自动特征选择：强化学习视角

摘要

特征选择是机器学习的关键步骤，它为后续的预测任务选择最重要的特征。有效的特征选择可以帮助降低维度、提高预测精度和提高结果可理解性。传统上，由于空间可能非常大，从特征子集空间找到最佳特征子集具有挑战性。虽然特征选择已经做了很多努力，但强化学习可以为更全局最优的搜索策略提供一个新的视角。在初步工作中，我们提出了一种用于特征选择问题的多智能体强化学习框架。具体来说，我们首先通过将每个特征视为代理来使用强化学习框架重新制定特征选择。此外，我们以描述性统计来作为状态表示，以推导出固定长度的状态表示作为强化学习的输入。此外，我们研究了如何通过更有效的奖励方案来提高特征代理之间的协调。此外，我们提出了一种基于 GMM 的生成校正采样策略来加速多智能体强化学习的收敛性。我们的方法更全局地搜索特征子集空间，并且由于强化学习的性质，可以很容易地适应实时场景。在扩展版本中，我们从两个方面进一步加快了框架。从采样方面，我们通过提出一种基于秩的 softmax 采样策略来展示间接加速。从探索方面，我们提出了一种基于交互式强化学习 (IRL) 的探索策略来展示直接加速。大量的实验结果表明，与传统方法相比，该方法有了显著的改进。

关键词：特征选择；多智能体强化学习；交互式强化学习

1 引言

在数据科学和机器学习领域，特征选择是一个关键的问题。随着数据集的不断增大和特征的增多，选择最重要的特征以提高模型性能和降低计算成本变得至关重要。传统的特征选择方法通常基于统计学或启发式规则，但它们可能无法处理高维数据或发现非线性关系。强化学习是一种通过代理与环境交互学习如何做出决策的方法。将强化学习应用于特征选择问题，可以使模型在不同特征之间进行权衡，自动学习适应于特定任务的最佳特征组合。因此在本次复现的工作中，选择一种基于强化学习的特征选择来作为复现的工作是比较有前景的工作。

在本次的复现工作中选择了以多智能体强化学习进行特征选择的工作，在这篇文章中，作者以 DQN 深度强化学习来构建特征选择的框架，并为每个特征分配了一个智能体，彼此之间相互联合来进行特征选择的工作。然而这种多智能体的强化学习方法有一个明显的弊端，太多的智能体占用了大量的空间，而且每个智能体都需要一定的计算资源来优化网络，为了解决这个问题，可以将特征选择问题重新定义为一种基于 push 和 pop 机制的单智能体决策，并引入了更为先进的优势演员评论家 (advantage actor critic) 算法，来对智能体进行优化，实验表明，新的算法的能够一定程度上提高算法的准确性。

2 相关工作

2.1 特征选择

特征选择可以根据特征选择算法与机器学习任务的结合方式分为三种类型，即过滤方法、包装方法和嵌入方法 [1]，[21]。过滤方法仅通过相关性分数对特征进行排名，仅选择排名靠前的特征。代表性的过滤方法包括单变量特征选择 [31] [6] 和基于相关性的特征选择 [32] [10]。过滤方法计算复杂度很低，非常快速，因此在高维数据集上非常高效。然而，它们忽视了特征之间的依赖关系，以及特征选择与后续预测器之间的相互作用。与过滤方法不同，包装方法利用预测器，并将预测性能视为目标函数 [8]。代表性的包装方法是分支定界算法 [19] [12]。包装方法应该比过滤方法具有更好的性能，因为它们在整个特征子集空间上进行搜索。然而，随着特征数量的增加，特征子集空间呈指数增长，使得遍历特征子集空间成为一个 NP 难题。进化算法 [29] [11]，[7] 降低了计算成本，但只能保证局部最优结果。嵌入方法比包装方法更紧密地将特征选择与预测器结合，实际上将特征选择作为预测器的一部分。最广泛使用的嵌入方法包括 LASSO [27]，决策树 [24] 和 SVM-RFE [9]。嵌入方法在合并的预测器上可能具有最优性能，但通常与其他预测器不太兼容。最近，一种新兴的技术称为强化特征选择通过采用强化学习技术解决特征选择问题，已经表现出显著的改进。

2.2 多智能体强化学习

这篇文章与多智能体工作有关，其中多个智能体共享一个复杂的环境并相互作用 [26]。在单智能体表述中，强化学习智能体采取行动来改变环境，并获得反馈奖励以评估其行动，从而改进其下一次行动决策 [25]。在多智能体表述中，智能体不仅需要与环境互动，还需要彼此之间进行互动。Stankovic 等人提出了新算法，用于多智能体分布式迭代价值函数近似，在此算法中，智能体在评估对单一目标政策的响应时被允许有不同的行为政策 [23]。Liao 等人提出了通过强化学习进行多目标优化 (MORL)，以解决最优电力系统调度和电压稳定问题，该问题通过由估计路径值选择的路径在高维空间的单个维度上进行，该路径值代表找到更好解决方案的潜力 [14]。Yang 等人开发了能够处理大规模智能体并具有有效通信协议的深度强化学习算法 [30] [20]。Lin 等人提出使用强化学习来解决大规模车队管理问题，并提出了一个成功应对出租车车队管理问题的上下文多智能体强化学习框架 [15]。然而，这些方法通过手工规则而不是通过表示学习来定义它们的状态，这可能会遗漏环境提供的重要信息。此外，我们知道由于动作空间大，多智能体强化学习的训练速度较慢，但这些方法很少研究如何提高训练效率。现有研究 [5] [13] 创建了一个单独的智能体来做决策。然而，这个智能体必须决定所有 N 个特征的选择或不选择。换句话说，这个智能体的动作空间是 2^N。这种表述类似于进化算法，倾向于获得局部最优解。

3 本文方法

3.1 框架概述

图1显示了论文提出的框架，由许多特征子空间探索步骤组成。每个探索步骤包括两个阶段，即控制阶段和训练阶段。在控制阶段，每个特征代理根据其策略网络采取行动，该网络

将当前状态作为输入并输出推荐的动作和下一个状态。每个特征代理的选择/取消选择动作将改变所选特征子集的大小和内容，从而导致一个新的选定特征子空间。文章将所选特征子集视为环境。以描述性统计作为状态表示同时，特征代理采取的行动会产生整体奖励。然后将此奖励分配给每个参与代理。在训练阶段，代理通过经验回放独立训练他们的策略。对于代理 i，在时间 t，新创建的元组 $\{s_i^t, a_i^t, r_i^t, s^t + 1_i\}$，包括状态 (s_i^t)、动作 (a_i^t)、奖励 (r_i^t) 和下一个状态 $(s^t + 1_i)$ 存储在每个代理的内存中。代理 i 使用其对应的小批量样本来训练其深度 Q 网络 (DQN)，以便基于贝尔曼方程 (1) 获得最大的长期奖励.

$$Q(s_i^t, a_i^t, \theta_i^t) = r_i^t + \gamma \max Q(s_i^{t+1}, a_i^{t+1} | \theta_{t+1}) \tag{1}$$

其中 θ 是 Q 网络的参数，g 是折扣因子。特征子空间会一直探索直到收敛或满足几个预定义的标准。

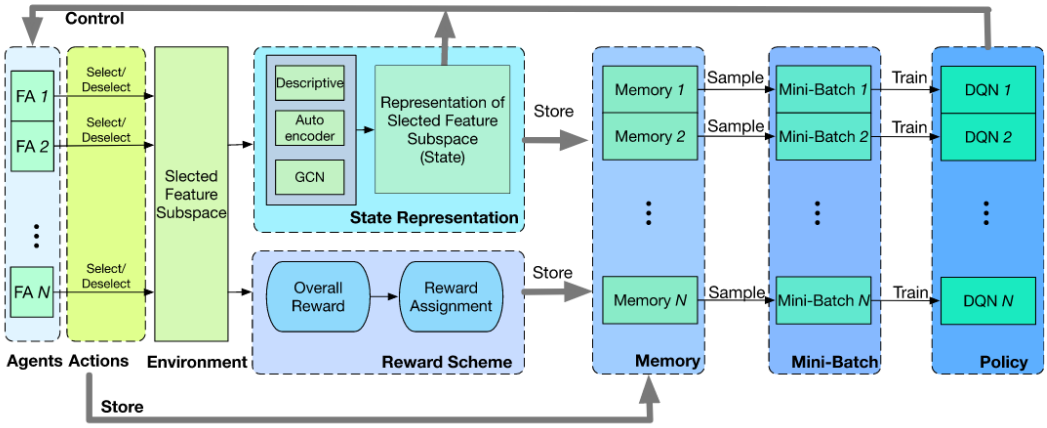


图 1. 框架示意图

3.2 强化学习建模

3.2.1 智能体

假设有 N 个特征，那么这 N 个特征定义为 N 个智能体。对于一个智能体，它旨在为相应的特征做出选择决策。

3.2.2 动作

对于第 i 个特征代理，特征动作 $a_i = 1$ 表示选择第 i 个特征，$a_i = 0$ 表示取消选择第 i 个特征。

3.2.3 环境

在文章的设计中，环境是特征子空间，表示选定的特征子集。每当特征代理发出选择或取消选择特征的动作时，特征子空间（环境）的状态会发生变化。

3.2.4 状态

状态 s 是描述所选特征子集。为了提取 s 的表示，以特征子集的元描述性统计作为状态表示。

3.2.5 奖励

文章设计了一个度量来量化所选特征子集生成的整体奖励 R，它被定义为 (i) 所选特征子集 Acc 的预测精度的加权和，(ii) 所选特征子集 Rv 的冗余，以及 (iii) 所选特征子集 Rd 的相关性。

3.3 奖励分配策略

这篇论文开发了一种策略，将整体奖励分配给每个特征代理。事实上，分配给每个代理的整体奖励分配显示了代理之间的协调和竞争关系。原则上，我们应该识别和奖励所有参与的特征代理。图2显示了奖励分配的示例。有四个具有四个相应特征代理的特征。在上一次迭代中，选择特征 1、2、3，不选择特征 4。在当前迭代中，特征代理 1 和特征代理 2 发出动作以选择特征 1 和特征 2；特征代理 3 发出去选择特征 3 的动作；特征代理 4 不参与并发出任何动作来改变特征 4 的状态。总之，只有三个特征代理（FA1、FA2、FA3）参与和问题动作。因此，这三个代理平等地共享当前奖励 R。

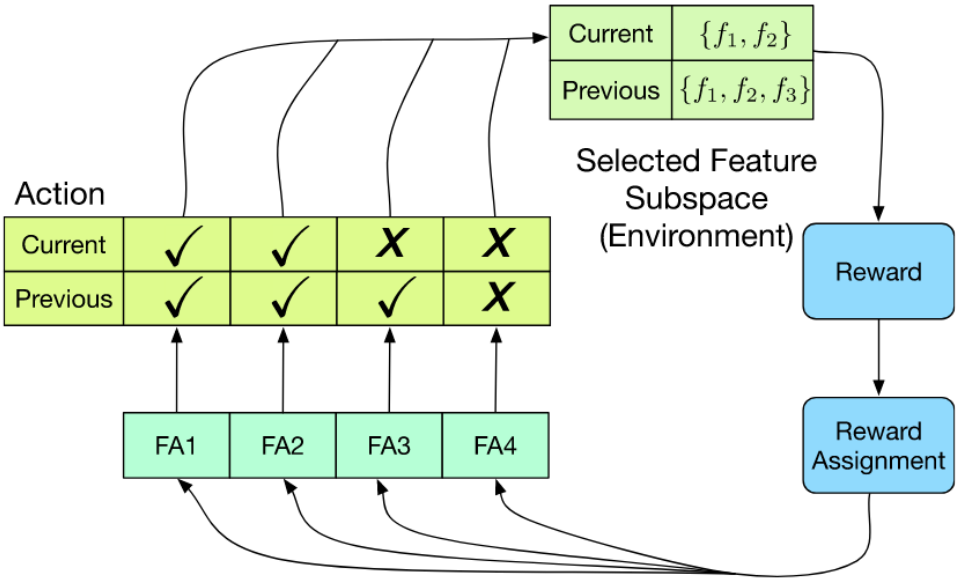


图 2. 奖励分配策略示意图

3.4 奖励测量

本文将预测精度 Acc、特征子空间相关性 Rv 和特征子空间冗余 Rd 结合起来作为动作的奖励 R。预测准确性。

预测准确性。由于特征选择的目标是探索和识别令人满意的特征子集，该子集将用于在下游任务（例如分类和异常值检测）中训练预测模型。作者建议使用准确度 Acc 预测模型量

化奖励。具体来说，如果预测精度较高，则产生所选特征子集的动作应该获得较高的奖励；如果预测精度较低，则产生所选特征子集的动作应该获得较低的奖励。

特征子空间特征。除了利用预测精度作为奖励外，建议考虑所选特征子集的关系。具体来说，合格的特征子集通常具有较低的信息冗余和与预测标签（响应）相关的信息相关性。信息相关性和冗余都可以通过互信息来量化，用 I 表示。具体定义如下：

$$I(x,y) = \sum_{i,j} p(x_i, y_j) log(\frac{p(x_i, y_j)}{p(x_i)p(y_j)} \tag{2}$$

其中 x_i, y_i 是第 i 个特征和第 j 个特征，$p(x_i, y_j)$ 是 x_i 和 y_i 的联合概率分布，而 $p(x_i)$ 和 $p(y_j)$ 是 x_i 和 y_j 的边缘概率分布。

特征子集的信息冗余，用 Rd 表示，可以通过特征之间的两两互信息之和来量化。形式上，Rd 由：

$$Rd = \frac{1}{|S|^2} \sum_{x_i, x_j \in \mathcal{S}} I(x_i; x_j) \tag{3}$$

其中 S 是特征子集,x_i 是第 i 个特征。

特征子集的信息相关性，用 Rv 表示，可以通过特征和标签之间的互信息来量化。形式上，Rv 由下式给出：

$$Rd = \frac{1}{|S|} \sum_{x_i \in \mathcal{S}} I(x_i; c) \tag{4}$$

其中 c 是标签向量。

3.5 描述性统计状态表示

假设有一个 $M * N$ 数据集 D，其中包括 M 个数据样本和 N 个特征。n_j 是第 j 个探索步骤中所选特征的数量。然后，因此所选数据矩阵 S 的维度为 $M * n_j$ ，不同的探索过程，其维度会有所变化。然而，DQN 中的策略网络和目标网络每次都需要状态表示向量 s 是一个固定长度的向量。因此，我们需要从选定的数据矩阵 S 推导出固定长度的状态向量 s，其维度随时间变化。为了推导出固定长度的精确状态表示，我们以元描述性统计作为状态的表示。图3显示了我们如何通过两步过程从所选数据矩阵中提取描述性统计的元数据。

第 1 步。提取所选数据矩阵 S 的描述性统计，包括标准偏差、最小值、最大值和 Q1（第一个四分位数）、Q2（第二个四分位数）和 Q3（第三个四分位数）。具体来说，我们提取了 S 中每个特征（列）的七个描述性统计数据，从而得到一个大小为 $7 * n_j$ 的描述性统计矩阵 D。

第 2 步：提取描述性统计矩阵 D 中每一行的七个描述性统计数据，并获得大小为 7*7 的元描述性统计矩阵 D'。

第 3 步，将每个列 D' 链接到固定长度为 49 的状态向量 s 中。

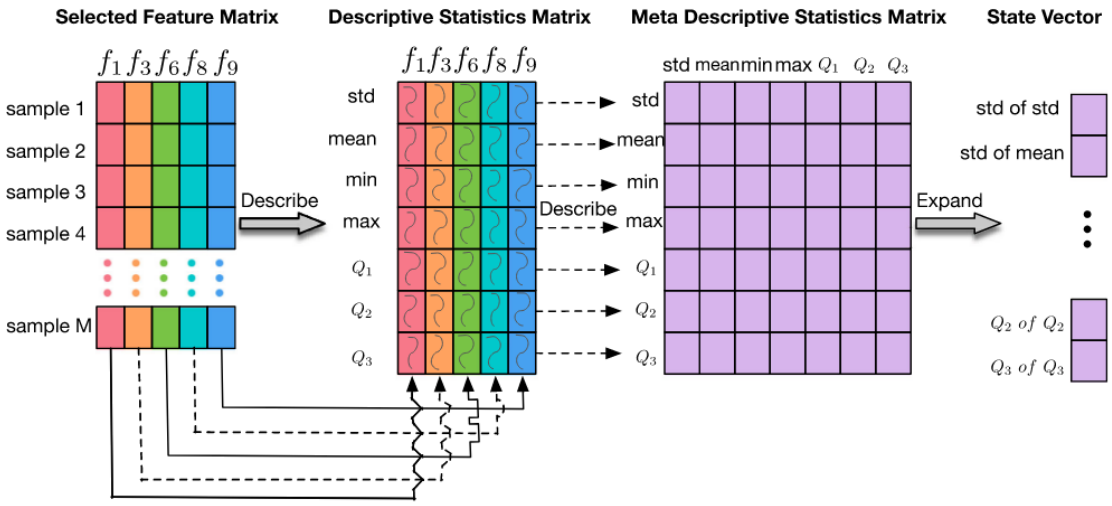


图 3. 描述性统计状态表示

3.6 生成校正采样改进的经验重放

经验回放被广泛用于提高神经网络在强化学习 [16] [17] 中的训练效率。在采取行动后，最新的样本以由动作 (a)、奖励 (r)、状态 (s) 和下一个状态 (s_0) 组成的元组的形式存储在内存中以替换最旧的样本。在训练过程中，选择小批量样本来更新策略网络。

在特征子空间探索任务中，作者希望利用高质量的样本加快勘探速度。先前的研究通过增加高质量样本 [22] [28] 的采样概率来解决这个问题。然而，这种策略创造了一个新问题：采样器反复选择有限数量的高质量样本。因此，先前的研究不能保证所选样本在不同训练步骤之间的独立性，并且不能覆盖未知高质量样本种群中。

Algorithm 1 GMM-Based Generative Rectified Sampling Algorithm

Input: Memory dataset T

Output: A mini-batch of samples B

Determine high-quality sample proportion p of T

Stratify T into two groups: Samples with a label 0 are assigned to group T_0 and samples with a label 1 are assigned to group T_1

for $i = 0$ to 1 **do**

  Determine sample number N_i of T_i

  Determine component number K_i of GMM model G_i

  Rank samples in T_i by their reward r, then select top $N_i \times p$ samples from T_i to form the high-quality dataset H_i

  Use H_i to train the GMM G_i as $\sum_{k=1}^{K_i} f_{i,N}(m_i, S_i)$ via EM algorithm

  Generate $N_i \times (1-p)$ samples from G_i to form the generated dataset G_i'

  Join H_i and G_i' to create high-quality dataset of action i, T_{0i}'

end for

Join T_0' and T_{01}' to get high-quality dataset T'

Sample a mini-batch of samples B from T' =0

为了解决这个问题，本文提出了一种基于高斯混合模型（GMM）的生成校正采样算法。对于每个代理，如算法1 所示，我们取一组内存样本 $T = \{< a, r, s, s' >\}$ 作为输入。我们首先将其分为两组:T_0 和 T_1。选择动作 (a=0) 的样本被分配到 T_0 组，而选择动作 (a=1) 的样本被分配到 T_1 组。之后，我们根据奖励 (r) 对 T 中的记忆样本进行排序，并选择每组中高奖励样本的前 p 个比例作为高质量的样本。然后使用选定的高质量样本通过期望最大化 (EM) 算法 [4] 为其对应的组训练两个基于 GMM 的生成模型。之后，对于每个组，我们使用其对应的训练有素的 GMM 模型来生成模拟样本，以替换相应组中低奖励样本的 (1-p) 比例的样本。通过这种方式，我们生成了高质量数据集。代理将从新的高质量数据集中获取小批量样本以加速训练。

3.7 基于排序的 Softmax 采样

基于 GMM 的生成校正采样策略可以充分利用经验回放中的样本，加快强化学习策略的训练过程。然而，它自然有三个潜在的缺点：1）它基于这样一个假设，即样本是由高斯混合模型生成的，而它们的实际分布可能不同；2）GMM 模型的拟合在计算上是昂贵的。更糟糕的是，每次采样时都需要拟合 GMM 模型;3) 样本中可能存在噪声，影响 GMM 模型的拟合精度。在这里，我们有一个问题：我们能否提出一种比 GMM 的生成校正采样策略更有效的更简单的采样策略，来降低计算负担和和获得更高的鲁棒性?

为了解决这些问题，我们引入了一种基于排名的 softmax 采样策略。在这个采样策略中，通过每个样本在经验回放记忆中的排名来衡量每个样本的重要性。然后根据每个样本的等级设计每个样本的采样概率。

$$P(i) = \frac{exp(p_i)}{\sum_{n-1}^{N_E} exp(p_n)} \tag{5}$$

其中 N_E 是经验重放记忆的大小，p_i 是第 i 个样本的优先级，我们做出 $p_i = \frac{1}{rank(i)}$ ，其中 $rank(i)$ 是样本 i 根据其奖励的排名。softmax 来说，所有概率的总和等于 1。由于 $rank(i)$ 是一个相对值，它对噪声具有很高的容忍度，并且可能非常稳健。相比于 GMM，该采样不需要拟合，使计算负担非常低。具体来说，如算法2 所示，我们首先推导出每个样本的排名，然后通过等式（5）获得它们的采样概率。代理将根据基于排名的采样概率采用小批量样本。与基于 GMM 的生成校正采样策略相比，由于计算负担低，并且由于对噪声的鲁棒性有效，它是有效的。

Algorithm 2 Softmax Sampling Algorithm

Input: Memory dataset T

Output: A mini-batch of samples B

1: Determine the size of experience replay memory N_E from T

2: Rank samples in T by their reward, and assign priority $p_i = \frac{1}{\text{rank}(i)}$ to each sample

3: Derive the sampling probability for each data sample by Equation (7)

4: Sample a mini-batch of samples B from T =0

3.8 通过交互式强化学习加速特征子空间探索

在经典的强化学习框架中，代理反复探索状态空间并获得奖励，之后它获得越来越多的经验并表现得更好和更好。这种探索策略是通用的和通用的，这意味着它可以应用于任何制定的强化学习问题。然而，在我们的例子中，状态空间非常大，如果我们简单地适应传统的探索策略，探索效率会相当低。在这里，我们有一个问题：我们能否提出一种更高级的探索策略，该策略可以沿着更有希望的方向探索，从而加快特征空间探索过程?

所提出的多智能体强化学习框架逐步改进了自身，当其性能非常差时，它在开始时有一个学徒期。为了减少其探索空间，我们引入了交互式强化学习 (IRL) [2] [3]。在 IRL 中，一种简单的特征选择方法，即 K-Best Selection [31]，作为“指导家”来指导强化学习沿着相对较好的方向探索。在预定义的步骤之后，强化学习放弃顾问并独立探索状态空间。

具体来说，如算法 3 所示，我们首先通过 K-Best Selection 推导出特征子集 S_K。在学徒步骤中，我们随机选择 S_K 中的一半特征将它们添加到选定的特征子集中。通过这个加法，状态表示发生了变化，从而将强化学习引导到更好的探索方向。我们在每一步不使用 S_K 中的所有特征的原因是为了避免过度拟合，并保持特征选择过程与 K-Best 选择不同。在学徒期之后，多智能体强化学习将独立进行特征选择。

Algorithm 3 Interactive Reinforcement Learning Enhanced Exploration Strategy

Input: Feature number K, apprenticeship step N_A, overall step N_O, feature set S

Output: Optimal feature subset S'

1: Derive a feature subset S_K via K-Best Selection
2: Randomly initialize selected feature subset S''
3: **for** $i = 1$ to N_A **do**
4: Derive a selected feature subset S'_i by multi-agent reinforcement (Note: This step relies on the selected feature subset S'_{i-1} from the last step, as S'_{i-1} decides the state representation.)
5: Randomly choose $\frac{K}{2}$ features from S_K, denoted as $S_{K/2i}$
6: Let $S'_i = S'_i + S_{K/2i}$
7: **end for**
8: **for** $i = N_A + 1$ to N_O **do**
9: Derive a selected feature subset S'_i by multi-agent reinforcement learning feature selection
10: **end for**=0

4 复现细节

4.1 复现工作主要内容

对于这篇论文，作者并没有给出相应的源代码，因此，在本次复现的工作中的代码都是基于作者的伪代码以及自己的理解进行编写。在此次复现中，我的主要工作是基于 Kaggle 的森林覆盖集的数据集，并将复现的工作分为几个步骤进行完成：首先是对整个多智能体的特征选择框架进行搭建，建立每个智能体的 DQN 网络，并将特征空间作为环境变量，对其进

行建模；其次，在此基础上我加入了基于高斯混合生成模型的经验采样，在该实验中，探索了高质量比例对于收敛效果的影响；而后，我加入了基于 softmax 的样本采样，将其与高斯混合生成模型的效果进行对比；最后，我构建了一个基于 K-best 的指导家，用于探索交互式强化学习加速特征子空间探索的效果。这便是对于整个论文的复现的主要工作。

在复现完主要的工作后，我发现了这篇论文的框架的主要弊端—每个特征都由一个智能体进行决策，而对于每个智能体来说，它有自己的神经网络，有自己的经验池，那么随着特征的增加，带来了资源和计算成本增加，尤其是在高维数据的情况下；其次，每个智能体都独立操作时，协调它们的行为以达到一个共同的目标或保持系统整体的一致性变得更加困难。基于这些问题，提出一个单智能体的特征选择框架是有必要的，于是，我将特征选择的强化学习框架重新进行定义，提出了一种基于 pop 和 push 动作的单智能体强化学习特征选择框架，并基于优势演员评论家算法实现，实现结果表明，这种框架是有效的。

4.2 论文的实验环境搭建

4.2.1 DQN 网络配置

经验回放池的配置

对于 DQN，采用的是带经验回放池的 DQN 网络，具体的参数如下：

- **经验回放池大小**：200
- **mini-batch**：32
- **学习率** (lr)：0.01

Q 网络结构

1. 第一层（全连接层）：输入维度为状态表示维度，输出维度为 64。
2. 第二层（全连接层）：将 64 维的表示映射到 8 维。
3. 第三层（全连接层）：将 8 维的表示映射到动作空间维度，每个维度对应一个动作的 Q 值。

对于第一层和第二层，都应用 ReLU 激活函数，用于非线性变换。

4.2.2 经验采样策略

(1) 高斯混合生成模型 (GMM) 采样

采用高斯混合生成模型对高质量样本，用高斯混合生成模型生成新的高质量样本，然后将这些样本进行混合，得到新的经验池，进行 mini-batch 采样。

输入参数

- **采样比例** p：确定从经验池中选取样本的比例。
- **经验回放池** $memo$：包含历史转换信息的集合。
- **样本数** $batch\_size$：输出样本的数量。

数据处理

1. 根据动作将经验分为两组：动作为 1 的一组，非 1 的另一组。
2. 对每组数据按奖励值排序，并根据比例 p 选取部分数据。
3. 使用选取的数据训练 GMM，并生成补充样本。
4. 处理后的数据被重新组合，随机选取 $batch\_size$ 个样本作为函数输出。

(2)Softmax 采样

采用 Softmax 采样策略，首先对经验池数据按奖励值进行排序，然后根据5, 生成概率分布，并对然后对其进行采样。此方法的优势在于不需要训练模型，计算成本更低。

输入参数

- **经验回放池** $memo$：包含历史转换信息的集合。
- **样本数** $batch\_size$：输出样本的数量。

数据处理

1. 根据奖励值对数据进行排序；
2. 根据5, 生成概率分布；
3. 根据概率分布进行采样，输出对应数量的样本。

4.2.3 k-best 指导家

该算法首先利用 K-Best 选择法从特征集合 S 中选出一个子集 S_K，然后将强化学习的学习分为两个阶段，第一个阶段为学徒期，此时智能体不仅利用强化学习结果，还会从 S_K 中随机选择一部分特征加入到当前的特征子集中。学徒阶段结束后，算法仅依赖多智能体强化学习来选择特征，直到达到设定的总步骤数。在实现过程中，需要调控学徒期的占比，并比较其收敛效果。

4.3 创新点–基于 pop-push 的单智能体特征选择

由于在本文的框架中，其采用的是多智能体的方法，每个特征都对应了一个智能体，对于每个智能体，都有一个神经网络需要训练，每个智能体都需要一个经验池，且每个智能体都是独立进行特征选择的，那么它们不一定能很好的协调彼此之间的关系，因此，为了解决以上问题，我将强化学习特征选择框架重新进行定义，并将传统的 DQN 算法换为优势演员评论家算法（A2C）[18]。

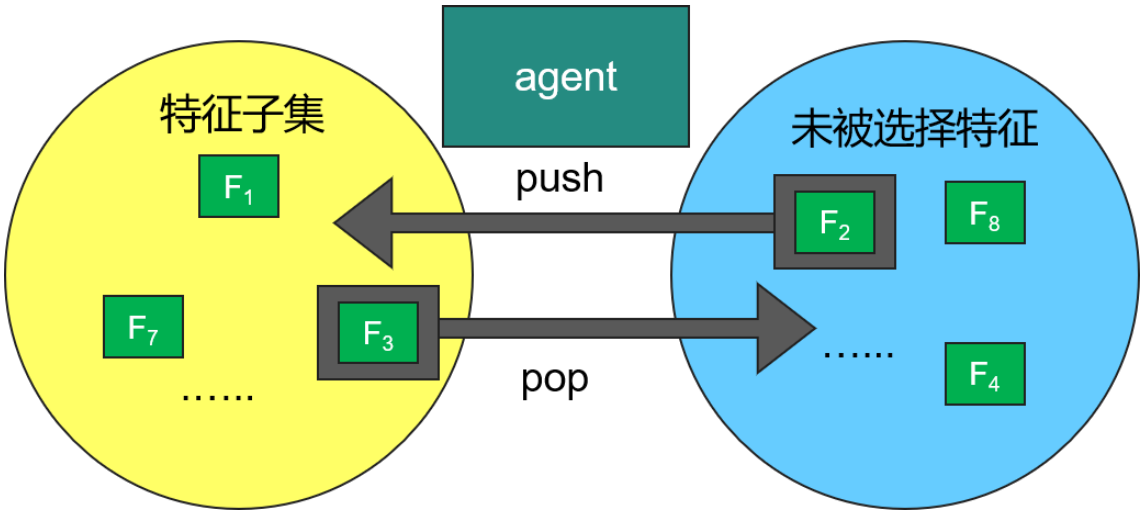


图 4. 基于 pop-push 的单智能体特征选择

4.3.1 基于 pop-push 的单智能体特征选框架

具体的，不再为每个特征分配一个智能体，而是对于整个特征空间分为两部分，然后用一个智能体进行决策。如图4, 我将特征分为两个部分，分别是已被选择的特征和未被选择的特征。那么对于智能体来说，其有两个动作维度，分别是 push 和 pop 动作，push 即从没被选择的特征集中选择一个特征加入特征集 ($a_{push} = 0$ 代表不做选择)，pop 则是从已经选择的动作中选择一个特征 ($a_{pop} = 0$ 代表不做选择)，并移除它，特别的是，每次只选择或者移除一个特征，这样可以将动作空间的维度下降到特征数量维，同时，对于 push 动作，它是从未被选择的特征中挑选一个特征，那么对于其动作的可选择特征个数并不是固定的，而对于 A2C 算法，其拟合的 Actor 网络的输出是一个关于各个动作可能值得概率分布，其维度是固定的，因此需要对网络的输出进行修改，具体的，将结合当前选择和没被选择的特征，分别生成一个 mask 层, 将无效的动作进行过滤从而得到了有效的动作概率分布，具体的算法如算法4。

4.3.2 状态表示

在新的设计中，状态 s 被表示为一个二进制向量 $[s_1, s_2, \ldots, s_n]$，其中 n 是特征的总数。每个元素 s_i 代表相应的特征是否被选中，具体表示为：1 表示特征被选中，0 表示未被选中。例如，状态向量 $[1, 0, 1, \ldots, 0]$ 表示第一个和第三个特征被选中，而其他特征未被选中。

Algorithm 4 基于 A2C 的特征选择算法

Input: 数据集 (X, Y), 最大步数 max_steps

Output: 最优特征子集 best_set, 最高准确率 best_acc

初始化环境和 A2C 代理
初始化状态 state, 准确率 acc, 最佳准确率 best_acc 和最优特征子集 best_set
for 每一步直到 max_steps **do**
    使用 A2C 生成动作 actions
    执行动作并获取新状态 next_state, 奖励 reward, 和准确率 acc
    更新 A2C 策略
    更新状态 state = next_state
    if acc > best_acc **then**
        更新 best_acc = acc 和 best_set = state
    end if
end for=0

4.3.3 奖励函数定义

我将根据预测模型的准确率 acc 来定义奖励函数。具体的奖励规则如下：

$$\text{Reward} = \begin{cases} 0 & \text{if acc} < 0.6, \\ 1 & \text{if } 0.6 \leq \text{acc} < 0.75, \\ (\text{acc} - 0.75) \times 100 + 1 & \text{if acc} \geq 0.75. \end{cases}$$

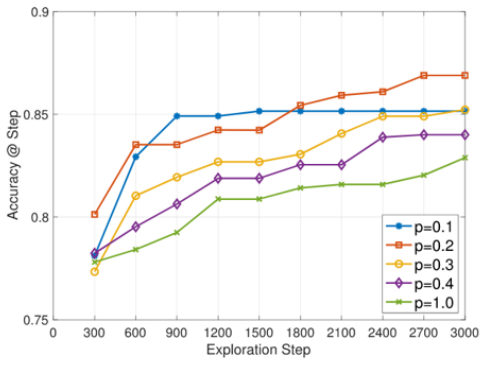
该奖励设计旨在鼓励模型实现更高的准确率。当准确率低于 0.6 时，奖励为 0，表示该准确率水平不可接受；准确率在 0.6 到 0.75 之间时，奖励为 1，鼓励模型达到基本的准确性水平；准确率超过 0.75 时，奖励将随准确率的提高而线性增加，进一步激励模型提升其精确度。

5 实验结果分析

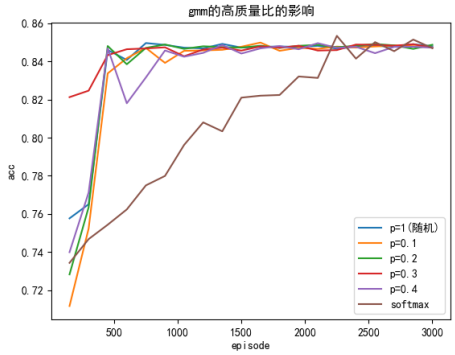
在实验中，与作者一样，选择来自 Kaggle 的森林覆盖集数据进行实验，其包含 15120 个样本，特征维度为 54，样本标签个数为 7。

5.1 对于高斯混合生成模型中的高质量样本比例的探索

我研究了基于 GMM 的生成校正采样的影响，其中高质量比例 $p = \{0, 0.1, 0.2, 0.3, 0.4\}$；。在这里，当 p=1 时,GMM 的方法简化为传统的采样策略。实验结果如5b, 可以看到对于 p=0.2 和 p=0.1 的效果最好，而对于 p=0.4 的情况，最开始在探索过程中有些不稳定，出现了较大幅度的下降，说明因为高质量样本的比例设置过高导致包含了较多的低质量样本，所以生成的样本质量较低导致的。因此对于高质量样本的比例设置也是比较重要的。而对于 softmax 采样策略来说，可以看到其收敛的曲线是比较缓慢的，特别是在前期，这是因为在前期的探索过程中，大部分的样本的质量都比较低，因此在即使是基于排序的 softmax 采样，其采样的样本质量也不一定高，随着探索的进行，样本的质量变得比较高，函数也逐渐收敛。在复现的结果中，最终特征子集的分类准确率基本上和论文的一致，达到 0.85 的水平。



(a) 原论文的收敛效果

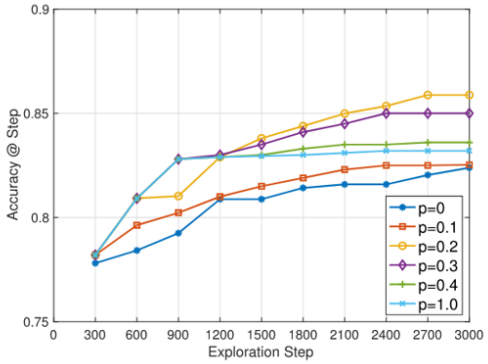


(b) 复现结果

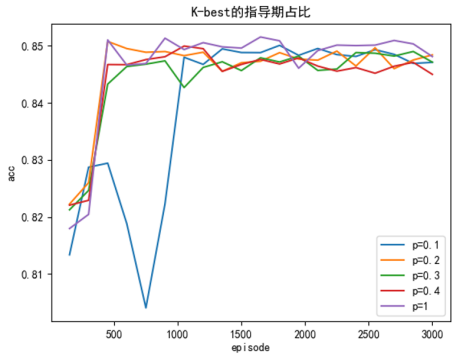
图 5. 高斯混合生成模型对比

5.2 对于基于 k-best 的指导家的学徒期比例的探索

在这一部分，我探索了交互式强化学习的影响，设置了学徒期占总的探索步数的比例分别为 p={0,0.1,0.2,0.3,0.4}, 总的探索步数为 3000。在 K-Best 选择上，设置 k=38。在学徒期，随机地将 k-best 的一半特征加入了特征子集中。实验结果如6b, 在学徒期为 p=0.1 的时候，曲线在离开指导家后出现了明显的下降，说明了指导家在一定程度上帮助了其进行探索；而对于 p=1 的情况，表现出了较好的准确率，因为这是在探索的整个阶段都是基于指导家辅助探索的，但是这并不代表了其训练的模型效果会更好，因为这其中加入了一半 k-best 的特征，离开的 k-best，其不一定能表现出较好的性能；而对于其他情况来说,p=0.2 表现出来的效果最好，在离开 K-best 指导家后，其性能并不会出现下滑，这说明了 k-best 指导家一定程度上帮助了其进行有效的探索。最终的复现结果上基本上持平了原文的收敛效果6a.



(a) 原论文的收敛效果



(b) 复现收敛结果结果

图 6. 高斯混合生成模型高质量比实验结果对比

5.3 pop-push 单智能体实现结果

我将基于 A2C 的单智能体应用于特征选择，其两个神经网络对应的参数为 $lr_{actor} = 0.001, lr_{critic} = 0.001$, 折扣系数 gamma=0.95。实验结果如表1, 可以看到对应实验结果的准确来说，超过了论文的方法。

表 1. 不同方法的分类准确率比较

原 GMM	原 Softmax	复现 GMM	复现 Softmax	A2C 单智能体
0.8690	0.8633	0.8570	0.8514	0.8697

在实验中，我们进行 1000 次的探索，每次探索都会从随机位置开始探索，然后每个探索都会和环境进行 50 次的交互，我们将所有的交互准确率的平均值作为实验结果，最终可以画出的实验结果如图7, 可以看到在 400 次探索的时候，平均的奖励就收敛到 0.854 的水平了，最终收敛到 0.8575 的水平。这的确是有效，且高效的策略。

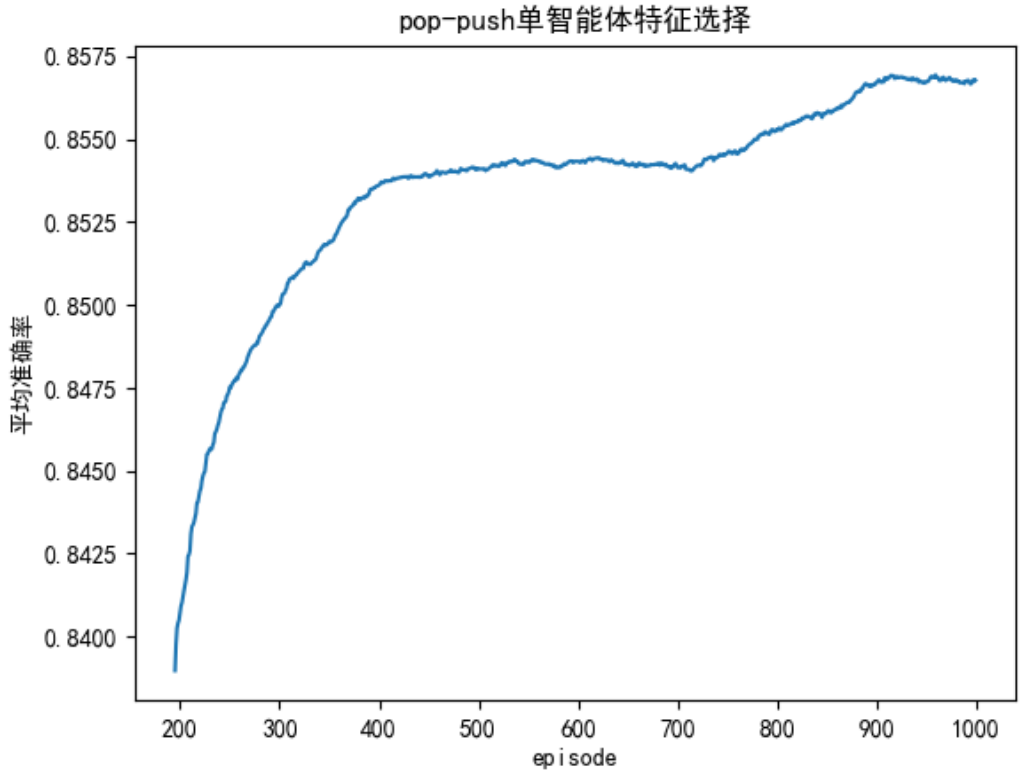


图 7. pop-push 的单智能体特征选择实验结果

在测试中我们随机初始化一个特征子集，让其进行 50 交互，8可以看到，智能体能有效的做出决策经过几次交互后，准确率便来到了 0.865 的水平，之后便在 0.86 到 0.87 之间波动，这说明智能体能够有效的识别出哪些是有用的特征，哪些是无效的特征并进行选择。

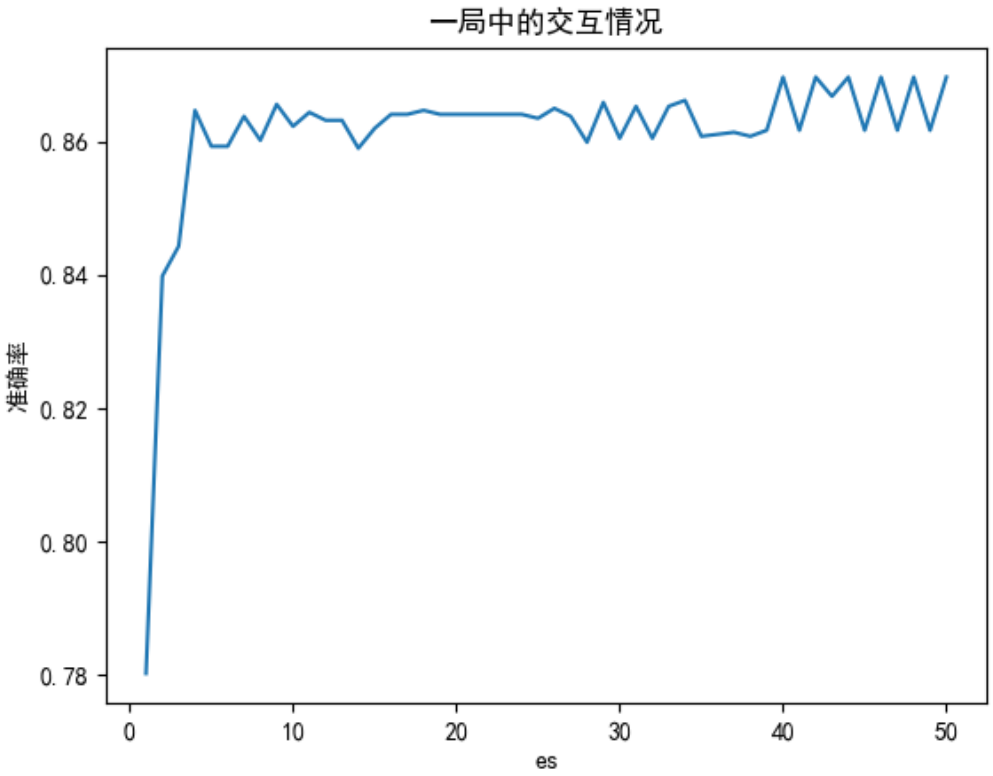


图 8. pop-push 的单智能体特征选择实验结果

6 总结与展望

总结

在此次复现工作中，我对于多智能体强化学习在特征选择上的实验进行了复现，并在其有。特征选择作为数据预处理的关键步骤，对提高模型性能至关重要。在此次复现中，我具体进行了以下工作：

- **高斯混合生成模型与 Softmax 采样策略的研究：**为了加速对特征空间的高效探索，我探讨了高斯混合模型生成采样和 Softmax 采样策略，在实验中，对于高斯生成模型的高质量比例进行探索，验证了该模型的有效性，此外还对基于 softmax 采样的效果进行谈论，明白了高效样本的重要性。
- **基于 K-Best 的指导家的探索：**在实验中我对基于 K-Best 的指导家策略进行探索，为其设定了不同的学徒期，实验结果证明该指导家能在一定程度上帮助智能体高效的探索。
- **A2C 单智能体特征选择方法的提出：**在多智能体方法基础上，我提出了一种基于优势演员-评论家（A2C）算法的单智能体特征选择方法，并在实验中证明了其有效性。

在未来的工作中，我期望在基于 pop 和 push 的单智能体特征选择方法上实现以下几点：

- **算法的进一步优化：**我们计划对现有的特征选择算法进行优化，提高其在不同类型数据集上的适用性和效率。
- **将其应用于局部特征选择工作：**局部特征选择也是数据处理的很重要的部分，我希望能够将该算法应用于局部特征选择领域，以加速局部特征选择。
- **多任务的智能体的开发：**对于局部特征选择，其每个局部都需要做出一个特征选择，我希望能够将该工作重新定义为多任务的特征选择，训练一个泛化性强的智能体，为每个局部做出特征选择。

参考文献

[1] Girish Chandrashekar and Ferat Sahin. A survey on feature selection methods. *Computers & Electrical Engineering*, 40(1):16–28, 2014.

[2] F Cruz, S Magg, Y Nagai, and S Wermter. Improving interactive reinforcement learning: What makes a good teacher? *Connection Sci.*, 30(3):306–325, 2018.

[3] F Cruz, J Twiefel, S Magg, C Weber, and S Wermter. Interactive reinforcement learning through speech guidance in a domestic scenario. In *Proceedings of the International Joint Conference on Neural Networks*, pages 1–8, 2015.

[4] AP Dempster, NM Laird, and DB Rubin. Maximum likelihood from incomplete data via the em algorithm. *J. Roy. Stat. Soc. Ser. Methodol.*, 39:1–22, 1977.

[5] SMH Fard, A Hamzeh, and S Hashemi. Using reinforcement learning to find an optimal set of features. *Comput. Math. Appl.*, 66(10):1892–1904, 2013.

[6] G. Forman. An extensive empirical study of feature selection metrics for text classification. *J. Mach. Learn. Res.*, 3(Mar.):1289–1305, 2003.

[7] F.-A. Fortin, F.-M. D. Rainville, M.-A. Gardner, M. Parizeau, and C. Gagn'e. Deap: Evolutionary algorithms made easy. *J. Mach. Learn. Res.*, 13(Jul.):2171–2175, 2012.

[8] D. Guo, H. Xiong, V. Atluri, and N. Adam. Semantic feature selection for object discovery in high-resolution remote sensing imagery. In *Proc. Pacific-Asia Conf. Knowl. Discov. Data Mining*, page 71–83, 2007.

[9] I Guyon, J Weston, S Barnhill, and V Vapnik. Gene selection for cancer classification using support vector machines. *Mach. Learn.*, 46(1-3):389–422, 2002.

[10] M. A. Hall. Feature selection for discrete and numeric class machine learning. *Dept. Comput. Sci., Univ. Waikato, Hamilton, New Zealand*, 1999.

[11] Y. Kim, W. N. Street, and F. Menczer. Feature selection in unsupervised learning via evolutionary search. In *Proc. 6th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, pages 365–369, 2000.

[12] R. Kohavi and G. H. John. Wrappers for feature subset selection. *Artif. Intell.*, 97(1/2):273–324, 1997.

[13] M Kroon and S Whiteson. Automatic feature selection for model-based reinforcement learning in factored mdps. In *Proceedings of the International Conference on Machine Learning Applications*, pages 324–330, 2009.

[14] H L Liao, Q H Wu, and L Jiang. Multi-objective optimization by reinforcement learning for power system dispatch and voltage stability. In *Innovative Smart Grid Technologies Conference, Europe*, pages 1–8, 2010.

[15] K Lin, R Zhao, Z Xu, and J Zhou. Efficient large-scale fleet management via multi-agent deep reinforcement learning. *arXiv preprint arXiv:1802.06444*, 2018.

[16] L-J Lin. Self-improving reactive agents based on reinforcement learning, planning and teaching. *Mach. Learn.*, 8(3/4):293–321, 1992.

[17] V Mnih, K Kavukcuoglu, D Silver, A Graves, I Antonoglou, D Wierstra, and M Riedmiller. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529, 2015.

[18] Volodymyr Mnih, Adria Puigdomenech Badia, Mehdi Mirza, Alex Graves, Timothy Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu. Asynchronous methods for deep reinforcement learning. In Maria Florina Balcan and Kilian Q. Weinberger, editors, *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 1928–1937, New York, New York, USA, 20–22 Jun 2016. PMLR.

[19] P. M. Narendra and K. Fukunaga. A branch and bound algorithm for feature subset selection. *IEEE Trans. Comput.*, C-26(9):917–922, 1977.

[20] P Peng, Y Wu, Y Zhu, Y Wang, and H Li. Multiagent bidirectionally-coordinated nets: Emergence of human-level coordination in learning to play starcraft combat games. *arXiv preprint arXiv:1703.10069*, 2017.

[21] Yvan Saeys, Inaki Inza, and Pedro Larranaga. A review of feature selection techniques in bioinformatics. *bioinformatics*, 23(19):2507–2517, 2007.

[22] T Schaul, J Quan, I Antonoglou, and D Silver. Prioritized experience replay. *arXiv preprint arXiv:1511.05952*, 2015.

[23] M Stankovic. Multi-agent reinforcement learning. In *Proceedings of the 13th Symposium on Neural Network Applications in Electrical Engineering*, pages 1–1, 2016.

[24] V. Sugumaran, V. Muralidharan, and K. Ramachandran. Feature selection using decision tree and classification through proximal support vector machine for fault diagnostics of roller bearing. *Mech. Syst. Signal Process.*, 21(2):930–942, 2007.

[25] Richard S Sutton and Andrew G Barto. *Reinforcement Learning: An Introduction*. MIT Press, 1998.

[26] A Tampuu, I Kovalenko, and K Tuyls. Multiagent cooperation and competition with deep reinforcement learning. *PLoS One*, 12(4):e0172395, 2017.

[27] R. Tibshirani. Regression shrinkage and selection via the lasso. *J. Roy. Stat. Soc. Ser. Methodol.*, 58:267–288, 1996.

[28] H Wei, G Zheng, H Yao, and Z Li. Intellilight: A reinforcement learning approach for intelligent traffic light control. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery Data Mining*, pages 2496–2505, 2018.

[29] J. Yang and V. Honavar. Feature subset selection using a genetic algorithm. In *Feature Extraction, Construction and Selection*, pages 117–136. Springer, 1998.

[30] Y Yang, R Luo, M Li, M Zhou, W Zhang, and J Wang. Mean field multi-agent reinforcement learning. *arXiv preprint arXiv:1802.05438*, 2018.

[31] Y. Yang and J. O. Pedersen. A comparative study on feature selection in text categorization. In *Proc. Int. Conf. Mach. Learn.*, pages 412–420, 1997.

[32] L. Yu and H. Liu. Feature selection for high-dimensional data: A fast correlation-based filter solution. In *Proc. 20th Int. Conf. Mach. Learn.*, pages 856–863, 2003.