cocycle equation* $(\delta D) = 0$, where

$$(\delta D)(a,b) := +a \cdot D(b) - D(ab) + D(a) \cdot b.$$

We write $D \in \mathcal{Z}^1(A;Y)$.
- An *inner derivation* is map given by a 1-*coboundary* of an element of Y

$$\delta y := (a \mapsto a \cdot y - y \cdot a)$$

these are always derivations. We write $\delta y \in \mathcal{B}^1(A;Y)$.
- We measure how far from being all derivations are the inner derivations by

$$\mathcal{H}^1(A;Y) = \frac{\mathcal{Z}^1(A;Y)}{\mathcal{B}^1(A;Y)}.$$

2.6 Examples of Derivations

- e.g. 1: $A = \ell^1(Z_+)$, $Y = \mathbf{C}_0$, given by the character at 0.

$$Df = f'(0)$$

is a derivation,

2.6 Examples of Derivations

- e.g. 1: $A = \ell^1(Z_+)$, $Y = \mathbf{C}_0$, given by the character at 0.

$$Df = f'(0)$$

is a derivation,

2.6 Examples of Derivations

- e.g. 1: $A = \ell^1(Z_+)$, $Y = \mathbf{C}_0$, given by the character at 0.

$$Df = f'(0)$$

is a derivation,
which cannot be inner as the module is commutative ($ay = ya$).

2.6 Examples of Derivations

- e.g. 1: $A = \ell^1(Z_+)$, $Y = \mathbf{C}_0$, given by the character at 0.

$$Df = f'(0)$$

  is a derivation,
  which cannot be inner as the module is commutative ($ay = ya$).
- e.g. 2: $A = \ell^1(Z_+)$, $Y = A\hat{\otimes}A$, where D is given by

$$Da = a \otimes 1 - 1 \otimes a.$$

  This derivation is always inner,

2.6 Examples of Derivations

- e.g. 1: $A = \ell^1(Z_+)$, $Y = \mathbf{C}_0$, given by the character at 0.

$$Df = f'(0)$$

  is a derivation,
  which cannot be inner as the module is commutative $(ay = ya)$.
- e.g. 2: $A = \ell^1(Z_+)$, $Y = A\hat{\otimes}A$, where D is given by

$$Da = a \otimes 1 - 1 \otimes a.$$

  This derivation is always inner,

2.6 Examples of Derivations

- e.g. 1: $A = \ell^1(Z_+)$, $Y = \mathbf{C}_0$, given by the character at 0.

$$Df = f'(0)$$

  is a derivation,
  which cannot be inner as the module is commutative ($ay = ya$).
- e.g. 2: $A = \ell^1(Z_+)$, $Y = A\hat{\otimes}A$, where D is given by

$$Da = a \otimes 1 - 1 \otimes a.$$

  This derivation is always inner, as

$$D(a) = a\cdot(1 \otimes 1) - (1 \otimes 1)\cdot a.$$

2.6 Examples of Derivations

- e.g. 1: $A = \ell^1(Z_+)$, $Y = \mathbf{C}_0$, given by the character at 0.

$$Df = f'(0)$$

  is a derivation,
  which cannot be inner as the module is commutative $(ay = ya)$.

- e.g. 2: $A = \ell^1(Z_+)$, $Y = A\hat{\otimes}A$, where D is given by

$$Da = a \otimes 1 - 1 \otimes a.$$

  This derivation is always inner, as

$$D(a) = a\cdot(1 \otimes 1) - (1 \otimes 1)\cdot a.$$

- e.g. 3: $A = \ell^1(Z_+)$, $Y = \mathrm{Ker}\,\pi \subset A\hat{\otimes}A$, where D is given by

$$Da = a \otimes 1 - 1 \otimes a.$$

  This derivation is *not* inner

3.1 Injective Motivation

If a derivation $D : A \to Y$ is not inner, then
Who is to blame: the derivation or the module?

If a derivation $D : A \to Y$ is not inner, then
Who is to blame: the derivation or the module?

The module is the blame

If a derivation $D : A \to Y$ is not inner, then
Who is to blame: the derivation or the module?

The module is the blame

- Every bimodule Y is canonically embedded in $L(A\hat{\otimes}A; Y)$,

If a derivation $D : A \to Y$ is not inner, then
Who is to blame: the derivation or the module?

The module is the blame

- Every bimodule Y is canonically embedded in $L(A \hat{\otimes} A; Y)$,
- ... by $y \mapsto ((a \otimes b) \mapsto b.y.a)$ [Ex]

3.1 Injective Motivation

If a derivation $D : A \to Y$ is not inner, then
Who is to blame: the derivation or the module?

The module is the blame

- Every bimodule Y is canonically embedded in $L(A\hat{\otimes}A; Y)$,
- ... by $y \mapsto ((a \otimes b) \mapsto b.y.a)$ [Ex]
- **Theorem:** Every derivation $D : A \to Y$ induces a derivation $\tilde{D} : A \to L(A\hat{\otimes}A; Y)$, which is inner

3.1 Injective Motivation

If a derivation $D : A \to Y$ is not inner, then
Who is to blame: the derivation or the module?

The module is the blame

- Every bimodule Y is canonically embedded in $L(A\hat{\otimes}A; Y)$,
- . . . by $y \mapsto ((a \otimes b) \mapsto b.y.a)$ [Ex]
- **Theorem:** Every derivation $D : A \to Y$ induces a derivation $\tilde{D} : A \to L(A\hat{\otimes}A; Y)$, which is inner
- *Proof:* [Ex] or by later theory "$L(A\hat{\otimes}A; Y)$ is cofree, so biinjective"

3.1 Injective Motivation

If a derivation $D : A \to Y$ is not inner, then
Who is to blame: the derivation or the module?

The module is the blame

- Every bimodule Y is canonically embedded in $L(A\hat{\otimes}A; Y)$,
- ... by $y \mapsto ((a \otimes b) \mapsto b.y.a)$ [Ex]
- **Theorem:** Every derivation $D : A \to Y$ induces a derivation $\tilde{D} : A \to L(A\hat{\otimes}A; Y)$, which is inner
- *Proof:* [Ex] or by later theory "$L(A\hat{\otimes}A; Y)$ is cofree, so biinjective"
- `Fact:` all dual modules for amenable algebras are biinjective

3.2 Injective modules - Definition

Defining properties

3.2 Injective modules - Definition

Defining properties

- The Hahn-Banach Theorem for $E \subseteq F$ says that every linear map

  $L(E, \mathbf{C})$ extends, or *lifts* to a map from F:

3.2 Injective modules - Definition

Defining properties

- The Hahn-Banach Theorem for $E \subseteq F$ says that every linear map

  $L(E, \mathbf{C})$ extends, or *lifts* to a map from F:

Defining properties

- The Hahn-Banach Theorem for $E \subseteq F$ says that every linear map $L(E, \mathbf{C})$ extends, or *lifts* to a map from F: $\begin{array}{ccc} F & & \\ \uparrow & \searrow & \\ E & \to & \mathbf{C} \end{array}$

3.2 Injective modules - Definition

Defining properties

- The Hahn-Banach Theorem for $E \subseteq F$ says that every linear map $L(E, \mathbf{C})$ extends, or *lifts* to a map from F:
$$\begin{array}{ccc} F & & \\ \uparrow & \searrow & \\ E & \rightarrow & \mathbf{C} \end{array}$$
- We say the module I is *injective*, if for every admissible $Z \hookrightarrow Y$, $\theta \in h_A(Z, I)$ lifts to $\tilde{\theta} \in h_A(Y, I)$:

3.2 Injective modules - Definition

Defining properties

- The Hahn-Banach Theorem for $E \subseteq F$ says that every linear map $L(E, \mathbf{C})$ extends, or *lifts* to a map from F:
$$\begin{array}{ccc} F & & \\ \uparrow & \searrow & \\ E & \rightarrow & \mathbf{C} \end{array}$$
- We say the module I is *injective*, if for every admissible $Z \hookrightarrow Y$, $\theta \in h_A(Z, I)$ lifts to $\tilde{\theta} \in h_A(Y, I)$:

3.2 Injective modules - Definition

Defining properties

- The Hahn-Banach Theorem for $E \subseteq F$ says that every linear map $L(E, \mathbf{C})$ extends, or *lifts* to a map from F:
$$\begin{array}{ccc} F & & \\ \uparrow & \searrow & \\ E & \rightarrow & \mathbf{C} \end{array}$$
- We say the module I is *injective*, if for every admissible $Z \hookrightarrow Y$, $\theta \in h_A(Z, I)$ lifts to $\tilde{\theta} \in h_A(Y, I)$:
$$\begin{array}{ccc} Y & & \\ \uparrow & \searrow & \\ Z & \rightarrow & I \end{array}$$

3.2 Injective modules - Definition

Defining properties

- The Hahn-Banach Theorem for $E \subseteq F$ says that every linear map $L(E, \mathbf{C})$ extends, or *lifts* to a map from F:
$$\begin{array}{ccc} F & & \\ \uparrow & \searrow & \\ E & \rightarrow & \mathbf{C} \end{array}$$
- We say the module I is *injective*, if for every admissible $Z \hookrightarrow Y$, $\theta \in h_A(Z, I)$ lifts to $\tilde{\theta} \in h_A(Y, I)$:
$$\begin{array}{ccc} Y & & \\ \uparrow & \searrow & \\ Z & \rightarrow & I \end{array}$$
- A module is called *projective* is we can lift maps, if for every admissible $Y \twoheadrightarrow X$:
$$\begin{array}{ccc} & & Y \\ & \nearrow & \downarrow \\ P & \rightarrow & X \end{array}$$

3.2 Injective modules - Definition

Defining properties

- The Hahn-Banach Theorem for $E \subseteq F$ says that every linear map $L(E, \mathbf{C})$ extends, or *lifts* to a map from F:
$$\begin{array}{ccc} F & & \\ \uparrow & \searrow & \\ E & \rightarrow & \mathbf{C} \end{array}$$
- We say the module I is *injective*, if for every admissible $Z \hookrightarrow Y$, $\theta \in h_A(Z, I)$ lifts to $\tilde{\theta} \in h_A(Y, I)$:
$$\begin{array}{ccc} Y & & \\ \uparrow & \searrow & \\ Z & \rightarrow & I \end{array}$$
- A module is called *projective* is we can lift maps, if for every admissible $Y \twoheadrightarrow X$:
$$\begin{array}{ccc} & & Y \\ & \nearrow & \downarrow \\ P & \rightarrow & X \end{array}$$

3.2 Injective modules - Definition

Defining properties

- The Hahn-Banach Theorem for $E \subseteq F$ says that every linear map $L(E, \mathbf{C})$ extends, or *lifts* to a map from F: $\begin{array}{ccc} F & & \\ \uparrow & \searrow & \\ E & \rightarrow & \mathbf{C} \end{array}$
- We say the module I is *injective*, if for every admissible $Z \hookrightarrow Y$, $\theta \in h_A(Z, I)$ lifts to $\tilde{\theta} \in h_A(Y, I)$: $\begin{array}{ccc} Y & & \\ \uparrow & \searrow & \\ Z & \rightarrow & I \end{array}$
- A module is called *projective* is we can lift maps, if for every admissible $Y \twoheadrightarrow X$: $\begin{array}{ccc} & & Y \\ & \nearrow & \downarrow \\ P & \rightarrow & X \end{array}$
- We say F is a *flat module* iff F' is injective.

3.2 Injective modules - Definition

Defining properties

- The Hahn-Banach Theorem for $E \subseteq F$ says that every linear map $L(E, \mathbf{C})$ extends, or *lifts* to a map from F:
$$\begin{array}{ccc} F & & \\ \uparrow & \searrow & \\ E & \rightarrow & \mathbf{C} \end{array}$$
- We say the module I is *injective*, if for every admissible $Z \hookrightarrow Y$, $\theta \in h_A(Z, I)$ lifts to $\tilde{\theta} \in h_A(Y, I)$:
$$\begin{array}{ccc} Y & & \\ \uparrow & \searrow & \\ Z & \rightarrow & I \end{array}$$
- A module is called *projective* is we can lift maps, if for every admissible $Y \twoheadrightarrow X$:
$$\begin{array}{ccc} & & Y \\ & \nearrow & \downarrow \\ P & \rightarrow & X \end{array}$$
- We say F is a *flat module* iff F' is injective.

3.2 Injective modules - Definition

Defining properties

- The Hahn-Banach Theorem for $E \subseteq F$ says that every linear map $L(E, \mathbf{C})$ extends, or *lifts* to a map from F:
$$\begin{array}{ccc} F & & \\ \uparrow & \searrow & \\ E & \rightarrow & \mathbf{C} \end{array}$$
- We say the module I is *injective*, if for every admissible $Z \hookrightarrow Y$, $\theta \in h_A(Z, I)$ lifts to $\tilde{\theta} \in h_A(Y, I)$:
$$\begin{array}{ccc} Y & & \\ \uparrow & \searrow & \\ Z & \rightarrow & I \end{array}$$
- A module is called *projective* is we can lift maps, if for every admissible $Y \twoheadrightarrow X$:
$$\begin{array}{ccc} & & Y \\ & \nearrow & \downarrow \\ P & \rightarrow & X \end{array}$$
- We say F is a *flat module* iff F' is injective. Projective implies flat [Ex]

3.2 Injective modules - Definition

Defining properties

- The Hahn-Banach Theorem for $E \subseteq F$ says that every linear map $L(E, \mathbf{C})$ extends, or *lifts* to a map from F:
$$\begin{array}{ccc} F & & \\ \uparrow & \searrow & \\ E & \rightarrow & \mathbf{C} \end{array}$$
- We say the module I is *injective*, if for every admissible $Z \hookrightarrow Y$, $\theta \in h_A(Z, I)$ lifts to $\tilde{\theta} \in h_A(Y, I)$:
$$\begin{array}{ccc} Y & & \\ \uparrow & \searrow & \\ Z & \rightarrow & I \end{array}$$
- A module is called *projective* is we can lift maps, if for every admissible $Y \twoheadrightarrow X$:
$$\begin{array}{ccc} & & Y \\ & \nearrow & \downarrow \\ P & \rightarrow & X \end{array}$$
- We say F is a *flat module* iff F' is injective. Projective implies flat [Ex]

3.3 Injective modules - Properties

Basic Properties of Injective modules

Basic Properties of Injective modules

- **Theorem:** The right module $I \cong L(A, E)$ is injective

3.3 Injective modules - Properties

Basic Properties of Injective modules

- **Theorem:** The right module $I \cong L(A, E)$ is injective
- PROOF . . . [Ex]

Basic Properties of Injective modules

- **Theorem:** The right module $I \cong L(A, E)$ is injective
- PROOF . . . [Ex]
- **Theorem:** The right module I is injective iff it is a module summand in $I \oplus I_2 \cong L(A, I)$

Basic Properties of Injective modules

- **Theorem:** The right module $I \cong L(A, E)$ is injective
- PROOF . . . [Ex]
- **Theorem:** The right module I is injective iff it is a module summand in $I \oplus I_2 \cong L(A, I)$
- PROOF . . . [Ex]

3.3 Injective modules - Properties

Basic Properties of Injective modules

- **Theorem:** The right module $I \cong L(A, E)$ is injective
- PROOF ... [Ex]
- **Theorem:** The right module I is injective iff it is a module summand in $I \oplus I_2 \cong L(A, I)$
- PROOF ... [Ex]
- `Aside:` the projection takes the place of *averaging* arguments in applications

3.3 Injective modules - Properties

Basic Properties of Injective modules

- **Theorem:** The right module $I \cong L(A, E)$ is injective
- PROOF . . . [Ex]
- **Theorem:** The right module I is injective iff it is a module summand in $I \oplus I_2 \cong L(A, I)$
- PROOF . . . [Ex]
- `Aside:` the projection takes the place of *averaging* arguments in applications

3.3 Injective modules - Properties

Basic Properties of Injective modules

- **Theorem:** The right module $I \cong L(A, E)$ is injective
- PROOF ... [Ex]
- **Theorem:** The right module I is injective iff it is a module summand in $I \oplus I_2 \cong L(A, I)$
- PROOF ... [Ex]
- `Aside:` the projection takes the place of *averaging* arguments in applications

Basic Properties of Projective modules

3.3 Injective modules - Properties

Basic Properties of Injective modules

- **Theorem:** The right module $I \cong L(A, E)$ is injective
- PROOF . . . [Ex]
- **Theorem:** The right module I is injective iff it is a module summand in $I \oplus I_2 \cong L(A, I)$
- PROOF . . . [Ex]
- `Aside:` the projection takes the place of *averaging* arguments in applications

Basic Properties of Projective modules

- **Theorem:** The left module $P \cong A\hat{\otimes}E$ is projective

3.3 Injective modules - Properties

Basic Properties of Injective modules

- **Theorem:** The right module $I \cong L(A, E)$ is injective
- PROOF ... [Ex]
- **Theorem:** The right module I is injective iff it is a module summand in $I \oplus I_2 \cong L(A, I)$
- PROOF ... [Ex]
- `Aside:` the projection takes the place of *averaging* arguments in applications

Basic Properties of Projective modules

- **Theorem:** The left module $P \cong A \hat{\otimes} E$ is projective
- PROOF ... [Ex]

3.3 Injective modules - Properties

Basic Properties of Injective modules

- **Theorem:** The right module $I \cong L(A, E)$ is injective
- PROOF . . . [Ex]
- **Theorem:** The right module I is injective iff it is a module summand in $I \oplus I_2 \cong L(A, I)$
- PROOF . . . [Ex]
- `Aside:` the projection takes the place of *averaging* arguments in applications

Basic Properties of Projective modules

- **Theorem:** The left module $P \cong A \hat{\otimes} E$ is projective
- PROOF . . . [Ex]
- **Theorem:** The left module P is projective iff it is a module summand in $P \oplus P_2 \cong A \hat{\otimes} P$ [Ex]

3.3 Injective modules - Properties

Basic Properties of Injective modules

- **Theorem:** The right module $I \cong L(A, E)$ is injective
- PROOF . . . [Ex]
- **Theorem:** The right module I is injective iff it is a module summand in $I \oplus I_2 \cong L(A, I)$
- PROOF . . . [Ex]
- `Aside:` the projection takes the place of *averaging* arguments in applications

Basic Properties of Projective modules

- **Theorem:** The left module $P \cong A\hat{\otimes}E$ is projective
- PROOF . . . [Ex]
- **Theorem:** The left module P is projective iff it is a module summand in $P \oplus P_2 \cong A\hat{\otimes}P$ [Ex]

3.3 Injective modules - Properties

Basic Properties of Injective modules

- **Theorem:** The right module $I \cong L(A, E)$ is injective
- PROOF . . . [Ex]
- **Theorem:** The right module I is injective iff it is a module summand in $I \oplus I_2 \cong L(A, I)$
- PROOF . . . [Ex]
- `Aside:` the projection takes the place of *averaging* arguments in applications

Basic Properties of Projective modules

- **Theorem:** The left module $P \cong A \hat{\otimes} E$ is projective
- PROOF . . . [Ex]
- **Theorem:** The left module P is projective iff it is a module summand in $P \oplus P_2 \cong A \hat{\otimes} P$ [Ex]

Biinjective definitions come for free

Biinjective definitions come for free

- Recall that $A\hat{\otimes}A = A^{ev}$ can be made into an algebra by using the product:

Biinjective definitions come for free

- Recall that $A \hat{\otimes} A = A^{ev}$ can be made into an algebra by using the product:

Biinjective definitions come for free

- Recall that $A\hat{\otimes}A = A^{ev}$ can be made into an algebra by using the product: $(a_1 \otimes b_1) \times (a_2 \otimes b_2) = (a_1 a_2 \otimes b_2 b_1)$

Biinjective definitions come for free

- Recall that $A\hat{\otimes}A = A^{ev}$ can be made into an algebra by using the product: $(a_1 \otimes b_1) \times (a_2 \otimes b_2) = (a_1 a_2 \otimes b_2 b_1)$ [Ex]

Biinjective definitions come for free

- Recall that $A\hat{\otimes}A = A^{ev}$ can be made into an algebra by using the product: $(a_1 \otimes b_1) \times (a_2 \otimes b_2) = (a_1 a_2 \otimes b_2 b_1)$ [Ex]
- Any A bimodule Y if now a right A^{ev} module by:

Biinjective definitions come for free

- Recall that $A\hat{\otimes}A = A^{ev}$ can be made into an algebra by using the product: $(a_1 \otimes b_1) \times (a_2 \otimes b_2) = (a_1 a_2 \otimes b_2 b_1)$ [Ex]
- Any A bimodule Y if now a right A^{ev} module by:

3.4 Bimodules

Biinjective definitions come for free

- Recall that $A\hat{\otimes}A = A^{ev}$ can be made into an algebra by using the product: $(a_1 \otimes b_1) \times (a_2 \otimes b_2) = (a_1 a_2 \otimes b_2 b_1)$ [Ex]
- Any A bimodule Y if now a right A^{ev} module by: $y(a \otimes b) = bya$

3.4 Bimodules

Biinjective definitions come for free

- Recall that $A\hat{\otimes}A = A^{ev}$ can be made into an algebra by using the product: $(a_1 \otimes b_1) \times (a_2 \otimes b_2) = (a_1 a_2 \otimes b_2 b_1)$ [Ex]
- Any A bimodule Y if now a right A^{ev} module by: $y(a \otimes b) = bya$
- We can now define a module to be *biinjective* if it is right injective as an A^{ev} module

3.4 Bimodules

Biinjective definitions come for free

- Recall that $A\hat{\otimes}A = A^{ev}$ can be made into an algebra by using the product: $(a_1 \otimes b_1) \times (a_2 \otimes b_2) = (a_1 a_2 \otimes b_2 b_1)$ [Ex]
- Any A bimodule Y if now a right A^{ev} module by: $y(a \otimes b) = bya$
- We can now define a module to be *biinjective* if it is right injective as an A^{ev} module
- *Similarly*, we can consider left module to be right modules over A^{op}, and vice versa

3.4 Bimodules

Biinjective definitions come for free

- Recall that $A\hat{\otimes}A = A^{ev}$ can be made into an algebra by using the product: $(a_1 \otimes b_1) \times (a_2 \otimes b_2) = (a_1 a_2 \otimes b_2 b_1)$ [Ex]
- Any A bimodule Y if now a right A^{ev} module by: $y(a \otimes b) = bya$
- We can now define a module to be *biinjective* if it is right injective as an A^{ev} module
- *Similarly*, we can consider left module to be right modules over A^{op}, and vice versa
- It is natural to ask:
  if I is left injective and right injective, then is it biinjective?

3.4 Bimodules

Biinjective definitions come for free

- Recall that $A\hat{\otimes}A = A^{ev}$ can be made into an algebra by using the product: $(a_1 \otimes b_1) \times (a_2 \otimes b_2) = (a_1 a_2 \otimes b_2 b_1)$ [Ex]
- Any A bimodule Y if now a right A^{ev} module by: $y(a \otimes b) = bya$
- We can now define a module to be *biinjective* if it is right injective as an A^{ev} module
- *Similarly*, we can consider left module to be right modules over A^{op}, and vice versa
- It is natural to ask:
  if I is left injective and right injective, then is it biinjective?
- $L(A\hat{\otimes}A, I) \cong L(A, L(A, I)) \cong L(A, I \oplus I_1) \cong L(A, I) \oplus L(A, I_1) \cong I \oplus I_2 \oplus L(A, I_1)$ [Ex]

3.4 Bimodules

Biinjective definitions come for free

- Recall that $A\hat{\otimes}A = A^{ev}$ can be made into an algebra by using the product: $(a_1 \otimes b_1) \times (a_2 \otimes b_2) = (a_1 a_2 \otimes b_2 b_1)$ [Ex]
- Any A bimodule Y if now a right A^{ev} module by: $y(a \otimes b) = bya$
- We can now define a module to be *biinjective* if it is right injective as an A^{ev} module
- *Similarly*, we can consider left module to be right modules over A^{op}, and vice versa
- It is natural to ask:
  if I is left injective and right injective, then is it biinjective?
- $L(A\hat{\otimes}A, I) \cong L(A, L(A, I)) \cong L(A, I \oplus I_1) \cong L(A, I) \oplus L(A, I_1) \cong I \oplus I_2 \oplus L(A, I_1)$ [Ex]

Biinjective definitions come for free

- Recall that $A\hat{\otimes}A = A^{ev}$ can be made into an algebra by using the product: $(a_1 \otimes b_1) \times (a_2 \otimes b_2) = (a_1 a_2 \otimes b_2 b_1)$ [Ex]
- Any A bimodule Y if now a right A^{ev} module by: $y(a \otimes b) = bya$
- We can now define a module to be *biinjective* if it is right injective as an A^{ev} module
- *Similarly*, we can consider left module to be right modules over A^{op}, and vice versa
- It is natural to ask:
  if I is left injective and right injective, then is it biinjective?
- $L(A\hat{\otimes}A, I) \cong L(A, L(A, I)) \cong L(A, I \oplus I_1) \cong L(A, I) \oplus L(A, I_1) \cong I \oplus I_2 \oplus L(A, I_1)$ [Ex]

3.5 Averaging with Biinjectives

Derivations into biinjective modules are inner

3.5 Averaging with Biinjectives

Derivations into biinjective modules are inner

- **Proof:**

3.5 Averaging with Biinjectives

Derivations into biinjective modules are inner

- **Proof:**
- $h_y(b, c) := cyb$ plays the role of the inner derivation

3.5 Averaging with Biinjectives

Derivations into biinjective modules are inner

- **Proof:**
- $h_y(b,c) := cyb$ plays the role of the inner derivation
- Recall that $\bar{h_y} = y$ for the bimodule map $\bar{\cdot} : L(A\hat{\otimes}A; Y) \to Y$

Derivations into biinjective modules are inner

- **Proof:**
- $h_y(b, c) := cyb$ plays the role of the inner derivation
- Recall that $\bar{h_y} = y$ for the bimodule map $\bar{\cdot} : L(A\hat{\otimes}A; Y) \to Y$
- $g(b, c) := c.D(b)$ plays the role of the linear map to be averaged

Derivations into biinjective modules are inner

- **Proof:**
- $h_y(b,c) := cyb$ plays the role of the inner derivation
- Recall that $\bar{h_y} = y$ for the bimodule map $\bar{\cdot} : L(A\hat{\otimes}A; Y) \to Y$
- $g(b,c) := c.D(b)$ plays the role of the linear map to be averaged
- $(a.g)(b,c) = g(b,ca) = ca.D(b)$

Derivations into biinjective modules are inner

- **Proof:**
- $h_y(b,c) := cyb$ plays the role of the inner derivation
- Recall that $\bar{h_y} = y$ for the bimodule map $\bar{\cdot} : L(A\hat{\otimes}A; Y) \to Y$
- $g(b,c) := c.D(b)$ plays the role of the linear map to be averaged
- $(a.g)(b,c) = g(b,ca) = ca.D(b)$
- $(g.a)(b,c) = g(ab,c) = c.D(ab) = ca.D(b) + c.D(a).b$

3.5 Averaging with Biinjectives

Derivations into biinjective modules are inner

- **Proof:**
- $h_y(b,c) := cyb$ plays the role of the inner derivation
- Recall that $\overline{h_y} = y$ for the bimodule map $\overline{\cdot} : L(A\hat{\otimes}A; Y) \to Y$
- $g(b,c) := c.D(b)$ plays the role of the linear map to be averaged
- $(a.g)(b,c) = g(b,ca) = ca.D(b)$
- $(g.a)(b,c) = g(ab,c) = c.D(ab) = ca.D(b) + c.D(a).b$
- $[(a.g) - (g.a)](b,c) = h_{D(a)}(b,c)$

Derivations into biinjective modules are inner

- **Proof:**
- $h_y(b,c) := cyb$ plays the role of the inner derivation
- Recall that $\bar{h_y} = y$ for the bimodule map $\bar{\cdot} : L(A\hat{\otimes}A; Y) \to Y$
- $g(b,c) := c.D(b)$ plays the role of the linear map to be averaged
- $(a.g)(b,c) = g(b,ca) = ca.D(b)$
- $(g.a)(b,c) = g(ab,c) = c.D(ab) = ca.D(b) + c.D(a).b$
- $[(a.g) - (g.a)](b,c) = h_{D(a)}(b,c)$
- $a.\bar{g} - \bar{g}.a = D(a)$

3.5 Averaging with Biinjectives

Derivations into biinjective modules are inner

- **Proof:**
- $h_y(b,c) := cyb$ plays the role of the inner derivation
- Recall that $\bar{h_y} = y$ for the bimodule map $\bar{\cdot} : L(A\hat{\otimes}A; Y) \to Y$
- $g(b,c) := c.D(b)$ plays the role of the linear map to be averaged
- $(a.g)(b,c) = g(b,ca) = ca.D(b)$
- $(g.a)(b,c) = g(ab,c) = c.D(ab) = ca.D(b) + c.D(a).b$
- $[(a.g) - (g.a)](b,c) = h_{D(a)}(b,c)$
- $a.\bar{g} - \bar{g}.a = D(a)$

3.5 Averaging with Biinjectives

Derivations into biinjective modules are inner

- **Proof:**
- $h_y(b,c) := cyb$ plays the role of the inner derivation
- Recall that $\bar{h_y} = y$ for the bimodule map $\bar{\cdot} : L(A\hat{\otimes}A; Y) \to Y$
- $g(b,c) := c.D(b)$ plays the role of the linear map to be averaged
- $(a.g)(b,c) = g(b,ca) = ca.D(b)$
- $(g.a)(b,c) = g(ab,c) = c.D(ab) = ca.D(b) + c.D(a).b$
- $[(a.g) - (g.a)](b,c) = h_{D(a)}(b,c)$
- $a.\bar{g} - \bar{g}.a = D(a)$

3.6 Amenable gives Biinjectives

Dual modules over amenable algebras are injective

Dual modules over amenable algebras are injective

- Recall a Banach algebra A is *amenable* if it has an `approximate diagonal`: a bounded net $m_\lambda \in A\hat{\otimes}A$, so that

3.6 Amenable gives Biinjectives

Dual modules over amenable algebras are injective

- Recall a Banach algebra A is *amenable* if it has an `approximate diagonal`: a bounded net $m_\lambda \in A \hat{\otimes} A$, so that
- $a.m_\lambda - m_\lambda.a \to 0$ and $\pi(m_\lambda).a \to a$

3.6 Amenable gives Biinjectives

Dual modules over amenable algebras are injective

- Recall a Banach algebra A is *amenable* if it has an `approximate diagonal`: a bounded net $m_\lambda \in A\hat{\otimes}A$, so that
- $a.m_\lambda - m_\lambda.a \to 0$ and $\pi(m_\lambda).a \to a$
- **Proof:**

3.6 Amenable gives Biinjectives

Dual modules over amenable algebras are injective

- Recall a Banach algebra A is *amenable* if it has an `approximate diagonal`: a bounded net $m_\lambda \in A\hat{\otimes}A$, so that
- $a.m_\lambda - m_\lambda.a \to 0$ and $\pi(m_\lambda).a \to a$
- **Proof:**

Dual modules over amenable algebras are injective

- Recall a Banach algebra A is *amenable* if it has an `approximate diagonal`: a bounded net $m_\lambda \in A\hat{\otimes}A$, so that
- $a.m_\lambda - m_\lambda.a \to 0$ and $\pi(m_\lambda).a \to a$
- **Proof:** Denote the approximate diagonal by $m_\lambda = \sum_{i=1}^{\infty} a_i^\lambda \otimes b_i^\lambda$ and assume the net to be an ultrafilter (or take subnets later)

3.6 Amenable gives Biinjectives

Dual modules over amenable algebras are injective

- Recall a Banach algebra A is *amenable* if it has an approximate diagonal: a bounded net $m_\lambda \in A\hat{\otimes}A$, so that
- $a.m_\lambda - m_\lambda.a \to 0$ and $\pi(m_\lambda).a \to a$
- **Proof:** Denote the approximate diagonal by $m_\lambda = \sum_{i=1}^{\infty} a_i^\lambda \otimes b_i^\lambda$ and assume the net to be an ultrafilter (or take subnets later)
- Given a left module X with dual right module X'

3.6 Amenable gives Biinjectives

Dual modules over amenable algebras are injective

- Recall a Banach algebra A is *amenable* if it has an `approximate diagonal`: a bounded net $m_\lambda \in A\hat{\otimes}A$, so that
- $a.m_\lambda - m_\lambda.a \to 0$ and $\pi(m_\lambda).a \to a$
- **Proof:** Denote the approximate diagonal by $m_\lambda = \sum_{i=1}^{\infty} a_i^\lambda \otimes b_i^\lambda$ and assume the net to be an ultrafilter (or take subnets later)
- Given a left module X with dual right module X'
- We define a module projection $\bar{\cdot} : L(A, X') \to X'$ by

3.6 Amenable gives Biinjectives

Dual modules over amenable algebras are injective

- Recall a Banach algebra A is *amenable* if it has an `approximate diagonal`: a bounded net $m_\lambda \in A\hat{\otimes}A$, so that
- $a.m_\lambda - m_\lambda.a \to 0$ and $\pi(m_\lambda).a \to a$
- **Proof:** Denote the approximate diagonal by $m_\lambda = \sum_{i=1}^{\infty} a_i^\lambda \otimes b_i^\lambda$ and assume the net to be an ultrafilter (or take subnets later)
- Given a left module X with dual right module X'
- We define a module projection $\bar{\cdot} : L(A, X') \to X'$ by
- $\bar{T}(x) = \lim_\lambda \sum_{i=1}^{\infty} T(a_i^\lambda)(b_i^\lambda x)$

3.6 Amenable gives Biinjectives

Dual modules over amenable algebras are injective

- Recall a Banach algebra A is *amenable* if it has an `approximate diagonal`: a bounded net $m_\lambda \in A\hat{\otimes}A$, so that
- $a.m_\lambda - m_\lambda.a \to 0$ and $\pi(m_\lambda).a \to a$
- **Proof:** Denote the approximate diagonal by $m_\lambda = \sum_{i=1}^{\infty} a_i^\lambda \otimes b_i^\lambda$ and assume the net to be an ultrafilter (or take subnets later)
- Given a left module X with dual right module X'
- We define a module projection $\bar{\cdot} : L(A, X') \to X'$ by
- $\bar{T}(x) = \lim_\lambda \sum_{i=1}^{\infty} T(a_i^\lambda)(b_i^\lambda x)$
- $\bar{\cdot}$ is a module map: $(\overline{T.c})(x) = \lim_\lambda \sum_{i=1}^{\infty}(T.c)(a_i^\lambda)(b_i^\lambda x) = \lim_\lambda \sum_{i=1}^{\infty} T(ca_i^\lambda)(b_i^\lambda x) = \lim_\lambda \sum_{i=1}^{\infty} T(a_i^\lambda)(b_i^\lambda cx) = \bar{T}(c.x)$

3.6 Amenable gives Biinjectives

Dual modules over amenable algebras are injective

- Recall a Banach algebra A is *amenable* if it has an `approximate diagonal`: a bounded net $m_\lambda \in A\hat{\otimes}A$, so that
- $a.m_\lambda - m_\lambda.a \to 0$ and $\pi(m_\lambda).a \to a$
- **Proof:** Denote the approximate diagonal by $m_\lambda = \sum_{i=1}^{\infty} a_i^\lambda \otimes b_i^\lambda$ and assume the net to be an ultrafilter (or take subnets later)
- Given a left module X with dual right module X'
- We define a module projection $\bar{\cdot} : L(A, X') \to X'$ by
- $\bar{T}(x) = \lim_\lambda \sum_{i=1}^{\infty} T(a_i^\lambda)(b_i^\lambda x)$
- $\bar{\cdot}$ is a module map: $(\overline{T.c})(x) = \lim_\lambda \sum_{i=1}^{\infty}(T.c)(a_i^\lambda)(b_i^\lambda x) = \lim_\lambda \sum_{i=1}^{\infty} T(ca_i^\lambda)(b_i^\lambda x) = \lim_\lambda \sum_{i=1}^{\infty} T(a_i^\lambda)(b_i^\lambda cx) = \bar{T}(c.x)$
- $\bar{\cdot}$ is a projection: recall $h_y(a) = ya$ $(\overline{h_y})(x) = \lim_\lambda \sum_{i=1}^{\infty}(h_y)(a_i^\lambda)(b_i^\lambda x) = \lim_\lambda \sum_{i=1}^{\infty}(y.a_i^\lambda)(b_i^\lambda x) = \lim_\lambda \sum_{i=1}^{\infty} y(a_i^\lambda.b_i^\lambda x) = \lim_\lambda y(\pi(m_\lambda)x) = y(x) = h_y(x)$

3.6 Amenable gives Biinjectives

Dual modules over amenable algebras are injective

- Recall a Banach algebra A is *amenable* if it has an `approximate diagonal`: a bounded net $m_\lambda \in A\hat{\otimes}A$, so that
- $a.m_\lambda - m_\lambda.a \to 0$ and $\pi(m_\lambda).a \to a$
- **Proof:** Denote the approximate diagonal by $m_\lambda = \sum_{i=1}^{\infty} a_i^\lambda \otimes b_i^\lambda$ and assume the net to be an ultrafilter (or take subnets later)
- Given a left module X with dual right module X'
- We define a module projection $\bar{\cdot} : L(A, X') \to X'$ by
- $\bar{T}(x) = \lim_\lambda \sum_{i=1}^{\infty} T(a_i^\lambda)(b_i^\lambda x)$
- $\bar{\cdot}$ is a module map: $(\overline{T.c})(x) = \lim_\lambda \sum_{i=1}^{\infty}(T.c)(a_i^\lambda)(b_i^\lambda x) = \lim_\lambda \sum_{i=1}^{\infty} T(ca_i^\lambda)(b_i^\lambda x) = \lim_\lambda \sum_{i=1}^{\infty} T(a_i^\lambda)(b_i^\lambda cx) = \bar{T}(c.x)$
- $\bar{\cdot}$ is a projection: recall $h_y(a) = ya$ $(\overline{h_y})(x) = \lim_\lambda \sum_{i=1}^{\infty}(h_y)(a_i^\lambda)(b_i^\lambda x) = \lim_\lambda \sum_{i=1}^{\infty}(y.a_i^\lambda)(b_i^\lambda x) = \lim_\lambda \sum_{i=1}^{\infty} y(a_i^\lambda.b_i^\lambda x) = \lim_\lambda y(\pi(m_\lambda)x) = y(x) = h_y(x)$

3.6 Amenable gives Biinjectives

Dual modules over amenable algebras are injective

- Recall a Banach algebra A is *amenable* if it has an `approximate diagonal`: a bounded net $m_\lambda \in A\hat{\otimes}A$, so that
- $a.m_\lambda - m_\lambda.a \to 0$ and $\pi(m_\lambda).a \to a$
- **Proof:** Denote the approximate diagonal by $m_\lambda = \sum_{i=1}^{\infty} a_i^\lambda \otimes b_i^\lambda$ and assume the net to be an ultrafilter (or take subnets later)
- Given a left module X with dual right module X'
- We define a module projection $\bar{\cdot} : L(A, X') \to X'$ by
- $\bar{T}(x) = \lim_\lambda \sum_{i=1}^{\infty} T(a_i^\lambda)(b_i^\lambda x)$
- $\bar{\cdot}$ is a module map: $(\overline{T.c})(x) = \lim_\lambda \sum_{i=1}^{\infty}(T.c)(a_i^\lambda)(b_i^\lambda x) = \lim_\lambda \sum_{i=1}^{\infty} T(ca_i^\lambda)(b_i^\lambda x) = \lim_\lambda \sum_{i=1}^{\infty} T(a_i^\lambda)(b_i^\lambda cx) = \bar{T}(c.x)$
- $\bar{\cdot}$ is a projection: recall $h_y(a) = ya$ $(\overline{h_y})(x) = \lim_\lambda \sum_{i=1}^{\infty}(h_y)(a_i^\lambda)(b_i^\lambda x) = \lim_\lambda \sum_{i=1}^{\infty}(y.a_i^\lambda)(b_i^\lambda x) = \lim_\lambda \sum_{i=1}^{\infty} y(a_i^\lambda.b_i^\lambda x) = \lim_\lambda y(\pi(m_\lambda)x) = y(x) = h_y(x)$

3.7 Averaging without amenability

Averaging with Bounded Approximate Identities (bai)

Averaging with Bounded Approximate Identities (bai)

- **Theorem:** Let J be a left ideal with a brai $\{e_\alpha\}$, then $(A/J)'$ is injective.

3.7 Averaging without amenability

Averaging with Bounded Approximate Identities (bai)

- **Theorem:** Let J be a left ideal with a brai $\{e_\alpha\}$, then $(A/J)'$ is injective.
- **Proof:** We can write $(A/J)'$ as a module direct summand of the injective module A' [Ex]

3.7 Averaging without amenability

Averaging with Bounded Approximate Identities (bai)

- **Theorem:** Let J be a left ideal with a brai $\{e_\alpha\}$, then $(A/J)'$ is injective.
- **Proof:** We can write $(A/J)'$ as a module direct summand of the injective module A' [Ex]
- using the module projection $f \mapsto (a \mapsto \lim_\alpha f(a - ae_\alpha))$ for $f \in A'$;

3.7 Averaging without amenability

Averaging with Bounded Approximate Identities (bai)

- **Theorem:** Let J be a left ideal with a brai $\{e_\alpha\}$, then $(A/J)'$ is injective.
- **Proof:** We can write $(A/J)'$ as a module direct summand of the injective module A' [Ex]
- using the module projection $f \mapsto (a \mapsto \lim_\alpha f(a - ae_\alpha))$ for $f \in A'$;
- There is a partial converse to this for (weakly) complemented ideals

3.7 Averaging without amenability

Averaging with Bounded Approximate Identities (bai)

- **Theorem:** Let J be a left ideal with a brai $\{e_\alpha\}$, then $(A/J)'$ is injective.
- **Proof:** We can write $(A/J)'$ as a module direct summand of the injective module A' [Ex]
- using the module projection $f \mapsto (a \mapsto \lim_\alpha f(a - ae_\alpha))$ for $f \in A'$;
- There is a partial converse to this for (weakly) complemented ideals
- if $(A/J)'$ is injective then J has a brai. [Ex]

3.7 Averaging without amenability

Averaging with Bounded Approximate Identities (bai)

- **Theorem:** Let J be a left ideal with a brai $\{e_\alpha\}$, then $(A/J)'$ is injective.
- **Proof:** We can write $(A/J)'$ as a module direct summand of the injective module A' [Ex]
- using the module projection $f \mapsto (a \mapsto \lim_\alpha f(a - ae_\alpha))$ for $f \in A'$;
- There is a partial converse to this for (weakly) complemented ideals
- if $(A/J)'$ is injective then J has a brai. [Ex]
- If $A' = (A/J)' \oplus J'$, then $A'' = (A/J)'' \oplus J''$ and $1 \in A''$ decomposes

3.7 Averaging without amenability

Averaging with Bounded Approximate Identities (bai)

- **Theorem:** Let J be a left ideal with a brai $\{e_\alpha\}$, then $(A/J)'$ is injective.
- **Proof:** We can write $(A/J)'$ as a module direct summand of the injective module A' [Ex]
- using the module projection $f \mapsto (a \mapsto \lim_\alpha f(a - ae_\alpha))$ for $f \in A'$;
- There is a partial converse to this for (weakly) complemented ideals
- if $(A/J)'$ is injective then J has a brai. [Ex]
- If $A' = (A/J)' \oplus J'$, then $A'' = (A/J)'' \oplus J''$ and $1 \in A''$ decomposes
- Interesting examples are the peak points in uniform algebras

3.7 Averaging without amenability

Averaging with Bounded Approximate Identities (bai)

- **Theorem:** Let J be a left ideal with a brai $\{e_\alpha\}$, then $(A/J)'$ is injective.
- **Proof:** We can write $(A/J)'$ as a module direct summand of the injective module A' [Ex]
- using the module projection $f \mapsto (a \mapsto \lim_\alpha f(a - ae_\alpha))$ for $f \in A'$;
- There is a partial converse to this for (weakly) complemented ideals
- if $(A/J)'$ is injective then J has a brai. [Ex]
- If $A' = (A/J)' \oplus J'$, then $A'' = (A/J)'' \oplus J''$ and $1 \in A''$ decomposes
- Interesting examples are the peak points in uniform algebras

3.7 Averaging without amenability

Averaging with Bounded Approximate Identities (bai)

- **Theorem:** Let J be a left ideal with a brai $\{e_\alpha\}$, then $(A/J)'$ is injective.
- **Proof:** We can write $(A/J)'$ as a module direct summand of the injective module A' [Ex]
- using the module projection $f \mapsto (a \mapsto \lim_\alpha f(a - ae_\alpha))$ for $f \in A'$;
- There is a partial converse to this for (weakly) complemented ideals
- if $(A/J)'$ is injective then J has a brai. [Ex]
- If $A' = (A/J)' \oplus J'$, then $A'' = (A/J)'' \oplus J''$ and $1 \in A''$ decomposes
- Interesting examples are the peak points in uniform algebras

4.1 Quotient and Submodules

How are derivations into modules related to derivations onto submodules and quotient modules?

Derivations into Submodules

How are derivations into modules related to derivations onto submodules and quotient modules?

Derivations into Submodules

- We have already seen that by enlarging the module we can make any derivation inner, $Y \to L(A\hat{\otimes}A, Y)$, but in general it will not become inner

How are derivations into modules related to derivations onto submodules and quotient modules?

Derivations into Submodules

- We have already seen that by enlarging the module we can make any derivation inner, $Y \to L(A\hat{\otimes}A, Y)$, but in general it will not become inner
- $0 \to X \hookrightarrow Y \to Z \to 0$, an admissible s.e.s.

How are derivations into modules related to derivations onto submodules and quotient modules?

Derivations into Submodules

- We have already seen that by enlarging the module we can make any derivation inner, $Y \to L(A\hat{\otimes}A, Y)$, but in general it will not become inner
- $0 \to X \hookrightarrow Y \to Z \to 0$, an admissible s.e.s.
- $\mathcal{H}^1(A, X) \to \mathcal{H}^1(A, Y) \to \mathcal{H}^1(A, Z)$

4.1 Quotient and Submodules

How are derivations into modules related to derivations onto submodules and quotient modules?

Derivations into Submodules

- We have already seen that by enlarging the module we can make any derivation inner, $Y \to L(A\hat{\otimes}A, Y)$, but in general it will not become inner
- $0 \to X \hookrightarrow Y \to Z \to 0$, an admissible s.e.s.
- $\mathcal{H}^1(A, X) \to \mathcal{H}^1(A, Y) \to \mathcal{H}^1(A, Z)$
- This is *exact* at $\mathcal{H}^1(A, Y)$

4.1 Quotient and Submodules

How are derivations into modules related to derivations onto submodules and quotient modules?

Derivations into Submodules

- We have already seen that by enlarging the module we can make any derivation inner, $Y \to L(A\hat{\otimes}A, Y)$, but in general it will not become inner
- $0 \to X \hookrightarrow Y \to Z \to 0$, an admissible s.e.s.
- $\mathcal{H}^1(A, X) \to \mathcal{H}^1(A, Y) \to \mathcal{H}^1(A, Z)$
- This is *exact* at $\mathcal{H}^1(A, Y)$
- **Proof:** ... [Ex]

4.1 Quotient and Submodules

How are derivations into modules related to derivations onto submodules and quotient modules?

Derivations into Submodules

- We have already seen that by enlarging the module we can make any derivation inner, $Y \to L(A\hat{\otimes}A, Y)$, but in general it will not become inner
- $0 \to X \hookrightarrow Y \to Z \to 0$, an admissible s.e.s.
- $\mathcal{H}^1(A, X) \to \mathcal{H}^1(A, Y) \to \mathcal{H}^1(A, Z)$
- This is *exact* at $\mathcal{H}^1(A, Y)$
- **Proof:** . . . [Ex]

How are derivations into modules related to derivations onto submodules and quotient modules?

Derivations into Submodules

- We have already seen that by enlarging the module we can make any derivation inner, $Y \to L(A\hat{\otimes}A, Y)$, but in general it will not become inner
- $0 \to X \hookrightarrow Y \to Z \to 0$, an admissible s.e.s.
- $\mathcal{H}^1(A, X) \to \mathcal{H}^1(A, Y) \to \mathcal{H}^1(A, Z)$
- This is *exact* at $\mathcal{H}^1(A, Y)$
- **Proof:** . . . [Ex]

4.2 Quotient and Submodules

Derivations to Quotients

4.2 Quotient and Submodules

Derivations to Quotients

- Given $D : A \to Y/X$, can we lift it to $\tilde{D} : A \to Y$?

Derivations to Quotients

- Given $D : A \to Y/X$, can we lift it to $\tilde{D} : A \to Y$?
- As a linear map we can do this, but can this be a derivation?
  $\phi(a, b) = a\tilde{D}(b) - \tilde{D}(ab) + \tilde{D}(a)b \in X$

4.2 Quotient and Submodules

Derivations to Quotients

- Given $D : A \to Y/X$, can we lift it to $\tilde{D} : A \to Y$?
- As a linear map we can do this, but can this be a derivation?
  $\phi(a, b) = a\tilde{D}(b) - \tilde{D}(ab) + \tilde{D}(a)b \in X$
- It should be noted for later, that ϕ satisfies the rule

Derivations to Quotients

- Given $D : A \to Y/X$, can we lift it to $\tilde{D} : A \to Y$?
- As a linear map we can do this, but can this be a derivation? $\phi(a, b) = a\tilde{D}(b) - \tilde{D}(ab) + \tilde{D}(a)b \in X$
- It should be noted for later, that ϕ satisfies the rule
- $a\phi(b, c) - \phi(ab, c) + \phi(a, bc) - \phi(a, b)c = 0$ [Ex]

4.2 Quotient and Submodules

Derivations to Quotients

- Given $D : A \to Y/X$, can we lift it to $\tilde{D} : A \to Y$?
- As a linear map we can do this, but can this be a derivation? $\phi(a, b) = a\tilde{D}(b) - \tilde{D}(ab) + \tilde{D}(a)b \in X$
- It should be noted for later, that ϕ satisfies the rule
- $a\phi(b, c) - \phi(ab, c) + \phi(a, bc) - \phi(a, b)c = 0$ [Ex]
- If there exists $\psi : A \to X$ such that

Derivations to Quotients

- Given $D : A \to Y/X$, can we lift it to $\tilde{D} : A \to Y$?
- As a linear map we can do this, but can this be a derivation? $\phi(a, b) = a\tilde{D}(b) - \tilde{D}(ab) + \tilde{D}(a)b \in X$
- It should be noted for later, that ϕ satisfies the rule
- $a\phi(b, c) - \phi(ab, c) + \phi(a, bc) - \phi(a, b)c = 0$ [Ex]
- If there exists $\psi : A \to X$ such that
- $\phi(a, b) = a\psi(b) - \psi(ab) + \psi(a)b$

4.2 Quotient and Submodules

Derivations to Quotients

- Given $D : A \to Y/X$, can we lift it to $\tilde{D} : A \to Y$?
- As a linear map we can do this, but can this be a derivation? $\phi(a, b) = a\tilde{D}(b) - \tilde{D}(ab) + \tilde{D}(a)b \in X$
- It should be noted for later, that ϕ satisfies the rule
- $a\phi(b, c) - \phi(ab, c) + \phi(a, bc) - \phi(a, b)c = 0$ [Ex]
- If there exists $\psi : A \to X$ such that
- $\phi(a, b) = a\psi(b) - \psi(ab) + \psi(a)b$
- then $\bar{D}(a) = \tilde{D}(a) - \psi(a)$ is a derivation $\bar{D} : A \to Y$ [Ex]

4.2 Quotient and Submodules

Derivations to Quotients

- Given $D : A \to Y/X$, can we lift it to $\tilde{D} : A \to Y$?
- As a linear map we can do this, but can this be a derivation? $\phi(a, b) = a\tilde{D}(b) - \tilde{D}(ab) + \tilde{D}(a)b \in X$
- It should be noted for later, that ϕ satisfies the rule
- $a\phi(b, c) - \phi(ab, c) + \phi(a, bc) - \phi(a, b)c = 0$ [Ex]
- If there exists $\psi : A \to X$ such that
- $\phi(a, b) = a\psi(b) - \psi(ab) + \psi(a)b$
- then $\bar{D}(a) = \tilde{D}(a) - \psi(a)$ is a derivation $\bar{D} : A \to Y$ [Ex]
- $\mathcal{H}^1(A, Y) \to \mathcal{H}^1(A, Y/X) \to \mathcal{H}^2(A, X)$

4.2 Quotient and Submodules

Derivations to Quotients

- Given $D : A \to Y/X$, can we lift it to $\tilde{D} : A \to Y$?
- As a linear map we can do this, but can this be a derivation? $\phi(a, b) = a\tilde{D}(b) - \tilde{D}(ab) + \tilde{D}(a)b \in X$
- It should be noted for later, that ϕ satisfies the rule
- $a\phi(b, c) - \phi(ab, c) + \phi(a, bc) - \phi(a, b)c = 0$ [Ex]
- If there exists $\psi : A \to X$ such that
- $\phi(a, b) = a\psi(b) - \psi(ab) + \psi(a)b$
- then $\bar{D}(a) = \tilde{D}(a) - \psi(a)$ is a derivation $\bar{D} : A \to Y$ [Ex]
- $\mathcal{H}^1(A, Y) \to \mathcal{H}^1(A, Y/X) \to \mathcal{H}^2(A, X)$
- This is *exact* at $\mathcal{H}^1(A, Y/X)$ [Ex]

Derivations to Quotients

- Given $D : A \to Y/X$, can we lift it to $\tilde{D} : A \to Y$?
- As a linear map we can do this, but can this be a derivation? $\phi(a,b) = a\tilde{D}(b) - \tilde{D}(ab) + \tilde{D}(a)b \in X$
- It should be noted for later, that ϕ satisfies the rule
- $a\phi(b,c) - \phi(ab,c) + \phi(a,bc) - \phi(a,b)c = 0$ [Ex]
- If there exists $\psi : A \to X$ such that
- $\phi(a,b) = a\psi(b) - \psi(ab) + \psi(a)b$
- then $\bar{D}(a) = \tilde{D}(a) - \psi(a)$ is a derivation $\bar{D} : A \to Y$ [Ex]
- $\mathcal{H}^1(A,Y) \to \mathcal{H}^1(A,Y/X) \to \mathcal{H}^2(A,X)$
- This is *exact* at $\mathcal{H}^1(A,Y/X)$ [Ex]
- **Proof:** ...

Derivations to Quotients

- Given $D : A \to Y/X$, can we lift it to $\tilde{D} : A \to Y$?
- As a linear map we can do this, but can this be a derivation? $\phi(a, b) = a\tilde{D}(b) - \tilde{D}(ab) + \tilde{D}(a)b \in X$
- It should be noted for later, that ϕ satisfies the rule
- $a\phi(b, c) - \phi(ab, c) + \phi(a, bc) - \phi(a, b)c = 0$ [Ex]
- If there exists $\psi : A \to X$ such that
- $\phi(a, b) = a\psi(b) - \psi(ab) + \psi(a)b$
- then $\bar{D}(a) = \tilde{D}(a) - \psi(a)$ is a derivation $\bar{D} : A \to Y$ [Ex]
- $\mathcal{H}^1(A, Y) \to \mathcal{H}^1(A, Y/X) \to \mathcal{H}^2(A, X)$
- This is *exact* at $\mathcal{H}^1(A, Y/X)$ [Ex]
- **Proof:** ...

Derivations to Quotients

- Given $D : A \to Y/X$, can we lift it to $\tilde{D} : A \to Y$?
- As a linear map we can do this, but can this be a derivation? $\phi(a, b) = a\tilde{D}(b) - \tilde{D}(ab) + \tilde{D}(a)b \in X$
- It should be noted for later, that ϕ satisfies the rule
- $a\phi(b, c) - \phi(ab, c) + \phi(a, bc) - \phi(a, b)c = 0$ [Ex]
- If there exists $\psi : A \to X$ such that
- $\phi(a, b) = a\psi(b) - \psi(ab) + \psi(a)b$
- then $\bar{D}(a) = \tilde{D}(a) - \psi(a)$ is a derivation $\bar{D} : A \to Y$ [Ex]
- $\mathcal{H}^1(A, Y) \to \mathcal{H}^1(A, Y/X) \to \mathcal{H}^2(A, X)$
- This is *exact* at $\mathcal{H}^1(A, Y/X)$ [Ex]
- **Proof:** ...

- We define

$$
\begin{aligned}
(\delta y)(f) &:= f\,.\,y - y\,.\,f \\
(\delta\psi)(f,g) &:= f\,.\,\psi(g) - \psi(fg) + \psi(f)\,.\,g \\
(\delta\phi)(f,g,h) &:= f\,.\,\phi(g,h) - \phi(fg,h) + \phi(f,gh) - \phi(f,g)\,.\,h
\end{aligned}
$$

5.1 Second cohomology from 2-forms

- We define

$$\begin{aligned} (\delta y)(f) &:= f.y - y.f \\ (\delta\psi)(f,g) &:= f.\psi(g) - \psi(fg) + \psi(f).g \\ (\delta\phi)(f,g,h) &:= f.\phi(g,h) - \phi(fg,h) + \phi(f,gh) - \phi(f,g).h \end{aligned}$$

- We say ϕ is a 2-*cocycle* if $\delta\phi = 0$

5.1 Second cohomology from 2-forms

- We define

$$\begin{aligned}(\delta y)(f) &:= f.y - y.f\\ (\delta\psi)(f,g) &:= f.\psi(g) - \psi(fg) + \psi(f).g\\ (\delta\phi)(f,g,h) &:= f.\phi(g,h) - \phi(fg,h) + \phi(f,gh) - \phi(f,g).h\end{aligned}$$

- We say ϕ is a 2-*cocycle* if $\delta\phi = 0$
- We say ϕ is a 2-*coboundary* if $\phi = \delta\psi$. (Think $\psi = D$.)

5.1 Second cohomology from 2-forms

- We define

$$\begin{aligned}(\delta y)(f) &:= f.y - y.f \\ (\delta\psi)(f,g) &:= f.\psi(g) - \psi(fg) + \psi(f).g \\ (\delta\phi)(f,g,h) &:= f.\phi(g,h) - \phi(fg,h) + \phi(f,gh) - \phi(f,g).h\end{aligned}$$

- We say ϕ is a 2-*cocycle* if $\delta\phi = 0$
- We say ϕ is a 2-*coboundary* if $\phi = \delta\psi$. (Think $\psi = D$.)
- How can such a 2-cocycle arise?

- We define

$$\begin{aligned}(\delta y)(f) &:= f.y - y.f\\ (\delta\psi)(f,g) &:= f.\psi(g) - \psi(fg) + \psi(f).g\\ (\delta\phi)(f,g,h) &:= f.\phi(g,h) - \phi(fg,h) + \phi(f,gh) - \phi(f,g).h\end{aligned}$$

- We say ϕ is a 2-*cocycle* if $\delta\phi = 0$
- We say ϕ is a 2-*coboundary* if $\phi = \delta\psi$. (Think $\psi = D$.)
- How can such a 2-cocycle arise?
- e.g. $A = \ell^1(Z_+^2)$, $Y = \mathbf{C}_0$, so $f(z,w)\cdot\lambda = f(0,0)\lambda$

$$\phi(f,g) = \left(\frac{\partial f}{\partial z}\frac{\partial g}{\partial w} - \frac{\partial f}{\partial w}\frac{\partial g}{\partial z}\right)\Big|_{(0,0)}.$$

5.1 Second cohomology from 2-forms

- We define

$$\begin{aligned}(\delta y)(f) &:= f\,.\,y - y\,.\,f\\(\delta\psi)(f,g) &:= f\,.\,\psi(g) - \psi(fg) + \psi(f)\,.\,g\\(\delta\phi)(f,g,h) &:= f\,.\,\phi(g,h) - \phi(fg,h) + \phi(f,gh) - \phi(f,g)\,.\,h\end{aligned}$$

- We say ϕ is a 2-*cocycle* if $\delta\phi = 0$
- We say ϕ is a 2-*coboundary* if $\phi = \delta\psi$. (Think $\psi = D$.)
- How can such a 2-cocycle arise?
- e.g. $A = \ell^1(Z_+^2)$, $Y = \mathbf{C}_0$, so $f(z,w)\cdot\lambda = f(0,0)\lambda$

$$\phi(f,g) = \left(\frac{\partial f}{\partial z}\frac{\partial g}{\partial w} - \frac{\partial f}{\partial w}\frac{\partial g}{\partial z}\right)\Big|_{(0,0)}.$$

- Calculation shows this to be a 2-cocycle, which is not zero, as $\phi(z,w) = +1$, and $\phi(w,z) = -1$,

5.1 Second cohomology from 2-forms

- We define

$$\begin{aligned}(\delta y)(f) &:= f.y - y.f\\(\delta\psi)(f,g) &:= f.\psi(g) - \psi(fg) + \psi(f).g\\(\delta\phi)(f,g,h) &:= f.\phi(g,h) - \phi(fg,h) + \phi(f,gh) - \phi(f,g).h\end{aligned}$$

- We say ϕ is a 2-*cocycle* if $\delta\phi = 0$
- We say ϕ is a 2-*coboundary* if $\phi = \delta\psi$. (Think $\psi = D$.)
- How can such a 2-cocycle arise?
- e.g. $A = \ell^1(Z_+^2)$, $Y = \mathbf{C}_0$, so $f(z,w)\cdot\lambda = f(0,0)\lambda$

$$\phi(f,g) = \left(\frac{\partial f}{\partial z}\frac{\partial g}{\partial w} - \frac{\partial f}{\partial w}\frac{\partial g}{\partial z}\right)|_{(0,0)}.$$

- Calculation shows this to be a 2-cocycle, which is not zero, as $\phi(z,w) = +1$, and $\phi(w,z) = -1$,
- Moreover, it cannot cobound, as if $\phi = \delta\psi$, then

$$\phi(f,g) = f.\psi(g) - \psi(fg) + \psi(f)g = \phi(g,f).$$

- We define higher cohomology, $\mathcal{H}^n(A, Y)$ by generalising the 2-cocycle formula to the *complex*

$$\to L^{n-1}(A, Y) \to L^n(A, Y) \to L^{n+1}(A, Y) \to$$

5.2 Higher Cohomology – a quick definition

- We define higher cohomology, $\mathcal{H}^n(A, Y)$ by generalising the 2-cocycle formula to the *complex*

$$\to L^{n-1}(A, Y) \to L^n(A, Y) \to L^{n+1}(A, Y) \to$$

- with coboundary maps

$$\begin{aligned}\delta(T)(a_1, a_2, \ldots, a_n) = \quad & + \quad a_1 T(a_2, a_3, \ldots, a_n) \\ & + \quad \sum_{j=1}^{n-1} (-1)^j T(a_1, \ldots, a_j a_{j+1}, \ldots) \\ & + \quad (-1)^n T(a_1, a_2, \ldots, a_{n-1}) a_n\end{aligned}$$

- We define higher cohomology, $\mathcal{H}^n(A, Y)$ by generalising the 2-cocycle formula to the *complex*

$$\to L^{n-1}(A, Y) \to L^n(A, Y) \to L^{n+1}(A, Y) \to$$

- with coboundary maps

$$\begin{aligned} \delta(T)(a_1, a_2, \ldots, a_n) = \; & + \; a_1 T(a_2, a_3, \ldots, a_n) \\ & + \; \sum_{j=1}^{n-1} (-1)^j T(a_1, \ldots, a_j a_{j+1}, \ldots) \\ & + \; (-1)^n T(a_1, a_2, \ldots, a_{n-1}) a_n \end{aligned}$$

- It is easy to check $\delta^2 = 0$ [Ex], which defines a *chain complex*

5.2 Higher Cohomology – a quick definition

- We define higher cohomology, $\mathcal{H}^n(A, Y)$ by generalising the 2-cocycle formula to the *complex*

$$\to L^{n-1}(A, Y) \to L^n(A, Y) \to L^{n+1}(A, Y) \to$$

- with coboundary maps

$$\begin{aligned}\delta(T)(a_1, a_2, \ldots, a_n) = \quad & + \quad a_1 T(a_2, a_3, \ldots, a_n) \\ & + \quad \sum_{j=1}^{n-1} (-1)^j T(a_1, \ldots, a_j a_{j+1}, \ldots) \\ & + \quad (-1)^n T(a_1, a_2, \ldots, a_{n-1}) a_n\end{aligned}$$

- It is easy to check $\delta^2 = 0$ [Ex], which defines a *chain complex*
- $\mathcal{Z}^n(A) = \mathrm{Ker}\ \delta$, are the *n*-cocycles

5.2 Higher Cohomology – a quick definition

- We define higher cohomology, $\mathcal{H}^n(A, Y)$
  by generalising the 2-cocycle formula to the *complex*

$$\to L^{n-1}(A, Y) \to L^n(A, Y) \to L^{n+1}(A, Y) \to$$

- with coboundary maps

$$\begin{aligned}\delta(T)(a_1, a_2, \ldots, a_n) = \; & + \; a_1 T(a_2, a_3, \ldots, a_n) \\ & + \; \sum_{j=1}^{n-1} (-1)^j T(a_1, \ldots, a_j a_{j+1}, \ldots) \\ & + \; (-1)^n T(a_1, a_2, \ldots, a_{n-1}) a_n\end{aligned}$$

- It is easy to check $\delta^2 = 0$ [Ex], which defines a *chain complex*
- $\mathcal{Z}^n(A) = \mathrm{Ker}\ \delta$, are the *n*-cocycles
- $\mathcal{B}^n(A) = \mathrm{Im}\ \delta$, are the *n*-cochains

5.2 Higher Cohomology – a quick definition

- We define higher cohomology, $\mathcal{H}^n(A, Y)$ by generalising the 2-cocycle formula to the *complex*

$$\to L^{n-1}(A, Y) \to L^n(A, Y) \to L^{n+1}(A, Y) \to$$

- with coboundary maps

$$\begin{aligned}\delta(T)(a_1, a_2, \ldots, a_n) = \; & + \; a_1 T(a_2, a_3, \ldots, a_n) \\ & + \; \sum_{j=1}^{n-1} (-1)^j T(a_1, \ldots, a_j a_{j+1}, \ldots) \\ & + \; (-1)^n T(a_1, a_2, \ldots, a_{n-1}) a_n\end{aligned}$$

- It is easy to check $\delta^2 = 0$ [Ex], which defines a *chain complex*
- $\mathcal{Z}^n(A) = \mathrm{Ker}\ \delta$, are the n-cocycles
- $\mathcal{B}^n(A) = \mathrm{Im}\ \delta$, are the n-cochains
- $\mathcal{H}^n(A, Y) = \mathcal{Z}^n(A, Y)/\mathcal{B}^n(A, Y)$, is the n-cohomology.

5.3 Using simpler cocycles

- It is often convenient to be able to assume that both your cocycles and coboundaries satisfy additional conditions

5.3 Using simpler cocycles

- It is often convenient to be able to assume that both your cocycles and coboundaries satisfy additional conditions
- e.g. that they vanish if any argument is a scalar multiple of $1 \in A$

5.3 Using simpler cocycles

- It is often convenient to be able to assume that both your cocycles and coboundaries satisfy additional conditions
- e.g. that they vanish if any argument is a scalar multiple of $1 \in A$
- This can be proved by induction, but there is a very flexible tool available to prove such results.

- It is often convenient to be able to assume that both your cocycles and coboundaries satisfy additional conditions
- e.g. that they vanish if any argument is a scalar multiple of $1 \in A$
- This can be proved by induction, but there is a very flexible tool available to prove such results.
- Define $P_n = A \hat{\otimes} \widehat{\bigotimes}^n A \hat{\otimes} A$ and $d : P_{n+1} \to P_n$ by

5.3 Using simpler cocycles

- It is often convenient to be able to assume that both your cocycles and coboundaries satisfy additional conditions
- e.g. that they vanish if any argument is a scalar multiple of $1 \in A$
- This can be proved by induction, but there is a very flexible tool available to prove such results.
- Define $P_n = A \hat{\otimes} \widehat{\bigotimes}^n A \hat{\otimes} A$ and $d : P_{n+1} \to P_n$ by
- $d(a_1 \otimes \cdots \otimes a_{n+1}) = \sum_{j=1}^{n} (-1)^{j+1} a_1 \otimes \cdots \otimes a_j a_{j+1} \otimes \cdots \otimes a_n$

- It is often convenient to be able to assume that both your cocycles and coboundaries satisfy additional conditions
- e.g. that they vanish if any argument is a scalar multiple of $1 \in A$
- This can be proved by induction, but there is a very flexible tool available to prove such results.
- Define $P_n = A \hat{\otimes} \widehat{\bigotimes}^n A \hat{\otimes} A$ and $d : P_{n+1} \to P_n$ by
- $d(a_1 \otimes \cdots \otimes a_{n+1}) = \sum_{j=1}^n (-1)^{j+1} a_1 \otimes \cdots \otimes a_j a_{j+1} \otimes \cdots \otimes a_n$
- $0 \leftarrow A \leftarrow P_0 \leftarrow P_1 \leftarrow P_2 \leftarrow \cdots$

5.3 Using simpler cocycles

- It is often convenient to be able to assume that both your cocycles and coboundaries satisfy additional conditions
- e.g. that they vanish if any argument is a scalar multiple of $1 \in A$
- This can be proved by induction, but there is a very flexible tool available to prove such results.
- Define $P_n = A \hat{\otimes} \widehat{\bigotimes}^n A \hat{\otimes} A$ and $d : P_{n+1} \to P_n$ by
- $d(a_1 \otimes \cdots \otimes a_{n+1}) = \sum_{j=1}^n (-1)^{j+1} a_1 \otimes \cdots \otimes a_j a_{j+1} \otimes \cdots \otimes a_n$
- $0 \leftarrow A \leftarrow P_0 \leftarrow P_1 \leftarrow P_2 \leftarrow \cdots$

5.3 Using simpler cocycles

- It is often convenient to be able to assume that both your cocycles and coboundaries satisfy additional conditions
- e.g. that they vanish if any argument is a scalar multiple of $1 \in A$
- This can be proved by induction, but there is a very flexible tool available to prove such results.
- Define $P_n = A \hat{\otimes} \widehat{\bigotimes}^n A \hat{\otimes} A$ and $d : P_{n+1} \to P_n$ by
- $d(a_1 \otimes \cdots \otimes a_{n+1}) = \sum_{j=1}^{n} (-1)^{j+1} a_1 \otimes \cdots \otimes a_j a_{j+1} \otimes \cdots \otimes a_n$
- $0 \leftarrow A \leftarrow P_0 \leftarrow P_1 \leftarrow P_2 \leftarrow \cdots$ is called the *Bar* resolution of A.

5.3 Using simpler cocycles

- It is often convenient to be able to assume that both your cocycles and coboundaries satisfy additional conditions
- e.g. that they vanish if any argument is a scalar multiple of $1 \in A$
- This can be proved by induction, but there is a very flexible tool available to prove such results.
- Define $P_n = A \hat{\otimes} \widehat{\bigotimes}^n A \hat{\otimes} A$ and $d : P_{n+1} \to P_n$ by
- $d(a_1 \otimes \cdots \otimes a_{n+1}) = \sum_{j=1}^n (-1)^{j+1} a_1 \otimes \cdots \otimes a_j a_{j+1} \otimes \cdots \otimes a_n$
- $0 \leftarrow A \leftarrow P_0 \leftarrow P_1 \leftarrow P_2 \leftarrow \cdots$ is called the *Bar* resolution of A.
- It is an exact complex of biprojective modules

5.3 Using simpler cocycles

- It is often convenient to be able to assume that both your cocycles and coboundaries satisfy additional conditions
- e.g. that they vanish if any argument is a scalar multiple of $1 \in A$
- This can be proved by induction, but there is a very flexible tool available to prove such results.
- Define $P_n = A \hat{\otimes} \widehat{\bigotimes}^n A \hat{\otimes} A$ and $d : P_{n+1} \to P_n$ by
- $d(a_1 \otimes \cdots \otimes a_{n+1}) = \sum_{j=1}^n (-1)^{j+1} a_1 \otimes \cdots \otimes a_j a_{j+1} \otimes \cdots \otimes a_n$
- $0 \leftarrow A \leftarrow P_0 \leftarrow P_1 \leftarrow P_2 \leftarrow \cdots$ is called the *Bar* resolution of A.
- It is an exact complex of biprojective modules
- Notice that ${}_A h_A(P_n, Y) = L^n(A, Y)$, and the induced maps are the same as the δ above [Ex]

5.3 Using simpler cocycles

- It is often convenient to be able to assume that both your cocycles and coboundaries satisfy additional conditions
- e.g. that they vanish if any argument is a scalar multiple of $1 \in A$
- This can be proved by induction, but there is a very flexible tool available to prove such results.
- Define $P_n = A \hat{\otimes} \widehat{\bigotimes}^n A \hat{\otimes} A$ and $d : P_{n+1} \to P_n$ by
- $d(a_1 \otimes \cdots \otimes a_{n+1}) = \sum_{j=1}^n (-1)^{j+1} a_1 \otimes \cdots \otimes a_j a_{j+1} \otimes \cdots \otimes a_n$
- $0 \leftarrow A \leftarrow P_0 \leftarrow P_1 \leftarrow P_2 \leftarrow \cdots$ is called the *Bar* resolution of A.
- It is an exact complex of biprojective modules
- Notice that ${}_A h_A(P_n, Y) = L^n(A, Y)$, and the induced maps are the same as the δ above [Ex]
- In fact we could take another resolution by biprojective A modules and get the same answer, i.e.

5.3 Using simpler cocycles

- It is often convenient to be able to assume that both your cocycles and coboundaries satisfy additional conditions
- e.g. that they vanish if any argument is a scalar multiple of $1 \in A$
- This can be proved by induction, but there is a very flexible tool available to prove such results.
- Define $P_n = A \hat{\otimes} \widehat{\bigotimes}^n A \hat{\otimes} A$ and $d : P_{n+1} \to P_n$ by
- $d(a_1 \otimes \cdots \otimes a_{n+1}) = \sum_{j=1}^n (-1)^{j+1} a_1 \otimes \cdots \otimes a_j a_{j+1} \otimes \cdots \otimes a_n$
- $0 \leftarrow A \leftarrow P_0 \leftarrow P_1 \leftarrow P_2 \leftarrow \cdots$ is called the *Bar* resolution of A.
- It is an exact complex of biprojective modules
- Notice that ${}_A h_A(P_n, Y) = L^n(A, Y)$, and the induced maps are the same as the δ above [Ex]
- In fact we could take another resolution by biprojective A modules and get the same answer, i.e.
- If we set $H^n(A, Y) = \mathrm{Ker}\, \delta_{[{}_A h_A(P_n, Y) \to {}_A h_A(P_{n-1}, Y)]} / \mathrm{Im}\, \delta$

5.3 Using simpler cocycles

- It is often convenient to be able to assume that both your cocycles and coboundaries satisfy additional conditions
- e.g. that they vanish if any argument is a scalar multiple of $1 \in A$
- This can be proved by induction, but there is a very flexible tool available to prove such results.
- Define $P_n = A \hat{\otimes} \widehat{\bigotimes}^n A \hat{\otimes} A$ and $d : P_{n+1} \to P_n$ by
- $d(a_1 \otimes \cdots \otimes a_{n+1}) = \sum_{j=1}^n (-1)^{j+1} a_1 \otimes \cdots \otimes a_j a_{j+1} \otimes \cdots \otimes a_n$
- $0 \leftarrow A \leftarrow P_0 \leftarrow P_1 \leftarrow P_2 \leftarrow \cdots$ is called the *Bar* resolution of A.
- It is an exact complex of biprojective modules
- Notice that ${}_A h_A(P_n, Y) = L^n(A, Y)$, and the induced maps are the same as the δ above [Ex]
- In fact we could take another resolution by biprojective A modules and get the same answer, i.e.
- If we set $H^n(A, Y) = \mathrm{Ker}\, \delta_{[{}_A h_A(P_n, Y) \to {}_A h_A(P_{n-1}, Y)]} / \mathrm{Im}\, \delta$
- then $H^n(A, Y) = \mathcal{H}^n(A, Y)$

5.3 Using simpler cocycles

- It is often convenient to be able to assume that both your cocycles and coboundaries satisfy additional conditions
- e.g. that they vanish if any argument is a scalar multiple of $1 \in A$
- This can be proved by induction, but there is a very flexible tool available to prove such results.
- Define $P_n = A\hat{\otimes}\widehat{\bigotimes}^n A\hat{\otimes}A$ and $d : P_{n+1} \to P_n$ by
- $d(a_1 \otimes \cdots \otimes a_{n+1}) = \sum_{j=1}^{n}(-1)^{j+1} a_1 \otimes \cdots \otimes a_j a_{j+1} \otimes \cdots \otimes a_n$
- $0 \leftarrow A \leftarrow P_0 \leftarrow P_1 \leftarrow P_2 \leftarrow \cdots$ is called the *Bar* resolution of A.
- It is an exact complex of biprojective modules
- Notice that ${}_A h_A(P_n, Y) = L^n(A, Y)$, and the induced maps are the same as the δ above [Ex]
- In fact we could take another resolution by biprojective A modules and get the same answer, i.e.
- If we set $H^n(A, Y) = \operatorname{Ker} \delta_{[{}_A h_A(P_n,Y) \to {}_A h_A(P_{n-1},Y)]} / \operatorname{Im} \delta$
- then $H^n(A, Y) = \mathcal{H}^n(A, Y)$

- It is often convenient to be able to assume that both your cocycles and coboundaries satisfy additional conditions
- e.g. that they vanish if any argument is a scalar multiple of $1 \in A$
- This can be proved by induction, but there is a very flexible tool available to prove such results.
- Define $P_n = A \hat{\otimes} \widehat{\bigotimes}^n A \hat{\otimes} A$ and $d : P_{n+1} \to P_n$ by
- $d(a_1 \otimes \cdots \otimes a_{n+1}) = \sum_{j=1}^{n} (-1)^{j+1} a_1 \otimes \cdots \otimes a_j a_{j+1} \otimes \cdots \otimes a_n$
- $0 \leftarrow A \leftarrow P_0 \leftarrow P_1 \leftarrow P_2 \leftarrow \cdots$ is called the *Bar* resolution of A.
- It is an exact complex of biprojective modules
- Notice that ${}_A h_A(P_n, Y) = L^n(A, Y)$, and the induced maps are the same as the δ above [Ex]
- In fact we could take another resolution by biprojective A modules and get the same answer, i.e.
- If we set $H^n(A, Y) = \mathrm{Ker}\, \delta_{[{}_A h_A(P_n, Y) \to {}_A h_A(P_{n-1}, Y)]} / \mathrm{Im}\, \delta$
- then $H^n(A, Y) = \mathcal{H}^n(A, Y)$ ... This is surprising!

- It is often convenient to be able to assume that both your cocycles and coboundaries satisfy additional conditions
- e.g. that they vanish if any argument is a scalar multiple of $1 \in A$
- This can be proved by induction, but there is a very flexible tool available to prove such results.
- Define $P_n = A\hat{\otimes}\widehat{\bigotimes}^n A\hat{\otimes}A$ and $d : P_{n+1} \to P_n$ by
- $d(a_1 \otimes \cdots \otimes a_{n+1}) = \sum_{j=1}^n (-1)^{j+1} a_1 \otimes \cdots \otimes a_j a_{j+1} \otimes \cdots \otimes a_n$
- $0 \leftarrow A \leftarrow P_0 \leftarrow P_1 \leftarrow P_2 \leftarrow \cdots$ is called the *Bar* resolution of A.
- It is an exact complex of biprojective modules
- Notice that ${}_A h_A(P_n, Y) = L^n(A, Y)$, and the induced maps are the same as the δ above [Ex]
- In fact we could take another resolution by biprojective A modules and get the same answer, i.e.
- If we set $H^n(A, Y) = \operatorname{Ker} \delta_{[{}_A h_A(P_n, Y) \to {}_A h_A(P_{n-1}, Y)]} / \operatorname{Im} \delta$
- then $H^n(A, Y) = \mathcal{H}^n(A, Y)$ . . . This is surprising!

5.4 Comparing Projective Resolutions

- The proof that $H^n(A, Y) = \mathcal{H}(A, Y)$ does to depend on the resolution involves several steps

5.4 Comparing Projective Resolutions

- The proof that $H^n(A, Y) = \mathcal{H}(A, Y)$ does to depend on the resolution involves several steps
- In general one compares two resolutions:

$$\begin{array}{ccccccccccc} 0 & \leftarrow & A & \leftarrow & P_0 & \leftarrow & P_1 & \leftarrow & P_2 & \leftarrow \cdots \\ \uparrow & & \uparrow & & \uparrow & & \uparrow & & \uparrow & \\ 0 & \leftarrow & A & \leftarrow & Q_0 & \leftarrow & Q_1 & \leftarrow & Q_2 & \leftarrow \cdots \end{array}$$

5.4 Comparing Projective Resolutions

- The proof that $H^n(A, Y) = \mathcal{H}(A, Y)$ does to depend on the resolution involves several steps
- In general one compares two resolutions:

$$\begin{array}{ccccccccccc} 0 & \leftarrow & A & \leftarrow & P_0 & \leftarrow & P_1 & \leftarrow & P_2 & \leftarrow \cdots \\ \uparrow & & \uparrow & & \uparrow & & \uparrow & & \uparrow & \\ 0 & \leftarrow & A & \leftarrow & Q_0 & \leftarrow & Q_1 & \leftarrow & Q_2 & \leftarrow \cdots \end{array}$$

- Using the fact that the Q_n are projective to build the comparison

5.4 Comparing Projective Resolutions

- The proof that $H^n(A, Y) = \mathcal{H}(A, Y)$ does to depend on the resolution involves several steps
- In general one compares two resolutions:

$$\begin{array}{ccccccccccc} 0 & \leftarrow & A & \leftarrow & P_0 & \leftarrow & P_1 & \leftarrow & P_2 & \leftarrow \cdots \\ \uparrow & & \uparrow & & \uparrow & & \uparrow & & \uparrow & \\ 0 & \leftarrow & A & \leftarrow & Q_0 & \leftarrow & Q_1 & \leftarrow & Q_2 & \leftarrow \cdots \end{array}$$

- Using the fact that the Q_n are projective to build the comparison
- This gives maps

$$\begin{array}{ccccccc} {}_A h_A(P_0, Y) & \to & {}_A h_A(P_1, Y) & \to & {}_A h_A(P_2, Y) & \cdots \\ \downarrow & & \downarrow & & \downarrow & \\ {}_A h_A(Q_0, Y) & \to & {}_A h_A(Q_1, Y) & \to & {}_A h_A(Q_2, Y) & \cdots \end{array}$$

5.4 Comparing Projective Resolutions

- The proof that $H^n(A, Y) = \mathcal{H}(A, Y)$ does to depend on the resolution involves several steps
- In general one compares two resolutions:

$$\begin{array}{ccccccccccc} 0 & \leftarrow & A & \leftarrow & P_0 & \leftarrow & P_1 & \leftarrow & P_2 & \leftarrow \cdots \\ \uparrow & & \uparrow & & \uparrow & & \uparrow & & \uparrow & \\ 0 & \leftarrow & A & \leftarrow & Q_0 & \leftarrow & Q_1 & \leftarrow & Q_2 & \leftarrow \cdots \end{array}$$

- Using the fact that the Q_n are projective to build the comparison
- This gives maps

$$\begin{array}{ccccccc} {}_Ah_A(P_0, Y) & \to & {}_Ah_A(P_1, Y) & \to & {}_Ah_A(P_2, Y) & \cdots \\ \downarrow & & \downarrow & & \downarrow & \\ {}_Ah_A(Q_0, Y) & \to & {}_Ah_A(Q_1, Y) & \to & {}_Ah_A(Q_2, Y) & \cdots \end{array}$$

- which gives maps:

5.4 Comparing Projective Resolutions

- The proof that $H^n(A, Y) = \mathcal{H}(A, Y)$ does to depend on the resolution involves several steps
- In general one compares two resolutions:

$$\begin{array}{ccccccccccc} 0 & \leftarrow & A & \leftarrow & P_0 & \leftarrow & P_1 & \leftarrow & P_2 & \leftarrow \cdots \\ \uparrow & & \uparrow & & \uparrow & & \uparrow & & \uparrow & \\ 0 & \leftarrow & A & \leftarrow & Q_0 & \leftarrow & Q_1 & \leftarrow & Q_2 & \leftarrow \cdots \end{array}$$

- Using the fact that the Q_n are projective to build the comparison
- This gives maps

$$\begin{array}{ccccccc} {}_A h_A(P_0, Y) & \rightarrow & {}_A h_A(P_1, Y) & \rightarrow & {}_A h_A(P_2, Y) & \cdots \\ \downarrow & & \downarrow & & \downarrow & \\ {}_A h_A(Q_0, Y) & \rightarrow & {}_A h_A(Q_1, Y) & \rightarrow & {}_A h_A(Q_2, Y) & \cdots \end{array}$$

- which gives maps:
  - $\mathcal{Z}^n_P(A, Y) \rightarrow \mathcal{Z}^n_Q(A, Y)$

5.4 Comparing Projective Resolutions

- The proof that $H^n(A, Y) = \mathcal{H}(A, Y)$ does to depend on the resolution involves several steps
- In general one compares two resolutions:

$$\begin{array}{ccccccccccc} 0 & \leftarrow & A & \leftarrow & P_0 & \leftarrow & P_1 & \leftarrow & P_2 & \leftarrow \cdots \\ \uparrow & & \uparrow & & \uparrow & & \uparrow & & \uparrow & \\ 0 & \leftarrow & A & \leftarrow & Q_0 & \leftarrow & Q_1 & \leftarrow & Q_2 & \leftarrow \cdots \end{array}$$

- Using the fact that the Q_n are projective to build the comparison
- This gives maps

$$\begin{array}{cccccc} {}_A h_A(P_0, Y) & \to & {}_A h_A(P_1, Y) & \to & {}_A h_A(P_2, Y) & \cdots \\ \downarrow & & \downarrow & & \downarrow & \\ {}_A h_A(Q_0, Y) & \to & {}_A h_A(Q_1, Y) & \to & {}_A h_A(Q_2, Y) & \cdots \end{array}$$

- which gives maps:
  - $\mathcal{Z}^n_P(A, Y) \to \mathcal{Z}^n_Q(A, Y)$
  - $\mathcal{B}^n_P(A, Y) \to \mathcal{B}^n_Q(A, Y)$

5.4 Comparing Projective Resolutions

- The proof that $H^n(A,Y) = \mathcal{H}(A,Y)$ does to depend on the resolution involves several steps
- In general one compares two resolutions:

$$\begin{array}{ccccccccccc} 0 & \leftarrow & A & \leftarrow & P_0 & \leftarrow & P_1 & \leftarrow & P_2 & \leftarrow \cdots \\ \uparrow & & \uparrow & & \uparrow & & \uparrow & & \uparrow & \\ 0 & \leftarrow & A & \leftarrow & Q_0 & \leftarrow & Q_1 & \leftarrow & Q_2 & \leftarrow \cdots \end{array}$$

- Using the fact that the Q_n are projective to build the comparison
- This gives maps

$$\begin{array}{ccccccc} {}_A h_A(P_0, Y) & \to & {}_A h_A(P_1, Y) & \to & {}_A h_A(P_2, Y) & \cdots \\ \downarrow & & \downarrow & & \downarrow & \\ {}_A h_A(Q_0, Y) & \to & {}_A h_A(Q_1, Y) & \to & {}_A h_A(Q_2, Y) & \cdots \end{array}$$

- which gives maps:
  - $\mathcal{Z}_P^n(A,Y) \to \mathcal{Z}_Q^n(A,Y)$
  - $\mathcal{B}_P^n(A,Y) \to \mathcal{B}_Q^n(A,Y)$
  - $\mathcal{H}_P^n(A,Y) \to \mathcal{H}_Q^n(A,Y)$

5.4 Comparing Projective Resolutions

- The proof that $H^n(A,Y) = \mathcal{H}(A,Y)$ does to depend on the resolution involves several steps
- In general one compares two resolutions:

$$\begin{array}{ccccccccccc} 0 & \leftarrow & A & \leftarrow & P_0 & \leftarrow & P_1 & \leftarrow & P_2 & \leftarrow \cdots \\ \uparrow & & \uparrow & & \uparrow & & \uparrow & & \uparrow & \\ 0 & \leftarrow & A & \leftarrow & Q_0 & \leftarrow & Q_1 & \leftarrow & Q_2 & \leftarrow \cdots \end{array}$$

- Using the fact that the Q_n are projective to build the comparison
- This gives maps

$$\begin{array}{ccccccc} {}_A h_A(P_0, Y) & \rightarrow & {}_A h_A(P_1, Y) & \rightarrow & {}_A h_A(P_2, Y) & \cdots \\ \downarrow & & \downarrow & & \downarrow & \\ {}_A h_A(Q_0, Y) & \rightarrow & {}_A h_A(Q_1, Y) & \rightarrow & {}_A h_A(Q_2, Y) & \cdots \end{array}$$

- which gives maps:
  - $\mathcal{Z}^n_P(A,Y) \to \mathcal{Z}^n_Q(A,Y)$
  - $\mathcal{B}^n_P(A,Y) \to \mathcal{B}^n_Q(A,Y)$
  - $\mathcal{H}^n_P(A,Y) \to \mathcal{H}^n_Q(A,Y)$
- To check the last is an isomorphism one begins with the case $Q_n = P_n$

5.4 Comparing Projective Resolutions

- The proof that $H^n(A, Y) = \mathcal{H}(A, Y)$ does to depend on the resolution involves several steps
- In general one compares two resolutions:

$$\begin{array}{ccccccccccc} 0 & \leftarrow & A & \leftarrow & P_0 & \leftarrow & P_1 & \leftarrow & P_2 & \leftarrow \cdots \\ \uparrow & & \uparrow & & \uparrow & & \uparrow & & \uparrow & \\ 0 & \leftarrow & A & \leftarrow & Q_0 & \leftarrow & Q_1 & \leftarrow & Q_2 & \leftarrow \cdots \end{array}$$

- Using the fact that the Q_n are projective to build the comparison
- This gives maps

$$\begin{array}{ccccccc} {}_Ah_A(P_0, Y) & \to & {}_Ah_A(P_1, Y) & \to & {}_Ah_A(P_2, Y) & \cdots \\ \downarrow & & \downarrow & & \downarrow & \\ {}_Ah_A(Q_0, Y) & \to & {}_Ah_A(Q_1, Y) & \to & {}_Ah_A(Q_2, Y) & \cdots \end{array}$$

- which gives maps:
  - $\mathcal{Z}^n_P(A, Y) \to \mathcal{Z}^n_Q(A, Y)$
  - $\mathcal{B}^n_P(A, Y) \to \mathcal{B}^n_Q(A, Y)$
  - $\mathcal{H}^n_P(A, Y) \to \mathcal{H}^n_Q(A, Y)$
- To check the last is an isomorphism one begins with the case $Q_n = P_n$
- Beware: the maps $\uparrow$, may not have been chosen to be isomorphisms

5.5 Using injective resolutions

We are also allowed to use biinjective resolutions of the bimodule

$$0 \to Y \to I_0 \to I_1 \to I_2 \to \cdots$$

5.5 Using injective resolutions

We are also allowed to use biinjective resolutions of the bimodule

$$0 \to Y \to I_0 \to I_1 \to I_2 \to \cdots$$

and then the cohomology if given by

5.5 Using injective resolutions

We are also allowed to use biinjective resolutions of the bimodule

$$0 \to Y \to I_0 \to I_1 \to I_2 \to \cdots$$

and then the cohomology if given by

$$H^n(A, Y) = \mathrm{Ker}\,\delta : {}_A h_A(A, I_n) \to {}_A h_A(A, I_{n+1}) / \mathrm{Im}\,\delta$$

5.5 Using injective resolutions

We are also allowed to use biinjective resolutions of the bimodule

$$0 \to Y \to I_0 \to I_1 \to I_2 \to \cdots$$

and then the cohomology if given by

$$H^n(A, Y) = \mathrm{Ker}\,\delta : {}_A h_A(A, I_n) \to {}_A h_A(A, I_{n+1}) / \mathrm{Im}\,\delta$$

Theorem A is amenable iff
A' is a bimodule direct summand of $(A \hat{\otimes} A)'$

5.5 Using injective resolutions

We are also allowed to use biinjective resolutions of the bimodule

$$0 \to Y \to I_0 \to I_1 \to I_2 \to \cdots$$

and then the cohomology if given by

$$H^n(A, Y) = \operatorname{Ker} \delta : {}_A h_A(A, I_n) \to {}_A h_A(A, I_{n+1}) / \operatorname{Im} \delta$$

Theorem A is amenable iff

A' is a bimodule direct summand of $(A \hat{\otimes} A)'$ iff A' is biinjective.

5.5 Using injective resolutions

We are also allowed to use biinjective resolutions of the bimodule

$$0 \to Y \to I_0 \to I_1 \to I_2 \to \cdots$$

and then the cohomology if given by

$$H^n(A, Y) = \operatorname{Ker} \delta : {}_A h_A(A, I_n) \to {}_A h_A(A, I_{n+1}) / \operatorname{Im} \delta$$

Theorem A is amenable iff

A' is a bimodule direct summand of $(A \hat{\otimes} A)'$ iff A' is biinjective.

$\mathcal{H}^n(A, I) = 0$, for biinjective modules

We are also allowed to use biinjective resolutions of the bimodule

$$0 \to Y \to I_0 \to I_1 \to I_2 \to \cdots$$

and then the cohomology if given by

$$H^n(A, Y) = \mathrm{Ker}\,\delta : {}_A h_A(A, I_n) \to {}_A h_A(A, I_{n+1}) / \mathrm{Im}\,\delta$$

Theorem A is amenable iff

A' is a bimodule direct summand of $(A \hat{\otimes} A)'$ iff A' is biinjective.

$\mathcal{H}^n(A, I) = 0$, for biinjective modules

- If the bimodule is already biinjective, then it has a short resolution

5.5 Using injective resolutions

We are also allowed to use biinjective resolutions of the bimodule

$$0 \to Y \to I_0 \to I_1 \to I_2 \to \cdots$$

and then the cohomology if given by

$$H^n(A, Y) = \operatorname{Ker} \delta : {}_A h_A(A, I_n) \to {}_A h_A(A, I_{n+1}) / \operatorname{Im} \delta$$

Theorem A is amenable iff

A' is a bimodule direct summand of $(A \hat{\otimes} A)'$ iff A' is biinjective.

$\mathcal{H}^n(A, I) = 0$, for biinjective modules

- If the bimodule is already biinjective, then it has a short resolution
- $$0 \to I \to I \to 0 \to \cdots$$

We are also allowed to use biinjective resolutions of the bimodule

$$0 \to Y \to I_0 \to I_1 \to I_2 \to \cdots$$

and then the cohomology if given by

$$H^n(A, Y) = \operatorname{Ker} \delta : {}_A h_A(A, I_n) \to {}_A h_A(A, I_{n+1}) / \operatorname{Im} \delta$$

Theorem A is amenable iff

A' is a bimodule direct summand of $(A \hat{\otimes} A)'$ iff A' is biinjective.

$\mathcal{H}^n(A, I) = 0$, for biinjective modules

- If the bimodule is already biinjective, then it has a short resolution
- $0 \to I \to I \to 0 \to \cdots$
- It is clear that all the $H^n(A, Y)$ defined above are trivial, for $n > 0$

We are also allowed to use biinjective resolutions of the bimodule

$$0 \to Y \to I_0 \to I_1 \to I_2 \to \cdots$$

and then the cohomology if given by

$$H^n(A, Y) = \mathrm{Ker}\,\delta : {}_A h_A(A, I_n) \to {}_A h_A(A, I_{n+1}) / \mathrm{Im}\,\delta$$

Theorem A is amenable iff
A' is a bimodule direct summand of $(A\hat{\otimes}A)'$ iff A' is biinjective.

$\mathcal{H}^n(A, I) = 0$, for biinjective modules

- If the bimodule is already biinjective, then it has a short resolution
- $0 \to I \to I \to 0 \to \cdots$
- It is clear that all the $H^n(A, Y)$ defined above are trivial, for $n > 0$
- So it is not just 'trivial' derivations which biinjective modules have

5.5 Using injective resolutions

We are also allowed to use biinjective resolutions of the bimodule

$$0 \to Y \to I_0 \to I_1 \to I_2 \to \cdots$$

and then the cohomology if given by

$$H^n(A, Y) = \operatorname{Ker} \delta : {}_A h_A(A, I_n) \to {}_A h_A(A, I_{n+1}) / \operatorname{Im} \delta$$

Theorem A is amenable iff
A' is a bimodule direct summand of $(A \hat{\otimes} A)'$ iff A' is biinjective.

$\mathcal{H}^n(A, I) = 0$, for biinjective modules

- If the bimodule is already biinjective, then it has a short resolution
- $0 \to I \to I \to 0 \to \cdots$
- It is clear that all the $H^n(A, Y)$ defined above are trivial, for $n > 0$
- So it is not just 'trivial' derivations which biinjective modules have
- All of the higher cohomology vanishes

5.6 Dimension Reduction

We have already seen $\mathcal{H}^1(A, X)$ is related to $\mathcal{H}^2(A, Z)$.

5.6 Dimension Reduction

We have already seen $\mathcal{H}^1(A, X)$ is related to $\mathcal{H}^2(A, Z)$. More Generally ...

5.6 Dimension Reduction

We have already seen $\mathcal{H}^1(A, X)$ is related to $\mathcal{H}^2(A, Z)$. More Generally . . .

Long Exact Sequences

5.6 Dimension Reduction

We have already seen $\mathcal{H}^1(A, X)$ is related to $\mathcal{H}^2(A, Z)$. More Generally . . .

Long Exact Sequences

- **Theorem** Given an *admissible* `short exact sequence`

5.6 Dimension Reduction

We have already seen $\mathcal{H}^1(A, X)$ is related to $\mathcal{H}^2(A, Z)$. More Generally ...

Long Exact Sequences

- **Theorem** Given an *admissible* `short exact sequence`
- $0 \to X \hookrightarrow Y \to Z \to 0$

5.6 Dimension Reduction

We have already seen $\mathcal{H}^1(A, X)$ is related to $\mathcal{H}^2(A, Z)$. More Generally ...

Long Exact Sequences

- **Theorem** Given an *admissible* `short exact sequence`
- $$0 \to X \hookrightarrow Y \to Z \to 0$$
- There is a `long exact sequence`

5.6 Dimension Reduction

We have already seen $\mathcal{H}^1(A, X)$ is related to $\mathcal{H}^2(A, Z)$. More Generally ...

Long Exact Sequences

- **Theorem** Given an *admissible* `short exact sequence`
- $0 \to X \hookrightarrow Y \to Z \to 0$
- There is a `long exact sequence`
- $\cdots \to \mathcal{H}^n(A, Y) \to \mathcal{H}^n(A, Z) \to \mathcal{H}^{n+1}(A, X) \to \mathcal{H}^{n+1}(A, Y) \to \cdots$

5.6 Dimension Reduction

We have already seen $\mathcal{H}^1(A, X)$ is related to $\mathcal{H}^2(A, Z)$. More Generally ...

Long Exact Sequences

- **Theorem** Given an *admissible* `short exact sequence`
- $$0 \to X \hookrightarrow Y \to Z \to 0$$
- There is a `long exact sequence`
- $\cdots \to \mathcal{H}^n(A, Y) \to \mathcal{H}^n(A, Z) \to \mathcal{H}^{n+1}(A, X) \to \mathcal{H}^{n+1}(A, Y) \to \cdots$
- If we select a bimodule Y so that $\mathcal{H}^n(A, Y) = 0$ then

5.6 Dimension Reduction

We have already seen $\mathcal{H}^1(A, X)$ is related to $\mathcal{H}^2(A, Z)$. More Generally . . .

Long Exact Sequences

- **Theorem** Given an *admissible* `short exact sequence`
- $0 \to X \hookrightarrow Y \to Z \to 0$
- There is a `long exact sequence`
- $\cdots \to \mathcal{H}^n(A, Y) \to \mathcal{H}^n(A, Z) \to \mathcal{H}^{n+1}(A, X) \to \mathcal{H}^{n+1}(A, Y) \to \cdots$
- If we select a bimodule Y so that $\mathcal{H}^n(A, Y) = 0$ then
- $\mathcal{H}^n(A, Z) \cong \mathcal{H}^{n+1}(A, X)$

5.6 Dimension Reduction

We have already seen $\mathcal{H}^1(A,X)$ is related to $\mathcal{H}^2(A,Z)$. More Generally ...

Long Exact Sequences

- **Theorem** Given an *admissible* `short exact sequence`
- $0 \to X \hookrightarrow Y \to Z \to 0$
- There is a `long exact sequence`
- $\cdots \to \mathcal{H}^n(A,Y) \to \mathcal{H}^n(A,Z) \to \mathcal{H}^{n+1}(A,X) \to \mathcal{H}^{n+1}(A,Y) \to \cdots$
- If we select a bimodule Y so that $\mathcal{H}^n(A,Y) = 0$ then
- $\mathcal{H}^n(A,Z) \cong \mathcal{H}^{n+1}(A,X)$
- We select $Y = L(A\hat{\otimes}A, X)$, are it is biinjective

5.6 Dimension Reduction

We have already seen $\mathcal{H}^1(A, X)$ is related to $\mathcal{H}^2(A, Z)$. More Generally . . .

Long Exact Sequences

- **Theorem** Given an *admissible* `short exact sequence`
- $0 \to X \hookrightarrow Y \to Z \to 0$
- There is a `long exact sequence`
- $\cdots \to \mathcal{H}^n(A, Y) \to \mathcal{H}^n(A, Z) \to \mathcal{H}^{n+1}(A, X) \to \mathcal{H}^{n+1}(A, Y) \to \cdots$
- If we select a bimodule Y so that $\mathcal{H}^n(A, Y) = 0$ then
- $\mathcal{H}^n(A, Z) \cong \mathcal{H}^{n+1}(A, X)$
- We select $Y = L(A\hat{\otimes}A, X)$, are it is biinjective
- Which gives $Z = L(A\hat{\otimes}A, X)/X$

5.6 Dimension Reduction

We have already seen $\mathcal{H}^1(A,X)$ is related to $\mathcal{H}^2(A,Z)$. More Generally . . .

Long Exact Sequences

- **Theorem** Given an *admissible* `short exact sequence`
- $0 \to X \hookrightarrow Y \to Z \to 0$
- There is a `long exact sequence`
- $\cdots \to \mathcal{H}^n(A,Y) \to \mathcal{H}^n(A,Z) \to \mathcal{H}^{n+1}(A,X) \to \mathcal{H}^{n+1}(A,Y) \to \cdots$
- If we select a bimodule Y so that $\mathcal{H}^n(A,Y) = 0$ then
- $\mathcal{H}^n(A,Z) \cong \mathcal{H}^{n+1}(A,X)$
- We select $Y = L(A\hat{\otimes}A, X)$, are it is biinjective
- Which gives $Z = L(A\hat{\otimes}A, X)/X$
- So we have that

5.6 Dimension Reduction

We have already seen $\mathcal{H}^1(A, X)$ is related to $\mathcal{H}^2(A, Z)$. More Generally ...

Long Exact Sequences

- **Theorem** Given an *admissible* `short exact sequence`
- $$0 \to X \hookrightarrow Y \to Z \to 0$$
- There is a `long exact sequence`
- $\cdots \to \mathcal{H}^n(A, Y) \to \mathcal{H}^n(A, Z) \to \mathcal{H}^{n+1}(A, X) \to \mathcal{H}^{n+1}(A, Y) \to \cdots$
- If we select a bimodule Y so that $\mathcal{H}^n(A, Y) = 0$ then
- $\mathcal{H}^n(A, Z) \cong \mathcal{H}^{n+1}(A, X)$
- We select $Y = L(A \hat{\otimes} A, X)$, are it is biinjective
- Which gives $Z = L(A \hat{\otimes} A, X)/X$
- So we have that
- $$\mathcal{H}^{n+1}(A, X) \cong \mathcal{H}^n(A, Z)$$

We have already seen $\mathcal{H}^1(A,X)$ is related to $\mathcal{H}^2(A,Z)$. More Generally . . .

Long Exact Sequences

- **Theorem** Given an *admissible* `short exact sequence`
- $$0 \to X \hookrightarrow Y \to Z \to 0$$
- There is a `long exact sequence`
- $\cdots \to \mathcal{H}^n(A,Y) \to \mathcal{H}^n(A,Z) \to \mathcal{H}^{n+1}(A,X) \to \mathcal{H}^{n+1}(A,Y) \to \cdots$
- If we select a bimodule Y so that $\mathcal{H}^n(A,Y) = 0$ then
- $\mathcal{H}^n(A,Z) \cong \mathcal{H}^{n+1}(A,X)$
- We select $Y = L(A\hat{\otimes}A, X)$, are it is biinjective
- Which gives $Z = L(A\hat{\otimes}A, X)/X$
- So we have that
- $$\mathcal{H}^{n+1}(A,X) \cong \mathcal{H}^n(A,Z)$$
- Hence it is isomorphic to some $\mathcal{H}^1(A,W)$

We have heard that the we can use other biprojective resolutions to compute cohomology:

Unit normalised resolution

5.7 Other Resolutions - I

We have heard that the we can use other biprojective resolutions to compute cohomology:

Unit normalised resolution

- We set $P_n = A\hat{\otimes}\widehat{\bigotimes}^n(A/1\mathbf{C})\hat{\otimes}A$ and use the $d : P_{n+1} \to P_n$ as above.

5.7 Other Resolutions - I

We have heard that the we can use other biprojective resolutions to compute cohomology:

Unit normalised resolution

- We set $P_n = A\hat{\otimes}\widehat{\bigotimes}^n(A/1\mathbf{C})\hat{\otimes}A$ and use the $d : P_{n+1} \to P_n$ as above.
- It is clear that these modules are of the correct form to be biprojective

We have heard that the we can use other biprojective resolutions to compute cohomology:

Unit normalised resolution

- We set $P_n = A\hat{\otimes}\widehat{\bigotimes}^n(A/1\mathbf{C})\hat{\otimes}A$ and use the $d : P_{n+1} \to P_n$ as above.
- It is clear that these modules are of the correct form to be biprojective
- To see that the complex is exact, we define $s : P_n \to P_{n+1}$ by

5.7 Other Resolutions - I

We have heard that the we can use other biprojective resolutions to compute cohomology:

Unit normalised resolution

- We set $P_n = A\hat{\otimes}\widehat{\bigotimes}^n(A/1\mathbf{C})\hat{\otimes}A$ and use the $d : P_{n+1} \to P_n$ as above.
- It is clear that these modules are of the correct form to be biprojective
- To see that the complex is exact, we define $s : P_n \to P_{n+1}$ by
- $s(\omega) = s(a_1 \otimes \cdots \otimes a_n) = 1 \otimes \omega$

5.7 Other Resolutions - I

We have heard that the we can use other biprojective resolutions to compute cohomology:

Unit normalised resolution

- We set $P_n = A \hat{\otimes} \widehat{\bigotimes}^n (A/1\mathbf{C}) \hat{\otimes} A$ and use the $d : P_{n+1} \to P_n$ as above.
- It is clear that these modules are of the correct form to be biprojective
- To see that the complex is exact, we define $s : P_n \to P_{n+1}$ by
- $s(\omega) = s(a_1 \otimes \cdots \otimes a_n) = 1 \otimes \omega$
- Observe $ds(\omega) = d(1 \otimes \omega) = 1.\omega - 1 \otimes d(\omega)$,

5.7 Other Resolutions - I

We have heard that the we can use other biprojective resolutions to compute cohomology:

Unit normalised resolution

- We set $P_n = A \hat{\otimes} \widehat{\bigotimes}^n (A/1\mathbf{C}) \hat{\otimes} A$ and use the $d : P_{n+1} \to P_n$ as above.
- It is clear that these modules are of the correct form to be biprojective
- To see that the complex is exact, we define $s : P_n \to P_{n+1}$ by
- $s(\omega) = s(a_1 \otimes \cdots \otimes a_n) = 1 \otimes \omega$
- Observe $ds(\omega) = d(1 \otimes \omega) = 1.\omega - 1 \otimes d(\omega)$,
- We usually write: $ds + sd = 1$, and call s a *contracting homotopy*

5.7 Other Resolutions - I

We have heard that the we can use other biprojective resolutions to compute cohomology:

Unit normalised resolution

- We set $P_n = A \hat{\otimes} \widehat{\bigotimes}^n (A/1\mathbf{C}) \hat{\otimes} A$ and use the $d : P_{n+1} \to P_n$ as above.
- It is clear that these modules are of the correct form to be biprojective
- To see that the complex is exact, we define $s : P_n \to P_{n+1}$ by
- $s(\omega) = s(a_1 \otimes \cdots \otimes a_n) = 1 \otimes \omega$
- Observe $ds(\omega) = d(1 \otimes \omega) = 1.\omega - 1 \otimes d(\omega)$,
- We usually write: $ds + sd = 1$, and call s a *contracting homotopy*
- Exactness:

5.7 Other Resolutions - I

We have heard that the we can use other biprojective resolutions to compute cohomology:

Unit normalised resolution

- We set $P_n = A\hat{\otimes}\widehat{\bigotimes}^n(A/1\mathbf{C})\hat{\otimes}A$ and use the $d : P_{n+1} \to P_n$ as above.
- It is clear that these modules are of the correct form to be biprojective
- To see that the complex is exact, we define $s : P_n \to P_{n+1}$ by
- $s(\omega) = s(a_1 \otimes \cdots \otimes a_n) = 1 \otimes \omega$
- Observe $ds(\omega) = d(1 \otimes \omega) = 1.\,\omega - 1 \otimes d(\omega)$,
- We usually write: $ds + sd = 1$, and call s a *contracting homotopy*
- Exactness:

We have heard that the we can use other biprojective resolutions to compute cohomology:

Unit normalised resolution

- We set $P_n = A\hat{\otimes}\widehat{\bigotimes}^n(A/1\mathbf{C})\hat{\otimes}A$ and use the $d : P_{n+1} \to P_n$ as above.
- It is clear that these modules are of the correct form to be biprojective
- To see that the complex is exact, we define $s : P_n \to P_{n+1}$ by
- $s(\omega) = s(a_1 \otimes \cdots \otimes a_n) = 1 \otimes \omega$
- Observe $ds(\omega) = d(1 \otimes \omega) = 1.\omega - 1 \otimes d(\omega)$,
- We usually write: $ds + sd = 1$, and call s a *contracting homotopy*
- Exactness: if η is in $\mathrm{Ker}\, d$, then $\eta = \eta + sd(\eta) = ds(\eta) \in \mathrm{Im} d$

5.7 Other Resolutions - I

We have heard that the we can use other biprojective resolutions to compute cohomology:

Unit normalised resolution

- We set $P_n = A\hat{\otimes}\widehat{\bigotimes}^n(A/1\mathbf{C})\hat{\otimes}A$ and use the $d : P_{n+1} \to P_n$ as above.
- It is clear that these modules are of the correct form to be biprojective
- To see that the complex is exact, we define $s : P_n \to P_{n+1}$ by
- $s(\omega) = s(a_1 \otimes \cdots \otimes a_n) = 1 \otimes \omega$
- Observe $ds(\omega) = d(1 \otimes \omega) = 1.\omega - 1 \otimes d(\omega)$,
- We usually write: $ds + sd = 1$, and call s a *contracting homotopy*
- Exactness: if η is in $\mathrm{Ker}\, d$, then $\eta = \eta + sd(\eta) = ds(\eta) \in \mathrm{Im} d$
- Now note that ${}_Ah_A(P_n, Y)$ is exactly the unit normalised maps [Ex]

5.7 Other Resolutions - I

We have heard that the we can use other biprojective resolutions to compute cohomology:

Unit normalised resolution

- We set $P_n = A\hat{\otimes}\widehat{\bigotimes}^n(A/1\mathbf{C})\hat{\otimes}A$ and use the $d : P_{n+1} \to P_n$ as above.
- It is clear that these modules are of the correct form to be biprojective
- To see that the complex is exact, we define $s : P_n \to P_{n+1}$ by
- $s(\omega) = s(a_1 \otimes \cdots \otimes a_n) = 1 \otimes \omega$
- Observe $ds(\omega) = d(1 \otimes \omega) = 1.\omega - 1 \otimes d(\omega)$,
- We usually write: $ds + sd = 1$, and call s a *contracting homotopy*
- Exactness: if η is in $\mathrm{Ker}\, d$, then $\eta = \eta + sd(\eta) = ds(\eta) \in \mathrm{Im} d$
- Now note that ${}_Ah_A(P_n, Y)$ is exactly the unit normalised maps [Ex]
- Hence $H^n(A, Y)$ is the (usual) unit normalised cohomology

5.8 Other Resolutions - II

In fact for dual modules we can use even more resolutions

5.8 Other Resolutions - II

In fact for dual modules we can use even more resolutions we only need to have a bi-injective resolution of A'.

5.8 Other Resolutions - II

In fact for dual modules we can use even more resolutions
we only need to have a bi-injective resolution of A'.
Note the $I_n = P_n'$ would be such a bi-injective resolution

Normalisation w.r.t. and amenable subalgebra

5.8 Other Resolutions - II

In fact for dual modules we can use even more resolutions we only need to have a bi-injective resolution of A'.
Note the $I_n = P_n'$ would be such a bi-injective resolution

Normalisation w.r.t. and amenable subalgebra

Let B be an amenable subalgebra of A

- We set $I_n = L_B^{n+2}(A, \mathbf{C})$ such that for $c \in B$ and $a_i \in A$

5.8 Other Resolutions - II

In fact for dual modules we can use even more resolutions we only need to have a bi-injective resolution of A'.
Note the $I_n = P_n'$ would be such a bi-injective resolution

Normalisation w.r.t. and amenable subalgebra

Let B be an amenable subalgebra of A

- We set $I_n = L_B^{n+2}(A, \mathbf{C})$ such that for $c \in B$ and $a_i \in A$
- $T(\cdots, a_j c, a_{j+1}, \cdots) = T(\cdots, a_j, c a_{j+1}, \cdots)$,

5.8 Other Resolutions - II

In fact for dual modules we can use even more resolutions we only need to have a bi-injective resolution of A'.
Note the $I_n = P'_n$ would be such a bi-injective resolution

Normalisation w.r.t. and amenable subalgebra

Let B be an amenable subalgebra of A

- We set $I_n = L_B^{n+2}(A, \mathbf{C})$ such that for $c \in B$ and $a_i \in A$
- $T(\cdots, a_j c, a_{j+1}, \cdots) = T(\cdots, a_j, c a_{j+1}, \cdots)$,

In fact for dual modules we can use even more resolutions we only need to have a bi-injective resolution of A'.
Note the $I_n = P_n'$ would be such a bi-injective resolution

Normalisation w.r.t. and amenable subalgebra

Let B be an amenable subalgebra of A

- We set $I_n = L_B^{n+2}(A, \mathbf{C})$ such that for $c \in B$ and $a_i \in A$
- $T(\cdots, a_j c, a_{j+1}, \cdots) = T(\cdots, a_j, c a_{j+1}, \cdots)$, δ as above

5.8 Other Resolutions - II

In fact for dual modules we can use even more resolutions we only need to have a bi-injective resolution of A'.
Note the $I_n = P_n'$ would be such a bi-injective resolution

Normalisation w.r.t. and amenable subalgebra

Let B be an amenable subalgebra of A

- We set $I_n = L_B^{n+2}(A, \mathbf{C})$ such that for $c \in B$ and $a_i \in A$
- $T(\cdots, a_j c, a_{j+1}, \cdots) = T(\cdots, a_j, c a_{j+1}, \cdots)$, δ as above
- Note these are indeed bimodules, which are submodules of $L^{n+2}(A, \mathbf{C})$

5.8 Other Resolutions - II

In fact for dual modules we can use even more resolutions we only need to have a bi-injective resolution of A'.
Note the $I_n = P'_n$ would be such a bi-injective resolution

Normalisation w.r.t. and amenable subalgebra

Let B be an amenable subalgebra of A

- We set $I_n = L_B^{n+2}(A, \mathbf{C})$ such that for $c \in B$ and $a_i \in A$
- $T(\cdots, a_j c, a_{j+1}, \cdots) = T(\cdots, a_j, c a_{j+1}, \cdots)$, δ as above
- Note these are indeed bimodules, which are submodules of $L^{n+2}(A, \mathbf{C})$
- We need to check that the modules are all bi-injective.

5.8 Other Resolutions - II

In fact for dual modules we can use even more resolutions we only need to have a bi-injective resolution of A'.
Note the $I_n = P_n'$ would be such a bi-injective resolution

Normalisation w.r.t. and amenable subalgebra

Let B be an amenable subalgebra of A

- We set $I_n = L_B^{n+2}(A, \mathbf{C})$ such that for $c \in B$ and $a_i \in A$
- $T(\cdots, a_j c, a_{j+1}, \cdots) = T(\cdots, a_j, c a_{j+1}, \cdots)$, δ as above
- Note these are indeed bimodules, which are submodules of $L^{n+2}(A, \mathbf{C})$
- We need to check that the modules are all bi-injective.
- We give a bimodule projection from $L^{n+2}(A, \mathbf{C})$ onto $L_B^{n+2}(A, \mathbf{C})$

5.8 Other Resolutions - II

In fact for dual modules we can use even more resolutions we only need to have a bi-injective resolution of A'.
Note the $I_n = P_n'$ would be such a bi-injective resolution

Normalisation w.r.t. and amenable subalgebra

Let B be an amenable subalgebra of A

- We set $I_n = L_B^{n+2}(A, \mathbf{C})$ such that for $c \in B$ and $a_i \in A$
- $T(\cdots, a_j c, a_{j+1}, \cdots) = T(\cdots, a_j, c a_{j+1}, \cdots)$, δ as above
- Note these are indeed bimodules, which are submodules of $L^{n+2}(A, \mathbf{C})$
- We need to check that the modules are all bi-injective.
- We give a bimodule projection from $L^{n+2}(A, \mathbf{C})$ onto $L_B^{n+2}(A, \mathbf{C})$
- Set $\bar{T}(\cdots, a_j, a_{j+1}, \cdots) = LIM_\lambda \sum_{i=1}^{\infty} T(\cdots, a_j a_i^\lambda, b_i^\lambda a_{j+1}, \cdots)$

5.8 Other Resolutions - II

In fact for dual modules we can use even more resolutions
we only need to have a bi-injective resolution of A'.
Note the $I_n = P'_n$ would be such a bi-injective resolution

Normalisation w.r.t. and amenable subalgebra

Let B be an amenable subalgebra of A

- We set $I_n = L_B^{n+2}(A, \mathbf{C})$ such that for $c \in B$ and $a_i \in A$
- $T(\cdots, a_j c, a_{j+1}, \cdots) = T(\cdots, a_j, c a_{j+1}, \cdots)$, δ as above
- Note these are indeed bimodules, which are submodules of $L^{n+2}(A, \mathbf{C})$
- We need to check that the modules are all bi-injective.
- We give a bimodule projection from $L^{n+2}(A, \mathbf{C})$ onto $L_B^{n+2}(A, \mathbf{C})$
- Set $\bar{T}(\cdots, a_j, a_{j+1}, \cdots) = LIM_\lambda \sum_{i=1}^{\infty} T(\cdots, a_j a_i^\lambda, b_i^\lambda a_{j+1}, \cdots)$
- This is a bimodule map, and $\bar{T}$ is normal between a_j and a_{j+1}

In fact for dual modules we can use even more resolutions we only need to have a bi-injective resolution of A'.
Note the $I_n = P'_n$ would be such a bi-injective resolution

Normalisation w.r.t. and amenable subalgebra

Let B be an amenable subalgebra of A

- We set $I_n = L_B^{n+2}(A, \mathbf{C})$ such that for $c \in B$ and $a_i \in A$
- $T(\cdots, a_j c, a_{j+1}, \cdots) = T(\cdots, a_j, c a_{j+1}, \cdots)$, δ as above
- Note these are indeed bimodules, which are submodules of $L^{n+2}(A, \mathbf{C})$
- We need to check that the modules are all bi-injective.
- We give a bimodule projection from $L^{n+2}(A, \mathbf{C})$ onto $L_B^{n+2}(A, \mathbf{C})$
- Set $\bar{T}(\cdots, a_j, a_{j+1}, \cdots) = LIM_\lambda \sum_{i=1}^{\infty} T(\cdots, a_j a_i^\lambda, b_i^\lambda a_{j+1}, \cdots)$
- This is a bimodule map, and $\bar{T}$ is normal between a_j and a_{j+1}
- Now repeat in each place to make fully B-normal

Extensions give Derivations

Extensions give Derivations

- Given an admissible short exact sequence, of left A-modules

Extensions give Derivations

- Given an admissible short exact sequence, of left A-modules
- $0 \leftarrow X \leftarrow Y \hookleftarrow Z \leftarrow 0$

Extensions give Derivations

- Given an admissible short exact sequence, of left A-modules
- $0 \leftarrow X \leftarrow Y \hookleftarrow Z \leftarrow 0$
- This is an extension on X by Z

Extensions give Derivations

- Given an admissible short exact sequence, of left A-modules
- $0 \leftarrow X \leftarrow Y \hookleftarrow Z \leftarrow 0$
- This is an extension on X by Z
- We can write the left module action on Y as

Extensions give Derivations

- Given an admissible short exact sequence, of left A-modules
- $0 \leftarrow X \leftarrow Y \hookleftarrow Z \leftarrow 0$
- This is an extension on X by Z
- We can write the left module action on Y as
- $\theta_Y(a) = \begin{pmatrix} \theta_Z(a) & D(a) \\ 0 & \theta_X(a) \end{pmatrix}$

Extensions give Derivations

- Given an admissible short exact sequence, of left A-modules
- $$0 \leftarrow X \leftarrow Y \hookleftarrow Z \leftarrow 0$$
- This is an extension on X by Z
- We can write the left module action on Y as
- $\theta_Y(a) = \begin{pmatrix} \theta_Z(a) & D(a) \\ 0 & \theta_X(a) \end{pmatrix}$
- where D is a derivation into $L(X, Z)$, [Ex]

Extensions give Derivations

- Given an admissible short exact sequence, of left A-modules
- $0 \leftarrow X \leftarrow Y \hookleftarrow Z \leftarrow 0$
- This is an extension on X by Z
- We can write the left module action on Y as
- $\theta_Y(a) = \begin{pmatrix} \theta_Z(a) & D(a) \\ 0 & \theta_X(a) \end{pmatrix}$
- where D is a derivation into $L(X, Z)$, [Ex]
- This is inner iff $Y \cong X \oplus Z$ as an A-module [Ex]

6.1 Extensions and Ext

Extensions give Derivations

- Given an admissible short exact sequence, of left A-modules
- $$0 \leftarrow X \leftarrow Y \hookleftarrow Z \leftarrow 0$$
- This is an extension on X by Z
- We can write the left module action on Y as
- $\theta_Y(a) = \begin{pmatrix} \theta_Z(a) & D(a) \\ 0 & \theta_X(a) \end{pmatrix}$
- where D is a derivation into $L(X, Z)$, [Ex]
- This is inner iff $Y \cong X \oplus Z$ as an A-module [Ex]

Extensions give Derivations

- Given an admissible short exact sequence, of left A-modules
- $0 \leftarrow X \leftarrow Y \hookleftarrow Z \leftarrow 0$
- This is an extension on X by Z
- We can write the left module action on Y as
- $\theta_Y(a) = \begin{pmatrix} \theta_Z(a) & D(a) \\ 0 & \theta_X(a) \end{pmatrix}$
- where D is a derivation into $L(X, Z)$, [Ex]
- This is inner iff $Y \cong X \oplus Z$ as an A-module [Ex]

6.2 Two variable cohomology

Ext as 2 variable cohomology

Ext as 2 variable cohomology

- We have seen $\mathcal{H}^1(A, L(X, Z))$ classifies extensions of X by Z

Ext as 2 variable cohomology

- We have seen $\mathcal{H}^1(A, L(X, Z))$ classifies extensions of X by Z
- This leads us to consider this special bimodule's higher cohomology

Ext as 2 variable cohomology

- We have seen $\mathcal{H}^1(A, L(X, Z))$ classifies extensions of X by Z
- This leads us to consider this special bimodule's higher cohomology
- We could make the definition $\mathrm{Ext}_A^n(X, Z) := \mathcal{H}^n(A, L(X, Z))$

6.2 Two variable cohomology

Ext as 2 variable cohomology

- We have seen $\mathcal{H}^1(A, L(X, Z))$ classifies extensions of X by Z
- This leads us to consider this special bimodule's higher cohomology
- We could make the definition $\mathrm{Ext}_A^n(X, Z) := \mathcal{H}^n(A, L(X, Z))$
- In fact these cohomology groups compare longer extensions

Ext as 2 variable cohomology

- We have seen $\mathcal{H}^1(A, L(X, Z))$ classifies extensions of X by Z
- This leads us to consider this special bimodule's higher cohomology
- We could make the definition $\mathrm{Ext}^n_A(X, Z) := \mathcal{H}^n(A, L(X, Z))$
- In fact these cohomology groups compare longer extensions
- $$0 \leftarrow X \leftarrow X_1 \leftarrow \cdots \leftarrow X_n \leftarrow Z \leftarrow 0$$

Ext as 2 variable cohomology

- We have seen $\mathcal{H}^1(A, L(X, Z))$ classifies extensions of X by Z
- This leads us to consider this special bimodule's higher cohomology
- We could make the definition $\mathrm{Ext}_A^n(X, Z) := \mathcal{H}^n(A, L(X, Z))$
- In fact these cohomology groups compare longer extensions
- $0 \leftarrow X \leftarrow X_1 \leftarrow \cdots \leftarrow X_n \leftarrow Z \leftarrow 0$
- But we are interested in another definition of $\mathrm{Ext}_A^n(X, Z)$

6.2 Two variable cohomology

Ext as 2 variable cohomology

- We have seen $\mathcal{H}^1(A, L(X, Z))$ classifies extensions of X by Z
- This leads us to consider this special bimodule's higher cohomology
- We could make the definition $\mathrm{Ext}_A^n(X, Z) := \mathcal{H}^n(A, L(X, Z))$
- In fact these cohomology groups compare longer extensions
- $0 \leftarrow X \leftarrow X_1 \leftarrow \cdots \leftarrow X_n \leftarrow Z \leftarrow 0$
- But we are interested in another definition of $\mathrm{Ext}_A^n(X, Z)$
- This allows us more resolutions to compute $\mathcal{H}^n(A, L(X, Z))$

6.2 Two variable cohomology

Ext as 2 variable cohomology

- We have seen $\mathcal{H}^1(A, L(X, Z))$ classifies extensions of X by Z
- This leads us to consider this special bimodule's higher cohomology
- We could make the definition $\mathrm{Ext}_A^n(X, Z) := \mathcal{H}^n(A, L(X, Z))$
- In fact these cohomology groups compare longer extensions
- $$0 \leftarrow X \leftarrow X_1 \leftarrow \cdots \leftarrow X_n \leftarrow Z \leftarrow 0$$
- But we are interested in another definition of $\mathrm{Ext}_A^n(X, Z)$
- This allows us more resolutions to compute $\mathcal{H}^n(A, L(X, Z))$

6.2 Two variable cohomology

Ext as 2 variable cohomology

- We have seen $\mathcal{H}^1(A, L(X, Z))$ classifies extensions of X by Z
- This leads us to consider this special bimodule's higher cohomology
- We could make the definition $\mathrm{Ext}^n_A(X, Z) := \mathcal{H}^n(A, L(X, Z))$
- In fact these cohomology groups compare longer extensions
- $$0 \leftarrow X \leftarrow X_1 \leftarrow \cdots \leftarrow X_n \leftarrow Z \leftarrow 0$$
- But we are interested in another definition of $\mathrm{Ext}^n_A(X, Z)$
- This allows us more resolutions to compute $\mathcal{H}^n(A, L(X, Z))$

Ext via left modules

Ext via left modules

- This allows us more resolutions to compute $\mathcal{H}^n(A, L(X, Z))$

Ext via left modules

- This allows us more resolutions to compute $\mathcal{H}^n(A, L(X, Z))$
- **Theorem** Given any injective resolution of the left module Y

6.3 One sided injective resolutions

Ext via left modules

- This allows us more resolutions to compute $\mathcal{H}^n(A, L(X, Z))$
- **Theorem** Given any injective resolution of the left module Y
- $0 \to Y \to I_0 \to I_1 \to \cdots,$

Ext via left modules

- This allows us more resolutions to compute $\mathcal{H}^n(A, L(X, Z))$
- **Theorem** Given any injective resolution of the left module Y
- $0 \to Y \to I_0 \to I_1 \to \cdots,$

Ext via left modules

- This allows us more resolutions to compute $\mathcal{H}^n(A, L(X, Z))$
- **Theorem** Given any injective resolution of the left module Y
- $0 \to Y \to I_0 \to I_1 \to \cdots$, one-sided here!
- $\quad {}_A h(X, I_0) \to {}_A h(X, I_1) \to {}_A h(X, I_2) \to$

Ext via left modules

- This allows us more resolutions to compute $\mathcal{H}^n(A, L(X, Z))$
- **Theorem** Given any injective resolution of the left module Y
- $0 \to Y \to I_0 \to I_1 \to \cdots$, one-sided here!
- $${}_Ah(X, I_0) \to {}_Ah(X, I_1) \to {}_Ah(X, I_2) \to$$
- Is a complex whose homology is $\mathcal{H}^n(A, L(X, Z))$

Ext via left modules

- This allows us more resolutions to compute $\mathcal{H}^n(A, L(X, Z))$
- **Theorem** Given any injective resolution of the left module Y
- $0 \to Y \to I_0 \to I_1 \to \cdots$, one-sided here!
- $${}_Ah(X, I_0) \to {}_Ah(X, I_1) \to {}_Ah(X, I_2) \to$$
- Is a complex whose homology is $\mathcal{H}^n(A, L(X, Z))$
- We are also allowed to use a projective resolution of X

6.3 One sided injective resolutions

Ext via left modules

- This allows us more resolutions to compute $\mathcal{H}^n(A, L(X,Z))$
- **Theorem** Given any injective resolution of the left module Y
- $0 \to Y \to I_0 \to I_1 \to \cdots$, one-sided here!
- $\quad {}_A h(X, I_0) \to {}_A h(X, I_1) \to {}_A h(X, I_2) \to$
- Is a complex whose homology is $\mathcal{H}^n(A, L(X,Z))$
- We are also allowed to use a projective resolution of X
- $0 \leftarrow X \leftarrow P_0 \leftarrow P_1 \leftarrow \cdots$

Ext via left modules

- This allows us more resolutions to compute $\mathcal{H}^n(A, L(X, Z))$
- **Theorem** Given any injective resolution of the left module Y
- $0 \to Y \to I_0 \to I_1 \to \cdots$, one-sided here!
- $_Ah(X, I_0) \to {}_Ah(X, I_1) \to {}_Ah(X, I_2) \to$
- Is a complex whose homology is $\mathcal{H}^n(A, L(X, Z))$
- We are also allowed to use a projective resolution of X
- $0 \leftarrow X \leftarrow P_0 \leftarrow P_1 \leftarrow \cdots$
- along with the homology of

6.3 One sided injective resolutions

Ext via left modules

- This allows us more resolutions to compute $\mathcal{H}^n(A, L(X, Z))$
- **Theorem** Given any injective resolution of the left module Y
- $0 \to Y \to I_0 \to I_1 \to \cdots$, one-sided here!
- $${}_A h(X, I_0) \to {}_A h(X, I_1) \to {}_A h(X, I_2) \to$$
- Is a complex whose homology is $\mathcal{H}^n(A, L(X, Z))$
- We are also allowed to use a projective resolution of X
- $0 \leftarrow X \leftarrow P_0 \leftarrow P_1 \leftarrow \cdots$
- along with the homology of
- $${}_A h(P_0, Z) \to {}_A h(P_1, Z) \to {}_A h(P_2, Z) \to$$

6.3 One sided injective resolutions

Ext via left modules

- This allows us more resolutions to compute $\mathcal{H}^n(A, L(X, Z))$
- **Theorem** Given any injective resolution of the left module Y
- $0 \to Y \to I_0 \to I_1 \to \cdots$, one-sided here!
- $\quad {}_A h(X, I_0) \to {}_A h(X, I_1) \to {}_A h(X, I_2) \to$
- Is a complex whose homology is $\mathcal{H}^n(A, L(X, Z))$
- We are also allowed to use a projective resolution of X
- $0 \leftarrow X \leftarrow P_0 \leftarrow P_1 \leftarrow \cdots$
- along with the homology of
- $\quad {}_A h(P_0, Z) \to {}_A h(P_1, Z) \to {}_A h(P_2, Z) \to$

6.3 One sided injective resolutions

Ext via left modules

- This allows us more resolutions to compute $\mathcal{H}^n(A, L(X, Z))$
- **Theorem** Given any injective resolution of the left module Y
- $0 \to Y \to I_0 \to I_1 \to \cdots$, one-sided here!
- $${}_A h(X, I_0) \to {}_A h(X, I_1) \to {}_A h(X, I_2) \to$$
- Is a complex whose homology is $\mathcal{H}^n(A, L(X, Z))$
- We are also allowed to use a projective resolution of X
- $0 \leftarrow X \leftarrow P_0 \leftarrow P_1 \leftarrow \cdots$
- along with the homology of
- $${}_A h(P_0, Z) \to {}_A h(P_1, Z) \to {}_A h(P_2, Z) \to$$

Application 1

- **Theorem** $\mathcal{H}^1(A, L(A, Y)) = 0$ [Ex]

Application 1

- **Theorem** $\mathcal{H}^1(A, L(A, Y)) = 0$ [Ex]
- **Proof** A has a short projective resolution: $0 \leftarrow A \leftarrow A \leftarrow 0 \leftarrow \cdots$

Application 1

- **Theorem** $\mathcal{H}^1(A, L(A, Y)) = 0$ [Ex]
- **Proof** A has a short projective resolution: $0 \leftarrow A \leftarrow A \leftarrow 0 \leftarrow \cdots$
- We are allowed to compute $\mathcal{H}^n(A, L(A, Y))$ using

6.4 Special bimodules

Application 1

- **Theorem** $\mathcal{H}^1(A, L(A, Y)) = 0$ [Ex]
- **Proof** A has a short projective resolution: $0 \leftarrow A \leftarrow A \leftarrow 0 \leftarrow \cdots$
- We are allowed to compute $\mathcal{H}^n(A, L(A, Y))$ using
- $${}_A h(A, Y) \to {}_A h(0, Y) \to {}_A h(0, Y) \to$$

Application 1

- **Theorem** $\mathcal{H}^1(A, L(A, Y)) = 0$ [Ex]
- **Proof** A has a short projective resolution: $0 \leftarrow A \leftarrow A \leftarrow 0 \leftarrow \cdots$
- We are allowed to compute $\mathcal{H}^n(A, L(A, Y))$ using
- $\quad {}_A h(A, Y) \to {}_A h(0, Y) \to {}_A h(0, Y) \to$
- This complex is almost all 0's, so $\mathcal{H}^n(A, L(A, Y)) = 0$

Application 1

- **Theorem** $\mathcal{H}^1(A, L(A, Y)) = 0$ [Ex]
- **Proof** A has a short projective resolution: $0 \leftarrow A \leftarrow A \leftarrow 0 \leftarrow \cdots$
- We are allowed to compute $\mathcal{H}^n(A, L(A, Y))$ using
- $\quad {}_Ah(A, Y) \to {}_Ah(0, Y) \to {}_Ah(0, Y) \to$
- This complex is almost all 0's, so $\mathcal{H}^n(A, L(A, Y)) = 0$

Application 1

- **Theorem** $\mathcal{H}^1(A, L(A, Y)) = 0$ [Ex]
- **Proof** A has a short projective resolution: $0 \leftarrow A \leftarrow A \leftarrow 0 \leftarrow \cdots$
- We are allowed to compute $\mathcal{H}^n(A, L(A, Y))$ using
- $\quad {}_A h(A, Y) \to {}_A h(0, Y) \to {}_A h(0, Y) \to$
- This complex is almost all 0's, so $\mathcal{H}^n(A, L(A, Y)) = 0$

Application 2

- If ψ is a character where $\mathrm{Ker}\,\psi$ has a b.a.i., then $\mathcal{H}^1(A, L(X, \mathbf{C}_\psi)) = 0$

Application 1

- **Theorem** $\mathcal{H}^1(A, L(A, Y)) = 0$ [Ex]
- **Proof** A has a short projective resolution: $0 \leftarrow A \leftarrow A \leftarrow 0 \leftarrow \cdots$
- We are allowed to compute $\mathcal{H}^n(A, L(A, Y))$ using
- $$_Ah(A, Y) \to {}_Ah(0, Y) \to {}_Ah(0, Y) \to$$
- This complex is almost all 0's, so $\mathcal{H}^n(A, L(A, Y)) = 0$

Application 2

- If ψ is a character where $\mathrm{Ker}\,\psi$ has a b.a.i., then $\mathcal{H}^1(A, L(X, \mathbf{C}_\psi)) = 0$
- **Proof** $\mathbf{C}_\psi$ has a short injective resolution: $0 \to \mathbf{C}_\psi \to \mathbf{C}_\psi \to 0 \to \cdots$

Application 1

- **Theorem** $\mathcal{H}^1(A, L(A, Y)) = 0$ [Ex]
- **Proof** A has a short projective resolution: $0 \leftarrow A \leftarrow A \leftarrow 0 \leftarrow \cdots$
- We are allowed to compute $\mathcal{H}^n(A, L(A, Y))$ using
- $${}_A h(A, Y) \to {}_A h(0, Y) \to {}_A h(0, Y) \to$$
- This complex is almost all 0's, so $\mathcal{H}^n(A, L(A, Y)) = 0$

Application 2

- If ψ is a character where $\mathrm{Ker}\,\psi$ has a b.a.i., then $\mathcal{H}^1(A, L(X, \mathbf{C}_\psi)) = 0$
- **Proof** $\mathbf{C}_\psi$ has a short injective resolution: $0 \to \mathbf{C}_\psi \to \mathbf{C}_\psi \to 0 \to \cdots$
- We are allowed to compute $\mathcal{H}^n(A, L(X, \mathbf{C}_\psi))$ using

6.4 Special bimodules

Application 1

- **Theorem** $\mathcal{H}^1(A, L(A, Y)) = 0$ [Ex]
- **Proof** A has a short projective resolution: $0 \leftarrow A \leftarrow A \leftarrow 0 \leftarrow \cdots$
- We are allowed to compute $\mathcal{H}^n(A, L(A, Y))$ using
- $${}_A h(A, Y) \to {}_A h(0, Y) \to {}_A h(0, Y) \to$$
- This complex is almost all 0's, so $\mathcal{H}^n(A, L(A, Y)) = 0$

Application 2

- If ψ is a character where $\mathrm{Ker}\,\psi$ has a b.a.i., then $\mathcal{H}^1(A, L(X, \mathbf{C}_\psi)) = 0$
- **Proof** $\mathbf{C}_\psi$ has a short injective resolution: $0 \to \mathbf{C}_\psi \to \mathbf{C}_\psi \to 0 \to \cdots$
- We are allowed to compute $\mathcal{H}^n(A, L(X, \mathbf{C}_\psi))$ using
- $${}_A h(X, \mathbf{C}_\psi) \to {}_A h(X, 0) \to {}_A h(X, 0) \to$$

Application 1

- **Theorem** $\mathcal{H}^1(A, L(A, Y)) = 0$ [Ex]
- **Proof** A has a short projective resolution: $0 \leftarrow A \leftarrow A \leftarrow 0 \leftarrow \cdots$
- We are allowed to compute $\mathcal{H}^n(A, L(A, Y))$ using
- $\qquad {}_A h(A, Y) \rightarrow {}_A h(0, Y) \rightarrow {}_A h(0, Y) \rightarrow$
- This complex is almost all 0's, so $\mathcal{H}^n(A, L(A, Y)) = 0$

Application 2

- If ψ is a character where $\operatorname{Ker}\psi$ has a b.a.i., then $\mathcal{H}^1(A, L(X, \mathbf{C}_\psi)) = 0$
- **Proof** $\mathbf{C}_\psi$ has a short injective resolution: $0 \rightarrow \mathbf{C}_\psi \rightarrow \mathbf{C}_\psi \rightarrow 0 \rightarrow \cdots$
- We are allowed to compute $\mathcal{H}^n(A, L(X, \mathbf{C}_\psi))$ using
- $\qquad {}_A h(X, \mathbf{C}_\psi) \rightarrow {}_A h(X, 0) \rightarrow {}_A h(X, 0) \rightarrow$
- This complex is almost all 0's, so $\mathcal{H}^n(A, L(X, \mathbf{C}_\psi)) = 0$

Application 1

- **Theorem** $\mathcal{H}^1(A, L(A, Y)) = 0$ [Ex]
- **Proof** A has a short projective resolution: $0 \leftarrow A \leftarrow A \leftarrow 0 \leftarrow \cdots$
- We are allowed to compute $\mathcal{H}^n(A, L(A, Y))$ using
- $\qquad {}_Ah(A, Y) \to {}_Ah(0, Y) \to {}_Ah(0, Y) \to$
- This complex is almost all 0's, so $\mathcal{H}^n(A, L(A, Y)) = 0$

Application 2

- If ψ is a character where $\operatorname{Ker}\psi$ has a b.a.i., then $\mathcal{H}^1(A, L(X, \mathbf{C}_\psi)) = 0$
- **Proof** $\mathbf{C}_\psi$ has a short injective resolution: $0 \to \mathbf{C}_\psi \to \mathbf{C}_\psi \to 0 \to \cdots$
- We are allowed to compute $\mathcal{H}^n(A, L(X, \mathbf{C}_\psi))$ using
- $\qquad {}_Ah(X, \mathbf{C}_\psi) \to {}_Ah(X, 0) \to {}_Ah(X, 0) \to$
- This complex is almost all 0's, so $\mathcal{H}^n(A, L(X, \mathbf{C}_\psi)) = 0$

6.4 Special bimodules

Application 1

- **Theorem** $\mathcal{H}^1(A, L(A, Y)) = 0$ [Ex]
- **Proof** A has a short projective resolution: $0 \leftarrow A \leftarrow A \leftarrow 0 \leftarrow \cdots$
- We are allowed to compute $\mathcal{H}^n(A, L(A, Y))$ using
- $\quad {}_Ah(A, Y) \to {}_Ah(0, Y) \to {}_Ah(0, Y) \to$
- This complex is almost all 0's, so $\mathcal{H}^n(A, L(A, Y)) = 0$

Application 2

- If ψ is a character where $\mathrm{Ker}\,\psi$ has a b.a.i., then $\mathcal{H}^1(A, L(X, \mathbf{C}_\psi)) = 0$
- **Proof** $\mathbf{C}_\psi$ has a short injective resolution: $0 \to \mathbf{C}_\psi \to \mathbf{C}_\psi \to 0 \to \cdots$
- We are allowed to compute $\mathcal{H}^n(A, L(X, \mathbf{C}_\psi))$ using
- $\quad {}_Ah(X, \mathbf{C}_\psi) \to {}_Ah(X, 0) \to {}_Ah(X, 0) \to$
- This complex is almost all 0's, so $\mathcal{H}^n(A, L(X, \mathbf{C}_\psi)) = 0$

6.5 Tensor bimodules

Homology groups and Tor

Homology groups and Tor

- If E is a left module, and F a right module

6.5 Tensor bimodules

Homology groups and Tor

- If E is a left module, and F a right module
- then $L(E, F') = (E\hat{\otimes}F)'$ is a bimodule of the sort considered above

6.5 Tensor bimodules

Homology groups and Tor

- If E is a left module, and F a right module
- then $L(E, F') = (E \hat{\otimes} F)'$ is a bimodule of the sort considered above
- and $\mathrm{Ext}^n(E, F')$ is easier to compute

6.5 Tensor bimodules

Homology groups and Tor

- If E is a left module, and F a right module
- then $L(E, F') = (E \hat{\otimes} F)'$ is a bimodule of the sort considered above
- and $\mathrm{Ext}^n(E, F')$ is easier to compute
- In fact we can define `homology groups` $\mathcal{H}_n(A, M)$ for bimodules

6.5 Tensor bimodules

Homology groups and Tor

- If E is a left module, and F a right module
- then $L(E, F') = (E \hat{\otimes} F)'$ is a bimodule of the sort considered above
- and $\mathrm{Ext}^n(E, F')$ is easier to compute
- In fact we can define `homology groups` $\mathcal{H}_n(A, M)$ for bimodules
- This is the homology of the predual of the complex for $\mathcal{H}_n(A, M')$

Homology groups and Tor

- If E is a left module, and F a right module
- then $L(E, F') = (E \hat{\otimes} F)'$ is a bimodule of the sort considered above
- and $\mathrm{Ext}^n(E, F')$ is easier to compute
- In fact we can define `homology groups` $\mathcal{H}_n(A, M)$ for bimodules
- This is the homology of the predual of the complex for $\mathcal{H}_n(A, M')$
- Recall $L(\widehat{\bigotimes}^n A, M') \cong (\widehat{\bigotimes}^n A \hat{\otimes} M)'$

Homology groups and Tor

- If E is a left module, and F a right module
- then $L(E, F') = (E\hat{\otimes}F)'$ is a bimodule of the sort considered above
- and $\mathrm{Ext}^n(E, F')$ is easier to compute
- In fact we can define `homology groups` $\mathcal{H}_n(A, M)$ for bimodules
- This is the homology of the predual of the complex for $\mathcal{H}_n(A, M')$
- Recall $L(\widehat{\bigotimes}^n A, M') \cong (\widehat{\bigotimes}^n A\hat{\otimes}M)'$
- Then define $\mathrm{Tor}_n^A(E, F) = \mathcal{H}_n(A, E\hat{\otimes}F)$

6.5 Tensor bimodules

Homology groups and Tor

- If E is a left module, and F a right module
- then $L(E, F') = (E\hat{\otimes}F)'$ is a bimodule of the sort considered above
- and $\mathrm{Ext}^n(E, F')$ is easier to compute
- In fact we can define `homology groups` $\mathcal{H}_n(A, M)$ for bimodules
- This is the homology of the predual of the complex for $\mathcal{H}_n(A, M')$
- Recall $L(\widehat{\bigotimes}^n A, M') \cong (\widehat{\bigotimes}^n A\hat{\otimes}M)'$
- Then define $\mathrm{Tor}_n^A(E, F) = \mathcal{H}_n(A, E\hat{\otimes}F)$
- Often $\mathcal{H}_n(A, M)$ is the predual of $\mathcal{H}^n(A, M')$,

Homology groups and Tor

- If E is a left module, and F a right module
- then $L(E, F') = (E \hat{\otimes} F)'$ is a bimodule of the sort considered above
- and $\mathrm{Ext}^n(E, F')$ is easier to compute
- In fact we can define `homology groups` $\mathcal{H}_n(A, M)$ for bimodules
- This is the homology of the predual of the complex for $\mathcal{H}_n(A, M')$
- Recall $L(\widehat{\bigotimes}^n A, M') \cong (\widehat{\bigotimes}^n A \hat{\otimes} M)'$
- Then define $\mathrm{Tor}_n^A(E, F) = \mathcal{H}_n(A, E \hat{\otimes} F)$
- Often $\mathcal{H}_n(A, M)$ is the predual of $\mathcal{H}^n(A, M')$,
- which makes is natural for dual modules

6.5 Tensor bimodules

Homology groups and Tor

- If E is a left module, and F a right module
- then $L(E, F') = (E \hat{\otimes} F)'$ is a bimodule of the sort considered above
- and $\mathrm{Ext}^n(E, F')$ is easier to compute
- In fact we can define `homology groups` $\mathcal{H}_n(A, M)$ for bimodules
- This is the homology of the predual of the complex for $\mathcal{H}_n(A, M')$
- Recall $L(\widehat{\bigotimes}^n A, M') \cong (\widehat{\bigotimes}^n A \hat{\otimes} M)'$
- Then define $\mathrm{Tor}_n^A(E, F) = \mathcal{H}_n(A, E \hat{\otimes} F)$
- Often $\mathcal{H}_n(A, M)$ is the predual of $\mathcal{H}^n(A, M')$,
- which makes is natural for dual modules
- However, sometimes neither is even a Banach Space

6.5 Tensor bimodules

Homology groups and Tor

- If E is a left module, and F a right module
- then $L(E, F') = (E\hat{\otimes}F)'$ is a bimodule of the sort considered above
- and $\mathrm{Ext}^n(E, F')$ is easier to compute
- In fact we can define `homology groups` $\mathcal{H}_n(A, M)$ for bimodules
- This is the homology of the predual of the complex for $\mathcal{H}_n(A, M')$
- Recall $L(\widehat{\bigotimes}^n A, M') \cong (\widehat{\bigotimes}^n A\hat{\otimes}M)'$
- Then define $\mathrm{Tor}_n^A(E, F) = \mathcal{H}_n(A, E\hat{\otimes}F)$
- Often $\mathcal{H}_n(A, M)$ is the predual of $\mathcal{H}^n(A, M')$,
- which makes is natural for dual modules
- However, sometimes neither is even a Banach Space

6.5 Tensor bimodules

Homology groups and Tor

- If E is a left module, and F a right module
- then $L(E, F') = (E \hat{\otimes} F)'$ is a bimodule of the sort considered above
- and $\mathrm{Ext}^n(E, F')$ is easier to compute
- In fact we can define `homology groups` $\mathcal{H}_n(A, M)$ for bimodules
- This is the homology of the predual of the complex for $\mathcal{H}_n(A, M')$
- Recall $L(\widehat{\bigotimes}^n A, M') \cong (\widehat{\bigotimes}^n A \hat{\otimes} M)'$
- Then define $\mathrm{Tor}_n^A(E, F) = \mathcal{H}_n(A, E \hat{\otimes} F)$
- Often $\mathcal{H}_n(A, M)$ is the predual of $\mathcal{H}^n(A, M')$,
- which makes is natural for dual modules
- However, sometimes neither is even a Banach Space

Cyclic Cohomology – a quick review

Recall

- We compute the simplicial cohomology, $\mathcal{HH}^n(A)$ using the complex

$$\to L^{n-1}(A; A') \to L^n(A; A') \to L^{n+1}(A; A') \to$$

Cyclic Cohomology – a quick review

Recall

- We compute the simplicial cohomology, $\mathcal{HH}^n(A)$ using the complex

$$\to L^{n-1}(A; A') \to L^n(A; A') \to L^{n+1}(A; A') \to$$

- which is actually just

$$\to (\widehat{\bigotimes}^n A)' \to (\widehat{\bigotimes}^{n+1} A)' \to (\widehat{\bigotimes}^{n+2} A)' \to$$

Cyclic Cohomology – a quick review

Recall

- We compute the simplicial cohomology, $\mathcal{HH}^n(A)$ using the complex

$$\to L^{n-1}(A; A') \to L^n(A; A') \to L^{n+1}(A; A') \to$$

- which is actually just

$$\to (\widehat{\bigotimes}^n A)' \to (\widehat{\bigotimes}^{n+1} A)' \to (\widehat{\bigotimes}^{n+2} A)' \to$$

- this extra symmetry allows us to impose an extra condition on our multilinear maps. We say a map T is *cyclic* if

$$T(a_1, a_2, \ldots, a_n)(a_0) = (-1)^n T(a_2, a_3, \ldots, a_0)(a_1)$$

Recall

- We compute the simplicial cohomology, $\mathcal{HH}^n(A)$ using the complex

$$\to L^{n-1}(A;A') \to L^n(A;A') \to L^{n+1}(A;A') \to$$

- which is actually just

$$\to (\widehat{\bigotimes}^n A)' \to (\widehat{\bigotimes}^{n+1} A)' \to (\widehat{\bigotimes}^{n+2} A)' \to$$

- this extra symmetry allows us to impose an extra condition on our multilinear maps. We say a map T is *cyclic* if

$$T(a_1, a_2, \ldots, a_n)(a_0) = (-1)^n T(a_2, a_3, \ldots, a_0)(a_1)$$

- Surprisingly if T is cyclic then so is δT, this allows us to make the definitions

Cyclic Cohomology – a quick review

Recall

- We compute the simplicial cohomology, $\mathcal{HH}^n(A)$ using the complex

$$\to L^{n-1}(A;A') \to L^n(A;A') \to L^{n+1}(A;A') \to$$

- which is actually just

$$\to (\widehat{\bigotimes}^n A)' \to (\widehat{\bigotimes}^{n+1} A)' \to (\widehat{\bigotimes}^{n+2} A)' \to$$

- this extra symmetry allows us to impose an extra condition on our multilinear maps. We say a map T is *cyclic* if

$$T(a_1, a_2, \ldots, a_n)(a_0) = (-1)^n T(a_2, a_3, \ldots, a_0)(a_1)$$

- Surprisingly if T is cyclic then so is δT, this allows us to make the definitions
  - $\mathcal{ZC}^n(A)$ the space of cocycles which are cyclic;

Cyclic Cohomology – a quick review

Recall

- We compute the simplicial cohomology, $\mathcal{HH}^n(A)$ using the complex

$$\to L^{n-1}(A; A') \to L^n(A; A') \to L^{n+1}(A; A') \to$$

- which is actually just

$$\to (\widehat{\bigotimes}^n A)' \to (\widehat{\bigotimes}^{n+1} A)' \to (\widehat{\bigotimes}^{n+2} A)' \to$$

- this extra symmetry allows us to impose an extra condition on our multilinear maps. We say a map T is *cyclic* if

$$T(a_1, a_2, \ldots, a_n)(a_0) = (-1)^n T(a_2, a_3, \ldots, a_0)(a_1)$$

- Surprisingly if T is cyclic then so is δT, this allows us to make the definitions
  - $\mathcal{ZC}^n(A)$ the space of cocycles which are cyclic;
  - $\mathcal{BC}^n(A)$ the space of boundaries of cyclic cochains; *(not coboundaries which are cyclic)*;

Cyclic Cohomology – a quick review

Recall

- We compute the simplicial cohomology, $\mathcal{HH}^n(A)$ using the complex

$$\to L^{n-1}(A; A') \to L^n(A; A') \to L^{n+1}(A; A') \to$$

- which is actually just

$$\to (\widehat{\bigotimes}^n A)' \to (\widehat{\bigotimes}^{n+1} A)' \to (\widehat{\bigotimes}^{n+2} A)' \to$$

- this extra symmetry allows us to impose an extra condition on our multilinear maps. We say a map T is *cyclic* if

$$T(a_1, a_2, \dots, a_n)(a_0) = (-1)^n T(a_2, a_3, \dots, a_0)(a_1)$$

- Surprisingly if T is cyclic then so is δT, this allows us to make the definitions
  - $\mathcal{ZC}^n(A)$ the space of cocycles which are cyclic;
  - $\mathcal{BC}^n(A)$ the space of boundaries of cyclic cochains; *(not coboundaries which are cyclic)*;
  - $\mathcal{HC}^n(A) = \mathcal{ZC}^n(A)/\mathcal{BC}^n(A)$.

- The Simplicial and the Cyclic cohomology groups are connected by the Connes-Tzygan long exact sequences.

$$0 \to \mathcal{HH}^1(A) \to \mathcal{HC}^0(A) \to \mathcal{HC}^2(A) \to \mathcal{HH}^2(A) \to \mathcal{HC}^1(A) \to \cdots$$
$$\to \mathcal{HH}^n(A) \to \mathcal{HC}^{n-1}(A) \to \mathcal{HC}^{n+1}(A) \to \mathcal{HH}^{n+1}(A) \to \cdots$$

7.1 Connes-Tzygan

- The Simplicial and the Cyclic cohomology groups are connected by the Connes-Tzygan long exact sequences.

$$0 \to \mathcal{HH}^1(A) \to \mathcal{HC}^0(A) \to \mathcal{HC}^2(A) \to \mathcal{HH}^2(A) \to \mathcal{HC}^1(A) \to \cdots$$
$$\to \mathcal{HH}^n(A) \to \mathcal{HC}^{n-1}(A) \to \mathcal{HC}^{n+1}(A) \to \mathcal{HH}^{n+1}(A) \to \cdots$$

- 2 Observations

7.1 Connes-Tzygan

- The Simplicial and the Cyclic cohomology groups are connected by the Connes-Tzygan long exact sequences.

$$0 \to \mathcal{HH}^1(A) \to \mathcal{HC}^0(A) \to \mathcal{HC}^2(A) \to \mathcal{HH}^2(A) \to \mathcal{HC}^1(A) \to \cdots$$
$$\to \mathcal{HH}^n(A) \to \mathcal{HC}^{n-1}(A) \to \mathcal{HC}^{n+1}(A) \to \mathcal{HH}^{n+1}(A) \to \cdots$$

- 2 Observations
  - If, for large n, $\mathcal{HH}^n(A) = 0$ then $\mathcal{HC}^{n-1}(A) \cong \mathcal{HC}^{n+1}(A)$ and so we only have $\mathcal{HC}^{odd}(A)$ and $\mathcal{HC}^{even}(A)$;

7.1 Connes-Tzygan

- The Simplicial and the Cyclic cohomology groups are connected by the Connes-Tzygan long exact sequences.

$$0 \to \mathcal{HH}^1(A) \to \mathcal{HC}^0(A) \to \mathcal{HC}^2(A) \to \mathcal{HH}^2(A) \to \mathcal{HC}^1(A) \to \cdots$$
$$\to \mathcal{HH}^n(A) \to \mathcal{HC}^{n-1}(A) \to \mathcal{HC}^{n+1}(A) \to \mathcal{HH}^{n+1}(A) \to \cdots$$

- 2 Observations
  - If, for large n, $\mathcal{HH}^n(A) = 0$ then $\mathcal{HC}^{n-1}(A) \cong \mathcal{HC}^{n+1}(A)$ and so we only have $\mathcal{HC}^{odd}(A)$ and $\mathcal{HC}^{even}(A)$;
  - If, for large n, $\mathcal{HC}^n(A) = 0$ then $\mathcal{HH}^n(A) = 0$;

7.1 Connes-Tzygan

- The Simplicial and the Cyclic cohomology groups are connected by the Connes-Tzygan long exact sequences.

$$0 \to \mathcal{HH}^1(A) \to \mathcal{HC}^0(A) \to \mathcal{HC}^2(A) \to \mathcal{HH}^2(A) \to \mathcal{HC}^1(A) \to \cdots$$
$$\to \mathcal{HH}^n(A) \to \mathcal{HC}^{n-1}(A) \to \mathcal{HC}^{n+1}(A) \to \mathcal{HH}^{n+1}(A) \to \cdots$$

- 2 Observations
  - If, for large n, $\mathcal{HH}^n(A) = 0$ then $\mathcal{HC}^{n-1}(A) \cong \mathcal{HC}^{n+1}(A)$ and so we only have $\mathcal{HC}^{odd}(A)$ and $\mathcal{HC}^{even}(A)$;
  - If, for large n, $\mathcal{HC}^n(A) = 0$ then $\mathcal{HH}^n(A) = 0$;
  - In fact it rarely happens like this as $\mathcal{HC}^{odd}(C) = 0$ and $\mathcal{HC}^{even}(C) = 0$,

7.1 Connes-Tzygan

- The Simplicial and the Cyclic cohomology groups are connected by the Connes-Tzygan long exact sequences.

$$0 \to \mathcal{HH}^1(A) \to \mathcal{HC}^0(A) \to \mathcal{HC}^2(A) \to \mathcal{HH}^2(A) \to \mathcal{HC}^1(A) \to \cdots$$
$$\to \mathcal{HH}^n(A) \to \mathcal{HC}^{n-1}(A) \to \mathcal{HC}^{n+1}(A) \to \mathcal{HH}^{n+1}(A) \to \cdots$$

- 2 Observations
  - If, for large n, $\mathcal{HH}^n(A) = 0$ then $\mathcal{HC}^{n-1}(A) \cong \mathcal{HC}^{n+1}(A)$ and so we only have $\mathcal{HC}^{odd}(A)$ and $\mathcal{HC}^{even}(A)$;
  - If, for large n, $\mathcal{HC}^n(A) = 0$ then $\mathcal{HH}^n(A) = 0$;
  - In fact it rarely happens like this as $\mathcal{HC}^{odd}(C) = 0$ and $\mathcal{HC}^{even}(C) = 0$,

- The Simplicial and the Cyclic cohomology groups are connected by the Connes-Tzygan long exact sequences.

$$0 \to \mathcal{HH}^1(A) \to \mathcal{HC}^0(A) \to \mathcal{HC}^2(A) \to \mathcal{HH}^2(A) \to \mathcal{HC}^1(A) \to \cdots$$
$$\to \mathcal{HH}^n(A) \to \mathcal{HC}^{n-1}(A) \to \mathcal{HC}^{n+1}(A) \to \mathcal{HH}^{n+1}(A) \to \cdots$$

- 2 Observations
  - If, for large n, $\mathcal{HH}^n(A) = 0$ then $\mathcal{HC}^{n-1}(A) \cong \mathcal{HC}^{n+1}(A)$ and so we only have $\mathcal{HC}^{odd}(A)$ and $\mathcal{HC}^{even}(A)$;
  - If, for large n, $\mathcal{HC}^n(A) = 0$ then $\mathcal{HH}^n(A) = 0$;
  - In fact it rarely happens like this as $\mathcal{HC}^{odd}(C) = 0$ and $\mathcal{HC}^{even}(C) = 0$, but $\mathcal{HC}^{n-1}(A) \cong \mathcal{HC}^{n+1}(A)$ is often enough to deduce the triviality of the higher simplicial cohomology groups.

7.2 Example of Cyclic Cohomology

- e.g. 1: The algebras $\ell^1(Z_+, +)$ has simplicial derivations, namely

$$D(z^n)(z^m) = nD(z^1)(z^{n+m-1}) = \frac{n}{n+m} D(z^{n+m})(1) = \tau_D(z^{n+m})$$

where τ_D is any element of A' (trace), which vanishes on 1.

7.2 Example of Cyclic Cohomology

- e.g. 1: The algebras $\ell^1(Z_+, +)$ has simplicial derivations, namely

$$D(z^n)(z^m) = nD(z^1)(z^{n+m-1}) = \frac{n}{n+m} D(z^{n+m})(1) = \tau_D(z^{n+m})$$

  where τ_D is any element of A' (trace), which vanishes on 1.
- However the following simple computation shows that $\ell^1(Z_+, +)$ has no simplicial derivations:

$$D(z^n)(z^m) = \frac{n}{n+m} D(z^{n+m})(1) = \frac{n}{n+m} D(1)(z^{n+m}) = 0.$$

7.2 Example of Cyclic Cohomology

- e.g. 1: The algebras $\ell^1(Z_+, +)$ has simplicial derivations, namely

$$D(z^n)(z^m) = nD(z^1)(z^{n+m-1}) = \frac{n}{n+m} D(z^{n+m})(1) = \tau_D(z^{n+m})$$

  where τ_D is any element of A' (trace), which vanishes on 1.
- However the following simple computation shows that $\ell^1(Z_+, +)$ has no simplicial derivations:

$$D(z^n)(z^m) = \frac{n}{n+m} D(z^{n+m})(1) = \frac{n}{n+m} D(1)(z^{n+m}) = 0.$$

- It then follows from the Connes-Tzygan long exact sequence

$$0 \to \mathcal{HH}^1(A) \to \mathcal{HC}^0(A) \to \mathcal{HC}^2(A) \to 0$$

  which gives $\mathcal{HC}^2(A) = \mathbf{C}$.

7.3 Vanishing Higher Cyclic Cohomology

- Recall a 2-cocycle ϕ is called cyclic if $\phi(f, g)(h) = +\phi(g, h)(f)$.

7.3 Vanishing Higher Cyclic Cohomology

- Recall a 2-cocycle ϕ is called cyclic if $\phi(f, g)(h) = +\phi(g, h)(f)$.
- Given any trace τ we can define a cyclic 2-cocycle by $\phi_\tau(f, g)(h) = \tau(fgh)$. [Check the cocycle identity.]

7.3 Vanishing Higher Cyclic Cohomology

- Recall a 2-cocycle ϕ is called cyclic if $\phi(f,g)(h) = +\phi(g,h)(f)$.
- Given any trace τ we can define a cyclic 2-cocycle by $\phi_\tau(f,g)(h) = \tau(fgh)$. [Check the cocycle identity.]
- Moreover, it is difficult for such cyclic 2-cocycles to cobound, for if $\phi = \delta\psi$, then given any idempotent $e \in A$, we have

$$\phi(e,e)(e) = \psi(e)(e^2) - \psi(e^2)(2) + \psi(e)(e^2) = \psi(e)(e),$$

but as $\tau(e) = \tau(eee) = \phi(e,e)(e) = -\psi(e)(e) = 0$.

7.3 Vanishing Higher Cyclic Cohomology

- Recall a 2-cocycle ϕ is called cyclic if $\phi(f,g)(h) = +\phi(g,h)(f)$.
- Given any trace τ we can define a cyclic 2-cocycle by $\phi_\tau(f,g)(h) = \tau(fgh)$. [Check the cocycle identity.]
- Moreover, it is difficult for such cyclic 2-cocycles to cobound, for if $\phi = \delta\psi$, then given any idempotent $e \in A$, we have

$$\phi(e,e)(e) = \psi(e)(e^2) - \psi(e^2)(2) + \psi(e)(e^2) = \psi(e)(e),$$

  but as $\tau(e) = \tau(eee) = \phi(e,e)(e) = -\psi(e)(e) = 0$.
- Thus we have a non-vanishing class in $\mathcal{HC}^2(A)$ whenever we have a trace which does not vanish on some idempotent.

7.3 Vanishing Higher Cyclic Cohomology

- Recall a 2-cocycle ϕ is called cyclic if $\phi(f,g)(h) = +\phi(g,h)(f)$.
- Given any trace τ we can define a cyclic 2-cocycle by $\phi_\tau(f,g)(h) = \tau(fgh)$. [Check the cocycle identity.]
- Moreover, it is difficult for such cyclic 2-cocycles to cobound, for if $\phi = \delta\psi$, then given any idempotent $e \in A$, we have

$$\phi(e,e)(e) = \psi(e)(e^2) - \psi(e^2)(2) + \psi(e)(e^2) = \psi(e)(e),$$

  but as $\tau(e) = \tau(eee) = \phi(e,e)(e) = -\psi(e)(e) = 0$.
- Thus we have a non-vanishing class in $\mathcal{HC}^2(A)$ whenever we have a trace which does not vanish on some idempotent.
- However, this is often the only way they can arise.

7.3 Vanishing Higher Cyclic Cohomology

- Recall a 2-cocycle ϕ is called cyclic if $\phi(f,g)(h) = +\phi(g,h)(f)$.
- Given any trace τ we can define a cyclic 2-cocycle by $\phi_\tau(f,g)(h) = \tau(fgh)$. [Check the cocycle identity.]
- Moreover, it is difficult for such cyclic 2-cocycles to cobound, for if $\phi = \delta\psi$, then given any idempotent $e \in A$, we have

$$\phi(e,e)(e) = \psi(e)(e^2) - \psi(e^2)(2) + \psi(e)(e^2) = \psi(e)(e),$$

  but as $\tau(e) = \tau(eee) = \phi(e,e)(e) = -\psi(e)(e) = 0$.
- Thus we have a non-vanishing class in $\mathcal{HC}^2(A)$ whenever we have a trace which does not vanish on some idempotent.
- However, this is often the only way they can arise.
- (Recall, we have already seen that $\mathcal{HC}^2(\ell^1(Z_+,+)) = \mathbf{C}$.)

- Recall a 2-cocycle ϕ is called cyclic if $\phi(f,g)(h) = +\phi(g,h)(f)$.
- Given any trace τ we can define a cyclic 2-cocycle by $\phi_\tau(f,g)(h) = \tau(fgh)$. [Check the cocycle identity.]
- Moreover, it is difficult for such cyclic 2-cocycles to cobound, for if $\phi = \delta\psi$, then given any idempotent $e \in A$, we have

$$\phi(e,e)(e) = \psi(e)(e^2) - \psi(e^2)(2) + \psi(e)(e^2) = \psi(e)(e),$$

  but as $\tau(e) = \tau(eee) = \phi(e,e)(e) = -\psi(e)(e) = 0$.
- Thus we have a non-vanishing class in $\mathcal{HC}^2(A)$ whenever we have a trace which does not vanish on some idempotent.
- However, this is often the only way they can arise.
- (Recall, we have already seen that $\mathcal{HC}^2(\ell^1(Z_+,+)) = \mathbf{C}$.)
- However, for C_2, the Hilbert-Schmidt operators, one can see that $\phi_\tau(f,g)(h) = \tau(fgh)$,

7.3 Vanishing Higher Cyclic Cohomology

- Recall a 2-cocycle ϕ is called cyclic if $\phi(f,g)(h) = +\phi(g,h)(f)$.
- Given any trace τ we can define a cyclic 2-cocycle by $\phi_\tau(f,g)(h) = \tau(fgh)$. [Check the cocycle identity.]
- Moreover, it is difficult for such cyclic 2-cocycles to cobound, for if $\phi = \delta\psi$, then given any idempotent $e \in A$, we have

$$\phi(e,e)(e) = \psi(e)(e^2) - \psi(e^2)(2) + \psi(e)(e^2) = \psi(e)(e),$$

  but as $\tau(e) = \tau(eee) = \phi(e,e)(e) = -\psi(e)(e) = 0$.
- Thus we have a non-vanishing class in $\mathcal{HC}^2(A)$ whenever we have a trace which does not vanish on some idempotent.
- However, this is often the only way they can arise.
- (Recall, we have already seen that $\mathcal{HC}^2(\ell^1(Z_+,+)) = \mathbf{C}$.)
- However, for C_2, the Hilbert-Schmidt operators, one can see that $\phi_\tau(f,g)(h) = \tau(fgh)$,

- Recall a 2-cocycle ϕ is called cyclic if $\phi(f, g)(h) = +\phi(g, h)(f)$.
- Given any trace τ we can define a cyclic 2-cocycle by $\phi_\tau(f, g)(h) = \tau(fgh)$. [Check the cocycle identity.]
- Moreover, it is difficult for such cyclic 2-cocycles to cobound, for if $\phi = \delta\psi$, then given any idempotent $e \in A$, we have

$$\phi(e, e)(e) = \psi(e)(e^2) - \psi(e^2)(2) + \psi(e)(e^2) = \psi(e)(e),$$

  but as $\tau(e) = \tau(eee) = \phi(e, e)(e) = -\psi(e)(e) = 0$.
- Thus we have a non-vanishing class in $\mathcal{HC}^2(A)$ whenever we have a trace which does not vanish on some idempotent.
- However, this is often the only way they can arise.
- (Recall, we have already seen that $\mathcal{HC}^2(\ell^1(Z_+, +)) = \mathbf{C}$.)
- However, for C_2, the Hilbert-Schmidt operators, one can see that $\phi_\tau(f, g)(h) = \tau(fgh)$, but C_2 has no non-trivial trace.

7.4 Idempotents and Cyclic Cohomology

- Let τ be a trace on A. So $\tau \in \mathcal{HC}^0(A)$

7.4 Idempotents and Cyclic Cohomology

- Let τ be a trace on A. So $\tau \in \mathcal{HC}^0(A)$
- Imagine $e(t)$ is a differentiable family of idempotents

7.4 Idempotents and Cyclic Cohomology

- Let τ be a trace on A. So $\tau \in \mathcal{HC}^0(A)$
- Imagine $e(t)$ is a differentiable family of idempotents
- Note $\tau(e)' = \tau(\dot{e})$

7.4 Idempotents and Cyclic Cohomology

- Let τ be a trace on A. So $\tau \in \mathcal{HC}^0(A)$
- Imagine $e(t)$ is a differentiable family of idempotents
- Note $\tau(e)' = \tau(\dot{e})$
- $e^2 = e$

7.4 Idempotents and Cyclic Cohomology

- Let τ be a trace on A. So $\tau \in \mathcal{HC}^0(A)$
- Imagine $e(t)$ is a differentiable family of idempotents
- Note $\tau(e)' = \tau(\dot{e})$
- $e^2 = e$
- $e\dot{e} + \dot{e}e = \dot{e}$

7.4 Idempotents and Cyclic Cohomology

- Let τ be a trace on A. So $\tau \in \mathcal{HC}^0(A)$
- Imagine $e(t)$ is a differentiable family of idempotents
- Note $\tau(e)' = \tau(\dot{e})$
- $e^2 = e$
- $e\dot{e} + \dot{e}e = \dot{e}$
- $e\dot{e} + e\dot{e}e = e\dot{e}$

7.4 Idempotents and Cyclic Cohomology

- Let τ be a trace on A. So $\tau \in \mathcal{HC}^0(A)$
- Imagine $e(t)$ is a differentiable family of idempotents
- Note $\tau(e)' = \tau(\dot{e})$
- $e^2 = e$
- $e\dot{e} + \dot{e}e = \dot{e}$
- $e\dot{e} + e\dot{e}e = e\dot{e}$
- $\tau(e\dot{e}) + \tau(e\dot{e}e) = \tau(e\dot{e})$

7.4 Idempotents and Cyclic Cohomology

- Let τ be a trace on A. So $\tau \in \mathcal{HC}^0(A)$
- Imagine $e(t)$ is a differentiable family of idempotents
- Note $\tau(e)' = \tau(\dot{e})$
- $e^2 = e$
- $e\dot{e} + \dot{e}e = \dot{e}$
- $e\dot{e} + e\dot{e}e = e\dot{e}$
- $\tau(e\dot{e}) + \tau(e\dot{e}e) = \tau(e\dot{e})$
- $\tau(e\dot{e}) + \tau(ee\dot{e}) = \tau(e\dot{e})$

7.4 Idempotents and Cyclic Cohomology

- Let τ be a trace on A. So $\tau \in \mathcal{HC}^0(A)$
- Imagine $e(t)$ is a differentiable family of idempotents
- Note $\tau(e)' = \tau(\dot{e})$
- $e^2 = e$
- $e\dot{e} + \dot{e}e = \dot{e}$
- $e\dot{e} + e\dot{e}e = e\dot{e}$
- $\tau(e\dot{e}) + \tau(e\dot{e}e) = \tau(e\dot{e})$
- $\tau(e\dot{e}) + \tau(ee\dot{e}) = \tau(e\dot{e})$
- $0 = \tau(ee\dot{e}) = \tau(e\dot{e}) = \tau(\dot{e}e) = 0$

- Let τ be a trace on A. So $\tau \in \mathcal{HC}^0(A)$
- Imagine $e(t)$ is a differentiable family of idempotents
- Note $\tau(e)' = \tau(\dot{e})$
- $e^2 = e$
- $e\dot{e} + \dot{e}e = \dot{e}$
- $e\dot{e} + e\dot{e}e = e\dot{e}$
- $\tau(e\dot{e}) + \tau(e\dot{e}e) = \tau(e\dot{e})$
- $\tau(e\dot{e}) + \tau(ee\dot{e}) = \tau(e\dot{e})$
- $0 = \tau(ee\dot{e}) = \tau(e\dot{e}) = \tau(\dot{e}e) = 0$
- $\tau(e)' = \tau(\dot{e}) = \tau(e\dot{e} + \dot{e}e) = 0$

7.4 Idempotents and Cyclic Cohomology

- Let τ be a trace on A. So $\tau \in \mathcal{HC}^0(A)$
- Imagine $e(t)$ is a differentiable family of idempotents
- Note $\tau(e)' = \tau(\dot{e})$
- $e^2 = e$
- $e\dot{e} + \dot{e}e = \dot{e}$
- $e\dot{e} + e\dot{e}e = e\dot{e}$
- $\tau(e\dot{e}) + \tau(e\dot{e}e) = \tau(e\dot{e})$
- $\tau(e\dot{e}) + \tau(ee\dot{e}) = \tau(e\dot{e})$
- $0 = \tau(ee\dot{e}) = \tau(e\dot{e}) = \tau(\dot{e}e) = 0$
- $\tau(e)' = \tau(\dot{e}) = \tau(e\dot{e} + \dot{e}e) = 0$
- So $\tau(e)$ is constant on components

- Let τ be a trace on A. So $\tau \in \mathcal{HC}^0(A)$
- Imagine $e(t)$ is a differentiable family of idempotents
- Note $\tau(e)' = \tau(\dot{e})$
- $e^2 = e$
- $e\dot{e} + \dot{e}e = \dot{e}$
- $e\dot{e} + e\dot{e}e = e\dot{e}$
- $\tau(e\dot{e}) + \tau(e\dot{e}e) = \tau(e\dot{e})$
- $\tau(e\dot{e}) + \tau(ee\dot{e}) = \tau(e\dot{e})$
- $0 = \tau(ee\dot{e}) = \tau(e\dot{e}) = \tau(\dot{e}e) = 0$
- $\tau(e)' = \tau(\dot{e}) = \tau(e\dot{e} + \dot{e}e) = 0$
- So $\tau(e)$ is constant on components
- Now, given $\phi \in \mathcal{HC}^2(A)$

7.4 Idempotents and Cyclic Cohomology

- Let τ be a trace on A. So $\tau \in \mathcal{HC}^0(A)$
- Imagine $e(t)$ is a differentiable family of idempotents
- Note $\tau(e)' = \tau(\dot{e})$
- $e^2 = e$
- $e\dot{e} + \dot{e}e = \dot{e}$
- $e\dot{e} + e\dot{e}e = e\dot{e}$
- $\tau(e\dot{e}) + \tau(e\dot{e}e) = \tau(e\dot{e})$
- $\tau(e\dot{e}) + \tau(ee\dot{e}) = \tau(e\dot{e})$
- $0 = \tau(ee\dot{e}) = \tau(e\dot{e}) = \tau(\dot{e}e) = 0$
- $\tau(e)' = \tau(\dot{e}) = \tau(e\dot{e} + \dot{e}e) = 0$
- So $\tau(e)$ is constant on components
- Now, given $\phi \in \mathcal{HC}^2(A)$
- Show $\phi(e, e)(e)' = 0$ [Ex]