

仅预测部分输入变量的步骤

本教程将指导您如何仅预测部分输入变量。您可以参考以下步骤进行操作：

1. 功能概述

本功能支持多变量输入，并允许用户通过 `target_channel` 参数灵活指定需要预测的目标变量。未被指定为目标变量的列将自动作为协变量（外生变量），参与模型的训练和预测过程。然而，在训练过程中计算损失值（`training loss`）以及评估指标时，仅使用用户指定的目标变量进行计算。

2. `target_channel` 参数详解

`target_channel` 是本功能的核心参数，用于指定需要预测的目标变量。未在 `target_channel` 中列出的列将作为协变量辅助预测。以下是详细说明：

- **类型：** `Optional[List]`
- **作用：** 定义目标变量的列索引，系统据此将数据分为：
 - **目标变量：** 由 `target_channel` 指定的列，是模型预测的对象。
 - **协变量：** 时间序列中的其他列，作为外生变量支持预测。
- **支持的格式：**
 - `[]`（空列表）：表示未指定目标变量，所有列均作为目标变量，无协变量（模型预测所有输入变量）。
 - **单个整数：**
 - `[0]`：选择第 1 列作为目标变量。
 - `[-1]`：选择最后一列作为目标变量。
 - **整数列表：**
 - `[0, 1]`：选择第 1 列和第 2 列作为目标变量。
 - `[-2, -1]`：选择最后两列作为目标变量。
 - **元组列表：**
 - `[[0, 3]]`：选择第 1 列到第 3 列（不包含第 4 列）作为目标变量。
 - `None`：所有列均作为目标变量，无协变量（模型预测所有输入变量）。
- **注意事项：**
 - 索引方式类似于 Python 切片：正数索引从 0 开始，负数索引从 -1 开始（例如，`0` 表示第一列，`-1` 表示最后一列，`-2` 表示倒数第二列）。
 - 如果 `target_channel` 未指定或设为 `None`，系统默认将所有列作为目标变量。

3. 使用示例

假设您的数据集为 `ETTh1.csv`，包含 7 列时间序列数据（列索引为 0 到 6），使用 DUET 模型。以下是 `target_channel` 的几种使用示例。

- 配置方式：
 - 命令行：通过 `--strategy-args` 指定 `target_channel`。
 - 配置文件：在 JSON 文件（例如 `rolling_forecast_config.json`）的 `"strategy_args"` 中定义。
 - 优先级：命令行参数将覆盖配置文件中的设置。
- 基础命令行示例：

```
python ./scripts/run_benchmark.py --config-path "rolling_forecast_config.json" --
data-name-list "ETTh1.csv" --strategy-args '{"horizon": 96, "target_channel": [-1]}' -
-model-name "duet.DUET" --model-hyper-params '{"CI": 1, "batch_size": 32, "d_ff":
512, "d_model": 512, "dropout": 0.5, "e_layers": 1, "factor": 3, "fc_dropout": 0.1,
"horizon": 96, "k": 1, "loss": "MAE", "lr": 0.0005, "lradj": "type1", "n_heads": 1,
"norm": true, "num_epochs": 100, "num_experts": 2, "patch_len": 48, "patience": 5,
"seq_len": 512}' --deterministic "full" --gpus 0 --num-workers 1 --timeout 60000 --
save-path "ETTh1/DUET"
```

- 基础配置文件示例（`rolling_forecast_config.json`）：

```
{
  "strategy_args": {
    "horizon": 96,
    "target_channel": [-1]
  }
}
```

示例 1：预测第 1 列和第 2 列

- 配置：
 - 命令行：

```
--strategy-args '{"horizon":96,"target_channel":[0,1]}'
```

- 配置文件：

```
{
  "strategy_args": {
    "horizon": 96,
    "target_channel": [0,1]
  }
}
```

- 结果：

- 目标变量：ETTh1数据集的第 1 列和第 2 列。
- 协变量：ETTh1数据集的第 3、4、5、6、7 列。

示例 2：预测第 3 列

- 配置：
 - 命令行：

```
--strategy-args '{"horizon":96,"target_channel":[2]}'
```

- 配置文件：

```
{
  "strategy_args": {
    "horizon": 96,
    "target_channel": [2]
  }
}
```

- 结果：
 - 目标变量：ETTh1数据集的第3 列。
 - 协变量：ETTh1数据集的第 1、2、4、5、6、7 列。

示例 3：预测第 2 列到第 4 列

- 配置：
 - 命令行：

```
--strategy-args '{"horizon":96,"target_channel":[[1,4]]}'
```

- 配置文件：

```
{
  "strategy_args": {
    "horizon": 96,
    "target_channel": [[1,4]]
  }
}
```

- 结果：
 - 目标变量：ETTh1数据集的第 2、3、4 列。
 - 协变量：ETTh1数据集的第 1、5、6、7 列。

示例 4：预测最后两列（使用负数索引）

- 配置：

- 命令行：

```
--strategy-args '{"horizon":96,"target_channel":[-2,-1]}'
```

- 配置文件：

```
{
  "strategy_args": {
    "horizon": 96,
    "target_channel": [-2, -1]
  }
}
```

- 结果：

- 目标变量：ETTh1数据集的第 6 列和第 7 列（最后两列）。
 - 协变量：ETTh1数据集的第 1、2、3、4、5 列。

示例 5：预测所有列

- 配置：

- 命令行（省略 `target_channel` 或设为 `null`）：

```
--strategy-args '{"horizon":96}'
```

- 配置文件：

```
{
  "strategy_args": {
    "horizon": 96,
    "target_channel": null
  }
}
```

- 结果：

- 目标变量：所有列（ETTh1数据集的第 1 到第 7 列）。
 - 协变量：无。
-

4. 注意事项

- **索引范围：**确保 `target_channel` 中指定的索引在数据集的列范围内。例如，对于 7 列数据，负数索引的有效范围为 -7 到 -1。
- **负数索引：**`-1` 对应第 7 列（最后一列），`-7` 对应第 1 列（第一列）。
- **元组格式：**在使用 `[[start, end]]` 时，`end` 索引不包含在内（例如，`[1, 4]` 包含第 2、3、4 列，不包含第 5 列）。
- **默认行为：**如果 `target_channel` 为 `None` 或未指定，系统将预测所有列，且无协变量。