The $\mathcal{ART}$ of Linear Algebra

Elisha Peterson

United States Military Academy

December 5, 2007

Outline

Motivating Questions

- Is there a way to generalize the cross product?
- Is there a quick way to generate inner/cross product identities such as the triple product?
- Why are the trace and determinant functions so special?
- What is the "best" way to compute the determinant?
- Why do the trace and determinant show up in the characteristic polynomial?
- What's with this *duality* thing?

Theme: notation is useful... but only if we can understand it!

Donning his new canine decoder, Professor Schwartzman becomes the first human being on Earth to hear what barking dogs are actually saying.

Symmetries in Linear Algebra

Let $\mathbf{u}$, $\mathbf{v}$, and $\mathbf{w}$ represent vectors, and let $\mathbf{A}$ represent a matrix.

Symmetries:

- The *inner product* $\langle \mathbf{u}, \mathbf{v} \rangle = \mathbf{u} \cdot \mathbf{v}$ satisfies $\langle \mathbf{u}, \mathbf{v} \rangle = \langle \mathbf{v}, \mathbf{u} \rangle$;
- Since $\langle \mathbf{u}, \mathbf{v} \rangle = \mathbf{u}^T \mathbf{v}$, it is also true that $\langle \mathbf{A}\mathbf{u}, \mathbf{v} \rangle = \langle \mathbf{u}, \mathbf{A}^T \mathbf{v} \rangle$.
- The matrix *trace* statisfies $\mathrm{tr}(\mathbf{A}) = \mathrm{tr}(\mathbf{A}^T)$;

Anti-Symmetries:

- The *cross product* satisfies $\mathbf{u} \times \mathbf{v} = -\mathbf{v} \times \mathbf{u}$;
- The matrix *determinant* satisfies $\det[\mathbf{u}\ \mathbf{v}\ \mathbf{w}] = -\det[\mathbf{v}\ \mathbf{u}\ \mathbf{w}]$;
- The *triple product identity* relates these two constructions:

$$\langle \mathbf{u} \times \mathbf{v}, \mathbf{w} \rangle = \det[\mathbf{u}\ \mathbf{v}\ \mathbf{w}].$$

Symmetries in Linear Algebra

Let $\mathbf{u}$, $\mathbf{v}$, and $\mathbf{w}$ represent vectors, and let $\mathbf{A}$ represent a matrix.

Symmetries:

- The *inner product* $\langle \mathbf{u}, \mathbf{v} \rangle = \mathbf{u} \cdot \mathbf{v}$ satisfies $\langle \mathbf{u}, \mathbf{v} \rangle = \langle \mathbf{v}, \mathbf{u} \rangle$;
- Since $\langle \mathbf{u}, \mathbf{v} \rangle = \mathbf{u}^T \mathbf{v}$, it is also true that $\langle \mathbf{A}\mathbf{u}, \mathbf{v} \rangle = \langle \mathbf{u}, \mathbf{A}^T \mathbf{v} \rangle$.
- The matrix *trace* statisfies $\mathrm{tr}(\mathbf{A}) = \mathrm{tr}(\mathbf{A}^T)$;

Anti-Symmetries:

- The *cross product* satisfies $\mathbf{u} \times \mathbf{v} = -\mathbf{v} \times \mathbf{u}$;
- The matrix *determinant* satisfies $\det[\mathbf{u}\ \mathbf{v}\ \mathbf{w}] = -\det[\mathbf{v}\ \mathbf{u}\ \mathbf{w}]$;
- The *triple product identity* relates these two constructions:

$$\langle \mathbf{u} \times \mathbf{v}, \mathbf{w} \rangle = \det[\mathbf{u}\ \mathbf{v}\ \mathbf{w}].$$

Symmetries in Linear Algebra

Let $\mathbf{u}$, $\mathbf{v}$, and $\mathbf{w}$ represent vectors, and let $\mathbf{A}$ represent a matrix.

Symmetries:

- The *inner product* $\langle \mathbf{u}, \mathbf{v} \rangle = \mathbf{u} \cdot \mathbf{v}$ satisfies $\langle \mathbf{u}, \mathbf{v} \rangle = \langle \mathbf{v}, \mathbf{u} \rangle$;
- Since $\langle \mathbf{u}, \mathbf{v} \rangle = \mathbf{u}^T \mathbf{v}$, it is also true that $\langle \mathbf{A}\mathbf{u}, \mathbf{v} \rangle = \langle \mathbf{u}, \mathbf{A}^T \mathbf{v} \rangle$.
- The matrix *trace* statisfies $\mathrm{tr}(\mathbf{A}) = \mathrm{tr}(\mathbf{A}^T)$;

Anti-Symmetries:

- The *cross product* satisfies $\mathbf{u} \times \mathbf{v} = -\mathbf{v} \times \mathbf{u}$;
- The matrix *determinant* satisfies $\det[\mathbf{u}\ \mathbf{v}\ \mathbf{w}] = -\det[\mathbf{v}\ \mathbf{u}\ \mathbf{w}]$;
- The *triple product identity* relates these two constructions:

$$\langle \mathbf{u} \times \mathbf{v}, \mathbf{w} \rangle = \det[\mathbf{u}\ \mathbf{v}\ \mathbf{w}].$$

Tensor Algebra

A *tensor product* of two vector spaces consists of pairs of elements such that

$$(\lambda \mathbf{v}, \mathbf{w}) = \lambda(\mathbf{v}, \mathbf{w}) = (\mathbf{v}, \lambda \mathbf{w})$$

where $\lambda \in \mathbb{C}$. It is usually written $\mathbf{v} \otimes \mathbf{w}$.

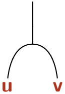
A *multilinear* function (one which is linear in each factor) can be thought of as a function on a tensor product space:

$$\langle \lambda \mathbf{u}, \mathbf{v} \rangle = \lambda \langle \mathbf{u} \times \mathbf{v} \rangle$$
$$(\lambda \mathbf{u}) \times \mathbf{v} = \lambda(\mathbf{u} \times \mathbf{v})$$
$$\det[(\lambda \mathbf{u})\ \mathbf{v}\ \mathbf{w}] = \lambda \det[\mathbf{u}\ \mathbf{v}\ \mathbf{w}].$$

So we could write $\cdot(\mathbf{u} \otimes \mathbf{v})$, $\times(\mathbf{u} \otimes \mathbf{v})$, and $\det(\mathbf{u} \otimes \mathbf{v} \otimes \mathbf{w})$ instead.

3-Vector Diagrams

Suppose the cross product and inner product are represented by

$\mathbf{u} \times \mathbf{v} =$ 

and $\langle \mathbf{u}, \mathbf{v} \rangle =$ 

.

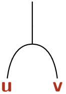
Exercise 1. How can you draw the identity

$$\langle \mathbf{u} \times \mathbf{v}, \mathbf{w} \times \mathbf{t} \rangle = \langle \mathbf{u}, \mathbf{w} \rangle \langle \mathbf{v}, \mathbf{t} \rangle - \langle \mathbf{u}, \mathbf{t} \rangle \langle \mathbf{v}, \mathbf{w} \rangle ?$$

Exercise 2. What does this diagram represent?

3-Vector Diagrams

Suppose the cross product and inner product are represented by

$\mathbf{u} \times \mathbf{v} =$ 

and $\langle \mathbf{u}, \mathbf{v} \rangle =$ 

.

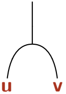
Exercise 1. How can you draw the identity

$$\langle \mathbf{u} \times \mathbf{v}, \mathbf{w} \times \mathbf{t} \rangle = \langle \mathbf{u}, \mathbf{w} \rangle \langle \mathbf{v}, \mathbf{t} \rangle - \langle \mathbf{u}, \mathbf{t} \rangle \langle \mathbf{v}, \mathbf{w} \rangle ?$$

Exercise 2. What does this diagram represent?

3-Vector Diagrams

Suppose the cross product and inner product are represented by

$\mathbf{u} \times \mathbf{v} =$ 

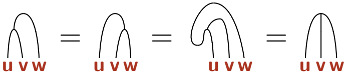
and $\langle \mathbf{u}, \mathbf{v} \rangle =$ 

.

Exercise 1. How can you draw the identity

$$\langle \mathbf{u} \times \mathbf{v}, \mathbf{w} \times \mathbf{t} \rangle = \langle \mathbf{u}, \mathbf{w} \rangle \langle \mathbf{v}, \mathbf{t} \rangle - \langle \mathbf{u}, \mathbf{t} \rangle \langle \mathbf{v}, \mathbf{w} \rangle ?$$

Exercise 2. What does this diagram represent?



Goals of the Talk

Goal

Describe how to create diagrams like these in *any* dimension, and how to translate them into traditional notation.

Diagrams $\longleftrightarrow$ Functions

$$\mathbf{u} \times \mathbf{v}$$

Goal

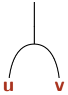
Be able to manipulate the diagrams topologically [as graphs], without worrying about what they represent. Make sure this manipulation doesn't change the underlying function.

Goals of the Talk

Goal

Describe how to create diagrams like these in *any* dimension, and how to translate them into traditional notation.

Diagrams $\longleftrightarrow$ Functions



$\mathbf{u} \times \mathbf{v}$

Goal

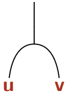
Be able to manipulate the diagrams topologically [as graphs], without worrying about what they represent. Make sure this manipulation doesn't change the underlying function.

Goals of the Talk

Goal

Describe how to create diagrams like these in *any* dimension, and how to translate them into traditional notation.

Diagrams $\longleftrightarrow$ Functions



$\mathbf{u} \times \mathbf{v}$

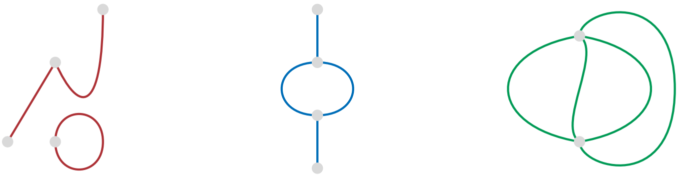
Goal

Be able to manipulate the diagrams topologically [as graphs], without worrying about what they represent. Make sure this manipulation doesn't change the underlying function.

What happens if...

Start with a graph with

- vertices of degree 1 or n;
- edges labelled by matrices,

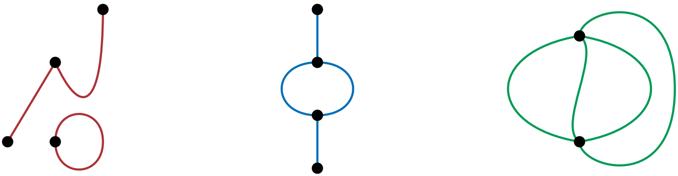


How do these translate to functions??
How do you even specify inputs/outputs??

What happens if...

Start with a graph with

- vertices of degree 1 or n;
- edges labelled by matrices,

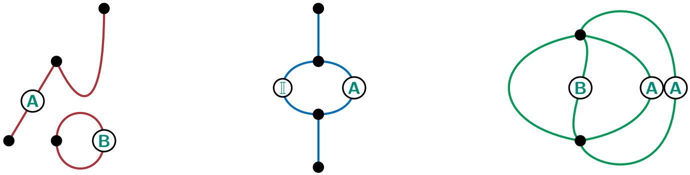


How do these translate to functions??
How do you even specify inputs/outputs??

What happens if...

Start with a graph with

- vertices of degree 1 or n;
- edges labelled by matrices,

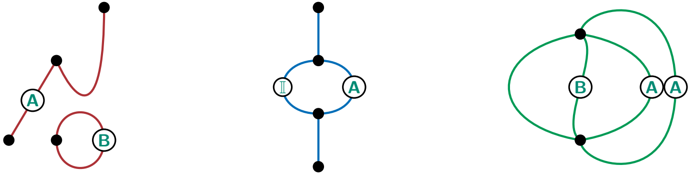


How do these translate to functions??
How do you even specify inputs/outputs??

What happens if...

Start with a graph with

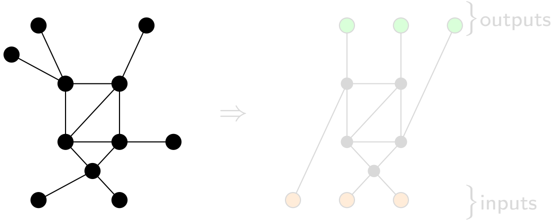
- vertices of degree 1 or n;
- edges labelled by matrices,



How do these translate to functions??
How do you even specify inputs/outputs??

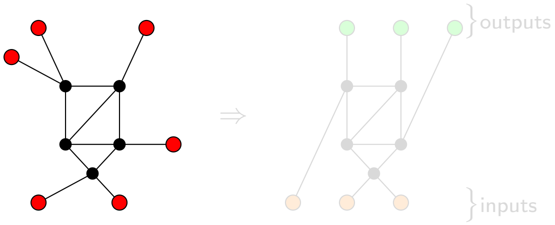
Making it Work: Inputs and Outputs

- Functions have inputs and outputs, whereas diagrams have degree 1 vertices.
- Partition these "leaves" into inputs and outputs.
- By convention, inputs are at the bottom, outputs at the top.
- Function is from $V^{\otimes |I|} \longrightarrow V^{\otimes |O|}$, where $V = \mathbb{C}^n$, $|I|$ is the number of inputs and $|O|$ the number of outputs.
- If there are no inputs, the domain is the scalars $V = \mathbb{C}^0 = \mathbb{C}$.



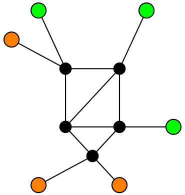
Making it Work: Inputs and Outputs

- Functions have inputs and outputs, whereas diagrams have degree 1 vertices.
- Partition these "leaves" into inputs and outputs.
- By convention, inputs are at the bottom, outputs at the top.
- Function is from $V^{\otimes|I|} \longrightarrow V^{\otimes|O|}$, where $V = \mathbb{C}^n$, $|I|$ is the number of inputs and $|O|$ the number of outputs.
- If there are no inputs, the domain is the scalars $V = \mathbb{C}^0 = \mathbb{C}$.



Making it Work: Inputs and Outputs

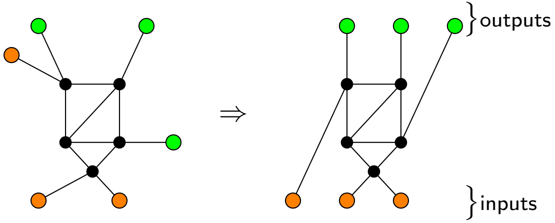
- Functions have inputs and outputs, whereas diagrams have degree 1 vertices.
- Partition these "leaves" into inputs and outputs.
- By convention, inputs are at the bottom, outputs at the top.
- Function is from $V^{\otimes |I|} \longrightarrow V^{\otimes |O|}$, where $V = \mathbb{C}^n$, $|I|$ is the number of inputs and $|O|$ the number of outputs.
- If there are no inputs, the domain is the scalars $V = \mathbb{C}^0 = \mathbb{C}$.



$\Rightarrow$

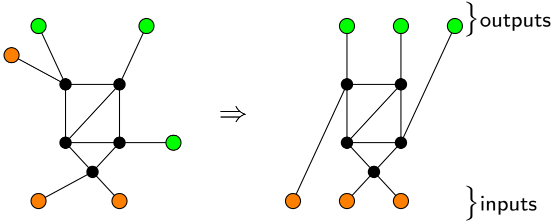
Making it Work: Inputs and Outputs

- Functions have inputs and outputs, whereas diagrams have degree 1 vertices.
- Partition these "leaves" into inputs and outputs.
- By convention, inputs are at the bottom, outputs at the top.
- Function is from $V^{\otimes|I|} \longrightarrow V^{\otimes|O|}$, where $V = \mathbb{C}^n$, $|I|$ is the number of inputs and $|O|$ the number of outputs.
- If there are no inputs, the domain is the scalars $V = \mathbb{C}^0 = \mathbb{C}$.



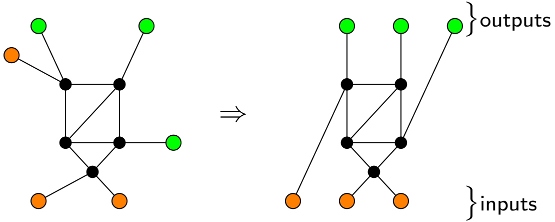
Making it Work: Inputs and Outputs

- Functions have inputs and outputs, whereas diagrams have degree 1 vertices.
- Partition these "leaves" into inputs and outputs.
- By convention, inputs are at the bottom, outputs at the top.
- Function is from $V^{\otimes|I|} \longrightarrow V^{\otimes|O|}$, where $V = \mathbb{C}^n$, $|I|$ is the number of inputs and $|O|$ the number of outputs.
- If there are no inputs, the domain is the scalars $V = \mathbb{C}^0 = \mathbb{C}$.



Making it Work: Inputs and Outputs

- Functions have inputs and outputs, whereas diagrams have degree 1 vertices.
- Partition these "leaves" into inputs and outputs.
- By convention, inputs are at the bottom, outputs at the top.
- Function is from $V^{\otimes|I|} \longrightarrow V^{\otimes|O|}$, where $V = \mathbb{C}^n$, $|I|$ is the number of inputs and $|O|$ the number of outputs.
- If there are no inputs, the domain is the scalars $V = \mathbb{C}^0 = \mathbb{C}$.



The Subtleties

Problem

According to earlier definitions, $= \langle \mathbf{u}, \mathbf{A}\mathbf{v} \rangle = \langle \mathbf{A}^T \mathbf{u}, \mathbf{v} \rangle$ while $= \langle \mathbf{A}\mathbf{u}, \mathbf{v} \rangle$. [They are the same graph.]

- **Solution:** Orient the edges; assume all vertices are *sources* or *sinks*.

Problem

$= \mathbf{v} \otimes \mathbf{w}$ and

$= \mathbf{w} \otimes \mathbf{v} = -\mathbf{v} \otimes \mathbf{w}$ are the same graph.

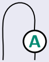
- **Solution:** Draw a "ciliation" on the graph to specify the order:

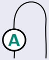
 implies the ordering

.

The Subtleties

Problem

According to earlier definitions, 

$= \langle \mathbf{u}, \mathbf{A}\mathbf{v} \rangle = \langle \mathbf{A}^T\mathbf{u}, \mathbf{v} \rangle$ while 

$= \langle \mathbf{A}\mathbf{u}, \mathbf{v} \rangle$. [They are the same graph.]

- **Solution:** Orient the edges; assume all vertices are *sources* or *sinks*.

Problem

$= \mathbf{v} \otimes \mathbf{w}$ and

$= \mathbf{w} \otimes \mathbf{v} = -\mathbf{v} \otimes \mathbf{w}$ are the same graph.

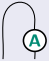
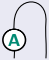
- **Solution:** Draw a "ciliation" on the graph to specify the order:

 implies the ordering

.

The Subtleties

Problem

According to earlier definitions,  $= \langle \mathbf{u}, \mathbf{A}\mathbf{v}\rangle = \langle \mathbf{A}^T\mathbf{u}, \mathbf{v}\rangle$ while  $= \langle \mathbf{A}\mathbf{u}, \mathbf{v}\rangle$. [They are the same graph.]

- **Solution:** Orient the edges; assume all vertices are *sources* or *sinks*.

Problem

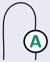
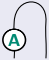
 $= \mathbf{v}\otimes\mathbf{w}$ and $= \mathbf{w}\otimes\mathbf{v} = -\mathbf{v}\otimes\mathbf{w}$ are the same graph.

- **Solution:** Draw a "ciliation" on the graph to specify the order:

 implies the ordering .

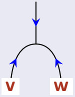
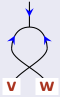
The Subtleties

Problem

According to earlier definitions,  $= \langle \mathbf{u}, \mathbf{A}\mathbf{v} \rangle = \langle \mathbf{A}^T \mathbf{u}, \mathbf{v} \rangle$ while  $= \langle \mathbf{A}\mathbf{u}, \mathbf{v} \rangle$. [They are the same graph.]

- **Solution:** Orient the edges; assume all vertices are *sources* or *sinks*.

Problem

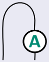
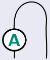
 $= \mathbf{v} \otimes \mathbf{w}$ and  $= \mathbf{w} \otimes \mathbf{v} = -\mathbf{v} \otimes \mathbf{w}$ are the same graph.

- **Solution:** Draw a "ciliation" on the graph to specify the order:

 implies the ordering .

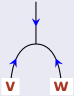
The Subtleties

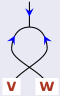
Problem

According to earlier definitions,  $= \langle \mathbf{u}, \mathbf{A}\mathbf{v} \rangle = \langle \mathbf{A}^T \mathbf{u}, \mathbf{v} \rangle$ while  $= \langle \mathbf{A}\mathbf{u}, \mathbf{v} \rangle$. [They are the same graph.]

- **Solution:** Orient the edges; assume all vertices are *sources* or *sinks*.

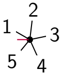
Problem



$= \mathbf{v} \otimes \mathbf{w}$ and 

$= \mathbf{w} \otimes \mathbf{v} = -\mathbf{v} \otimes \mathbf{w}$ are the same graph.

- **Solution:** Draw a "ciliation" on the graph to specify the order:

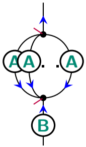
 implies the ordering 

.

The BIG Theorem

Definition

An *n-trace diagram* is an oriented graph with edges labeled by $n \times n$ matrices whose vertices (i) have degree 1 or n only, (ii) are sources or sinks, and (iii) are ciliated.



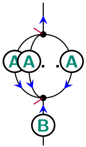
Theorem

The function of a trace diagram is well-defined; in particular, every decomposition into simpler maps gives the same function.

The BIG Theorem

Definition

An *n-trace diagram* is an oriented graph with edges labeled by $n \times n$ matrices whose vertices (i) have degree 1 or n only, (ii) are sources or sinks, and (iii) are ciliated.



Theorem

The function of a trace diagram is well-defined; in particular, every decomposition into simpler maps gives the same function.

Making it Work: Translation

Let $\mathbf{v}$ represent a vector in $\mathbb{C}^n$, and let $\mathbf{A}$ represent an $n \times n$ matrix.

- **Identity Rule**: : $\mathbf{v} \mapsto \mathbf{v}$;
- **Cup Rule**: : $1 \mapsto \hat{\mathbf{e}}^1 \otimes \hat{\mathbf{e}}_1 + \cdots + \hat{\mathbf{e}}^n \otimes \hat{\mathbf{e}}_n$;
- **Cap Rule**: : $\mathbf{v} \otimes \mathbf{w}^T \mapsto \langle \mathbf{v}, \mathbf{w} \rangle$;
- **Vertex Rule**:   : $\mathbf{v_1} \otimes \cdots \otimes \mathbf{v_n} \mapsto \det[\mathbf{v_1} \ \cdots \ \mathbf{v_n}]$;
- **Matrix Rule**:   : $\mathbf{v} \mapsto \mathbf{A}\mathbf{v}$,   : $\mathbf{v}^T \mapsto \mathbf{v}^T\mathbf{A}$.

Making it Work: Translation

Let $\mathbf{v}$ represent a vector in $\mathbb{C}^n$, and let $\mathbf{A}$ represent an $n \times n$ matrix.

- **Identity Rule**: : $\mathbf{v} \mapsto \mathbf{v}$;
- **Cup Rule**: : $1 \mapsto \hat{\mathbf{e}}^1 \otimes \hat{\mathbf{e}}_1 + \cdots + \hat{\mathbf{e}}^n \otimes \hat{\mathbf{e}}_n$;
- **Cap Rule**: : $\mathbf{v} \otimes \mathbf{w}^T \mapsto \langle \mathbf{v}, \mathbf{w} \rangle$;
- **Vertex Rule**:   : $\mathbf{v}_1 \otimes \cdots \otimes \mathbf{v}_n \mapsto \det[\mathbf{v}_1 \ \cdots \ \mathbf{v}_n]$;
- **Matrix Rule**:   : $\mathbf{v} \mapsto \mathbf{A}\mathbf{v}$,   : $\mathbf{v}^T \mapsto \mathbf{v}^T\mathbf{A}$.

Making it Work: Translation

Let $\mathbf{v}$ represent a vector in $\mathbb{C}^n$, and let $\mathbf{A}$ represent an $n \times n$ matrix.

- **Identity Rule**: : $\mathbf{v} \mapsto \mathbf{v}$;
- **Cup Rule**: : $1 \mapsto \hat{\mathbf{e}}^{1} \otimes \hat{\mathbf{e}}_{1} + \cdots + \hat{\mathbf{e}}^{n} \otimes \hat{\mathbf{e}}_{n}$;
- **Cap Rule**: : $\mathbf{v} \otimes \mathbf{w}^T \mapsto \langle \mathbf{v}, \mathbf{w} \rangle$;
- **Vertex Rule**: : $\mathbf{v}_1 \otimes \cdots \otimes \mathbf{v}_n \mapsto \det[\mathbf{v}_1 \ \cdots \ \mathbf{v}_n]$;
- **Matrix Rule**: : $\mathbf{v} \mapsto \mathbf{A}\mathbf{v}$, : $\mathbf{v}^T \mapsto \mathbf{v}^T\mathbf{A}$.

Making it Work: Translation

Let $\mathbf{v}$ represent a vector in $\mathbb{C}^n$, and let $\mathbf{A}$ represent an $n \times n$ matrix.

- **Identity Rule**: : $\mathbf{v} \mapsto \mathbf{v}$;
- **Cup Rule**: : $1 \mapsto \hat{\mathbf{e}}^{\mathbf{1}} \otimes \hat{\mathbf{e}}_{\mathbf{1}} + \cdots + \hat{\mathbf{e}}^{\mathbf{n}} \otimes \hat{\mathbf{e}}_{\mathbf{n}}$;
- **Cap Rule**: : $\mathbf{v} \otimes \mathbf{w}^T \mapsto \langle \mathbf{v}, \mathbf{w} \rangle$;
- **Vertex Rule**: n : $\mathbf{v_1} \otimes \cdots \otimes \mathbf{v_n} \mapsto \det[\mathbf{v_1} \ \cdots \ \mathbf{v_n}]$;
- **Matrix Rule**: A : $\mathbf{v} \mapsto \mathbf{A}\mathbf{v}$, A : $\mathbf{v}^T \mapsto \mathbf{v}^T\mathbf{A}$.

Making it Work: Translation

Let $\mathbf{v}$ represent a vector in $\mathbb{C}^n$, and let $\mathbf{A}$ represent an $n \times n$ matrix.

- **Identity Rule**:  : $\mathbf{v} \mapsto \mathbf{v}$;
- **Cup Rule**:  : $1 \mapsto \hat{\mathbf{e}}^{1} \otimes \hat{\mathbf{e}}_{1} + \cdots + \hat{\mathbf{e}}^{n} \otimes \hat{\mathbf{e}}_{n}$;
- **Cap Rule**:  : $\mathbf{v} \otimes \mathbf{w}^T \mapsto \langle \mathbf{v}, \mathbf{w} \rangle$;
- **Vertex Rule**: : $\mathbf{v_1} \otimes \cdots \otimes \mathbf{v_n} \mapsto \det[\mathbf{v_1} \cdots \mathbf{v_n}]$;
- **Matrix Rule**: : $\mathbf{v} \mapsto \mathbf{A}\mathbf{v}$, : $\mathbf{v}^T \mapsto \mathbf{v}^T\mathbf{A}$.

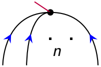
Making it Work: Translation

Let $\mathbf{v}$ represent a vector in $\mathbb{C}^n$, and let $\mathbf{A}$ represent an $n \times n$ matrix.

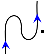
- **Identity Rule**: : $\mathbf{v} \mapsto \mathbf{v}$;
- **Cup Rule**: : $1 \mapsto \mathbf{\hat{e}^1} \otimes \mathbf{\hat{e}_1} + \cdots + \mathbf{\hat{e}^n} \otimes \mathbf{\hat{e}_n}$;
- **Cap Rule**: : $\mathbf{v} \otimes \mathbf{w}^T \mapsto \langle \mathbf{v}, \mathbf{w} \rangle$;
- **Vertex Rule**: $\cdot_n \cdot$: $\mathbf{v_1} \otimes \cdots \otimes \mathbf{v_n} \mapsto \det[\mathbf{v_1} \ \cdots \ \mathbf{v_n}]$;
- **Matrix Rule**: A : $\mathbf{v} \mapsto \mathbf{A}\mathbf{v}$, A : $\mathbf{v}^T \mapsto \mathbf{v}^T\mathbf{A}$.

Making it Work: Translation

Let $\mathbf{v}$ represent a vector in $\mathbb{C}^n$, and let $\mathbf{A}$ represent an $n \times n$ matrix.

- **Identity Rule**:  : $\mathbf{v} \mapsto \mathbf{v}$;
- **Cup Rule**:  : $1 \mapsto \hat{\mathbf{e}}^{\mathbf{1}} \otimes \hat{\mathbf{e}}_{\mathbf{1}} + \cdots + \hat{\mathbf{e}}^{\mathbf{n}} \otimes \hat{\mathbf{e}}_{\mathbf{n}}$;
- **Cap Rule**:  : $\mathbf{v} \otimes \mathbf{w}^T \mapsto \langle \mathbf{v}, \mathbf{w} \rangle$;
- **Vertex Rule**:  : $\mathbf{v_1} \otimes \cdots \otimes \mathbf{v_n} \mapsto \det[\mathbf{v_1} \ \cdots \ \mathbf{v_n}]$;
- **Matrix Rule**:  : $\mathbf{v} \mapsto \mathbf{Av}$,  : $\mathbf{v}^T \mapsto \mathbf{v}^T\mathbf{A}$.

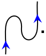
Example: Kinks

Problem. Compute the function corresponding to .

Solution.

- Input and output are both $V = \mathbb{C}^n$;
- Use decomposition to compute:

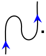
Example: Kinks

Problem. Compute the function corresponding to .

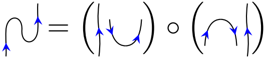
Solution.

- Input and output are both $V = \mathbb{C}^n$;
- Use decomposition to compute:

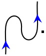
Example: Kinks

Problem. Compute the function corresponding to .

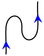
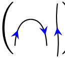
Solution.

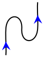
- Input and output are both $V = \mathbb{C}^n$;
- Use decomposition  to compute:

Example: Kinks

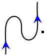
Problem. Compute the function corresponding to .

Solution.

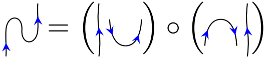
- Input and output are both $V = \mathbb{C}^n$;
- Use decomposition  $=$ $\Big($  $\Big) \circ \Big($  $\Big)$ to compute:

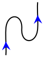
 $: \mathbf{v}$  $\longmapsto \sum_i \mathbf{v} \otimes \hat{\mathbf{e}}^{\mathbf{i}} \otimes \hat{\mathbf{e}}_{\mathbf{i}}$

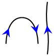
Example: Kinks

Problem. Compute the function corresponding to .

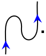
Solution.

- Input and output are both $V = \mathbb{C}^n$;
- Use decomposition  to compute:

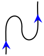
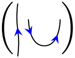
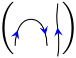
 $: \mathbf{v}$  $\displaystyle\longmapsto \sum_i \mathbf{v} \otimes \hat{\mathbf{e}}^{\mathbf{i}} \otimes \hat{\mathbf{e}}_{\mathbf{i}}$

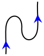
 $\displaystyle\longmapsto \sum_i \langle \mathbf{v}, \hat{\mathbf{e}}_{\mathbf{i}} \rangle \hat{\mathbf{e}}_{\mathbf{i}}$

Example: Kinks

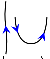
Problem. Compute the function corresponding to .

Solution.

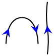
- Input and output are both $V = \mathbb{C}^n$;
- Use decomposition  $=$  $\circ$  to compute:



$$: \mathbf{v} \longmapsto \sum_i \mathbf{v} \otimes \hat{\mathbf{e}}^{\mathbf{i}} \otimes \hat{\mathbf{e}}_{\mathbf{i}}$$

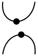


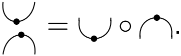
$$\longmapsto \sum_i \langle \mathbf{v}, \hat{\mathbf{e}}_{\mathbf{i}} \rangle \hat{\mathbf{e}}_{\mathbf{i}}$$



$$= \sum_i v_i \hat{\mathbf{e}}_{\mathbf{i}} = v.$$

The Binor Identity

Compute  for 2×2 matrices.

First, decompose it .

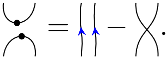
Second, use the fact that  $: 1 \mapsto \hat{\mathbf{e}}_1 \otimes \hat{\mathbf{e}}_2 - \hat{\mathbf{e}}_2 \otimes \hat{\mathbf{e}}_1$ to compute:

$\mathbf{v} \otimes \mathbf{w}$  $\det[\mathbf{v}\ \mathbf{w}] = v_1 w_2 - v_2 w_1$

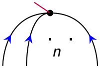
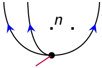
 $(v_1 w_2 - v_2 w_1)(\hat{\mathbf{e}}_1 \otimes \hat{\mathbf{e}}_2 - \hat{\mathbf{e}}_2 \otimes \hat{\mathbf{e}}_1)$

$= \mathbf{v} \otimes \mathbf{w} - \mathbf{w} \otimes \mathbf{v}.$

This proves the *binor identity*:



Caps and Cups

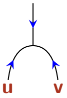
We already know how to compute , but what about ?

Proposition

 $: 1 \mapsto \sum_{\sigma \in \Sigma_n} \mathrm{sgn}(\sigma) \hat{\mathbf{e}}_{\sigma(1)} \otimes \cdots \otimes \hat{\mathbf{e}}_{\sigma(n)}.$

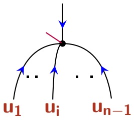
Generalizing the Cross Product

Given that the cross product in three dimensions is

$$\mathbf{u} \times \mathbf{v} =$$ 

,

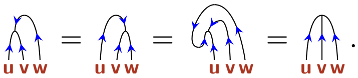
the natural extension to n dimensions is a product of $n-1$ vectors:

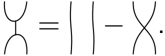
$$\mathbf{u_1} \times \cdots \times \mathbf{u_{n-1}} =$$ 

.

3-Vector Identities

The simplest identity is trivial:

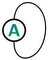


Four-vector identities depend on the relation 

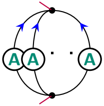
Importance of Trace and Determinant

A *closed* diagram represents a function $\mathbb{C} \to \mathbb{C}$, or a function from a product of matrices to $\mathbb{C}$.

The diagrams for trace and determinant are the simplest closed diagrams:

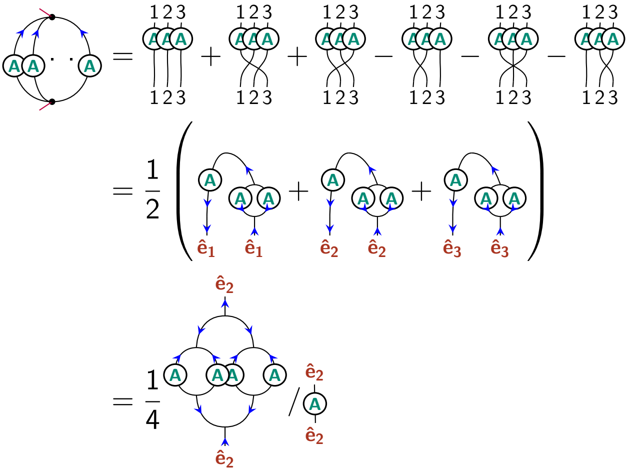
$\mathrm{tr}(\mathbf{A}) =$ 

and

$\det(\mathbf{A}) =$ .

Computing the Determinant

Three techniques for computing the determinant:

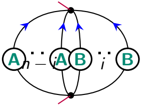


Which is the direct method? cofactor expansion? Dodgson condensation?

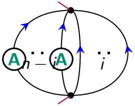
The Characteristic Polynomial

The easy answer to why $\mathrm{tr}(\mathbf{A})$, $\det(\mathbf{A})$ are part of the characteristic polynomial: they are the sum and product of the eigenvalues. Diagrammatically, the n coefficients of the polynomial are the n "simplest" diagrams.

Expand $\det(\mathbf{A}+\mathbf{B})$ in terms of diagrams:

$$\det(\mathbf{A}+\mathbf{B}) = \frac{1}{n!}\sum_{i=0}^{n}\binom{n}{i}$$  .

Applying to the case $\det(\mathbf{A}-\lambda\mathbb{I})$ gives:

$$\det(\mathbf{A}-\lambda\mathbb{I}) = \frac{1}{n!}\sum_{i=0}^{n}\binom{n}{i}(-1)^i\lambda^i$$  .

Duality

From a diagram's point-of-view:

- Inputs and outputs are "artificial";
- Switching orientations corresponds to transposing the whole calculation.

Lineage of Trace Diagrams

Ancestors:

- Euler: graph theory
- Frege: Begriffsschrift c. 1890
- Feynman: Feynman diagrams
- Penrose: spin networks

Early sources:

- Stedman: group theory
- Cvitanovic: 'bird tracks'

Siblings:

- Kauffman: Kauffman bracket
- IHX, STU Relations
- Kuperberg Spiders/Webs

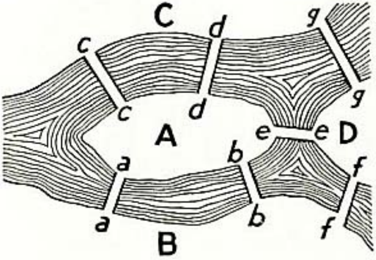


FIGURE 98. *Geographic Map: The Königsberg Bridges.*

Lineage of Trace Diagrams

Ancestors:

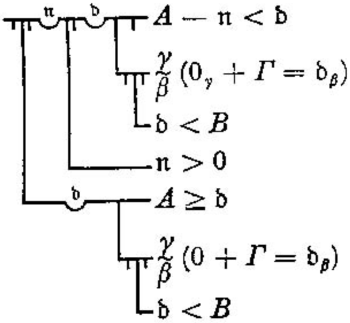
- Euler: graph theory
- Frege: Begriffsschrift c. 1890
- Feynman: Feynman diagrams
- Penrose: spin networks

Early sources:

- Stedman: group theory
- Cvitanovic: ‘bird tracks’

Siblings:

- Kauffman: Kauffman bracket
- IHX, STU Relations
- Kuperberg Spiders/Webs



Lineage of Trace Diagrams

Ancestors:

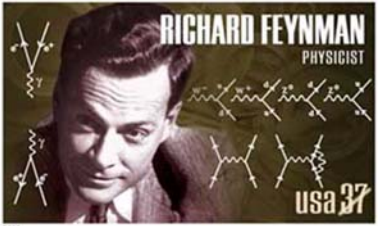
- Euler: graph theory
- Frege: Begriffsschrift c. 1890
- Feynman: Feynman diagrams
- Penrose: spin networks

Early sources:

- Stedman: group theory
- Cvitanovic: 'bird tracks'

Siblings:

- Kauffman: Kauffman bracket
- IHX, STU Relations
- Kuperberg Spiders/Webs



Lineage of Trace Diagrams

Ancestors:

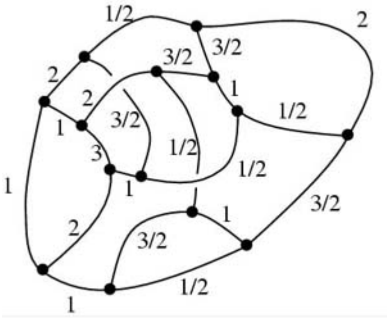
- Euler: graph theory
- Frege: Begriffsschrift c. 1890
- Feynman: Feynman diagrams
- Penrose: spin networks

Early sources:

- Stedman: group theory
- Cvitanovic: 'bird tracks'

Siblings:

- Kauffman: Kauffman bracket
- IHX, STU Relations
- Kuperberg Spiders/Webs



Lineage of Trace Diagrams

Ancestors:

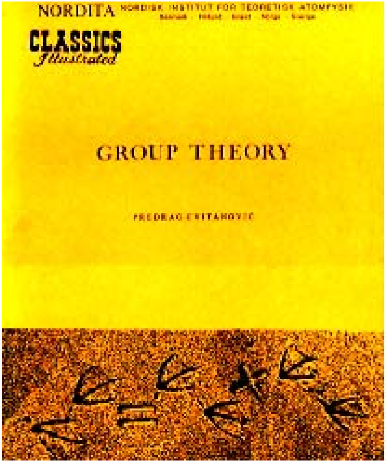
- Euler: graph theory
- Frege: Begriffsschrift c. 1890
- Feynman: Feynman diagrams
- Penrose: spin networks

Early sources:

- Stedman: group theory
- Cvitanovic: 'bird tracks'

Siblings:

- Kauffman: Kauffman bracket
- IHX, STU Relations
- Kuperberg Spiders/Webs



Lineage of Trace Diagrams

Ancestors:

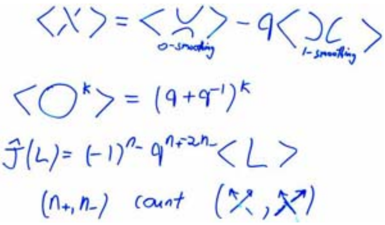
- Euler: graph theory
- Frege: Begriffsschrift c. 1890
- Feynman: Feynman diagrams
- Penrose: spin networks

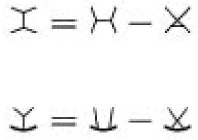
Early sources:

- Stedman: group theory
- Cvitanovic: 'bird tracks'

Siblings:

- Kauffman: Kauffman bracket
- IHX, STU Relations
- Kuperberg Spiders/Webs





Application to Invariant Theory

- A similarity transformation of a matrix is $\mathbf{A} \mapsto \mathbf{B}\mathbf{A}\mathbf{B}^{-1}$ for some nonsingular matrix B.
- Both $\mathrm{tr}(\mathbf{A})$ and $\det(\mathbf{A})$ are invariant under this transformation: $\mathrm{tr}(\mathbf{A}) = \mathrm{tr}(\mathbf{B}\mathbf{A}\mathbf{B}^{-1})$ and $\det(\mathbf{A}) = \det(\mathbf{B}\mathbf{A}\mathbf{B}^{-1})$.
- Diagrams labeled by several matrices are invariant under this transformation *if the same matrix* $\mathbf{B}$ *is used for all transformations*.

Application to Invariant Theory

- A similarity transformation of a matrix is $\mathbf{A} \mapsto \mathbf{BAB}^{-1}$ for some nonsingular matrix B.
- Both $\mathrm{tr}(\mathbf{A})$ and $\det(\mathbf{A})$ are invariant under this transformation: $\mathrm{tr}(\mathbf{A}) = \mathrm{tr}(\mathbf{BAB}^{-1})$ and $\det(\mathbf{A}) = \det(\mathbf{BAB}^{-1})$.
- Diagrams labeled by several matrices are invariant under this transformation *if the same matrix* $\mathbf{B}$ *is used for all transformations*.

Application to Invariant Theory

- A similarity transformation of a matrix is $\mathbf{A} \mapsto \mathbf{BAB}^{-1}$ for some nonsingular matrix B.
- Both $\mathrm{tr}(\mathbf{A})$ and $\det(\mathbf{A})$ are invariant under this transformation: $\mathrm{tr}(\mathbf{A}) = \mathrm{tr}(\mathbf{BAB}^{-1}$ and $\det(\mathbf{A}) = \det(\mathbf{BAB}^{-1})$.
- Diagrams labeled by several matrices are invariant under this transformation *if the same matrix* $\mathbf{B}$ *is used for all transformations*.

Application to Invariant Theory

One aspect of *Invariant Theory* is the classification of functions $M^{\times r} \to \mathbb{C}$ invariant under this simultaneous transformation. Often, the invariants can be linearly reduced to a few simple invariants.

For example, 2×2 matrices satisfy:

$$\mathbf{A}^2 = \mathrm{tr}(\mathbf{A})\mathbf{A} + \det(\mathbf{A})\mathbb{I},$$

and so

$$\mathrm{tr}(\mathbf{A}^2) = \mathrm{tr}(\mathbf{A})^2 + 2\det(\mathbf{A}).$$

For this reason, if there is only one matrix, $\mathrm{tr}(\mathbf{A})$ and $\det(\mathbf{A})$ are the simplest invariants.

Major Open Question: Achieve a complete understanding of all invariants of k $n \times n$ matrices (usually, restricted to either the *nonsingular* matrices or those with determinant 1).

Application to Graph Coloring

Application to Surface Geometry

Application to Representation Theory

References

Trace Diagrams for *any* Lie group:

- Predrag Cvitanovic, *Group Theory*, http://chaosbook.org/GroupTheory/