arXiv:2303.01433v2 [cs.LG] 6 Jun 2023

机器学习　模型会学习从数据中推断出的统计规则吗?

Aaditya Naik [1] **Yinjun Wu** [1] **Mayur Naik** [1] **Eric Wong** [1]

摘要

机器学习模型可能会犯下关键错误，而这些错误很容易被海量数据掩盖。这类错误往往与基于人类直觉的规则相悖。然而，基于人类知识的规则难以扩展，甚至难以形式化。因此，我们旨在从数据中推断统计规则，并量化模型对这些规则的学习程度。我们提出了一个框架SQRL，将基于逻辑的方法与统计推断相结合，无需监督即可从模型的训练数据中推导出这些规则。我们进一步展示了如何在测试时调整模型，以减少规则违反并生成更连贯的预测。SQRL 在视觉、表格数据和语言任务的数据集上生成多达 30 万条规则。我们发现，分类、目标检测和数据填补领域的最先进模型违反了其中多达 15.8 万条规则。测试时自适应可将这些违反行为减少多达 68.7%，相对性能提升最高达 32%。SQRL 的网址为 https://github.com/DebugML/sqrl。

1. 引言

机器学习模型可能会因噪声数据、模型泛化能力欠佳和领域偏移等因素而产生各种错误。要了解模型在这些错误方面的表现，通常需要使用准确率或 F1分数等定量指标。

虽然这些指标能够从整体上评估模型的表现，但它们无法区分根本性错误和普通错误。事实上，准确率较低的模型甚至可能比准确率较高的模型做出更连贯的预测。为了进一步提升这些指标而不断扩大数据集和模型，反而会加

[1]宾夕法尼亚大学计算机与信息科学系，宾夕法尼亚州，美国。通信作者：Aaditya Naik<asnaik@seas.upenn.edu>。

论文集：40th 国际机器学习会议，美国夏威夷州檀香山。PMLR 202，2023年。版权所有 © 2023，归作者所有。

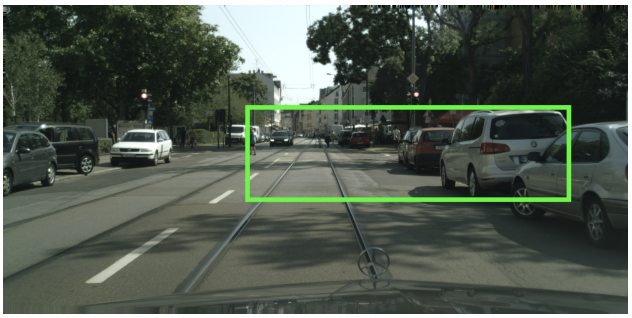


图 1：被预测为汽车的边界框展示了 EfficientPS 的一种基本错误。EfficientPS 是 Cityscapes 挑战赛中表现出色的全景分割模型。

剧这一问题。最终，最关键的错误很容易被淹没在海量数据之中。

例如，图 1 中横跨画面的绿色边界框被 EfficientPS 预测为一辆汽车。EfficientPS 是 Cityscapes数据集上表现出色的全景分割模型（Mohan 和 Valada，2021年；Cordts 等人，2016年）。这一预测属于一种根本性错误，因为它显然违背了人们对汽车形状的合理认知。这类错误甚至可能导致自动驾驶汽车的控制器突然停车，造成有害后果。

这些错误往往与基于人类直觉的规则相悖（例如，人们对于汽车典型宽度相对于身高的直觉）。因此，我们可以将估算这类错误的问题转化为寻找规则违反的问题。这些规则不仅有助于估算错误，还能通过改进模型遵循规则的预测来防止错误。

一个核心挑战在于如何大规模识别这类规则。一方面，规则必须在统计上有效且含义明确，以避免报告无意义或错误的违反情况（即假阳性）。另一方面，规则必须超越基本概念，表达数据中的复杂现象，以免遗漏值得关注的违反情况（即假阴性）。即使是人类专家，也可能无法提出或就这样一组规则达成一致，其中包括识别相关统计量、其界限以及关于这些统计量的逻辑谓词。

我们的关键洞见是，通过统计推导训练数据所遵循的规则，合成一组统计分位数

规则。我们的方法不仅能自动化指定大量规则的工作，还能确定规则的优劣。我们还实现了一个通用框架——统计分位数规则学习（SQRL），将基于逻辑的方法与统计推断相结合，无需监督即可从给定数据集中合成此类规则。我们还提出在测试时调整模型，以减少对合成规则的违反。

我们将该方法应用于来自五个不同领域的数据集和模型，以对其进行评估：心血管疾病数据集（Ulianova）上的表格分类、ImageNet（Deng 等人，2009年）上的图像分类、Cityscapes（Cordts 等人，2016年）和 KITTI（Geiger 等人，2013年）数据集上的目标检测、Physionet 数据集（Silva 等人，2012年）上的时间序列数据插补，以及 Financial PhraseBank 数据集（Malo 等人，2014年）上的情感分析。我们的框架在这些数据集上生成了约 35 条至 30 万条规则，并在模型预测中发现了约 578 条至 15.8 万条规则违反。为纠正这些规则违反，我们在测试时对模型进行调整，从而将规则违反最多减少 68.7%，并使相对性能最多提升 32%。

我们总结本研究的贡献如下：

1. 我们提出了统计分位数规则，用于估计机器学习数据集和模型中的基本错误。2. 我们开发了一个框架 SQRL，无需额外监督即可从数据中合成有效且表达力强的规则。3. 我们提出通过测试时自适应减少规则违规，从而改进模型。4. 我们将这些规则用作指标，对针对五种不同应用训练的模型进行评估。

2. 我们的方法

使用规则来刻画机器学习中的基本错误，是提升模型可信度和保真度的一种常见策略。例如，知识图谱和逻辑约束已被用于将领域知识整合到机器学习管道中。尽管这些方法在小规模且受控的场景中行之有效，但要应用于大规模数据场景仍面临诸多挑战。即使对于小型数据集，构建知识图谱也需要投入大量人力；而硬性逻辑约束对于往往充满例外情况的噪声数据集而言，可能过于简单。

因此，我们寻求一种适用于现代机器学习数据集的替代方法。为回答这一问题，我们首先确定这类规则表示的关键设计要求：

1. **有效性：** 它们必须对大多数数据有效，但要允许存在例外或噪声，
2. **表达能力：** 它们必须能够表达或捕捉复杂现象及其关系；3. **可扩展性：** 生成它们时不得要求对单条规则进行人工监督。

我们用图 1 中的示例来说明这些标准。有效性要求，正如人类直觉所示，任何能够检测出这一特定错误的规则，对于大多数其他图像也应成立，从而避免产生误报。在我们的示例中，认为边界框的形状不太可能代表汽车是合理的。表达能力则是一项相互制衡的要求，即表示必须足够灵活，不能只包含那些最基础或空洞、无法发现这些错误的规则。例如，如果仅使用汽车的形状并不足够，就应当能够结合多个属性，例如根据汽车所在位置采用不同的形状范围。最后，可扩展性意味着，对于丰富的数据集，手动编写这类规则并不可行，因为这会导致需要考虑的特征组合爆炸，还要为这些特征确定统计上有效的界限。

2.1. 统计分位数规则

我们提出基于分位数的统计推断，将其作为表示和生成此类规则的通用框架。具体而言，令 X 为某个数据点的随机变量，令 $\phi(X)$ 为 X 的某个统计量。那么，如果

$$\mathbb{P}\left(a \leq \phi(X)\right) = 1 - \delta, \tag{1}$$

对于某些 δ 和 a，我们称 $a \leq \phi(X)$ 为 $1-\delta$ 分位数规则。换言之，这意味着所有数据中有 $1-\delta$ 的数据，其统计量 $\phi(X)$ 高于 a。

例如，假设 X 是一个随机变量，表示图 1 中 Cityscapes 挑战赛里的汽车边界框。汽车的可能统计量 $\phi(X)$ 包括其宽度、身高或纵横比。要获得纵横比统计量的分位数规则，我们可以计算训练数据中每辆汽车的纵横比，并求出 $1-\delta$ 分位数，从而得到阈值 a。

分位数规则满足上述设计要求。首先，对于 $1-\delta$ 的数据，它们都是有效的，其中 δ 的比例允许存在例外。其次，分位数规则可针对任何 类型的统计量计算，而统计量可以任意复杂。虽然示例展示了针对身高和宽度等基本形状属性的分位数规则，但也可以考虑更复杂的统计量，例如光照、纹理，甚至来自其他机器学习模型的预测。最重要的是，无需人工检查给定规则的好坏。根据构造，该规则就是训练数据中某个统计量的有效 $1-\delta$ 分位数。[1] A

[1] 也可以轻松检查该分位数规则能否泛化，方法是

“好”的分位数规则会给出紧致的阈值 a，而“差”的分位数规则则会导致非常保守的阈值a。尽管过于保守的阈值可能用处不大（甚至毫无意义），但所有分位数规则都无需人工检查其正确性，因为它们在构造上就是正确的。

2.2. 分位数规则类别

通过改变统计量 ϕ，分位数规则框架能够捕捉广泛的知识。为了突出该框架的表达能力，我们讨论几类通用的分位数规则；它们可以视为本文所用的式 (1) 中原始分位数规则的变体。

双侧分位数规则。对分位数规则的一种简单推广是使用双侧分位数，从而为给定统计量求得上界和下界。具体来说，如果

$$\mathbb{P}\left(a \leq \phi(X) \leq b\right) = 1 - \delta, \tag{2}$$

那么 $\phi(X) \in [a, b]$ 是一个 $1-\delta$ 分位数规则。通常，我们可以计算 a 和 b，分别作为下界和上界分位数。例如，我们不仅可以计算纵横比的下界，还可以得到常见纵横比的完整区间。

小批量分位数规则。分位数规则的一种推广，是考虑一小批随机变量的统计量，而非仅考虑一个随机变量。具体来说，假设 $\mathbf{X} = (X_1, \ldots, X_m)$ 是一个大小为 m 的小批量。那么，如果

$$\mathbb{P}\left(a \leq \phi(\mathbf{X})\right) = 1 - \delta, \tag{3}$$

那么 $a \leq \phi(\mathbf{X})$ 是一个小批量分位数规则，其中 ϕ 可以是任何小批量统计量，例如均值或标准差。[2]

逻辑分位数规则。统计量 ϕ 可以远不止是数学公式——它也可以涉及离散的逻辑表达式。具体而言，令 ψ 为一个布尔逻辑公式，令 $\psi(\mathbf{X})$ 为在小批量$\mathbf{X}$的每个样本上求值的公式。那么，如果

$$\mathbb{P}\left(a \leq \phi(\psi(\mathbf{X}))\right) = 1 - \delta, \tag{4}$$

那么 $a \leq \phi(\psi(\mathbf{X}))$ 就是一个逻辑分位数规则。例如，ψ 可以是逻辑公式 $\text{smoke}(x) \Rightarrow \text{has cardio disease}(x)$，而 ϕ 可以是该规则的任意小批量汇总统计量，例如 F1 分数。

神经分位数规则。为突出该统计量的表达能力，我们可以考虑使用另一模型（即神经网络）统计量的分位数规则。具体而言，设 f 为一个神经模型，用于提取一些特征（例如目标检测器或属性提取器）。那么 , if

$$\mathbb{P}\left(a \leq \phi(f(X))\right) = 1 - \delta, \tag{5}$$

那么 $a \leq \phi(f(X))$ 就是一个神经分位数规则。例如，f可以是一个目标检测器，用于提取汽车、标志和行人等各种物体，而 ϕ 可以是对所得物体进行的任意统计量，例如数量或大小。

最后，我们可以结合上述各类规则中的概念，生成表达能力更强的规则。本文后续部分将展示不同的组合方式。

3. SQRL框架

对于一个数据集而言，可能存在的统计分位数规则空间是无界的。即使对规则规模设置较小的上限，具有丰富特征的大型数据集也可能诱导出海量此类规则。因此，依靠人工监督逐条手动编写并检查每条规则并不可行。

因此，我们开发了一个端到端框架——统计分位数规则学习（SQRL），用于大规模生成规则。SQRL的工作流程如图2所示。我们首先介绍它如何从数据中生成规则，然后说明如何利用生成的规则来评估并改进模型对这些规则的违反情况。

3.1. 生成统计分位数规则

我们对统计分位数规则的形式化定义旨在适用于种类丰富的机器学习数据集和模型。因此，这一定义对于任何特定场景而言都过于宽泛，我们希望建立一种机制，使领域专家能够引导规则的生成。例如，选择生成逻辑规则还是神经分位数规则，取决于采用逻辑公式上的统计量，还是神经模型的输出。同样，选择基于样本的分位数规则还是基于小批次的分位数规则，取决于个体样本或小批次的统计量哪种更合适。

统计分位数规则的强大之处可归因于两个截然不同的无界性来源：规则的结构和统计界限。考虑以下针对样本（$\mathbf{x}$，$\mathbf{y}$）的具体规则：

$$\mathbf{y} = \text{person} \Rightarrow (0.35 \leq \text{aspect_ratio}(\mathbf{x}) \leq 0.57),$$

其中，$\mathbf{x}$ 是由神经模型生成、带有单个边界框的图像，纵横比是该框的纵横比，而$\mathbf{y}$ 是对象标签。此规则展示了双侧神经分位数规则的一种条件扩展。当学习得到的界限适用于满足某个条件的样本子集时，我们会使用这种形式的规则，例如，$\mathbf{y} = \text{人}$。

就结构（例如神经模型、特征和逻辑谓词的选择）和统计界限而言，这条规则只是数量无限的可能规则之一

在验证集上检查该规则。[2]随机重新采样的小批量中某个小批量统计量的 $1-\delta$ 分位数，可以解释为经典的 $1-\delta$ 置信区间。

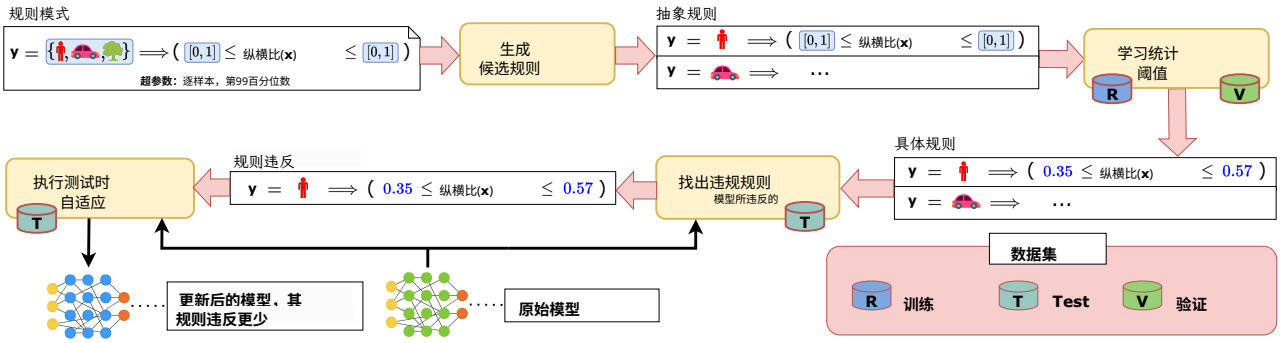


图2：我们 的 SQRL 框架工作流程：生成规则，并评估和调整模型以处理规则违反 .

（例如纵横比的区间）。

SQRL 允许用户简洁地指定一个规则模式S，以指导规则生成。规则模式充当生成规则的先验，与归纳逻辑编程中的元规则（Cropper & Muggleton, 2015）、答案集编程中的模式声明（Law 等，2020）以及逻辑程序合成中的规则模板（Raghothaman 等，2020）类似。接下来，我们希望解决的问题可以更正式地表述如下：

给定规则模式 S 和数据集 D，输出一组具体规则 R，这些规则：

1. 与 S一致：生成的规则不得包含 S中未指定的构造；2. 对 S完备：必须生成每一条与 S一致的规则；3. 相对于 D有效：每条生成的规则都必须在 D的 $1-\delta$ 上成立。

SQRL 分两个阶段解决这一问题，结合使用基于逻辑的方法和统计推断。第一阶段使用基于逻辑的方法，根据用户提供的规则模式 S枚举一组抽象规则 。第二阶段使用统计推断计算每条抽象规则的界限，从而得出一条符合 D的具体规则。

输入规则模式 S 是一组形式为 $T[\Theta]$ 的规则模板，其中 T 指定规则的结构，包括任何超参数（例如小批量大小和百分位数阈值），而 Θ 指定统计界限的占位符。第 2.2 节中定义的不同类别的规则需要不同形式的模式。我们的示例使用以下双侧分位数规则模式：

$$Y \in \{\text{car}, \text{person}, \text{rider}, \ldots\},\ \phi \in \{\text{aspect_ratio}, \ldots\},$$
$$\mathbf{y} = Y \Rightarrow \theta_{\text{lb}} \le \phi(\mathbf{x}) \le \theta_{\text{ub}},\ 1-\delta = 0.98,$$

其中，Y是对象标签集合，ϕ 是统计量集合，$\Theta = [\theta_{\text{lb}}, \theta_{\text{ub}}]$ 是给定标签对应统计量的下界和上界的占位符，其对应于 $1-\delta$ 指定的百分位数阈值。

上述模式会生成一组抽象规则：

$$\mathbf{y} = \text{car} \Rightarrow \theta_{\text{lb,car}} \le \text{aspect_ratio}(\mathbf{x}) \le \theta_{\text{ub,car}},$$
$$\mathbf{y} = \text{person} \Rightarrow \theta_{\text{lb,person}} \le \text{aspect_ratio}(\mathbf{x}) \le \theta_{\text{ub,person}},$$
$$\mathbf{y} = \text{rider} \Rightarrow \theta_{\text{lb,rider}} \le \text{aspect_ratio}(\mathbf{x}) \le \theta_{\text{ub,rider}}.$$

生成这些抽象规则的算法需要$O(mn^k)$时间，其中 m 是 Y中的标签数量， n是 ϕ中的统计量数量，而 k 是单条规则中使用的统计量最大数量。

随后，通过学习给定数据集上的第 98 百分位数界限，从每条抽象规则实例化出具体规则。在 KITTI 数据集（Geiger 等人，2013年）上生成的此类规则示例如下：

$$\mathbf{y} = \text{car} \Rightarrow 0.07 \le \text{aspect_ratio}(\mathbf{x}) \le 2.77.$$

算法 1 概述了获取每条抽象规则的此类界限的方法。该方法在从训练集中随机抽取的多个样本（或小批量）上计算每条抽象规则的统计量，并据此获取统计界限。由于这些界限在未见过的数据集上可能不成立，因此我们只保留那些在训练集与留出验证集之间统计界限保持一致的规则；一致性根据杰卡德指数判定，其定义见附录 A。对于每条规则，设 n 为每个小批量的大小。要推导该小批量的分位数界限，我们必须先对统计量排序，这需要$O(n\log(n))$的时间。我们使用 numpy.percentile 函数来学习界限本身。虽然该函数的复杂度未有说明，但假设为 $O(n)$，则为每个小批量推导界限需要 $O(n\log(n))$的时间。

3.2. 使用统计分位数规则

SQRL 生成的规则有许多潜在应用。在本文中，我们重点讨论两个重要用途：根据统计

算法 1计算抽象规则的统计量**输入**：训练集 R、验证集 V、一组抽象规则 $\mathcal{R}_{\text{abs}}$、小批量大小 B、分位数阈值 $1-\delta$、验证阈值 ϵ**输出**：一组候选规则： $\mathcal{R}_{\text{cand}}$1：从 R 中随机抽取 N_1 个大小为 B 的小批次；2：从 V 中随机抽取 N_2 个大小为 B 的小批次；3：将输出规则集初始化为空：$R_{\text{cand}} = \{\}$4：**对于**每条规则 $r \in R_{\text{abs}}$ **执行**5：收集上述每个 N_1 训练小批次上的统计量，用于 r；6：在这些样本上计算 $1-\delta$ 百分位数界限 ϕ_{train}；7：收集上述每个 N_2 验证小批次上的统计量，用于 r；8：在这些样本上计算 $1-\delta$ 百分位数界限 ϕ_{valid}；9：**如果** |杰卡德(ϕ_{train}， ϕ_{valid})| $> 1-\epsilon$ **则** 10：将 r 添加到 $\mathcal{R}_{\text{cand}}$11：**结束如果**12：**结束对于**

分位数规则违反。

为了评估规则违反情况，我们设定了一项指标，用于计算模型在测试集上的预测中每条规则被违反的次数。如果$\phi(X)$的值落在学习得到的界限之外，则视为违反了规则；其中， X 根据规则的不同，可以是单个样本，也可以是一个小批量。

我们还可以通过加入一种能够捕捉规则违反情况的辅助语义损失，尝试减少给定模型违反统计分位数规则的次数。然而，使用这种损失重新训练或微调大型模型可能并不切实际，因为我们的框架通常会生成大量规则。因此，我们转而在测试时调整模型的性能。

考虑一条形式为 $\theta_{\text{lb}} \leq \phi(X)$的统计分位数规则 r。对于模型参数 W 所决定的模型在一个样本（或一个小批次的样本）上的输出 X，我们定义如下损失：

$$l_W(r, X) = \begin{cases} 0 & \text{if } X \text{ satisfies } r \\ \min\{\theta_{\text{lb}} - \phi(X), 1\} & \text{otherwise} \end{cases} \quad (6)$$

本质上，损失是分配给满足和违反 r 的样本的零值或非零值。在直觉上，非零损失是 a 与在样本 X上评估规则 r 所得结果 ϕ（X）之间的距离。为避免损失值可能过大，损失被截断为不超过 1。其他类型规则的损失函数也以类似方式定义，见附录B。给定一组具体规则 $\mathcal{R}$，我们对所有规则的公式（6）取平均，作为测试时自适应的目标函数：

$$\mathcal{L}_W(\mathcal{R}, \mathbf{X}) = \frac{1}{|\mathcal{R}|} \sum\nolimits_{r \in \mathcal{R}} l_W(r, \mathbf{X}). \quad (7)$$

对于某些统计量 ϕ，例如 F1 分数，上述损失可能不可微。在这种情况下，我们使用它们的共轭，例如共轭F1分数（B´en´edict 等，2021年）。

算法2概述了用于减少规则违反的整体测试时自适应算法。该算法在每次迭代中计算公式（7），并对批归一化层中的所有参数执行反向传播。此过程总共重复 N 次迭代。

算法2基于规则的测试时自适应算法

输入：测试数据集 T、一组统计分位数规则 $\mathcal{R}$，以及预训练模型 M、迭代次数 N

初始化 从所有批次归一化层中收集参数，将其作为集合 W，并且只微调 W，同时保持中所有其他参数固定不变
当 迭代次数小于 N **时**
  从 T中随机抽取一个小批次$\mathbf{X}$
  使用公式（7）计算损失 $\mathcal{L}_W(\mathcal{R}, \mathbf{X})$
  如果 $\mathcal{L}_W(\mathcal{R}, \mathbf{X}) > 0$ **那么**
    对 M 执行反向传播并更新 W
  结束如果
结束循环

4. 评估

我们在用于五种不同任务的数据集和模型上评估 SQRL。我们展示了框架根据各种数据集生成的统计分位数规则的指标和定性特征。我们衡量在这些数据集上训练的现有模型在生成的规则方面的表现。我们还评估了在测试时调整模型以减少规则违反的效果。附录 E 中还讨论了生成的规则的用途及相关问题。

基准任务和模型。 我们考虑五种不同的任务：表格分类、图像分类、目标检测、时间序列插补，最后是语义分析。下面介绍这些数据集，并在附录D中进一步详细说明。

对于表格分类，我们使用心血管疾病数据集（Ulianova）中的数据训练 FT-Transformer 模型（Gorishniy 等人，2021年）。

对于图像分类，我们使用在原始 ImageNet 数据集（Deng 等人，2009年）上训练的 ResNet-34 模型，但使用 ImageNet-X（Idrissi 等人，2022年）生成规则并评估违规情况。ResNet-34 模型经过训练，可对 ImageNet-X 定义的 17 个元类别进行分类。

对于目标检测，我们考察 EfficientPS 模型中的目标检测器组件（Mohan 和 Valada，2021年）。具体而言，我们使用在 KITTI 自动驾驶数据集上预训练的 EfficientPS 模型版本



（a）时间序列插补。　　（b）情感分析。　　（c）目标检测。

图 3：时间序列插补、情感分析和目标检测任务的定性结果（表格数据和图像分类任务的结果见附录 C.2）。对于每项任务，我们展示：（a）SQRL 生成的一条规则；（b）违反该规则的模型预测；以及（c）经过测试时自适应后满足该规则的模型预测。这些规则均满足我们的三项设计要求：根据构造，它们对 98% 的数据成立，因此是有效的；它们具有足够的表达能力，能够发现模型的缺陷；并且无需人工监督每条规则，即可大规模生成。

（Geiger 等人，2013年）[3]。我们在 KITTI、Cityscapes（Cordts 等人，2016年）和 $\text{Cityscapes}_{\text{rain}}$（Tremblay 等人，2020年）的验证集划分上评估该模型；后者是经大量雨水效果增强的 Cityscapes 版本。

对于时间序列插补，我们在 Physionet 挑战赛 2012 数据集（Silva 等人，2012年）（简称 Physionet）上训练了一个最先进的时间序列插补模型 SAITS（Du 等人，2023年）。使用 Tashiro 等人（2021年）的预处理脚本后，得到的时间序列样本中约有 80% 是缺失条目。此任务的目标是对这些缺失值进行插补。

对于语义分析，我们在金融短语库数据集（Malo 等人，2014年）（简称 PhraseBank）上使用预训练的 FinBERT 模型（Araci，2019年）。此任务的目标是预测该数据集中每条关于财经新闻的陈述的情感（负面、中性或正面）。

4.1. 使用统计分位数规则评估模型

为了评估机器学习模型对统计分位数规则的捕捉能力，我们针对相应的训练数据合成了一系列规则，并统计违反这些规则的总次数。下面我们将介绍各项任务的设置，并在表 1 中总结结果。

表格分类。 不同任务所考虑的规则类型各不相同，并通过规则模式指定。对于表格分类任务，我们考虑一种双侧形式的逻辑分位数规则，其中统计量 ϕ 为该规则在一个小批量上的 F1 分数。如第 2.2 节所述，这些规则表示为基于相应数据集特征的布尔公式。

我们将心血管数据集中的各列作为生成公式所依据的特征。虽然“smoke”等列的取值为布尔值，但“age”等列并非如此。我们将这类列的取值划分为 8 个区间，从而为每个区间内的取值生成 8 个布尔值特征。这样，生成规则所依据的特征总数为 52 个。我们生成 300K 条规则，其中选取 400 条（每个类别 200 条）进行评估，方法如算法 1 所述。我们在附录 C.2 的图 4a 中展示了一条规则及其违反该规则的小批量示例。

图像分类。 与前一个任务类似，我们考虑双侧逻辑分位数规则，并将 F1 分数作为每个小批量的汇总统计量。特征取自 ImageNet-X 中布尔值标注，例如背景或姿势。由此得到 16 个特征，

[3]预训练模型下载自 https://github.com/DeepSceneSeg/EfficientPS

并据此生成规则。我们在附录 C.2 的图4b中展示了规则及其违反情况的示例。

由于我们在 ImageNet 验证集上评估规则违反情况，因此将 ImageNet-X数据集的训练集按 80%/20% 的比例划分为规则训练集和规则验证集。对于每个元类别，我们还按照算法 1 选取 20 条规则（即总计 340 条规则，约占 7.3 万条规则中的一部分）。

目标检测。鉴于此任务中预测结果的特点，我们不采用逻辑分位数规则，而是针对模型生成的边界框预测所涉及的 6 个统计量，考虑双侧形式的神经分位数规则。这些统计量包括纵横比、宽度、高度、面积、边界框中心的 x坐标，以及边界框底部的 y坐标。每个统计量均针对单个边界框定义。我们也会成对考虑统计量。对于每个类别，我们考虑所有统计量对（s_1，s_2）。对于每一对，我们按 s_1 将该类别的物体分为四个区间。在每个区间内，我们学习统计量 s_2 的第 98 百分位数界限。由此生成的规则如图 3c(a) 所示，其中 s_1 是纵横比，s_2 是每个边界框的宽度。图 3c(b) 展示了违反该规则的情况。我们使用这 6 个统计量，为目标检测器训练来预测的 7 个类别学习了 252 条规则。在所有情况下，这些规则的第 98 百分位数界限均通过KITTI训练数据集学习得到。

时间序列插补。 插补模型会输出对缺失条目的估计值。因此，对于这项任务，我们将数据集中的特征视为统计量集合。对于每个特征，我们针对插补值采用双侧形式的神经分位数规则 。如附录D所述，Physionet 数据集中有 35 个特征，因此生成了 35 条逻辑规则。对于每条规则，均使用 Physionet 数据集训练划分中非缺失值的统计量，学习其第 98 百分位数界限。图 3a(a)展示了一条生成规则的示例。该规则给出了“格拉斯哥昏迷评分”特征的学习界限，这也与医学领域普遍认可的该特征取值范围一致（Teasdale & Jennett, 1974年）。图3a(b)展示了一个违反此规则的插补值示例，该值超出了上界。

语义分析。 与前两个任务类似，我们针对 PhraseBank数据集中的每个句子考虑双侧神经分位数规则。对于每个句子，我们使用预训练模型提取特征，以构成生成规则所需的统计量。我们使用预训练的 DistilRoBERTa 模型（Hartmann，2022年）计算 7 种基本情绪中每一种的可能性，并使用在 11,267 条推文上预训练的 RoBERTa 模型（Antypas等人，2022年）计算 19 个 Twitter主题中每一个的可能性对于每个句子。这为每个句子生成了 26 个特征。与目标检测任务类似，对于每个类别，我们针对每个特征以及每对特征生成规则，总计生成 7.8K 条规则。随后，我们根据算法 1 选取 158 条规则。图 3b(a) 和 3b(b) 分别展示了一条规则及其违反情况示例。

结果。 表 1显示了规则违反的总数以及每个样本的规则违反数。对于时间序列插补任务，尽管规则数量较少（35 条），规则违反数仍然很高（最多约 158K）。这表明最先进的时间序列插补模型仍远非完美。此外，图像分类和表格分类任务中，每个样本违反的规则也更多。请注意，表格分类和图像分类任务中的规则主要用于衡量一批预测与训练数据的一致性。表格分类中每个类别有 200 条规则用于衡量这种一致性，而图像分类中每个类别只有 20 条，因此分类错误可能会导致前一项任务中出现更多规则违反。

另一方面，在目标检测和情感分析任务中，每个样本的违规情况较少。目标检测器的预测结果仅违反了少数生成的规则。表 1 展示了 Cityscapes-rainy 数据集上的违规结果，KITTI 和 Cityscapes 上的结果见附录 C.2。

4.2. 使用统计分位数规则适应模型

接下来，我们考虑减少表 1 所示规则在测试时的违规次数。为此，我们采用第 3 节介绍的规则驱动测试时自适应技术。我们将 SQRL 与最先进的测试时自适应方法进行比较，并将其作为基线方法。

基线方法。 第 5 节综述的大多数测试时自适应方法与我们的工作正交。因此，我们只与以下最相关的基线方法进行比较：

- **批归一化**：批归一化（BN）（Ioffe 和 Szegedy, 2015）在测试时计算批归一化层的统计量，而不是在训练阶段重用已学习的统计量。
- **基于熵最小化的方法**：Tent（Wang 等，2020年）通过最小化模型预测的熵进行优化，但仅关注分类问题。因此，我们遵循（Goyal 等人）的分析，在目标检测任务的回归问题中最小化模型输出的负 L2 范数。
- **基于伪标签的方法**：鲁棒伪标签（RPL）（Rusak 等人，2021）和共轭伪标签（CPL）（Goyal 等人）生成伪标签，作为测试时自适应的监督信号。

Task	表格数据 分类	图像 分类	目标检测	时间序列 插补	情感 分析
模型	FT-Transformer	ResNet-34	EfficientPS	SAITS	FinBERT
规则类别	逻辑	逻辑	神经网络	神经网络	逻辑
样本量	4096	256	1	1	128
数据集 **R**	心血管$_{\text{train}}$	ImageNet$_{\text{rule-train}}$	KITTI$_{\text{train}}$	Physionet$_{\text{train}}$	PhraseBank$_{\text{train}}$
数据集 **V**	心血管$_{\text{valid}}$	ImageNet$_{\text{rule-valid}}$	-	Physionet$_{\text{valid}}$	PhraseBank$_{\text{valid}}$
数据集 **T**	心血管$_{\text{test}}$	ImageNet$_{\text{valid}}$	Cityscapes$_{\text{rainy, valid}}$	Physionet$_{\text{test}}$	PhraseBank$_{\text{test}}$
# 规则总数	292,129	73,032	252	35	7878
# 选定规则	400	340	252	35	158
# 违规总数	26083±56	7716±318	8,108	157772±18697.80	578
# 违规规则 每个样本	389.29±0.83	42.62±1.76	16.21	197.46±23.40	0.59

表 1：生成规则并据此评估模型的结果。针对每个训练数据集，使用指定的样本量生成规则。样本量大于 1 时，我们从数据集中随机抽取一个批次。使用验证数据集从生成的规则中抽取一个子集进行评估。我们统计模型在每个测试样本上违反的选定规则数量。最后两行分别展示了所有测试样本中这些数量的总和与平均值。

Task	表格数据 分类		图像 分类		物体 检测		时间序列 插补		情感 分析	
数据集	心血管		ImageNet-X		Cityscapes-rainy		Physionet		PhraseBank	
指标	AUC	减少百分比	准确率	减少百分比	mAP	减少百分比	MSE ($\times 10^{-3}$)	减少百分比	准确率	减少百分比
不适应	80.10	-	80.47	-	25.49	-	7.7	-	84.02	-
BN	80.10	0.0%	80.47	2.9±3.4%	25.49	-0.0±0.0%	7.7	0.0±0.0%	84.02	0.0±0.0%
Tent	80.10	-0.1±0.0%	**80.51**	0.7±5.3%	25.52	-1.4±0.8%	8.0	-0.2±0.9%	**86.08**	14.7±0.0%
RPL	80.09	-0.1±0.0%	80.48	-0.8±6.5%	25.70	0.2±0.3%	7.4	18.7±16.2%	84.85	7.3±0.2%
CPL	80.10	-0.1±0.0%	80.44	-0.1±6.0%	24.95	-0.1±0.1%	5.6	12.7±6.5%	85.77	**15.7±0.1%**
MEMO	80.10	1.8±0.0%	80.47	11.6±4.1%	-	-	-	-	**86.08**	15.0±0.2%
Ours	**80.12**	**38.1±0.0%**	**80.51**	**22.2±4.5%**	**27.01**	**31.4±0.2%**	**5.2**	**68.7±8.1%**	**86.08**	**15.7±0.1%**

表 2：Res 使用不同测试时自适应方法减少规则违反的结果，涵盖我们的三个任务 .

• **基于数据增强的方法**：MEMO（Zhang 等人，2021）会为单个测试样本生成多个增强版本，并最小化模型在这些样本上的输出熵。由于它主要针对分类模型，将其推广到目标检测模型和时间序列插补模型需要付出相当大的努力。因此，我们不比较 MEMO 与 SQRL 在目标检测和时间序列插补任务上的表现。

对于基线方法，我们遵循默认设置，仅进行少数轮次的测试时自适应，因为轮次过多可能导致过拟合。对于 SQRL，我们最多使用 60 轮次。我们遵循测试时自适应的默认设置，只微调批归一化层的统计量，而不是整个模型。我们还在附录 C.1 中进行了一项消融研究，以分析使用 SQRL 微调整个模型的效果。

结果。评估结果见表 2。我们报告所有方法的两个指标：模型性能以及适应后相对于表 1 中报告的违规总数而言，规则违反减少的百分比（“减少百分比”）。对于模型性能指标，我们报告表格分类的 AUC 分数（因为这是一项二元分类任务）、图像分类的预测准确率、目标检测的平均精度均值（mAP）、时间序列插补的均方误差（MSE），以及情感分析的准确率。SQRL 能够在测试时显著减少规则违反的数量（在 Physionet 数据集中最多减少 68.7%），这也能使模型在 Physionet 数据集上的相对性能提升最多达 32%（将 MSE 从 7.7×10^{-3} 降至 5.2×10^{-3}），并使其在其他数据集上的性能略有提升。图 3 展示了时间序列插补、情感分析和目标检测的示例；在每个子图中，(a) 所示规则遭到违反时产生的错误预测显示于 (b)，并在测试时自适应后于 (c) 得到纠正。附录 C.2 的图4 中还提供了更多表格分类和图像分类任务的示例。例如，

如图 3a(c) 所示，测试时自适应后，SQRL 能够使 1_{st} 小时处的插补值（黑点）更接近真实值（蓝点）。相比之下，基线方法对规则违反的影响微乎其微，且产生的模型表现略逊于 SQRL。

5. 相关工作

测试时自适应。测试时自适应旨在分布发生偏移时使模型适应新的数据分布，这通常假设无法访问源数据或目标分布数据的真实标签。这些方法包括最小化模型输出的熵（Wang 等，2020；Goyal 等人）、生成伪标签以在测试时进行监督训练（Goyal 等人；Rusak 等人，2021）或自监督（Zhang 等人；Chen 等人，2022）；强制模型输出在同一个测试样本的不同数据增强下保持一致（Zhang 等人，2021）；构建生成模型以生成带标签的测试样本（Li 等人，2020）；等等。据我们所知，现有的测试时自适应方法中没有一种会考虑强制模型输出与统计分位数规则保持一致。此外，这些技术中的大多数采用了与我们正交的策略，例如元学习（Xiao 等人，2021）或生成模型（Li 等人，2020）。此外，它们中的大多数仅限于分类任务，而我们的方法适用于通用任务，包括分类和回归任务。

将规则注入机器学习模型。人们已尝试多种方法，将规则或领域知识有效地注入神经网络。一种广泛采用的策略是添加一项，用于编码违反逻辑规则的情况，从而对目标函数进行正则化。例如，Hu 等人（2016年）设计了一个教师-学生框架，以联合利用带标签样本和逻辑规则进行学习；Seo 等人（2021年）联合学习逻辑规则和训练数据的嵌入；Ganchev 等人（2010年）则利用数据约束对模型后验进行正则化。该领域的其他一些解决方案包括通过对抗学习惩罚偏置规则（Zhang 等人，2018年），以及在训练过程中将逻辑规则视为约束来求解约束优化问题（Fioretto 等人，2021年；Narasimhan，2018年）。然而，这些方法都要求将人工指定的规则作为输入，而我们的方法能够自动学习符号规则及其统计界限。此外，将规则整合进机器学习模型的最先进解决方案主要着眼于增强监督学习，而我们的方法能够处理更具挑战性的场景，即测试时自适应。如上所述，测试时自适应期间没有来自数据的监督，因此无法利用规则对模型训练进行正则化。

规则学习。从数据中综合一阶逻辑规则的问题，已在归纳逻辑编程领域得到广泛研究。ILASP（Law 等，2020）和 Prosynth（Raghothaman 等，2020）等技术利用纯符号推理，在给定数据库中搜索能够产生预期答案的逻辑规则。NeuralLP（Yang 等，2017）和 NLIL（Yang 和 Song，2019）等工作则尝试利用神经网络来引导对可行逻辑规则的搜索。NLIL 还可以为每条综合出的规则学习一种简单的统计量，即置信度分数。然而，与我们的框架所捕获的任意统计量相比，它的表达能力较弱。此外，置信度分数是在整个数据集上计算的，而不是在数据的小批次上计算的。因此，我们无法检查它们在数据不同部分之间是否一致。

基于逻辑规则的弱监督学习。与测试时自适应类似，在不存在真实标签时，也可以采用弱监督学习。一种弱监督策略是使用逻辑规则为未标注样本添加标签，然后再进行常规监督学习（Ratner 等，2016；2017）。然而，据我们所知，最先进的基于规则的弱监督学习方法只能处理分类任务。相比之下，我们的方法适用于非常广泛的机器学习场景，包括分类和回归任务。此外，尽管 Varma 和 R´e（2018）等研究能够自动推导出逻辑规则作为弱监督信号，但它们仅限于对符号规则进行推理。因此，这些方法不适用于需要统计量的场景。

6. 结论与未来工作

我们将统计分位数规则形式化，用以刻画与训练数据不一致的基本错误，并定义了大规模提取此类规则的问题。我们提出了 SQRL，这是一个通用框架，可针对给定数据集生成大量此类规则，并评估模型对这些规则的违反情况。通过广泛的实证研究，我们发现，机器学习模型并不总能学习从数据中推断出的统计规则，但可以利用基于规则的测试时自适应来纠正这些规则违反。未来，我们计划在现代模型上更广泛地评估统计分位数规则。我们还计划探索这些规则的更多用途，例如寻找更有效的方法来训练模型以减少规则违反，并将这些规则用于无监督学习。

致谢

我们感谢审稿人提出的意见。本研究部分由美国国家科学基金会资助，项目编号为 #1836936 和 #2107429。

参考文献

Antypas, D., Ushio, A., Camacho-Collados, J., Neves, L., Silva, V. 和 Barbieri, F. Twitter主题分类。载于第*29*届国际计算语言学会议论文集，大韩民国庆州，2022年10月。国际计算语言学委员会。Araci, D. Finbert：基于预训练语言模型的金融情感分析。*arXiv*预印本 *arXiv:1908.10063*，2019年。B´en´edict, G., Koops, V., Odijk, D. 和 de Rijke, M. sigmoidf1：一种用于多标签分类的平滑F1分数替代损失。*arXiv*预印本 *arXiv:2108.10566*，2021年。Chen, D., Wang, D., Darrell, T. 和 Ebrahimi, S. 对比式测试时自适应。载于*IEEE/CVF*计算机视觉与模式识别会议论文集，第295–305页，2022年。Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S. 和 Schiele, B. 用于语义城市场景理解的Cityscapes数据集。载于*IEEE*计算机视觉与模式识别会议论文集，第3213–3223页，2016年。Cropper, A. 和 Muggleton, S. 元解释学习中的元规则逻辑最小化。载于归纳逻辑编程，2015年。Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K. 和 Fei-Fei, L. ImageNet：大规模分层图像数据库。载于*2009*年*IEEE*计算机视觉与模式识别会议，第248–255页。IEEE，2009年。Du, W., Cˆot´e, D. 和 Liu, Y. SAITS：用于时间序列的基于自注意力的插补。*ExpertSystems withApplica-tions*，219:119619，2023年。Fioretto, F., Van Hentenryck, P., Mak, T. W., Tran, C., Baldo, F. 和 Lombardi, M. 约束深度学习中的拉格朗日对偶。载于机器学习与数据库知识发现。应用数据科学与演示专题：欧洲会议，*ECML PKDD 2020*，比利时根特，*2020*年*9*月*14*日至*18*日，会议论文集，第五部分，第118–135页。施普林格，2021年。Ganchev, K., Grac¸a, J., Gillenwater, J. 和 Taskar, B. 结构化潜变量模型的后验正则化。《机器学习研究杂志》，11:2001–2049，2010年。Geiger, A., Lenz, P., Stiller, C. 和 Urtasun, R. 视觉与机器人技术相遇：KITTI数据集。《国际机器人研究杂志》，32(11):1231–1237，2013年。Gorishniy, Y., Rubachev, I., Khrulkov, V., and Babenko, A. 重新审视表格数据的深度学习模型。神经信息处理系统进展，34：18932–18943，2021年。Goyal, S., Sun, M., Raghunathan, A., and Kolter, J. Z. 通过共轭伪标签实现测试时自适应。载于神经信息处理系统进展。Hartmann, J. emotion-english-distilroberta-base。https://huggingface.co/j-hartmann/emotion-english-distilroberta-base/，2022年。He, K., Zhang, X., Ren, S., and Sun, J. 用于图像识别的深度残差学习。载于*IEEE*计算机视觉与模式识别会议论文集，第770–778页，2016年。Hu, Z., Ma, X., Liu, Z., Hovy, E., and Xing, E. 利用逻辑规则增强深度神经网络。载于计算语言学协会第*54*届年会论文集（第*1*卷：长篇论文），第2410–2420页，2016年。Idrissi, B. Y., Bouchacourt, D., Balestriero, R., Evtimov, I., Hazirbas, C., Ballas, N., Vincent, P., Drozdzal, M., Lopez-Paz, D., and Ibrahim, M. ImageNet-X：通过变化因素标注理解模型错误。*arXiv*预印本 *arXiv:2211.01866*，2022年。Ioffe, S. and Szegedy, C. 批归一化：通过减少内部协变量偏移加速深度网络训练。载于国际机器学习会议，第448–456页。PMLR，2015年。Law, M., Russo, A., and Broda, K. 用于归纳学习答案集程序的ILASP系统。*arXiv*预印本 *arXiv:2005.00904*，2020年。Li, R., Jiao, Q., Cao, W., Wong, H.-S., and Wu, S. 模型自适应：无需源数据的无监督领域自适应。载于*IEEE/CVF*计算机视觉与模式识别会议论文集，第9641–9650页，2020年。Malo, P., Sinha, A., Korhonen, P., Wallenius, J., and Takala, P. 好债还是坏债：检测经济文本中的语义倾向。《信息科学与技术协会期刊》，65(4)：782–796，2014年。Mohan, R. and Valada, A. EfficientPS：高效全景分割。《国际计算机视觉杂志》，129(5)：1551–1579，2021年。Narasimhan, H. 使用复杂损失函数与约束进行学习。载于国际人工智能与统计会议，第1646–1654页。PMLR，2018年。

Raghothaman, M., Mendelson, J., Zhao, D., Naik, M., and Scholz, B. 溯源引导的Datalog程序合成。载于*ACM*编程语言原理研讨会论文集（*POPL*），2020年。

Ratner, A., Bach, S. H., Ehrenberg, H., Fries, J., Wu, S., and R´e, C. Snorkel：利用弱监督快速创建训练数据。载于超大型数据库国际会议论文集（*VLDB* 基金会），第 11 卷，第 269 页。美国国立卫生研究院公共访问计划，2017年。

Ratner, A. J., De Sa, C. M., Wu, S., Selsam, D., and R´e, C. 数据编程：快速创建大型训练集。神经信息处理系统进展，29，2016年。

Rusak, E., Schneider, S., Pachitariu, G., Eck, L., Gehler, P. V., Bringmann, O., Brendel, W., 和 Bethge, M. 如果数据分布发生偏移，请使用自学习。机器学习研究汇刊，2021年。Seo, S., Arik, S., Yoon, J., Zhang, X., Sohn, K., 和 Pfister, T. 使用规则表示控制神经网络。神经信息处理系统进展，34：11196–11207，2021年。Silva, I., Moody, G., Scott, D. J., Celi, L. A., 和 Mark, R. G. 预测重症监护病房患者的院内死亡率：Physionet／2012年心脏病学计算挑战赛。载于*2012*年心脏病学计算，第245–248页。IEEE，2012年。Tashiro, Y., Song, J., Song, Y., 和 Ermon, S. CSDI：用于概率时间序列插补的条件分数扩散模型。神经信息处理系统进展，34：24804–24816，2021年。Teasdale, G. 和 Jennett, B. 昏迷和意识障碍评估：实用量表。《柳叶刀》，304（7872）：81–84，1974年。Tremblay, M., Halder, S. S., de Charette, R., 和 Lalonde, J.- F. 用于评估并提升对恶劣天气鲁棒性的雨景渲染。《国际计算机视觉杂志》，2020年。Ulianova, S. 心血管疾病数据集。https: // www.kaggle.com/datasets/sulianova/cardiovascular-disease-dataset。Varma, P. 和 R´e, C. Snuba：自动化弱监督以标注训练数据。载于超大型数据库国际会议论文集，第12卷，第223页。美国国立卫生研究院公共访问计划，2018年。Wang, D., Shelhamer, E., Liu, S., Olshausen, B., 和 Darrell, T. Tent：通过熵最小化实现完全测试时自适应。载于国际学习表征会议，2020年。

Xiao, Z., Zhen, X., Shao, L., 和 Snoek, C. G. 在单个测试样本上学习跨域泛化。载于国际学习表征会议，2021年。

Yang, F., Yang, Z., and Cohen, W. W. 用于知识库推理的逻辑规则可微学习。神经信息处理系统进展，30，2017年。

Yang, Y. and Song, L. 通过神经逻辑归纳学习高效学习解释。*arXiv* 预印本 *arXiv:1910.02481*，2019年。

Zhang, B. H., Lemoine, B. 和 Mitchell, M. 使用对抗学习缓解不受欢迎的偏见。载于*2018*年*AAAI/ACM*人工智能、伦理与社会会议论文集，第 335–340 页，2018年。Zhang, M. M., Levine, S. 和 Finn, C. MEMO：通过适应和增强实现测试时鲁棒性。载于*NeurIPS 2021*年分布偏移研讨会：连接方法与应用，2021年。Zhang, Z., Chen, W., Cheng, H., Li, Z., Li, S., Lin, L. 和 Li, G. 划分与对比：通过自适应对比学习实现无源域适应。载于神经信息处理系统进展。

A. 杰卡德指数的补充细节

如第3.1节所述，杰卡德指数用于从训练集和验证集中选取一致的统计规则。为说明如何计算杰卡德指数，我们重新看一下第3.1节中的示例，其中定义了以下模式：

$$
\begin{aligned}
&Y \in \{\mathsf{car},\ \mathsf{person},\ \mathsf{rider},\ \ldots\},\\
&\phi \in \{\mathsf{aspect_ratio},\ \ldots\},\quad \gamma = 0.98,\\
&\mathbf{y} = Y \Rightarrow \theta_{\mathsf{lb}} \leq \phi(\mathbf{x}) \leq \theta_{\mathsf{ub}},
\end{aligned}
$$

该方法可在训练集和验证集上分别进行评估，从而得到以下在训练集和验证集上的双侧分位数规则实例：

$$
\begin{aligned}
&\mathbf{y} = Y \Rightarrow \theta_{\mathsf{lb,train}} \leq \phi_{\mathsf{train}}(\mathbf{x}) \leq \theta_{\mathsf{ub,train}}, \text{ for training set,}\\
&\mathbf{y} = Y \Rightarrow \theta_{\mathsf{lb,valid}} \leq \phi_{\mathsf{valid}}(\mathbf{x}) \leq \theta_{\mathsf{ub,valid}}, \text{ for validation set.}
\end{aligned}
$$

因此，上述两条规则之间的杰卡德指数可表示如下：

$$
\begin{aligned}
\text{Jaccard}(\phi_{\mathsf{train}}, \phi_{\mathsf{valid}}) &= \frac{[\theta_{\mathsf{lb,train}}, \theta_{\mathsf{ub,train}}] \bigcap [\theta_{\mathsf{lb,valid}}, \theta_{\mathsf{ub,valid}}]}{[\theta_{\mathsf{lb,train}}, \theta_{\mathsf{ub,train}}] \bigcup [\theta_{\mathsf{lb,valid}}, \theta_{\mathsf{ub,valid}}]}\\
&= \frac{\min\{\theta_{\mathsf{ub,train}}, \theta_{\mathsf{ub,valid}}\} - \max\{\theta_{\mathsf{lb,train}}, \theta_{\mathsf{lb,valid}}\}}{\max\{\theta_{\mathsf{ub,train}}, \theta_{\mathsf{ub,valid}}\} - \min\{\theta_{\mathsf{lb,train}}, \theta_{\mathsf{lb,valid}}\}}.
\end{aligned}
$$

对于单侧分位数规则，即 $\theta_{\mathsf{lb}} = -\infty$ 或 $\theta_{\mathsf{ub}} = \infty$，我们取由整个数据集计算得出的统计量$\phi(\mathbf{x})$的下界或上界，以替代 θ_{lb} 或 θ_{ub}。

B. 测试时自适应损失函数

回顾公式6中提到的损失函数背后的直觉：对于参数为 W、输出为 X 的模型，其规则 r 的损失 $l_W(r, X)$在 X 满足 r时为零，否则非零。非零值的大小取决于分位数规则的类别。公式6展示了基本形式的分位数规则对应的损失。现在，我们为本文中使用的分位数规则的若干扩展定义测试时自适应所用的损失函数。

1. **双侧分位数规则**。我们将（6）中的定义扩展至双侧分位数规则。假设有一个双侧分位数规则 $r := \theta_{\mathsf{lb}} \leq \phi(X) \leq \theta_{\mathsf{ub}}$。我们将此类规则的损失定义如下：

$$
l(r; X) = \begin{cases} 0 & X \text{satisfies } r \\ \min\{(\theta_{\mathsf{lb}} - \phi(X))(\theta_{\mathsf{ub}} - \phi(X)), 1\} & \text{otherwise} \end{cases} \tag{8}
$$

请注意，非零损失是 $\phi(X)$的二次函数，其上界和下界为方程的根，但 $\phi(X)$的输出会截断至1。这确保当 $\phi(X)$的值小于 θ_{lb} 或大于 θ_{ub}时，违反规则会产生正的非零损失。

2. **逻辑分位数规则**。假设有一个双侧逻辑分位数规则 $r := \theta_{\mathsf{lb}} \leq \phi(\psi(X)) \leq \theta_{\mathsf{ub}}$，其中 X 是模型在一批样本上的输出，该模型的参数为 W。此类规则的损失取决于统计量 ϕ，因为它是一个汇总 $\psi(X)$结果的统计量。在我们的实验中，我们使用 F1 分数作为该统计量。然而，正如第3.2节所述，F1 分数不可微，因此我们改用 Benedict 等人（2021）定义的替代F1损失。将该损失函数记为 L_{F1}。于是，我们将逻辑分位数规则的损失定义如下：

$$
l(r; X) = \begin{cases} 0 & X \text{satisfies } r \\ \min\{(\theta_{\mathsf{lb}} - L_{\mathsf{F1}}(\psi(X)))(\theta_{\mathsf{ub}} - L_{\mathsf{F1}}(\psi(X))), 1\} & \text{otherwise} \end{cases} \tag{9}
$$

	表格分类		图像分类		目标检测 (Cityscapes-rainy)	
	AUC 分数	减少百分比	准确率	减少百分比	mAP	减少百分比
我们的方法（全模型微调）	71.17	30.61±3.5%	80.40	23.28±1.40	26.87	31.3±0.5%
我们的方法	80.12	38.1±0.0%	80.51	22.17±4.5%	27.01	31.4±0.2%

表3：对整个模型与批归一化层进行微调的结果对比

Task	**目标检测**	
模型	EfficientPS	EfficientPS
规则类别	神经网络	神经网络
样本量	1	1
数据集 **R**	$\text{KITTI}_{\text{train}}$	$\text{KITTI}_{\text{train}}$
数据集 **V**	-	-
数据集 **T**	$\text{KITTI}_{\text{valid}}$	$\text{Cityscapes}_{\text{valid}}$
# 规则总数	252	252
# 选定规则	252	252
# 违规总数	1,561	7,673
# 违规规则 每个样本	7.80	15.34

表4：ge 的结果　　在 KITTI 和 Cityscapes 数据集上生成规则，并据此评估模型　　ly.

C. 补充实验

C.1. 消融研究

请注意，在实验中，在基于规则的测试时自适应阶段，我们仅微调模型的批归一化层。许多最先进的测试时自适应方法都研究了微调整个模型与微调批归一化层之间的比较，例如（Wang 等，2020）。因此，我们还针对表格分类任务、图像分类任务以及 Cityscapes-rainy 数据集上的目标检测任务进行了这项消融研究。

结果见表3。该表清楚地表明，在基于规则的测试时自适应阶段微调整个模型并不理想，因为这可能会使模型性能显著下降，最高可达9%。因此，与先前的测试时自适应方法相同，对于基于规则的测试时自适应，微调批归一化层而非整个模型是合理的。

C.2. 更多定量结果

在表4中，我们分别展示了在 KITTI 和 Cityscapes 数据集上生成和评估规则的结果。将 Cityscapes-rainy 数据集的结果与上述结果进行比较，我们可以观察到，在诸如 Cityscapes 这类分布不同的数据集上，规则违反会增加。这表明，在可能属于分布外的数据集上，模型预测与训练数据的一致性会变差。在表5中，我们展示了在 KITTI 和 Cityscapes 数据集上执行测试时自适应后的结果，这仍然表明，我们的方法能够比其他方法减少更多规则违反，同时不会损害模型性能。

C.3. 其他定性结果

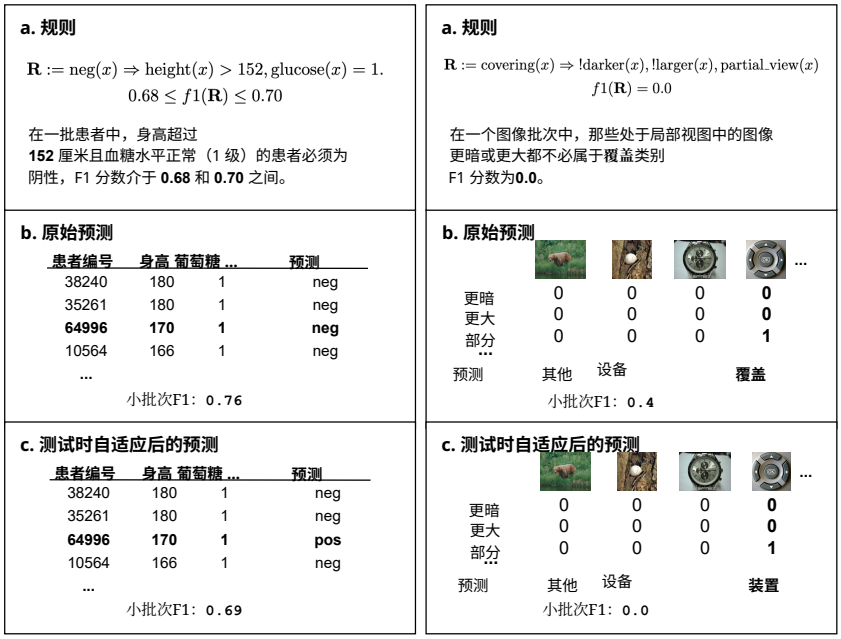
在图4 中，我们展示了表格分类和图像分类任务的定性结果　　s.

D. 基准任务和模型

我们考虑三种不同的任务：表格分类、图像分类和物体检测　　n.

Task	目标检测			
数据集	KITTI		Cityscapes	
指标	平均精度均值	减少百分比	平均精度均值	减少百分比
不适应	44.23	-	56.15	-
BN	44.23	0.0±0.0%	56.15	0.0±0.0%
Tent	44.84	0.2±0.2%	56.24	0.0±0.0%
RPL	44.26	0.4±0.1%	56.67	-0.7±0.2%
CPL	44.62	-3.7±0.0%	57.13	0.2±0.1%
MEMO	-	-	-	-
Ours	44.27	0.1±0.2%	56.91	6.1±1.2%

表5：结果 使用不同的测试时自适应方法减少我们三个任务中的规则违反的结果 .



（a）表格分类任务。 （b）图像分类任务。

图4：表格分类和图像分类任务的其他定性结果。对于三个任务中的每一个，我们展示了：（a）由 SQRL 生成的一条规则；（b）违反该规则的模型预测；以及（c）测试时自适应后，模型在该样本上满足规则的预测。这些规则均满足我们的三个设计要求：由于其构造方式，它们对98%的数据成立，因此是有效的；它们具有足够的表达能力，能够发现模型中的错误；并且无需对单条规则进行人工监督，即可大规模生成。

表格分类。 这项任务涉及对心血管疾病数据集中的表格数据进行分类。该数据集包含约 7 万名患者的信息，以 11 个属性的形式记录（6 个数值属性和 5 个离散属性），这些信息是在体检时收集的；此外，还有一个二元标签，用于指示患者是否患有心血管疾病。我们按 65%/15%/20% 的比例划分数据集，分别用于训练、验证和测试。我们在训练集上训练 FT-Transformer（Gorishniy 等人，2021年），这是一种先进的表格分类模型。随后，在测试时自适应阶段，我们将该模型适配到测试数据集。为学习和评估规则，我们分别使用 67 个训练样本、22 个验证样本和 6 个测试样本，每个样本集的大小均为 4096。

图像分类。 我们使用 ImageNet-X（Idrissi 等人，2022年）来定义这项任务，它是 ImageNet 数据集（Deng 等人，2009年）的一个变体。该数据集包含 ImageNet 数据集中的 1.2 万个训练样本子集以及全部验证样本。ImageNet-X数据集中的每张图像还附有 16 项二元人工标注，用于标示图像中物体的特征，例如物体的姿态或整张图像的光照情况，据此我们可以定义逻辑规则。我们使用在原始 ImageNet 数据集训练数据上训练的 ResNet-34 模型（He 等人，2016年），预测（Idrissi 等人，2022年）定义的 17 个元类别。我们在该数据集的验证样本上进行测试时自适应。为了学习和评估规则，我们分别使用 181 个训练样本、79 个验证样本和 9 个测试样本，每个样本的大小均为 256。

目标检测。 这项任务涉及在 Cityscapes（Cordts 等人，2016年）和 KITTI 数据集（Geiger 等人，2013年）上进行自动驾驶应用中的目标检测。我们首先考虑不同的场景。首先，我们将预先在 KITTI 数据集（Geiger 等人，2013年）训练数据上训练的 EfficientPS 模型（Mohan 和 Valada，2021年）[4] 适应到该数据集的验证样本。此外，我们还在分布偏移场景下评估我们的方法：将预训练的 EfficientPS 模型适应到两个版本的 Cityscapes 数据集（Cordts 等人，2016年），一个包含正常场景，另一个则加入了强降雨（Tremblay 等人，2020年）。请注意，EfficientPS 是一种全景分割模型，可处理目标检测和语义分割等多项任务。我们仅评估该模型的目标检测组件。训练和测试时均使用单个样本。为学习规则，我们使用 855 个训练样本。在 KITTI 基准上测试时使用 200 个测试样本，在 Cityscapes 和 Cityscapes^{rainy}上测试时则分别使用 500 个样本。

时间序列插补。 通过复用（Tashiro 等人，2021年）的预处理脚本，我们获得了 4000 个时间序列样本，每个样本都包含一名患者在重症监护室住院 48 小时期间每小时采集的 35 个特征。我们将这些时间序列样本随机划分为训练集、验证集和测试集，比例为 70:10:20。为评估插补性能，我们将测试集划分中 40% 的非缺失观测值输入插补模型，由模型输出其余 60% 非缺失观测值的插补值。随后，我们将插补值与这 60% 的真实值进行比较，以评估模型性能。

情感分析。 此任务旨在预测 PhraseBank数据集（Malo 等人，2014年）中每个句子的情感。该数据集包含 3487 个训练集句子、387 个验证集句子和 969 个测试集句子。请注意，与统计量由目标检测模型本身收集的目标检测任务不同，情感分析任务所用的统计量是通过使用其他预训练模型获得的。因此，为了保证损失可微，我们考虑以下形式的神经分位数规则：

$$Y \in \{\text{正面，负面，中性}\},\ \ \phi_1,\ \phi_2 \in \{\text{健身与健康，新闻与社会关注}\ \ldots\},\ \theta_{\text{lb}} \leq \phi_1(\mathbf{x}) \leq \theta_{\text{ub}} \Rightarrow \mathbf{y} = Y,\ \ 1-\delta = 0.98,\ \text{或}\theta_{\text{lb},\,1} \leq \phi_1(\mathbf{x}) \leq \theta_{\text{ub},\,1},\ \theta_{\text{lb},\,2} \leq \phi_2(\mathbf{x}) \leq \theta_{\text{ub},\,2} \Rightarrow \mathbf{y} = Y,\ 1-\delta = 0.98,$$

其中，健身与健康、新闻与社会关注是从预训练模型中提取的特征。

E. 讨论

E.1. 捕捉统计相关性。

通过构建统计分位数规则，并以可扩展的方式生成这些规则，生成的规则中有些可能会捕捉到虚假相关性。然而，由于这些规则在98%以上的数据中均有效，因此只有那些在98%的数据中同样有效的虚假相关性才会被捕捉到。实践中，许多虚假相关性并不会在如此大比例的数据中持续存在（例如，狗可能与户外场景相关，但被拍摄在户外的狗不到98%），因此我们的方法不会识别出这些相关性。

E.2. 生成的规则的冗余性。

根据模式的不同，SQRL 可能会生成冗余规则。不过，这种情况仅出现在逻辑分位数规则中；在不使用逻辑规则的模式中不会发生，例如目标检测、语义分析和插补任务。表格分类和图像分类任务中经过验证的规则里，冗余规则最多占生成规则的 2%。此外，冗余规则不会对测试时自适应产生负面影响，只会略微增加计算成本。

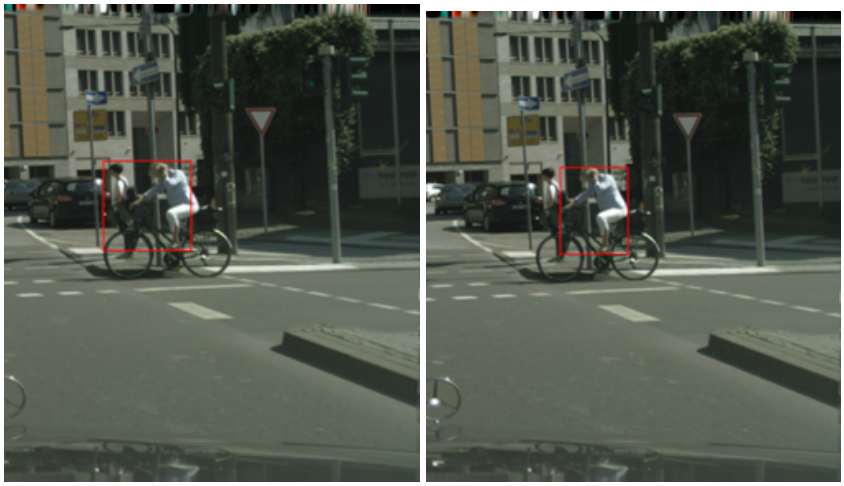
E.3. 统计分位数规则的其他用途。

为错误预测提供上下文。由于统计分位数规则可用于评估模型，因此也可用于在测试时识别模型的错误预测，此时真实标签不可用。我们使用某个样本对规则的违反情况作为特征，并以该样本是否被模型错误预测作为标签，来训练线性分类器。直接解读线性分类器的权重，可以看出与模型最相关的规则

[4]The 从 https://github.com/DeepSceneSeg/EfficientPS 下载预训练模型



图5：模型错误预测违反公式（10）中规则的一个示例



(a) 预测边界框 (b) 真实边界框

图6：根据交并比得分，预测边界框（见图（6a））被视为正确预测的一个示例。但这个预测边界框与真实边界框（见图（6b））在视觉上仍有很大差异，并且违反了公式（11）中的规则

错误。以 ins 为例例如，在目标检测任务中，违反以下规则的预测中有 92% 是错误的 s:

$$\begin{aligned}\text{person}(\mathbf{x}) \Rightarrow &(\text{size}(\mathbf{x}) < 4172 \wedge 302.6 \leq \text{pos}_\text{y}(\mathbf{x}) < 514.1) \vee \\ &(4172 \leq \text{size}(\mathbf{x}) < 13680.5 \wedge 348.76 \leq \text{pos}_\text{y}(\mathbf{x}) < 513.24) \vee \\ &(13680.5 \leq \text{size}(\mathbf{x}) < 47795.75 \wedge 393.44 \leq \text{pos}_\text{y}(\mathbf{x}) < 541.90) \vee \\ &(\text{size}(\mathbf{x}) \geq 47795.75 \wedge 407.45 \leq \text{pos}_\text{y}(\mathbf{x}) < 764.09).\end{aligned} \tag{10}$$

图5展示了一个违反上述规则的示例。在该图中，被预测为人的边界框违反了上述规则，因为该边界框的位置所对应的尺寸过大。这也解释了该预测为何错误。

除了传统指标之外，规则还能为错误预测提供更多背景信息。它们可以指出错误的严重程度、某些类型错误相较于其他类型的发生频率，而类似的规则违反也可能表明存在类似类型的错误。

发现传统指标无法识别的错误。即使传统指标判定预测正确，该指标也可能过于粗略，无法充分评估该预测。例如，平均精度均值指标会检查预测边界框与真实值边界框之间的重叠度（交并比得分）。如果它发现某个真实值边界框的重叠度超过阈值（通常为 0.5），就会检查预测框的标签；如果标签匹配，则判定预测正确。这里展示了这样一个例子：图6a中显示了预测边界框，而

图6b中显示了真实值。直观地说，预测边界框是不正确的，因为与真实值相比，它几乎覆盖了图中的两个人。然而，根据自动驾驶模型使用的指标，这一预测是正确的，尽管它违反了以下关于行人的分位数规则，因为几乎所有行人都比边界框所暗示的要窄：

$$\begin{aligned}\text{person}(\mathbf{x}) \Rightarrow\ & (\text{y}(\mathbf{x}) <= 433 \wedge 0.70 < \text{aspect_ratio}(\mathbf{x}) < 4.4) \\ & \vee (433 < \text{y}(\mathbf{x}) <= 468 \wedge 0.85 < \text{aspect_ratio}(\mathbf{x}) < 4.81) \\ & \vee (468 < \text{y}(\mathbf{x}) <= 513 \wedge 1.07 < \text{aspect_ratio}(\mathbf{x}) < 5.0) \\ & \vee (\text{y}(\mathbf{x}) > 513 \wedge 1.20 < \text{aspect_ratio}(\mathbf{x}) < 5.61).\end{aligned} \tag{11}$$