Kaehler geometry of quiver moduli in application to machine learning

Friday, March 26, 2021   9:05 PM

with George Jeffreys

I. Neural network and quiver representation

New ingredient: non-linear functions at vertices.
We observe many of these functions are closely related with symplectomorphisms.

Thm: $\frac{e^{x_i}}{1+\sum_j e^{x_j}}: \mathbb{R}^n \to \mathbb{R}^n$ that comes from the symplectomorphism $(\mathbb{C}^n, \omega_{\mathbb{P}^n}) \to (\mathbb{C}^n, \omega_{\text{std}})$ satisfies the universal approximation property.

II. An AG formulation of computing machine

New ingredient: $\mathbb{C}$-near-ring.
Want to make well-defined action on $[R(A)/G]$.

Thm: there exists a chain map $DR^{\cdot}(\tilde{A}) \to \left(\Omega^{\cdot}(R(A))\right)^G$.

III. Metrics on framed quiver moduli

Formulate learning algorithm over framed quiver moduli.
Metrics on the universal bundles and moduli space would be important.

Thm:

$$H_i = \left(\sum_{h(\gamma)=i} (w_\gamma e_{t(\gamma)})(w_\gamma e_{t(\gamma)})^*\right)^{-1}$$

gives a well-defined metric on $\mathcal{V}_i \to \mathcal{M}$, whose Ricci curvature induces a Kaehler metric on $\mathcal{M}_{\vec{n},\vec{d}}$.

IV. Uniformization of metrics

Want to relate with usual algorithm over Euclidean space.
Use duality of symmetric spaces to construct non-compact duals of quiver moduli.

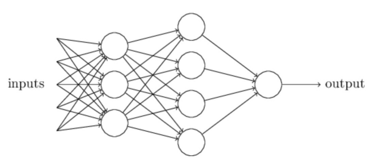
Thm:

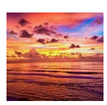
We have the non-compact dual moduli $\mathcal{M}^- \subset \mathcal{M}$ with metric $H_{\mathcal{M}^-} := -\sqrt{-1}\sum_i \partial\bar{\partial} \log\det H_i^-$.
There exists a (non-holomorphic) isometry, which respects the real structure:

$$(M^-, H_{M^-}) \cong \left(\prod_i \mathrm{Gr}^-(m_i, d_i), \bigoplus_i H_{\mathrm{Gr}^-(m_i,d_i)}\right)$$

where $m_i = n_i + \sum_{a:h(a)=i} \dim V_{t(a)}$.

Neural network and quiver representation





$\mapsto 8/10$

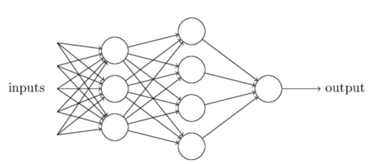
Fix a directed graph Q. Associate to
vertex: vector space
arrow: linear map.

That is, a quiver representation w.

Fix a collection of vertices $i_{\text{in}}, i_{\text{out}}$, and $V_{i_{\text{in}}}, V_{i_{\text{out}}}$.

To approximate any given continuous function $f: K \to V_{i_{\text{out}}}$ (where $K \overset{\text{cpt}}{\subset} V_{i_{\text{in}}}$) by using a representation w.

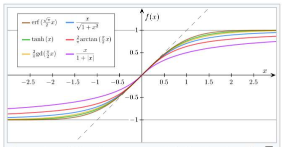
Fix $\gamma \in i_{\text{out}} \cdot \mathbb{C}Q \cdot i_{\text{in}}$.



Get a linear function $f_{\gamma,w}: V_{i_{\text{in}}} \to V_{i_{\text{out}}}$.

Linear approximation $f_{\gamma,w}$ is not good enough!

Introduce non-linear `activation functions' at vertices.



Some sigmoid functions compared. In the drawing all functions are normalized in such a way that their slope at the origin is 1.

Compose with these activation functions and get
network function
$f_{\tilde{\gamma},w}: V_{i_{\text{in}}} \to V_{i_{\text{out}}}$
for every $w \in \operatorname{Rep}(Q)$.

Minimize
$C(V) = \left|f_{\tilde{\gamma},w} - f\right|^2_{L^2(K)}$
by taking a (stochastic) gradient descent on the vector space $\operatorname{Rep}(Q)$.

So a neural network is essentially:
a quiver representation, together with
a fixed choice of non-linear functions on the representing vector spaces, and a fixed path.

Relation between quiver representations and neural network was observed by **[Armenta-Jodoin 20]**.

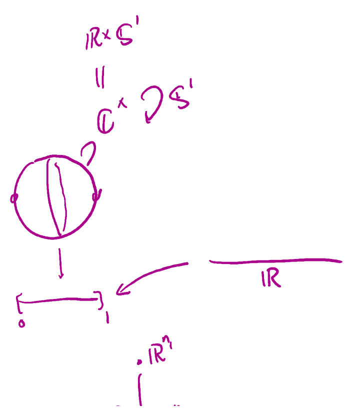
AI neural network has achieved great success in many fields of science and daily life.

Related to lot of areas in math:
Representation theory, stochastic analysis, Riemannian geometry, Morse theory, mathematical physics…

Fundamental motivating questions:
1. Are there any deeper geometric structures in the subject?
2. Can modern geometry provide new insight for the theory and find enhancement of methods?

Main difference between neural network and quiver representations is:
there are non-linear activation functions.

Interesting relation with toric geometry:
The sigmoid function $\frac{1}{1+e^{-x}}$ can be obtained from the moment map $\mathbb{P}^1 \to \mathbb{R}$.

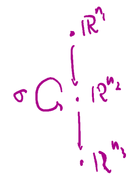


Similarly the algebraic sigmoid $\frac{z}{\sqrt{1+|z|^2}}$ is a symplectomorphism $(\mathbb{C}, \omega_{\mathbb{P}^1}) \to (\mathbb{C}, \omega_{\text{std}})$.

Using tropical rescaling, we show that a higher dimensional

$(\mathbb{C}, \omega_{\mathbb{P}^1}) \to (\mathbb{C}, \omega_{\text{std}})$.

Using tropical rescaling, we show that a higher dimensional analog can also be used as activation functions [**arXiv:2101.11487**].



Another gap between quiver and neural network:

In math, we work with **moduli space of representations**:
$\mathcal{M} := \text{Rep}(Q)//_{\chi}\,\text{Aut}$.

Isomorphic objects should produce the same result.

However, **this is not true for $f_{\widetilde{\gamma},w}$ given as above**:
Any useful non-linear functions $\sigma: V_i \to V_i$ are NOT equivariant under $\text{GL}(V_i)$:
$\sigma(g \cdot v) \neq g \cdot \sigma(v)$.
Then $f_{\widetilde{\gamma},w}$ does not descend to $[w] \in \mathcal{M}$.

A crucial gap between neural network and representation theory!

It poses an obstacle for carrying out machine learning using moduli space of quiver representations.

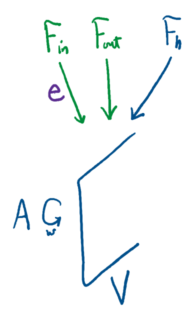
[**arXiv:2101.11487**] provided a simple solution to overcome this.

An AG formulation of computing machine

***A*: associative algebra.**
• consisting of *linear operations* of the machine.

***V*: a vector space (basis-free).**
• States of the machine(before observation).

Consider ***A*-module structures *w***: $A \to \mathfrak{gl}(V)$.
• Linear operations on the state space.



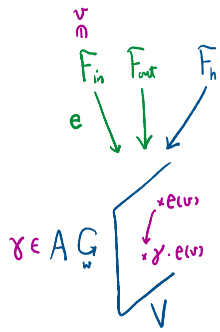
In reality, data are observed and recorded in fixed basis!
Framing *e*:
$F = F_{\text{in}} \oplus F_{\text{out}} \oplus F_h$ (with fixed basis), with linear maps $e: F \to V$.
• $F_{\text{in}} \oplus F_{\text{out}}$: vector spaces of all possible inputs and outputs.
• F_h: Physical memory for the machine.
• e: to observe and record the states.

The triple (V, w, e) is called a framed A-module.

• e: to observe and record the states.

The triple (V, w, e) is called a framed A-module.

Fix $\boldsymbol{\gamma} \in \boldsymbol{A}$.
• An algorithm.

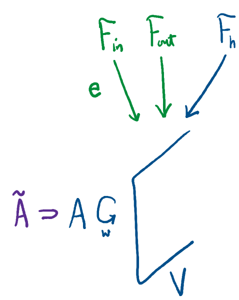


For each $v \in F_{\text{in}}$, have $f^{\gamma}: F_{\text{in}} \to F_{\text{out}}$,
$\boldsymbol{f^{\gamma}(v) \coloneqq e_{\text{out}}^{*}(\gamma \cdot e_{\text{in}}(v))}$.
• The `machine function'.
Given an input signal v, send it to machine by e_{in};
then perform operations according to γ;
then output by the adjoint of e_{out}.)
• **Metric is needed** to define the adjoint.

So far, everything is linear.

In real applications, need
non-linear operations $\sigma_1, \dots, \sigma_N: V \to V$.
Then enlarge A to a near-ring $\tilde{A} = A\{\sigma_1, \dots, \sigma_N\}$;
take $\tilde{\gamma} \in \tilde{A}$.
• A non-linear algorithm.

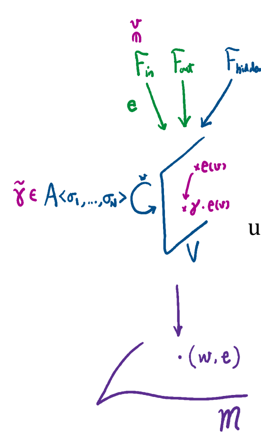
For every $(w, e) \in R$,
have $f_{(w,e)}^{\tilde{\gamma}}: F_{\text{in}} \to F_{\text{out}}$,
$f_{(w,e)}^{\tilde{\gamma}}(v) = e_{\text{out}}^{*}\left(\tilde{\gamma} \cdot_{(w,e)} e_{\text{in}}(v)\right)$.
• Network function.



Can do this over
vector space of framed modules:
$R \coloneqq \{(w, e): w: A \to \mathfrak{gl}(V) \text{ alg. homo.};\ e: F \to V\}$.
• Parameter space of the machine.

Math. and physics principle:
Isomorphic objects should produce the same $f_{(w,e)}^{\tilde{\gamma}}$.

If so, $f_{(w,e)}^{\tilde{\gamma}}$ is defined for $[w, e] \in \mathcal{M} = [R/G]$.



universal bundle

Isomorphism here is $G = \mathrm{GL}(V)$:
$(V, w, e) \cong (V, g \cdot w, g \cdot e)$.

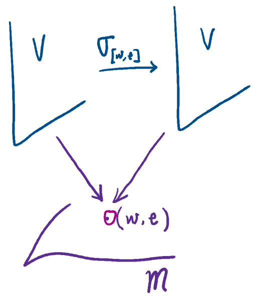
To satisfy this principle, need:
the non-linear operations $\sigma: V \to V$ to be $\mathrm{GL}(V)$-equivariant:
$g \cdot (\sigma(v)) = \sigma(g \cdot v)$.

Impossible for any useful function σ!

A **main obstacle** to realize machine learning using moduli theory.

Solution:

Rather than taking σ as a single linear map $V \to V$,
σ is instead a fiber-bundle map on $\mathcal{V} \to \mathcal{V}$, where $\mathcal{V}$ is the universal bundle over $\mathcal{M}$.

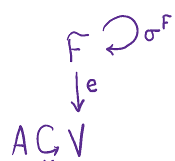


Then the equivariance equation is indeed for $\sigma_{(w,e)}$ over $R \times V$:
$g \cdot \left(\sigma_{(w,e)}(v)\right) = \sigma_{(g\cdot w, g\cdot e)}(g \cdot v)$
instead of
$g \cdot \left(\sigma(v)\right) = \sigma(g \cdot v)$.

(No hope if we look for σ independent of (w, e) since $\mathcal{V}$ is non-trivial.)

To cook up an explicit fiber bundle map,
use equivariant metric:
Equip V with Hermitian metric $h_{(w,e)}$ for every $(w, e) \in R$, in a GL_d-equivariant way:
$h_{(g\cdot w, g\cdot e)}(g \cdot u, g \cdot v) = h_{(w,e)}(u, v)$.
(Note that we are NOT asking for GL_d-invariance $h(g \cdot u, g \cdot v) = h(u, v)$ for a single h, which is IMPOSSIBLE!)

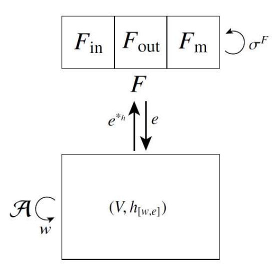
Given ANY $\sigma^F: F \to F$, we cook up $\sigma_{(w,e)}$
using the equivariant metric and framing:
$\sigma_{(w,e)}(v) \coloneqq e \cdot \sigma^F\left(h_{(w,e)}(e_1, v), \ldots, h_{(w,e)}(e_n, v)\right)$.



- Observe and record the state using e, do the non-linear operation, and then send it back as state.
- So the non-linear operation is on F rather than on V!

Let's conclude with the following definition.

Def.
An **activation module** consists of:
(1) a noncommutative algebra A and vector spaces V, F;
(2) a collection of possibly non-linear functions
$\sigma_j^F: F \to F$;
(3) A family of metrics $h_{(w,e)}$ on V over the space R of framed A-modules which is $\mathrm{GL}(V)$-equivariant.



Now we need a near-ring to encapsulate all possible operations.

Definition 1.11. *A near-ring is a set $\tilde{A}$ with two binary operations $+, \circ$ called addition and multiplication such that*

(1) $\tilde{A}$ is a group under addition.
(2) Multiplication is associative.

(3) *Right multiplication is distributive over addition:*

$$(x+y)\circ z = x\circ z + y\circ z$$

for all $x, y, z \in \tilde{A}$.

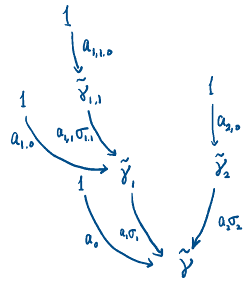
In this paper, the near-ring we use will be required to satisfy that:

(4) $(\tilde{A}, +)$ is a vector space over $\mathbb{F} = \mathbb{C}$, with $c\cdot(x\circ y) = (c\cdot x)\circ y$ for all $c \in \mathbb{C}$ and $x, y \in \tilde{A}$.

(5) There exists $1 \in \tilde{A}$ such that $1\circ x = x = x\circ 1$.

Given an algebra A, we can form a near ring $A\{\sigma_1, \dots, \sigma_N\}$.

ex. $\tilde{\gamma} = a_0 + a_1\sigma_1 \circ \left(a_{1,0} + a_{1,1}\sigma_{1,1} \circ a_{1,1,0}\right) + a_2\sigma_2 \circ a_{2,0}$.



Assume the above setting of activation module. Set

$$\tilde{A} := \left(\mathrm{Mat}_F\left(\hat{A}^{\mathrm{double}}\right)\right)\{\sigma_1, \dots, \sigma_N\}$$

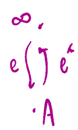
where

$\hat{A}^{\mathrm{double}}$ is the doubling of $\mathbb{C}\hat{Q}$; (so has e^*, a^*)

$\mathrm{Mat}_F\left(\hat{A}^{\mathrm{double}}\right)$ is algebra of n-by-n matrices, whose entries are cycles in $\mathbb{C}\hat{Q}$ based at the framing vertex ∞.

Doubling is a standard procedure in construction of Nakajima's quiver variety.

We consider $\tilde{A}$-module struture on F.



Prop.
Each point in $[R/G]$ gives a well-defined map
$\tilde{A} \to \mathrm{Map}(F)$.
That is, we have
$\tilde{A} \to \Gamma(\mathcal{M}, \mathrm{Map}(F))$.

Note: $[R/G]$ is moduli of A-modules, NOT the doubling.
The actions of e^*, a^* on $F \oplus V$ are produced by the adjoint with respect to
h (the equivariant family of metrics on V).

Have **differential forms for nc algebra** $\boldsymbol{A}$
[**Connes; Cuntz-Quillen; Kontsevich**; **Ginzburg…**].
$\mathrm{DR}^*(A) \to \Omega^*\left(R(A)\right)^G$.

Study moduli spaces for all dimension vectors at the same time!

The noncommutative differential forms can be described as follows. Consider the quotient vector space $\overline{A} = A/\mathbb{K}$ (which is no longer an algebra). We think of elements in $\overline{A}$ as differentials. Define

$$D(A) := \bigoplus_{n \in \mathbb{Z}_{\geq 0}} D(A)_n, \ D(A)_n := A \otimes \overline{A} \otimes \ldots \otimes \overline{A}$$

where n copies of $\overline{A}$ appear in $D(A)_n$, and the tensor product is over the ground field $\mathbb{K}$. We should think of elements in $\overline{A}$ as *matrix-valued* differential one-forms. Note that $X \wedge X$ may not be zero, and $X \wedge Y \neq -Y \wedge X$ in general for matrix-valued differential forms X, Y.

The differential $d_n : D(A)_n \to D(A)_{n+1}$ is defined as

$$d_n(a_0 \otimes \overline{a_1} \otimes \ldots \otimes \overline{a_n}) := 1 \otimes \overline{a_0} \otimes \ldots \otimes \overline{a_n}.$$

The product $D(A)_n \otimes D(A)_{m-1-n} \to D(A)_{m-1}$ is more tricky:

$$(a_0 \otimes \overline{a_1} \otimes \ldots \otimes \overline{a_n}) \cdot (a_{n+1} \otimes \overline{a_{n+2}} \otimes \ldots \otimes \overline{a_m})$$

(9) $$:= (-1)^n a_0 a_1 \otimes \overline{a_2} \otimes \ldots \otimes \overline{a_m} + \sum_{i=1}^{n} (-1)^{n-i} a_0 \otimes \overline{a_1} \otimes \ldots \otimes \overline{a_i a_{i+1}} \otimes \ldots \otimes \overline{a_m}$$

which can be understood by applying the Leibniz rule on the terms $\overline{a_i a_{i+1}}$. Note that we have chosen representatives $a_i \in A$ for $i = 1, \ldots, n+1$ on the RHS, but the sum is independent of choice of representatives (while the product $\overline{a_i a_{i+1}}$ itself depends on representatives).

$d^2 = 0.$

The Karoubi-de Rham complex is defined as

(10) $$DR^\bullet(A) := \Omega^\bullet(A)/[\Omega^\bullet(A), \Omega^\bullet(A)]$$

where $[a, b] := ab - (-1)^{ij} ba$ is the graded commutator for a graded algebra. d descends to be a well-defined differential on $DR^\bullet(A)$. Note that $DR^\bullet(A)$ is not an algebra since $[\Omega^\bullet(A), \Omega^\bullet(A)]$ is not an ideal. $DR^\bullet(A)$ is the non-commutative analog for the space of de Rham forms. Moreover, there is a natural map by taking trace to the space of G-invariant differential forms on the space of representations $R(A)$:

(11) $$DR^\bullet(A) \to \Omega^\bullet(R(A))^G.$$

We extend such notions to the near-ring $\tilde{A}$.

Theorem 1.40. *There exists a degree-preserving map*

$$DR^\bullet(\widetilde{\mathcal{A}}) \to (\Omega^\bullet(R, \mathbf{Map}\,(F, F)))^G$$

which commutes with d on the two sides, and equals to the map (14)*:* $DR^\bullet(\mathrm{Mat}_F(\hat{\mathcal{A}})) \to (\Omega^\bullet(R, \mathrm{End}\,(F)))^G$ *when restricted to* $DR^\bullet(\mathrm{Mat}_F(\hat{\mathcal{A}}))$*. Here,* $\mathbf{Map}\,(F, F)$ *denotes the trivial bundle* $\mathrm{Map}\,(F, F) \times R$*, and the action of* $G = \mathrm{GL}(V)$ *on fiber direction is trivial.*

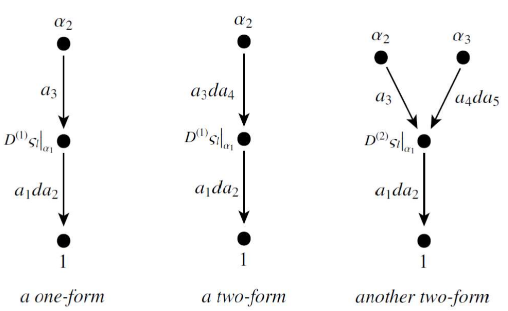


FIGURE 3.

(The number of leaves is required to be $\leq$ form degree.)
Also have $d^2 = 0$.

In particular, the function

$$\int_K \left| f^{\tilde{\gamma}}_{(w,e)}(v) - f(v) \right|^2 dv$$

and its differential are induced from 0-form and 1-form on $\tilde{A}$.
Central object in machine learning.

Thus the learning is governed by geometric objects on $\tilde{A}$!

Metric for framed quiver moduli

Fix Q. $A = \mathbb{C}Q$.

Framed representation:
Vertex: V_i
Arrow: w_a
together with $e_i \colon \mathbb{C}^{n_i} \to V_i$ (called framing).

$$R = \mathrm{Rep}_{\vec{n},\vec{d}} \coloneqq \mathrm{Rep}_{\vec{d}} \times \bigoplus_{i \in Q_0} \mathrm{Hom}_k(\mathbb{C}^{n_i}, V_i)\,.$$

$\vec{d}$ is dim. of rep.
$\vec{n}$ is dim. of framing.

$\mathcal{M} \coloneqq \mathrm{Rep}_{\vec{n},\vec{d}} /\!/_{\chi} \mathrm{GL}_{\vec{d}}$.

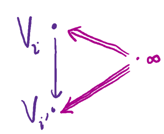
Smooth moduli space of framed quiver representations.
[Kings; Nakajima; Crawley-Boevey; Reineke]

Stability condition:
no proper subrepresentation of V contains Im e.

$\mathcal{M}_{\vec{n},\vec{d}} \coloneqq \{\text{stable framed rep. } (V, e)\}/GL_{\vec{d}}$.

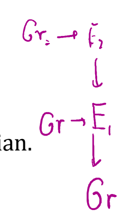
Typical example:
$Gr(n, d)$.

Remark: $\mathcal{M}_{\vec{n},\vec{d}}$ is the usual GIT quotient for a bigger quiver $\hat{Q}$
which has one more vertex ∞ than Q,
together with n_i arrows from ∞ to i.



Put dim=1 over the vertex ∞. Then take the character
$\Theta = -\infty^*$ for slope stability $\Theta(\vec{\alpha})/\Sigma\vec{\alpha}$.

Topology of $\mathcal{M}_{\vec{n},\vec{d}}$ is well-known.



Thm. [Reineke]
Suppose Q has no oriented cycle. Then $\mathcal{M}_{\vec{n},\vec{d}}$ is an iterated
Grassmannian bundle, and it embeds to quiver Grassmannian.

$\mathcal{V}_i$: universal bundle over vertex i.

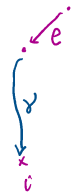
Thm.
Fix $i \in Q_0$.
$H_i \colon \mathrm{Rep}_{\vec{n},\vec{d}} \to \mathrm{End}(\mathbb{C}^{d_i})$,

e

Fix $i \in Q_0$.

$$H_i\colon \mathrm{Rep}_{\vec{n},\vec{d}} \to \mathrm{End}\left(\mathbb{C}^{d_i}\right),$$

$$(w,e) \mapsto \left(\sum_{h(\gamma)=i} \left(\boxed{w_\gamma e_{t(\gamma)}}\right)\left(w_\gamma e_{t(\gamma)}\right)^*\right)^{-1}$$



gives a well-defined metric on $\mathcal{V}_i \to \mathcal{M}$.

Moreover, the Ricci curvature $i \sum_i \partial\bar{\partial} \log\det H_i$
of the resulting metric on $\otimes_{i\in Q_0} U_i$ defines a Kaehler metric on
$\mathcal{M}_{\vec{n},\vec{d}}$.

Then we can run learning algorithm over
$\mathcal{M}_{\vec{n},\vec{d}}$.

$$\tilde{A} \to \Gamma\left(\prod_k \mathcal{M}^{(k)}, \mathrm{Map}(F)\right),$$

$$\tilde{\gamma} \mapsto f^{\tilde{\gamma}}_{(w,e)}(v) = H_i\left(e_{\mathrm{out}}, \tilde{\gamma} \circ_{(w,e)} e_{\mathrm{in}} \cdot v\right)$$

Question:
How to relate this moduli formulation
back to the original setup over
Euclidean space of representations?

In application, take $\vec{n} \geq \vec{d}$.
Write the framing as $e^{(i)} = \left(\epsilon^{(i)} \;\; b^{(i)}\right)$.

By using the quiver automorphism, $\epsilon^{(i)}$ can be made as Id.
whenever $\epsilon^{(i)}$ is invertible.

This gives a chart:
$\mathrm{Rep}_{\vec{n}-\vec{d},\vec{d}} \hookrightarrow \mathcal{M}_{\vec{n},\vec{d}}$.

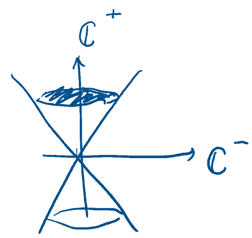
Restricting the above $H_i\left(e_{\mathrm{out}}, \tilde{\gamma} \circ_{[w,e]} e_{\mathrm{in}} \cdot v\right)$ to this chart,
pretending the metrics are all trivial,
it recovers the usual Euclidean setup!

Question:
How to give the vector space $\mathrm{Rep}_{\vec{n}-\vec{d},\vec{d}}$ have a more intrinsic
interpretation?

Yes, by considering uniformization.

Uniformization

For $\mathrm{Gr}(n,d) = U(n)/U(d)U(n-d)$,
has Hermitian symmetric dual
$\mathrm{Gr}^-(n,d) = U(d,n-d)/U(d)U(n-d)$
$= \{\text{Spacelike subspace in } \mathbb{R}^{d,n-d}\} \overset{\textbf{Borel}}{\subset} \mathrm{Gr}(n,d)$.



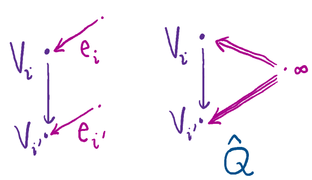
Ex. Hyperbolic disc $D \subset \mathbb{CP}^1$.
Hyperbolic <--> spherical.

Such symmetric dual and embedding was studied uniformly for general symmetric spaces by [**Chen-Huang-Leung**].

By [**Reineke**], framed quiver moduli $\mathcal{M}_{n,d}$ is an iterated Grassmannian bundle.

What is its `non-compact dual'?

$\hat{Q}$: the quiver with one more vertex denoted as ∞.



Assume $\vec{n} > \vec{d}$. Write $e^{(i)} = (\epsilon^{(i)}\ b^{(i)})$.

For each i, define

$$H_i^- = \left(\sum_{h(\gamma)=i} (-1)^{s(\gamma)} \gamma\gamma^* \right)^{-1} = \left(\rho_i \begin{pmatrix} I_{d_i} & 0 \\ 0 & -I_{N_i - d_i} \end{pmatrix} \rho_i^* \right)^{-1}$$

where γ is a path in $\hat{Q}$ with $t(\gamma) = \infty$;
$s(\gamma) = 1$ for $\gamma = \epsilon_j^{(i)}$, and -1 for all other γ.

$R^- := \{(w,e) \in R_{n,d} \colon H_i^- \text{ is positive definite for all } i\}$.

Lemma.
$\emptyset \neq R^- \subset \{(w,e) \colon \epsilon^{(i)} \text{ is invertible } \forall i\} \subset R^s$.

Lemma.
R^- is G_d-invariant.

$\mathcal{M}^- := R^-/G_d$.
The moduli of space-like framed representations.

Theorem 1.
• H_i^- defines Hermitian metric on the universal bundle

$\mathcal{V}_i \to \mathcal{M}^-$.

- $H_{M^-} := -i \sum \partial\bar{\partial} \log \det H_i^-$ defines a Kaehler metric on M^-.

- There exists a (non-holomorphic) isometry, which respects the real structure:
  $(M^-, H_{M^-}) \cong \left(\prod_i \mathrm{Gr}^-(m_i, d_i), \bigoplus_i H_{\mathrm{Gr}^-(m_i,d_i)} \right)$
  where $m_i = n_i + \sum_{a:h(a)=i} \dim V_{t(a)}$.
- There is a canonical identification of $\mathcal{V}_i \to \mathcal{M}^-$ with $\mathcal{V}_{\mathrm{Gr}^-(m_i,d_i)} \to \prod_i \mathrm{Gr}^-(m_i, d_i)$ covering the isometry.

Remark:
$\mathrm{Gr}^-(m,d) = \{b \in \mathrm{Mat}_{d\times(m-d)}: bb^* < I_d\}$ has non-positive curvature (invariant under parallel transport).

In the same manner like before, have network function
$f_{(w,e)}^{\tilde{\gamma}}(v) = H_i\left(e_{out}, \tilde{\gamma} \circ_{[w,e]} e_{in} \cdot v\right)$
over (M^-, H_T).

Remark:
Machine learning using hyperbolic geometry has recently attracted a lot of research in learning graphs and word embeddings.
Most has focused on taking hyperbolic metric in the fiber direction.

Homogeneous spaces have also been introduced in the fiber direction [**Cohen; Geiger; Weiler**], to make use of symmetry of input data.

Here, we introduce ML over the **moduli space** and its non-compact dual, which universally exists for all neural network models.

A parallel Euclidean story:
Take
$H_i^0 = \left(\rho_i \begin{pmatrix} I_{d_i} & 0 \\ 0 & 0 \end{pmatrix} \rho_i^*\right)^{-1}$.
That is, we assign positive sign to $\epsilon_j^{(i)}$ and 0 (instead of -1) to all other paths of $\hat{Q}$.

$R^0 := \{(w,e) \in R_{n,d}:\ H_i^0 \text{ is positive definite for all } i\}$.

Prop.
$R^0 //_\chi G_d = \mathrm{Rep}_{n-d,d}$
a vector space.
Also H_i defines trivial metric on $V_i|_{M^0}$.

That is,
$\mathbf{Rep}_{n-d,d} \subset \mathcal{M}_{n,d}$ is the moduli of framed positive-def. representations with respect to H_i^0.

This recovers the usual Euclidean machine learning.

Conclusion:
$\mathcal{M}, \mathcal{M}^-, \mathcal{M}^0$ (spherical, hyperbolic, Euclidean) are the moduli of framed positive-definite representations with respect to
$H_i = (\rho_i \rho_i^*)^{-1}$,
$H_i^- = \left(\rho_i \begin{pmatrix} I_{d_i} & 0 \\ 0 & -I \end{pmatrix} \rho_i^*\right)^{-1}$,
$H_i^0 = H_i^0 = \left(\rho_i \begin{pmatrix} I_{d_i} & 0 \\ 0 & 0 \end{pmatrix} \rho_i^*\right)^{-1}$ respectively.

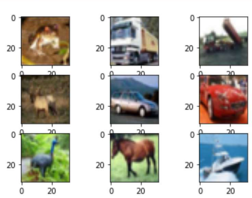
Can connect them in a family:
$\left(\rho_i \begin{pmatrix} I_{d_i} & 0 \\ 0 & tI \end{pmatrix} \rho_i^*\right)^{-1}$.

Experiments

Let's experiment with metrics on the moduli space of representations.

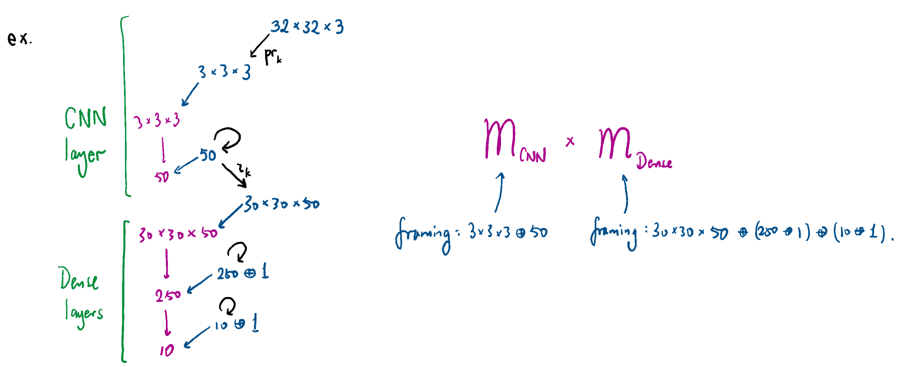
```
from keras.datasets import cifar10
```

```
Using TensorFlow backend.
```



To train machine to classify these pictures into 10 classes.
Want to compare the results of using trivial and non-trivial metrics in the moduli space of framed quiver representations.

ex.



Metric on universal bundles:
$$H_i = (\rho_i \mathcal{I} \rho_i^*)^{-1} = \left(I_{d_i} - \frac{\widetilde{W_i}\widetilde{W_i}^*}{M}\right)^{-1}.$$
Metrics on moduli spaces:
$$h_{\mathcal{M}} = -M\left(\sum_i tr\big((\rho_i \mathcal{I} \rho_i^*)^{-1}(\partial\rho_i)\mathcal{I}(\partial\rho_i)^*\big) - \sum_i tr\big((\rho_i \mathcal{I} \rho_i^*)^{-1}\rho_i \mathcal{I}(\partial\rho_i)^*(\rho_i \mathcal{I} \rho_i^*)^{-1}(\partial\rho_i)\mathcal{I}\rho_i^*\big)\right).$$
($M = \infty$ <-> Euclidean; $M > 0$ <-> the non-compact dual $\mathcal{M}^-$; $M < 0$ <-> $\mathcal{M}$.)

Abelianize to simplify the computation:
Take $(\mathbb{C}^\times)^d$ in place of $GL(d)$ in $\mathcal{M} = R//GL(d)$.
This means taking rep. (of a bigger quiver) with dimension vector $(1, \dots, 1)$.
Then metrics on universal bundles are recorded as 1×1 matrices.

The actual model in the experiment:

```
inputs = keras.Input(shape=input_shape)
y = hypConv2D(50, kernel_size=(3, 3),padding='same')(inputs)
y = layers.MaxPooling2D(pool_size=(2, 2))(y)
y = hypConv2D(75, kernel_size=(3, 3),padding='same')(y)
y = layers.MaxPooling2D(pool_size=(2, 2))(y)
y = Dropout(0.25)(y)
y = hypConv2D(125, kernel_size=(3, 3),padding='same')(y)
y = layers.MaxPooling2D(pool_size=(2, 2))(y)
y = Dropout(0.25)(y)
y = layers.Flatten()(y)
y = hypMDenseb(500)(y)
y = Activation(activations.relu)(y)
y = Dropout(0.4)(y)
y = hypMDenseb(250)(y)
y = Activation(activations.relu)(y)
y = Dropout(0.3)(y)
y = hypMDenseb(n_classes)(y)
outputs = layers.Softmax()(y)
model = hypModel(inputs=inputs, outputs=outputs)
model.compile(optimizer="adam", loss="categorical_crossentropy", metrics=["accuracy"])
history = model.fit(x_train, y_train, batch_size=128, epochs=50, validation_split=0.1)
```

```
inputs = keras.Input(shape=input_shape)
y = EuclidConv2D(50, kernel_size=(3, 3),padding='same')(inputs)
y = layers.MaxPooling2D(pool_size=(2, 2))(y)
y = EuclidConv2D(75, kernel_size=(3, 3),padding='same')(y)
y = layers.MaxPooling2D(pool_size=(2, 2))(y)
y = Dropout(0.25)(y)
y = EuclidConv2D(125, kernel_size=(3, 3),padding='same')(y)
y = layers.MaxPooling2D(pool_size=(2, 2))(y)
y = Dropout(0.25)(y)
y = layers.Flatten()(y)
y = Denseb(500)(y)
y = Activation(activations.relu)(y)
y = Dropout(0.4)(y)
y = Denseb(250)(y)
y = Activation(activations.relu)(y)
y = Dropout(0.3)(y)
y = Denseb(num_classes)(y)
outputs = layers.Softmax()(y)
model = EuclidModel(inputs=inputs, outputs=outputs)
model.compile(optimizer="adam", loss="categorical_crossentropy", metrics=["accuracy"])
history = model.fit(x_train, y_train, batch_size=128, epochs=50, validation_split=0.1)
```

```
def call(self, x):
    Hinv = 1 - tf.math.reduce_sum(tf.math.square(self.kernel),[0,1,2]) / self.M
    y = K.conv2d(x, self.kernel,padding=self.padding)
    return keras.activations.relu(y/Hinv)
```

```
#hyperbolic gradient for 1st conv2d layer
#g_i = H_i (Id - H_i wtilde_i wtilde_i^*)
#g_i^(-1) wtilde_i = partial_i /H_i - (partial_i dot wtilde_i) wtilde_i/(M+|wtilde_i|
H1inv = 1 - tf.math.reduce_sum(tf.math.square(trainable_vars[0]),[0,1,2]) / M1
grads[0] = grads[0] * H1inv \
- tf.multiply(tf.reduce_sum(tf.multiply(trainable_vars[0],grads[0]),[0,1,2]),\
              trainable_vars[0]) \
/ (M1+tf.divide(tf.reduce_sum(tf.square(trainable_vars[0]),[0,1,2]),H1inv))
```

$$H_i = (\rho_i \mathcal{J} \rho_i^*)^{-1} = \left(I_{d_i} - \frac{\widetilde{w_i}\widetilde{w_i}^*}{M}\right)^{-1}$$
$$= \left(1 - \frac{|\widetilde{w_i}|^2}{M}\right)^{-1} \text{ if } d_i = 1.$$

$$h_{\mathcal{M}} = -M \cdot \left(\sum_i tr\big((\rho_i \mathcal{J} \rho_i^*)^{-1}(\partial\rho_i)\mathcal{J}(\partial\rho_i)^*\big) - \sum_i tr\big((\rho_i \mathcal{J} \rho_i^*)^{-1}\rho_i \mathcal{J}(\partial\right.$$

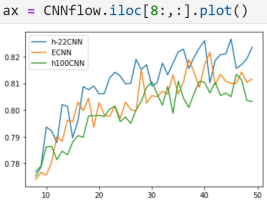
$$h^{\mathcal{M}}_{\widetilde{w}^{0,1}_{kj}\widetilde{w}^{1,0}_{qp}} = \bigoplus_i H^{(i)}_{kj}\left(\delta_{jp} + \frac{1}{M}\cdot \widetilde{w}_p^* \cdot H^{(i)} \cdot \widetilde{w}_j\right).$$

After Abelianize:

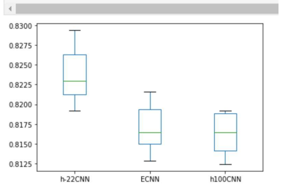
$$h^{\mathcal{M}}_l = H_l\left(I + \frac{1}{M} H_l \widetilde{w}_l^* \widetilde{w}_l\right).$$

$$(\text{grad } f)_l = \frac{1}{H_l}\partial_{\widetilde{w}_l} f - \frac{\left(\partial_{\widetilde{w}_l} f \cdot \widetilde{w}_l^*\right)\widetilde{w}_l}{M + |\widetilde{w}_l|^2 H_l}.$$

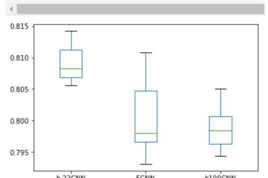
```
ax = CNNflow.iloc[8:,:].plot()
```




```
ax = CNNValAcc.plot.box()
```



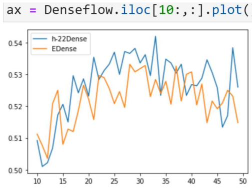

```
ax = CNNtestAcc.plot.box()
```



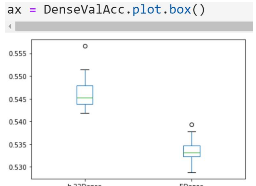
Another test:
Use only dense layers for the same dataset.
Compare trivial and non-trivial metrics.

```
initM = float(-30)
inputs = keras.Input(shape=input_shape)
y = layers.Flatten()(inputs)
y = hypMDenseb(500)(y)
y = Activation(activations.relu)(y)
y = hypMDenseb(250)(y)
y = Activation(activations.relu)(y)
y = hypMDenseb(n_classes)(y)
outputs = layers.Softmax()(y)
model = hypModel(inputs=inputs, outputs=outputs)
model.compile(optimizer="adam", loss="categorical_crossentropy", metrics=["accuracy"])
history = model.fit(x_train, y_train, batch_size=128, epochs=50, validation_split=0.1)
```

```
ax = Denseflow.iloc[10:,:].plot()
```




```
ax = DenseValAcc.plot.box()
```



Conclusion:
in this case, $M < 0$ (curvature ≥ 0) behaves around 1% better than $M = 0$ and $M > 0$.

- The method of running ML over moduli spaces and their non-compact duals is UNIVERSAL and works in practice
- Geometric structures on near-ring $\tilde{A}$ is a new subject and govern machine learning over the moduli
- To lay the algebraic foundation of computing machine, and find new applications of geometry.