

# Raddy Documentation

## Conventions

- If const generics N and L appear in the same type:
  - N is for the dimension of the vector that **the derivative is w.r.t.**;
  - L is the length of the vector.
  - When creating active variable, L == N **must** be satisfied.

## Core data structure: Ad<const N: usize>

- Generics
  - N: # dimension of **variable** (that requires derivatives).
- Notes
  - Computes on precision f64.

## 1. Constructors

### 1.1. pub fn ad(value: f64) -> Ad<1>

- Initializes a differentiated scalar.

### 1.2. pub fn vec<const L: usize>(values: &[f64]) -> SVector<Ad<L>, L>

- Initializes a differentiated vector.
- *Panics* if size mismatch.

### 1.3. pub fn val<const L: usize>(value: f64) -> Ad<L>

- Initializes a constant scalar.

### 1.4. pub fn valvec<const N: usize, const L: usize>(values: &[f64]) -> SVector<Ad<N>, L>

- Initializes a constant vector.
- *Panics* if size mismatch.

## 2. Supports

- Elementary function (sin, cosh, exp, ln, *etc.*)
  - Does **not** support atan by design, please use atan2 instead.
- Norms and determinant for matrices.
- Matrix multiplication.
- SVD, although you should not use it (for some numerical problems).