Mix & match your model, algorithm, and task and start to BenchMARL!

**Models**
Models are neural network blueprints that can be chained to make actors and critics with or without parameter sharing.

<table>
<thead>
<tr>
<th>Model</th>
<th>Policy</th>
<th>Critic</th>
</tr>
</thead>
<tbody>
<tr>
<td>MLP</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>CNN</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>GNN</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>Deep Sets</td>
<td>✔️</td>
<td>✔️</td>
</tr>
</tbody>
</table>

**Algorithms**
Algorithms are an ensemble of components (e.g., loss, replay buffer, exploration strategy) that determine the training strategy.

**Tasks**
Tasks are scenarios from a simulated environment which constitute the MARL challenge to solve.

- **Environment**
  - VMAS: 27
  - SISL: 3
  - SMACv2: 15
  - MPE: 8
  - Melting Pot: 49

### Reporting and plotting
Automatically report and plot your BenchMARL results using statistically-robust integrated tools [1, 2].

Interactively log your results on Wandb and compare with the publicly available benchmarks:

We are benchmarking and releasing results for BenchMARL tasks. Check out the results in the VMAS environment:

---

**Features**
BenchMARL’s features focus on enabling its core tenants: standardization and reproducibility.

- Reproducibility
- Standardized reporting
- TorchRL backend
- Experiment independence
- Easy integration of new solutions

The TorchRL backend allows BenchMARL to re-use extensively-benchmarked single-agent implementations.