Pinning-Sympy
Development roadmap

Joshua Ruiter

October 27, 2024

Contents

1 To do list

1.1 High priority

- In order to calculate commutator coefficients, Weyl group elements, and Weyl group coefficients, perhaps we should write something akin to calculate_root_spaces which can do this programmatically and extract the coefficients.

- Complete group builder functions for special unitary groups.

  - Write various internal methods for build_special_unitary_group
    ~~is_lie_algebra_element_SU~~
    ~~is_group_element_SU~~
    ~~is_torus_element_SU~~
    ~~root_space_dimension_SU~~
    root_space_map_SU
    root_subgroup_map_SU
    ~~torus_element_map_SU~~
  - Work out commutator coefficients for special unitary groups,
    and write commutator_coefficient_map_SU
  - Work out Weyl group elements for special unitary groups,
    and write weyl_group_element_map_SU
  - Work out Weyl group conjugation coefficients for special unitary groups,
    and write weyl_group_coefficient_map_SU

1.2 Medium priority

- nondegenerate_isotropic_form

  - Implement an "evaluate this form on two vectors" method
  - Implement an "evaluate the anisotropic part of this form on two vectors" method

- Complete group builder functions for special orthogonal groups.

  - ~~Write various internal methods for build_special_orthogonal_group, from is_lie_algebra_element_SO down to torus_element_map~~
  - Work out commutator coefficients for special orthogonal groups,
    and write commutator_coefficient_map_SO
  - Work out Weyl group elements for special orthogonal groups,
    and write weyl_group_element_map_SO
  - Work out Weyl group conjugation coefficients for special orthogonal groups,
    and write weyl_group_coefficient_map_SO

1.3 Low priority

- Implement constructor/builders for types D, E, F, and G root systems.

1.4 Completed

- Implement a custom root system class (still lacks type D, E, F, G).
- Implement something to automatically calculate and output roots and root spaces for special linear, special orthogonal, and special unitary groups.
- Create a custom class to represent pinned groups and run tests to verify pinning properties.
- Create a group builder method for special linaer groups.

2 Files

Color coding: Black = mostly done, blue = some work to be done, red = far from done.

Name: build_group

Purpose: Methods for quickly constructing pinned_group objects of special linear, special orthogonal, and special unitary groups.

Technical notes: The special linear group is just $\mathrm{SL}_n(K) = \{X \in \mathrm{GL}_n(K) : \det(X) = 1\}$. The special orthogonal groups of interest are as follows. Fix a field K, and a vector space V over K, and a symmetric nondegenerate isotropic bilinear form b on V. Without loss of generality, we can assume that the matrix of b has the form

$$B = \begin{pmatrix} 0 & I_q & 0 \\ I_q & 0 & 0 \\ 0 & 0 & C \end{pmatrix}$$

The special orthogonal group of b is $\{\tau \in SL(V) : b(\tau v, \tau w) = b(v, w) \text{ for all } v, w\}$. More concretely, the matrix version of this is the group

$$\mathrm{SO}_{n,q}(K, B) = \left\{X \in \mathrm{SL}_n(K) : X^T B X = B\right\}$$

Technically, $\mathrm{SO}_{n,q}(-, B)$ is a functor from K-algebras to groups, which sends a K-algebra R to the group

$$\mathrm{SO}_{n,q}(R, B) = \left\{X \in \mathrm{SL}_n(R) : X^T B X = B\right\}$$

But for the most part, we can just work with the group of K-points.
The special unitary groups of interest are as follows. Fix a quadratic field extension L/K, and a vector space V over L, and a nondegenerate hermitian or skew-hermitian isotropic form h on V. Without loss of generality, we can assume that the matrix of h has the form

$$H = \begin{pmatrix} 0 & I_q & 0 \\ \varepsilon I_q & 0 & 0 \\ 0 & 0 & C \end{pmatrix}$$

The special unitary group of h is $\{\tau \in SL(L) : h(\tau v, \tau w) = h(v, w) \text{ for all } v, w\}$. More concretely, the matrix version of this is the group

$$\mathrm{SU}_{n,q}(L, H) = \{X \in \mathrm{SL}_n(L) : X^* H X = H\}$$

where X^* is the conjugate transpose. As with the orthogonal groups, technically $\mathrm{SU}_{n,q}(-, H)$ is a functor from K-algebras to groups, which sends a K-algebra R to the group

$$\mathrm{SU}_{n,q}(R, H) = \{X \in \mathrm{SL}_n(R \otimes_K L) : X^* H X = H\}$$

But as with orthogonal groups, it is mostly enough to consider the group of K-points. (This is potentially confusing – the group of K-points of $\mathrm{SU}_{n,q}(-, H)$ consists of matrices with entries in L, not just in K, because of the tensor up to L.)

Status and future work: Implementation for special linear groups is essentially complete. Special orthogonal groups are next up, then special unitary groups.

Name: calculate_root_spaces

Purpose: Do root space calculations for special linear, special orthogonal, and special unitary groups.

Technical notes: Let G be a algebraic K-group with Lie algebra $\mathfrak{g}$, and let $S \subset G$ be a maximal k-split torus. Then S acts on $\mathfrak{g}$ by conjugation.

$$S \times \mathfrak{g} \to \mathfrak{g} \qquad s \cdot X = sXs^{-1}$$

A character of S is a morphism of algebraic k-groups $S \to \mathbb{G}_m$. The characters of S form an abelian group, $X(S)$. In all examples dealt with in Pinning-Sympy, S is a subset of the diagonal subgroup of G, so $X(S)$ is a subgroup of the free abelian group generated by $\alpha_1, \dots, \alpha_n$ where $\alpha_i : S \to k^\times$ is the map which picks off the ith diagonal entry. For $\alpha \in X(S)$, define

$$\mathfrak{g}_\alpha = \left\{ X \in \mathfrak{g} : sXs^{-1} = \alpha(s)X \text{ for all } s \in S \right\}$$

Think of $\mathfrak{g}_\alpha$ as a generalized eigenspace. Elemetents of $\mathfrak{g}_\alpha$ are like simultaneous eigenvectors for all elements $s \in S$, except that the eigenvalue is allowed to vary with s. For all α, $\mathfrak{g}_\alpha$ is a vector subspace of $\mathfrak{g}$. For all but finitely many α, $\mathfrak{g}_\alpha$ is just the zero subspace. The $\alpha \in X(S)$ so that $\mathfrak{g}_\alpha \neq \{0\}$ are called roots. If α is a root, the associated space $\mathfrak{g}_\alpha$ is called the root space.
If the field K is algebraically closed and characteristic zero, then root spaces are always one-dimensional (over K). This fact is critical to the classification of semisimple Lie algebras over $\mathbb{C}$, for example. However, in situations dealt with in Pinning-Sympy, the root spaces may have dimension greater than one.
The purpose of calculate_root_spaces is to run calculations to determine the nonzero $\mathfrak{g}_\alpha$ for special linear, special orthogonal, and special unitary groups.

Status and future work: Probably complete. Mostly useful at this point as a reference for example code blocks.

Name: matrix_utility
Purpose: Miscellaneous stand-alone functions related to matrices.
Status and future work: Everything currently works as intended. Add functions as needed.

Name: nondegenerate_isotropic_form

Purpose: Custom class/data type to represent a nondegenerate isotropic form on a vector space.

Technical notes: Let V be a finite-dimensional vector space over a field K. A symmetric bilinear form on V is a K-bilinear map $b : V \times V \to K$ such that $b(v, w) = b(w, v)$. It is isotropic if $b(v, v) = 0$ for some $v \neq 0$. It is degenerate if the linear function $b_v : V \to K, b_v(w) = b(v, w)$ is identically zero for some $v \in V$, and nondegenerate if not. Given a basis $\{e_1, \dots, e_n\}$ of V, there is a matrix associated to b, given by $B_{ij} = b(e_i, e_j)$. B_{ij} is symmetric if and only if b, and B is invertible if and only if b is nondegenerate.

Now consider a quadratic extension L/K, and assume V is a vector space over L. A hermitian form on V is a K-bilinear map $h : V \times V \to L$ such that $h(v, w) = \overline{h(w, v)}$, where the bar denotes conjugation on L. The form is instead skew-hermitian if $h(v, w) = -\overline{h(w, v)}$. Degeneracy and isotropy are defined as for symmetric bilinear forms.

Using some theory, every nondegenerate isotropic symmetric bilinear form is equivalent to one whose matrix has the block form below, where q is the Witt index of b, I_q is the $(q \times q)$ identity matrix, and C is an invertible diagonal matrix (with entries in K).

$$B = \begin{pmatrix} 0 & I_q & 0 \\ I_q & 0 & 0 \\ 0 & 0 & C \end{pmatrix}$$

Similarly, every nondegenerate isotropic (skew)-hermitian form is equivalent to one whose matrix has the block form below, where q is again the Witt index, $\varepsilon = 1$ for hermitian and $\varepsilon = -1$ for skew-hermitian, and C is an invertible diagonal matrix (with entries in L). More precisely, $\overline{C} = \varepsilon C$, so entries of C are "purely real" in the hermitian case and "purely imaginary" in the skew-hermitian case.

$$H = \begin{pmatrix} 0 & I_q & 0 \\ \varepsilon I_q & 0 & 0 \\ 0 & 0 & C \end{pmatrix}$$

Status and future work: Achieves all needed functions for the moment, but will definitely need expanding.

Name: pinned_group

Purpose: Custom class/data type to represent a matrix group and a large amount of associated data.

Technical notes: Let G be an algebraic K-group with root system Φ. A pinning of G is a collection of morphisms $X_\alpha : V_\alpha \to G$ for each root $\alpha \in \Phi$, satisfying several conditions. X_α is a morphism of schemes, and V_α is a scheme such that $V_\alpha(K)$ is a K-vector space and G is an algebraic K-group scheme so $G(K)$ is a group. But it is oke for the notation to get sloppy and identify X_α (the morphism of schemes) with the map $X_\alpha(K) : V_\alpha(K) \to G(K)$, which is more concrete.
The dimension of $V_\alpha(K)$ is equal to the dimension of the root space $\mathfrak{g}_\alpha$. The conditions on X_α are as follows:

1. $X_\alpha(0) = 1$
2. (X_α is roughly a homomorphism) For $u, v \in V_\alpha(K)$
$$X_\alpha(u) \cdot X_\alpha(v) = X_\alpha(u + v)$$
(This is not true all the time, sometimes there is an extra term) INCOMPLETE
3. (Torus conjugation formula) For $s \in S(K)$ and $u \in V_\alpha(K)$,
$$s \cdot X_\alpha(u) \cdot s^{-1} = X_\alpha\Big(\alpha(s)u\Big)$$
4. (Commutator formula) For every $\alpha, \beta \in \Phi$ with $\alpha \neq c\beta$ for any scalar c, there are homogeneous polynomial maps
$$N_{ij}^{\alpha\beta} : V_\alpha(K) \times V_\beta(K) \to V_{\alpha+\beta}(K)$$

such that for any $u \in V_\alpha(K)$ and $v \in V_\beta(K)$ we have

$$\Big[X_\alpha(u), X_\beta(v)\Big] = \prod_{\substack{i,j\geq 1 \\ i\alpha+j\beta\in\Phi}} X_{i\alpha+j\beta}\Big(N_{ij}^{\alpha\beta}(u,v)\Big)$$

There is also another condition which is not strictly necessary for a pinning, but very useful to have if possible.

5. (Weyl group conjugation formula) For every $\alpha \in \Phi$, there is a Weyl group element $w_\alpha \in G(K)$ such belongs to the subgroup generated by $X_\alpha(V_\alpha(K))$ and $X_{-\alpha}(V_{-\alpha}(K))$ and such that for every $\beta \in \Phi$ and $v \in V_\beta(K)$, we have

$$w_\alpha \cdot X_\beta(v) \cdot w_\alpha^{-1} = X_{\sigma_\alpha(\beta)}\Big(\varphi_{\alpha\beta}(v)\Big)$$

where $\sigma_\alpha : \Phi \to \Phi$ is reflection across the hyperplane perpendicular to α, and $\varphi_{\alpha\beta} : V_\beta \to V_{\sigma_\alpha(\beta)}$ is some invertible linear function. (Note that σ_α permutes roots of the same length, which means that β and $\sigma_\alpha(\beta)$ have root spaces of equal dimension.)

The tests in pinned_group are designed to check all of these formulas hold.

Status and future work: Implementation for special linear groups is essentially complete. Special orthogonal groups are next up, then special unitary groups.

Name: quadratic_field

Purpose: Custom class/data type to model an element of a quadratic field extension.

Technical notes: A quadratic extension of a field K is a field $L = K(\sqrt{d})$ for some non-square $d \in K$, i.e. $L = \Big\{a + b\sqrt{d} : a, b \in K\Big\}$. The Galois group $\mathrm{Gal}(L/K)$ is order two, with elements Id_L and σ where $\sigma(a + b\sqrt{d}) = a - b\sqrt{d}$. The nontrivial automorphism σ is typically called conjugation. Informally speaking, elements of the form $a + 0\sqrt{d}$ are called "real" and elements of the form $0 + b\sqrt{d}$ are called "purely imaginary." The general setting of a quadratic extension is not too different from examples like $K = \mathbb{R}$ and $L = \mathbb{C}$, or $K = \mathbb{Q}$ and $L = \mathbb{Q}(\sqrt{d})$.

Status and future work: Basic operations such as addition, negation, subtraction, multiplication, and conjugation are implemented. Division not yet implemented. Minimal tests are implemented, but more could be good.