arXiv:2305.05575v1 [stat.AP] 9 May 2023

电力负荷与峰值预测：特征工程、概率 LightGBM 与时间层级

Nicol`o Rubattu[a]、Gabriele Maroni[a] 和 Giorgio Corani[a]

[a] Dalle Molle 人工智能研究所（IDSIA）、USI-SUPSI，CH-6962，卢加诺，瑞士{nicolo.rubattu、gabriele.maroni、giorgio.corani}@idsia.ch

摘要

我们介绍了开发预测模型的经验，该模型在 BigDeal 挑战赛 2022（一项关于负荷与峰值预测的能源竞赛）中取得了较高名次。我们提出了一种新颖的特征工程与特征选择流程，该流程基于对温度和日历变量进行聚类置换。我们采用梯度提升树，并通过趋势建模和分布预测增强其能力。我们还纳入了一种称为时间层级的预测组合方法，进一步提高了预测准确性。

关键词 负荷预测 特征工程 梯度提升 层级预测 时间协调 BigDEAL 挑战赛 2022年

引言

负荷预测是对电力需求未来变化曲线进行预测的问题，而峰值预测则是预测需求的最大值（例如每日最大值）及其出现时间的问题。峰值预测十分重要，因为决策往往基于峰值预测，而非整个负荷曲线的预测。

在这项工作中，我们介绍了一种成功参与 BIGDeal 挑战赛 2022 [21]，的方法，该挑战赛旨在预测能源负荷和峰值。竞赛于2022年10月至12月举行；121名参赛者参与其中，分为78个团队。预测结果使用不同指标进行评估，竞赛分为资格赛和决赛。我们在资格赛中获得了第 3^{rd} 名，晋级决赛，最终排名为 6^{th} [22]。

资格赛中，我们使用了基于树的梯度提升（GB），并结合一种原创的特征工程与特征选择方法。决赛中，我们开发了更复杂的方法。具体而言，我们采用了近期推出的 LightGBM 概率版本 [38]；使用 DART算法 [45] 进行正则化；还采用了时间层级结构 [3] ，通过组合不同时间尺度上的预测结果来改进预测。尽管竞赛只对点预测进行评分，我们的方法仍是概率型的，因此能够量化预测的不确定性。这对于支持决策确实不可或缺。

本文结构如下。接下来将概述长期负荷预测并阐述我们的研究动机。我们在“梯度提升与分布预测”节介绍树的梯度提升（GB）及其概率扩展。我们在“特征工程”节介绍用于负荷预测的特征工程方法，并在第“特征选择”节介绍特征选择。在“时间层级”节介绍时间层级。在“实验”节中，我们将结合技术见解详细回顾整个流程及竞赛结果。最后，我们以批判性的结论为本文作结。

方法论

长期负荷预测

负荷预测是对未来 H个时间步的电力需求进行预测的问题，记为 $[y_{T+1}, \dots, y_{T+H}]$。当 H 的数量级达到数百或更多时，我们称之为长期预测。例如，以每小时为尺度进行提前一年预测，意味着需要生成 $24 \times 365 = 8760$个预测值。经典预测策略 [4]会根据时间序列的最近观测值来生成预测。然而，这不适用于长期预测，因为在这种情况下， y_{T+H}与 y_T无关。长期预测更适合被视为回归问题，并采用一组丰富的解释变量（特征）[27, 15, 8, 46]，涵盖日历效应（星期几、一天中的小时等）、节假日、温度等。这种方法使我们能够采用树的梯度提升（GB）等回归方法 [16], ，而该方法在长期能源预测中确实表现出色 [44]。

梯度提升与分布预测

事实上，GB 在 2012, 2014, 2017 [24, 25, 26], M5预测竞赛 [35, 29]、表格数据竞赛 [7, 42]以及全球能源预测竞赛（GEFCom）中均取得了顶尖名次。最流行的实现包括 XGBoost [9], LightGBM [30] 和 CatBoost

[12]。它们的准确率相当，但 LightGBM 通常速度更快，在大型数据集上的扩展性也更好。

GB 可以使用传统最小二乘法之外的不同损失函数进行训练。例如，采用分位数回归训练的 GB 在 GEFCom2014 概率预测竞赛中获胜 [17]。然而，即使是分位数回归，也只能返回点预测，无法提供预测分布。可以针对每个所需的分位数分别训练不同的 GB 模型；但如果预测的分位数发生交叉，预测分布就会失效 [39, 41]。近期的概率树 GB 版本为概率预测提供了一种更可靠的方法 [13, 37, 38]。本文采用 M¨arz 等人提出的模型 [38], ，该模型会返回多元输出，其中包含预测分布的各阶矩。

无论如何，要成功实现 GB，都需要留意一些潜在问题。例如，GB 通常无法对长期趋势进行建模。如果时间序列存在趋势，建议先对其去趋势，再拟合 GB 模型，最后将预测的趋势加到 GB 预测中[50, 33, 47]。另一种有时会有所帮助的预处理步骤是对目标时间序列进行对数变换，以稳定其方差 [43]。此外，GB 也容易过拟合。DART算法[45]通过引入 Dropout 正则化来解决这一问题，其原理与神经网络类似。

特征工程

负荷预测中常用的外生变量与日历和温度有关。

基于日历的特征 日历变量有助于捕捉季节性模式。通常使用分类变量对其建模，并为特征的每个取值设置一个类别。例如，星期几由一个包含七个类别的分类变量表示。节假日则由一个包含两个类别的二元变量表示：节假日为 1，非节假日为 0。

滞后和滚动温度 温度会影响能源消耗，因为它会驱动供暖、通风和空调（HVAC）系统的使用。不过，温度变化与能源消耗变化之间通常存在延迟。因此，我们考虑滞后每小时温度：

$$T(t-h), \qquad h = 1, 2, \ldots, L \tag{1}$$

其中 L 是最大滞后值，以及移动/滚动温度统计量：

$$T_f^w(t) = f(T(t-1), \ldots, T(t-w)) \tag{2}$$

其中 $f(\cdot)$ 是某种统计函数， w 表示所考虑的过去温度值窗口宽度（按小时计）。例如，过去 24 小时温度值的移动平均为 $T_{avg}^{24}(t) = \frac{1}{24}\sum_{h=1}^{24} T(t-h)$。

温度的聚合指标 聚合特征能够捕捉温度对能源负荷的长期影响，同时平滑数据中的噪声。它们可以表示为 $\tilde{T}_f^g(t)$，其中 g 是聚合周期， $f(\cdot)$ 是聚合函数。这些特征包括例如温度的每日最大值和最小值，或温度的月度标准差。

本文提出了其他借鉴自信号处理的聚合函数（表 1） [14, 49], ，据我们所知，这些函数尚未用于能源预测。信号处理特征应根据温度时间序列计算，并在选定的聚合周期内重新缩放，使其均值为零。它们能够揭示聚合周期内温度曲线的变异性和形状。例如，峰值因数衡量信号的峰值与均值之比。较高的日峰值因数对应着日间温度的大幅变化，这通常会增加能源需求。另一方面，较低的日峰值因数对应着日间温度较为稳定，这通常会降低能源需求。

表 1：用于负荷预测的信号处理特征。

RMS	$x_{RMS} = \sqrt{\frac{1}{N}\sum_{i=1}^{N} x_i^2}$
峰值	$x_p = \max(\lvert x_i \rvert)$
峰值因数	$x_{crest} = \frac{x_p}{x_{RMS}}$
脉冲因数	$x_{if} = \frac{x_p}{\frac{1}{N}\sum_{i=1}^{N} \lvert x_i \rvert}$
裕度因数	$x_{mf} = \frac{x_p}{\left(\frac{1}{N}\sum_{i=1}^{N} \lvert x_i \rvert^{1/2}\right)^2}$
波形因数	$x_{sf} = \frac{x_{RMS}}{\frac{1}{N}\sum_{i=1}^{N} \lvert x_i \rvert}$
峰峰值	$x_{pp} =$ 最大值$(x_i) -$ 最小值(x_i)

特征选择

特征工程通常会生成大量特征，其中许多最终都是冗余的。因此，需要进行特征选择，以缩短

训练时间 [32], 减少过拟合 [6], 提高模型的可解释性[28]。

我们基于特征的层次聚类和两两相关性进行特征选择。我们方法的核心模块是置换特征重要性（PFI），它通过随机打乱某个特征，衡量模型性能的下降幅度。其理论基础见 [5, 40]。性能下降的幅度表明模型在进行预测时对该特征的依赖程度。PFI 具有以下优势：

- 可应用于任何模型；- 易于实现（算法 1）；- 可根据竞赛指标衡量特征重要性；- 可在样本外计算。

算法 1 置换特征重要性

要求： 一个经过训练的模型，以及在评估数据集上记录的分数 s **对于**特征 $x_j,\ j = 1,\ \ldots,\ d$ **执行以下操作，遍历**每次重复 $k,\ k = 1,\ \ldots,\ K$ **执行**随机打乱原始评估集中的列 j。计算模型在扰动后的数据集上的新分数 $s_{k,\ j}$。

循环结束

计算特征x_j 的重要性：

$$I_j = s - \frac{1}{K}\sum_{k=1}^{K} s_{k,j}$$

循环结束

然而，如果特征之间存在依赖关系，打乱单个特征可能会产生不切实际的样本。此外，相关特征会分摊重要性，因此其相关性可能被低估（替代效应） [20,40]。

聚类置换特征重要性 为克服这些问题，我们提出了一种新方法，称为聚类置换特征重要性（CPFI）。该方法的工作流程如下。首先，通过对特征的相关矩阵进行层次聚类，识别高度相关的特征组。为构建相关矩阵，我们使用相关性指标计算每对特征之间的依赖程度，例如皮尔逊相关系数或斯皮尔曼相关系数。然后，我们将同一聚类中的所有变量一起打乱，并计算随后得到的性能

下降幅度。不同聚类所含信息的正交性越高，重要性估计就越可靠。

最后，我们从模型中剔除非信息性特征簇。相反，我们会将相关簇中的特征保留在模型中。此外，可根据特征对目标的解释能力，或依据专家建议，从每组中仅选择一个或少数几个特征。我们在“实验”一节提出了一项准则，用于区分信息性与非信息性特征簇。类似方法已在金融 [11]和生物化学 [36]领域提出。

时间层级

作为进一步提高预测准确度的工具，我们考虑时间层级 [3]。例如，假设我们要生成每小时尺度的预测（称为底层）。时间层级会在更粗的时间尺度（例如，每2小时和每4小时）上生成预测，然后综合不同时间尺度生成的预测中的信息。这一过程通常能提高所有层级的预测准确度 [3, 31]。时间层级的工作方式如下。首先，在不同时间尺度上分别生成预测（基础预测）。例如，图1展示了一个用于预测4小时^ ^之后情况的时间层级。它包含4个按每小时频率计算的预测（h_1，……，h_4，^ ^底层）；两个按每2小时频率计算的预测（h_{12}，……，h_{34}，中间层）；以及一个按每4小时频率计算的预测（h_{1234}，顶层）。

通常，基础预测无法正确求和，因此被称为不协调。例如：$h_{12} \neq h_1 + h_2$, $h_{34} \neq h_3 + h_4$ 等。协调 [48] 是调整基础预测，使其变得协调，即能够正确求和的过程。协调后的预测用波浪号表示，因此，在图 1 的示例中，协调后˜ ˜ ˜ ˜ ˜ ˜ ˜ ˜ ˜我们有：$h_{12} = h_1 + h_2$, $h_{34} = h_3 + h_4$, $h_{1234} = h_{12} + h_{34}$。

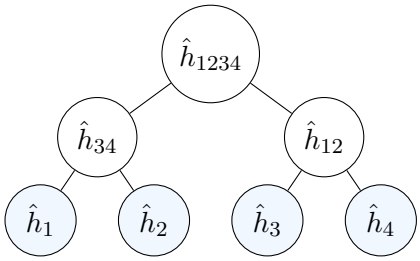


图 1：预测未来 4 小时时的时间层级，使用小时预测（底层）、每2小时预测和每4小时预测。

时间层级需要基础预测的均值和方差。

原始算法 [3]仅提供协调后的点预测，而 [10] 提出的方法还可生成协调后的预测分布。

实验

BigDEAL 挑战赛 2022 分为资格赛和决赛。

资格赛提供了 2002年至2006年期间四个气象站的每小时负荷和每小时温度统计数据（均值、中位数、最小值、最大值）；示例见图 2。参赛者需要根据实际温度预测 2007 年。这被称为事后设定 [27]。

决赛提供了美国三家地方配电公司（LDC）三年（2015—2017 年）的每小时负荷数据，以及六个气象站的每小时温度数据。2018 年提前一天的预测温度（事前设定 [27]）按滚动方式发布，每次发布两个月的数据。这些时段的预测需分六轮连续提交。

两项赛事均要求按每小时粒度预测 24 小时的负荷、每天的峰值及其出现时间（即 1 到 24 之间的离散数值）。

组织者运行了两种基线方法。第一种是 *Tao* 的基础基准方法 [24], ，即一个多元线性回归模型，其特征包括趋势、日历效应，以及仅一个与温度相关的特征。第二种是 近期性基准方法 [46], ，它在前一种模型的基础上进行了扩展，加入了许多与温度相关的特征，其中包括滞后值和移动平均值。

性能指标

组织者使用三种不同的轨迹评估了每场比赛的预测。

在资格赛中，使用平均绝对百分比误差 （平均绝对百分比误差）对每小时预测（$24 \times 365 = 8760$）进行了评分：

$$\mathrm{MAPE} = \frac{1}{H} \sum_{t=T+1}^{T+H} \frac{|y_t - \hat{y}_t|}{|y_t|} \times 100, \tag{3}$$

其中， y_t 和$\hat{}y_t$ 分别表示时间 t的实际值和预测值。第二个指标是幅值（M）；它是实际每日峰值与预测每日峰值之间的平均绝对百分比误差（MAPE）（即指一年预测期内的 365 个预测值）。不过，我们需要指出，MAPE 在预测领域一直受到批评：它对高估误差的惩罚大于对低估误差的惩罚 [2] ，而且在处理接近 0 的值时，其数值不稳定。

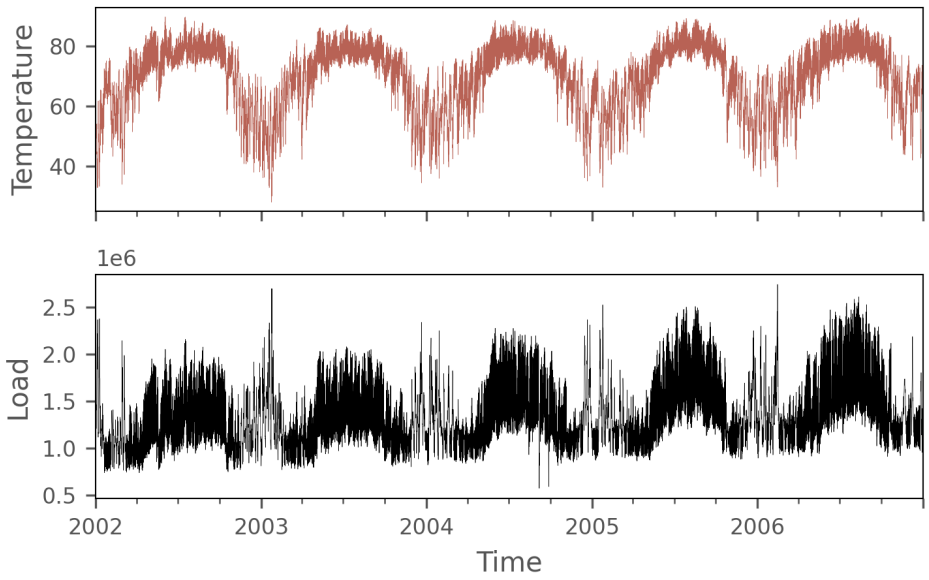


图 2：BigDEAL 挑战赛资格赛的负荷（MW）和温度 T^{avg} （°F）。为便于阅读，我们展示了按 12 小时聚合的数据。

为了评估峰值时段的预测，组织者使用了第三个指标，称为时序 （T），用于计算平均绝对误差。例如，如果实际峰值出现在下午 6 点，而预测峰值时间为晚上 8 点，则当天的误差为 $|6-8|=2$。

决赛使用幅值 (M)和 时序(T)对预测进行评分，并额外采用一项名为形状 (S)的指标。不过，时序的定义经过了修改，对误差引入了非均匀代价。设 $\mathcal{T}_d$ 和 $\mathcal{T}_d$ 分别表示某日 d的实际峰值小时和预测峰值小时。时序的定义如下：

$$\mathrm{T} = \frac{1}{|days|} \sum_{d \text{ in } days} w(\mathcal{T}_d, \hat{\mathcal{T}}_d), \text{with}$$

$$w(\mathcal{T}_d, \hat{\mathcal{T}}_d) = \begin{cases} |\mathcal{T}_d - \hat{\mathcal{T}}_d|, & \text{if } |\mathcal{T}_d - \hat{\mathcal{T}}_d| = 1, \\ 2|\mathcal{T}_d - \hat{\mathcal{T}}_d|, & \text{if } 2 \le |\mathcal{T}_d - \hat{\mathcal{T}}_d| \le 4, \\ 10, & \text{if } |\mathcal{T}_d - \hat{\mathcal{T}}_d| \ge 5 \end{cases} \tag{4}$$

形状 (S)用于评估峰值附近的预测形状。计算时，将某一天的24小时负荷预测除以该日的峰值预测进行归一化，实际负荷也采用相同的处理方式。然后，计算每天5小时峰值时段（实际峰值小时 $\pm$ 2 小时）内绝对误差的总和。我们用 $\bar{y}_d$ 和 $\hat{}y_d$ 表示归一化后的实际负荷和预测

– ·负荷 d; $\bar{y}_d = \frac{y_d}{\text{最大值 } y_d}$，$\hat{y}_d = \frac{y_d}{\text{最大值} \hat{y}_d}$。形状定义如下:

$$\mathrm{S} = \frac{1}{|days|} \sum_{d \text{ in } days} \sum_{t \text{ in } \{\mathcal{T}_d, \mathcal{T}_d \pm 1, \mathcal{T}_d \pm 2\}} |\bar{y}_d(t) - \bar{\hat{y}}_d(t)| \tag{5}$$

预测分布评分 虽然竞赛只评估了点预测，我们也对概率模型得到的分布预测进行了评分。具体而言，我们比较了 GB 模型的概率预测（基于 [38]）与应用时间层级后得到的预测。

我们使用连续秩概率得分 （CRPS） [19]对模型的预测分布进行了评分。设$\hat{}\mathbf{F}$ 为预测累积分布函数， y 为实际值:

$$\mathrm{CRPS}(\hat{\mathbf{F}}, y) = \int_{-\infty}^{\infty} (\hat{\mathbf{F}}(x) - \mathbb{1}(x \geq y))^2 \, dx \tag{6}$$

当$\hat{}\mathbf{F}$为高斯分布时，积分可以通过闭式形式计算 [19]。

随后，我们使用区间得分 （IS） [18]对预测区间进行了评分。设 $1-\alpha$ 为区间的名义覆盖率（本文假定为 0.9），$\mathbf{l}$ 和$\mathbf{u}$ 分别为其下界和上界。因此，我们使用模型为每个小时计算了一个 90% 的预测区间，并计算其得分:

$$\mathrm{IS}(\mathbf{l}, \mathbf{u}, y) = (\mathbf{u} - \mathbf{l}) + \frac{2}{\alpha}(\mathbf{l} - y)\mathbb{1}(y < \mathbf{l}) + \frac{2}{\alpha}(y - \mathbf{u})\mathbb{1}(y > \mathbf{u}) \,. \tag{7}$$

我们还报告了区间（$\mathbf{l}$，$\mathbf{u}$）包含y的案例比例。

技能评分 设 m_{origin} 和 m_{new} 为两个可比较的指标。我们将正向或负向的百分比改进表示为以下定义的技能评分:

$$\mathrm{Skill}_{\%}(m_{origin}, m_{new}) = \frac{m_{origin} - m_{new}}{(m_{origin} + m_{new})/2} \times 100 \tag{8}$$

资格赛

下面介绍我们实现方案的构成模块。

交叉验证策略鉴于数据具有序列特性，我们采用时间序列交叉验证来评估各个模型的性能、超参数调优和特征选择。该方法将数据划分为样本内数据集（包含较早的观测值）和样本外数据集（包含最新的观测值）。模型在样本内部分进行训练，并在样本外部分进行评估。为获得多次评估结果，样本外数据集会迭代地

向前移动一个时间窗口，并在更新后的样本内数据集上重新训练模型。对所有样本外数据集重复此过程，确保评估时模型是在未见过的观测序列上接受测试，从而得出更贴近实际的评估结果。

时间窗口的大小通常设定为与最终进行预测时所用测试集的大小相同。因此，在资格赛阶段，我们将2004年、2005年和2006年作为样本外折。

基线模型 我们首先对基本日历特征（年、月、周、日、星期、小时）以及当前时刻的温度（T^{avg}、T$_{med}$、T$_{min}$、T^{max}）进行建模。日历变量采用标签编码。我们测试了更复杂的编码方式（目标编码、使用正弦/余弦变换的周期性编码，以及使用径向基函数的周期性编码），但未见明显改进。

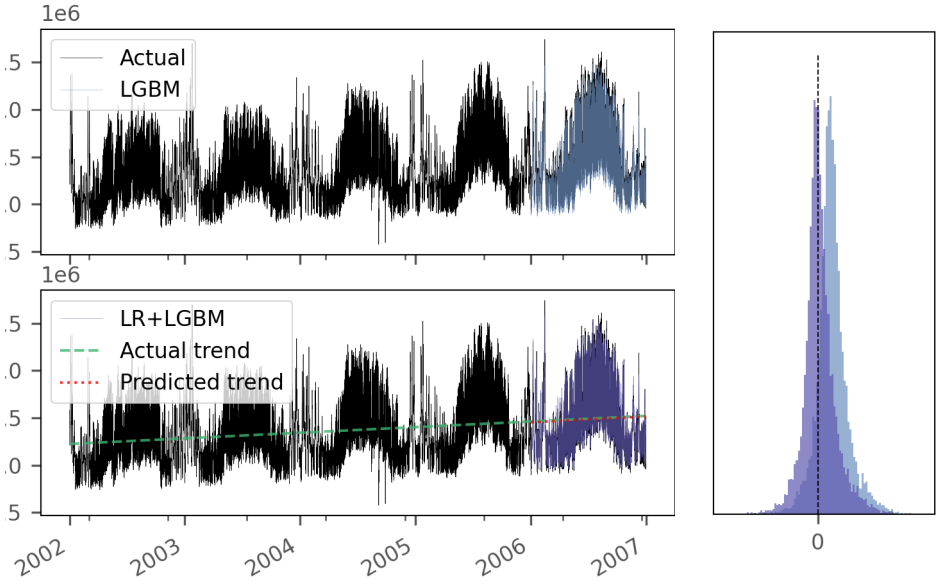
我们对目标变量进行了对数变换，以稳定其方差。此外，由于目标变量呈长期上升趋势，我们进行了去趋势处理。在每一轮交叉验证中，我们都使用样本内数据拟合线性回归（LR）模型（$y_i = \beta_0 + \beta_1 x_i$，其中 x_i 为递增的时间索引，取值为 $i = 1$、.....、T）。在拟合LightGBM模型之前，我们从训练数据中减去了线性趋势。在预测阶段，我们将预测的线性趋势加到样本外预测结果中，然后进行指数变换，从而得到最终预测。图3展示了去趋势处理的效果。经过去趋势处理后，模型的残差均值如预期为0；否则，残差会严重偏离。通过去趋势处理，我们将基线模型的平均绝对百分比误差（MAPE）（H）从6.18降至4.81。

损失函数 我们使用 L2 损失函数训练了 LightGBM 模型。

异常值 为防止模型受到异常值的影响而产生偏差，我们将包含这些值的时间步的权重设为零。我们从目标分布的左侧识别出这些异常值，我们认为它们对应于极端天气条件下的时段。

特征工程 特征工程以逐步递增的方式开展，每次添加一个相关的特征区块。这样既能系统地评估每个区块对模型性能的影响，也能确保每组新增的相关特征都提供额外信息。我们发现，以下特征对本次竞赛具有预测作用：

- 额外的日历特征：节假日、节假日名称、周末、月份中的第几周、季节、一年中的第几天、距上一个节假日已过／距下一个节假日还有的天数。



(a) 预测　　(b) 残差

图3：去趋势的效果。线性回归模型（$\beta_1 = 7.05$）修正了低估问题。残差图（右侧）显示，去趋势后残差分布正确地以 0 为中心。

- 滞后小时温度：针对每个温度变量（T_{avg}、T_{med}、T_{min}、T_{max}），将滞后 1 小时至最大 48 小时的每小时温度纳入模型，共新增 192 个特征。

- 基于温度的滚动统计量：针对每个温度变量，采用 4 种不同的窗口宽度（3 小时、1 天、1 周、1 个月），计算 5 种统计函数（均值、最大值、最小值、中位数、标准差），共生成 80 个新特征。

- 温度聚合统计量：对于每个温度变量，针对 2 个不同的聚合时间段（年-月-日、月-小时）和 11 个聚合函数（均值、最大值、最小值、中位数，以及居中的 *RMS*、峰值因数、峰值、脉冲因数、裕度因数、波形因数、峰峰值），计算实际温度值与聚合值之间的差异，共得到 $88 \times 2 = 176$ 个新特征。为便于说明，我们以 $\tilde{T}^{Year,}_{max}{}_{Month\ Day,}(t)$ 表示日最大值，其中年 - 月 - 日表示从左侧层级到右侧层级读取的聚合方式。

特征选择 为评估我们的特征选择策略，我们开展了多项实验。首先，我们评估了未进行任何特征选择时的模型性能（实验 **a**）。随后，在完成全部特征工程后，针对添加到基线模型中的全部特征，应用了第“特征选择”节所述的特征选择策略（实验 **b**）。最后，每当向模型中添加一个新的特征块时，我们都执行逐步 特征选择，即在添加滞后变量、添加滚动变量等操作之后进行特征选择（实验 **c**）。为提高可靠性，我们执行了 100 次簇置换，并根据所有样本外折，计算性能下降的均值和标准差。如果某个特征簇的重要性值高于 0 且处于均值以上三个标准差范围内，我们将其视为具有信息量的 特征簇。

结果见表 2。具体而言，MAPE、幅值和时序列展示了根据相应竞赛指标得出的结果，而**a**、**b**和**c** 列则对应所采用的 3 种实验策略。

需要注意的是，与实验**a**的结果来自单次训练运行不同，实验**b** 和**c** 的结果来自三次不同的训练运行，每次训练均以最大化感兴趣的指标为目标。

表 2：采用特征选择方法得到的折外资格评估结果。

	MAPE（H）			幅值			时序		
	a	**b**	**c**	**a**	**b**	**c**	**a**	**b**	**c**
基线	4.81	-	-	4.43	-	-	1.42	-	-
日历	4.83	-	4.78	4.48	-	4.46	1.39	-	1.39
Lags	3.33	-	3.24	3.29	-	3.20	0.94	-	0.92
滚动滞后项	3.28	-	3.16	3.22	-	3.20	1.06	-	0.94
聚合统计量	3.24	3.16	3.09	3.21	3.10	3.08	0.91	0.95	0.91

为了进行说明，在图4中，我们展示了将滞后每小时温度纳入模型后获得的特征选择结果。图4a展示了基于图4b所示的斯皮尔曼相关矩阵进行层次聚类后得到的树状图。通过选择0.1作为阈值，我们识别出36个聚类。图4c展示了分别使平均绝对百分比误差（MAPE）、幅值（Magnitude）和时序（Timing）性能最大化的聚类排名。对于所有指标，聚类8均表现出最显著的重要性，其次是聚类31、7、2和12。这表明，信息量最大的滞后项位于 t-$\{1, 2, 3, 4, 5, 6\}$、t-$\{11, 12\}$ 和 t-$\{25, 26\}$。表3展示了与这些聚类相关联的特征集。

表 3：聚类置换特征重要性：使性能指标达到最大值的滞后温度前 5 个聚类。

簇 ID	特征集
8	$\mathrm{T}_{avg,\ med,\ min}(t-1)$, $\mathrm{T}_{avg,med,min}(t-2)$
31	$\mathrm{T}_{avg,\ med,\ min}(t-11)$, $\mathrm{T}_{avg,med,min}(t-12)$
7	$\mathrm{T}_{avg,med,min}(t-3)$, $\mathrm{T}^{avg}(t-4)$
2	$\mathrm{T}_{avg,\ med,\ min}(t-5)$, $\mathrm{T}_{avg,med}(t-6)$
12	$\mathrm{T}_{max}(t-1)$, $\mathrm{T}_{max}(t-2)$, $\mathrm{T}_{max}(t-25)$, $\mathrm{T}_{max}(t-26)$

超参数优化 我们使用 Optuna [1]来调优 LightGBM 的学习参数。它实现了时间预算优化，在竞赛期限较短的情况下非常实用。Optuna 的另一项优势是，它可以针对竞赛的指标优化参数。

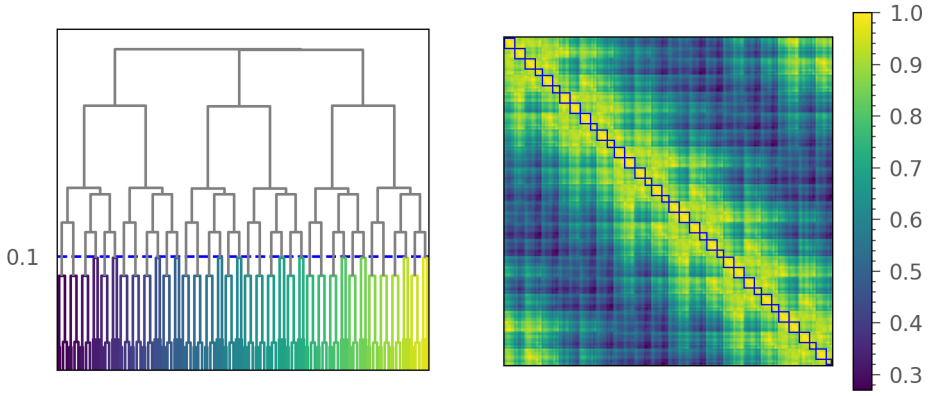
结果 我们的团队名为“swissknife”；如表 4 所示，我们在每小时预测（H）中排名 8^{th}，在幅值（M）中排名 3^{rd}，在时序（T）中排名 3^{rd}。

表 4：资格赛 [23] 排行榜。

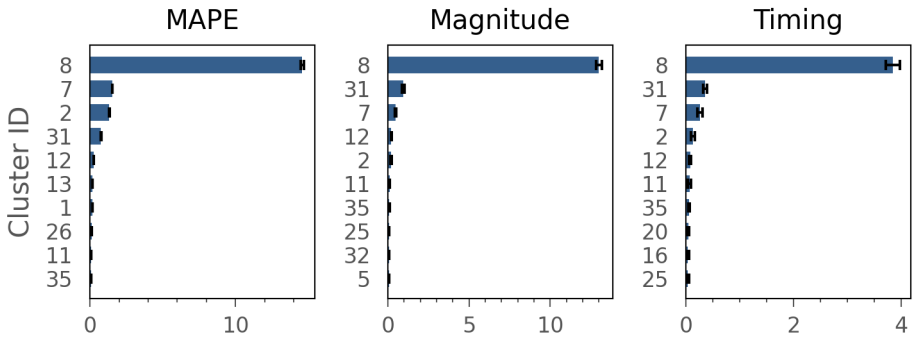
Team	排名 H。	Team	M.排名	Team	T.排名
X-Mines	1	Amperon	1	RandomForecast	1
Amperon	2	Team SGEM KIT	2	Amperon	2
Yike Li	3	swissknife	3	swissknife	3
peaky-finders	4	peaky-finders	4	freshlobster	4
KIT-IAI	5	KIT-IAI	5	peaky-finders	5
Overfitters	6	EnergyHACker	6	*Recency Benchmark*	
BelindaTrotta	7	BelindaTrotta	7	X-Mines	6
swissknife	8	Overfitters	8	BrisDF	7
Recency Benchmark		VinayakSharma	9	BelindaTrotta	8
RandomForecast	9	SheenJavan	10	KIT-IAI	9
Team SGEM KIT	10	...		SheenJavan	10
...		*Recency Benchmark*	13	...	
Tao's Vanilla Benchmark	27	*Tao's Vanilla Benchmark*	25	*Tao's Vanilla Benchmark*	30

决赛

决赛中，我们沿用了资格赛阶段调优的同一套流程，但无需进行目标变量变换，因为目标变量本身已经平稳。此外，需要预测三个 LDC 负荷（LDC1、LDC2、LDC3），而非一个；温度变量则直接来自六个气象站（T1、T2、T3、T4、T5、T6），不包含汇总统计数据，也不包含地理位置信息。



(a) 树状图 (b) 相关矩阵



(c) 聚类置换特征重要性

图4：层次聚类（阈值为0.1）（a）和 Spearman 相关矩阵（b）。蓝色方块标出了36个聚类。（c）报告了每条轨迹的聚类置换特征重要性（CPFI）值。前5个特征集见表3。

为了进一步提升性能，我们采用了多种技术，包括 DART、概率 LightGBM 和时间层级。

特征选择 最重要的滞后温度出现在时刻 t-$\{1, 2, 3, 4, 5\}$ 和 t-$\{10, 11, 12\}$，而最重要的滚动滞后温度对应的窗口为 $w = \{3\,hours, 1\,day\}$。图 5 显示，在这六个气象站中，温度 {T1、T2、T5} 能更好地解释 LDC1。因此，我们的框架能够很好地处理包含多个气象站的数据集。

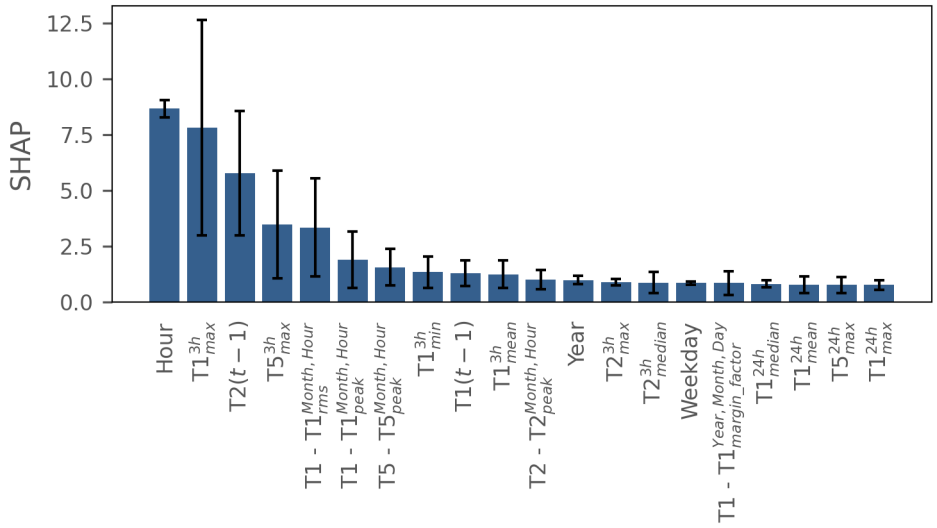


图 5：在第 1 轮中，对 LDC1 完成特征工程（聚合特征）和特征选择的最后一个增量步骤后，得到的折外前 20 个重要特征。在y轴上，我们报告了 LightGBM 模型的 SHAP（SHapley Additive exPlanations） [34] 值。

正则化 使用 DART 提升器，通过丢弃降低了 LightGBM 标准提升器容易出现的过拟合。然而，由于需要更多次提升迭代，训练速度会变慢。此外，由于算法会在训练期间更新先前学习到的树，因此无法使用提前停止。尽管存在这些问题，使用 DART 后我们降低了预测误差。我们还发现，即使提升迭代次数很多，损失也会稳步下降并渐近收敛。我们仅在资格赛数据处理完成后才对其进行了 DART 测试。经过 30,000 次迭代后，平均绝对百分比误差 (H) 从 3.24 降至 2.83，幅值从 3.21 降至 3.09。因此，我们在决赛模型中加入了 DART。

时间层级 我们通过对每小时负荷时间序列及相关温度时间序列按以下尺度求和，构建时间层级：*2* 小时、*4* 小时、*6* 小时、*12* 小时。我们在每个时间尺度上训练一个独立的概率 LightGBM-LSS [38] 模型，并使用 DART 提升器进行 $30'000$ 次提升。LightGBM-LSS 将负对数似然损失函数降至最低。在每个时间尺度上，针对相同的预测时域 H，我们得到了高斯分布预测。协调步骤使基础预测保持一致。

在表5（负荷曲线）和表6（峰值）中，我们比较了基础预测与协调后的预

测，使用技巧评分（$S_{\%}$）；在图6中，我们展示了一些预测结果。时间层级仅略微改善了点预测，但更重要的是改善了预测分布：连续秩概率得分的技巧评分约为5%，区间得分约为10%。

我们还测试了1天聚合，但底层时间序列并未得到进一步改善。正如先前的特征重要性分析所示，接近条件时点的过去值是预测中最重要的变量。我们的解释是，过度聚合（经验上大于1天）会使这些变量消失，从而使时间层级不再有效。相反，如表6和图6所示，较小的层级也能改善峰值。

鉴于数据的可用性，我们针对决赛所呈现的指标，采用的是第1至第5轮（2018年1月至10月）的实际竞赛值，而不是折外集。

表5：负荷曲线的协调指标；基准（$\hat{y}$）预测和协调后（$\tilde{y}$）预测，以及技巧评分（$S_{\%}$）。使用每小时（底层）、每2小时、每4小时、每6小时和每12小时聚合进行预测的时间层级。

	MAPE			CRPS			区间得分90%			区间覆盖率90% (%)	
	$\hat{y}$	$\tilde{y}$	$\mathbf{S}_{\%}$	$\hat{y}$	$\tilde{y}$	$\mathbf{S}_{\%}$	$\hat{y}$	$\tilde{y}$	$\mathbf{S}_{\%}$	$\hat{y}$	$\tilde{y}$
LDC1	4.87	**4.84**	0.75	6.35	**6.03**	5.16	61.62	**55.01**	11.34	99.24	98.81
LDC2	5.02	**4.99**	0.52	10.92	**10.44**	4.49	101.35	**90.39**	11.43	99.07	98.49
LDC3	4.51	**4.5**	0.05	45.99	**43.84**	4.78	446.49	**398.37**	11.39	98.85	98.14

表6：峰值的协调指标；基准（$\hat{y}$）和协调后（$\tilde{y}$）的预测，以及技巧评分（$S_{\%}$）。采用每小时（底层）、每2小时、每4小时、每6小时和每12小时聚合的预测时间层级。

	幅值			时序			形状			连续秩概率得分 peak		
	$\hat{y}$	$\tilde{y}$	$\mathbf{S}_{\%}$	$\hat{y}$	$\tilde{y}$	$\mathbf{S}_{\%}$	$\hat{y}$	$\tilde{y}$	$\mathbf{S}_{\%}$	$\hat{y}$	$\tilde{y}$	$\mathbf{S}_{\%}$
LDC1	4.97	**4.90**	1.34	1.22	**1.13**	7.93	0.088	**0.086**	2.16	8.33	**7.89**	5.46
LDC2	5.51	**5.48**	0.52	1.26	**1.23**	1.87	0.102	**0.101**	1.11	15.73	**15.13**	3.85
LDC3	4.83	**4.79**	0.95	1.19	**1.09**	8.80	0.079	**0.078**	1.56	60.83	**57.97**	4.82

结果 我们放置了 6^{th} (M)、 6^{th} (T)、 7^{th} (S)，见表7。

结论

在这项工作中，我们分享了自身经验，并凭借这些经验在国际能源预测竞赛中取得了优异成绩。我们的主要贡

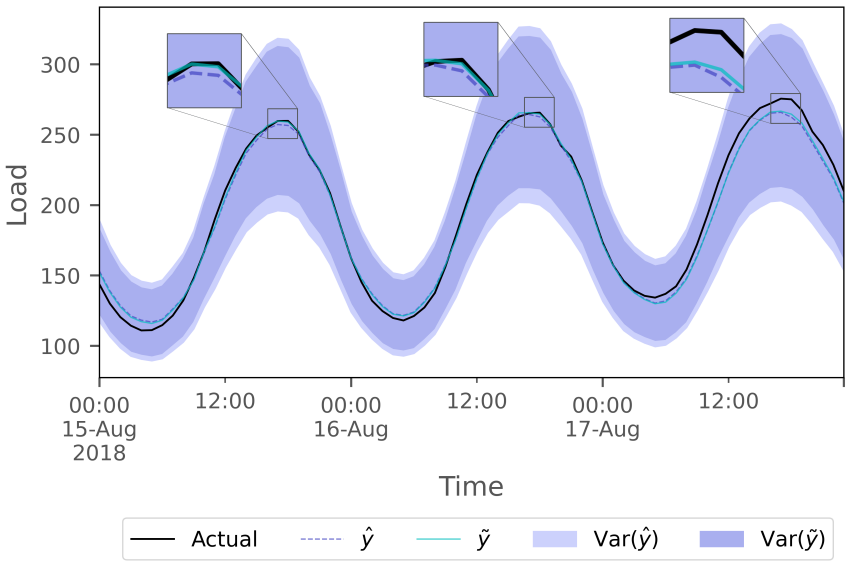


图6：应用时间层级前后的概率预测比较。时间层级略微改善了点预测，也缩短了预测区间，同时不影响其可靠性。样本为 LDC1 的三天数据（2018年8月15日至17日）。

表7：决赛排行榜 [22]。

Team	M.排名
Amperon	1
Overfitters	2
peaky-finders	3
Team SGEM KIT	4
KIT-IAI	5
swissknife	6
Recency Benchmark	7
Energy HACker	8
Rajnish Deo	9
X-mines	10
...	
Tao's Vanilla Benchmark	17.5

Team	T.排名
KIT-IAI	1
Amperon	2
BelindaTrotta	3
Overfitters	4
X-mines	5
swissknife	6
peaky-finders	7
Rajnish Deo	8
Team SGEM KIT	9
SheenJavan	10
...	
Recency Benchmark	14
Tao's Vanilla Benchmark	18

Team	S.排名
KIT-IAI	1
Amperon	2
Overfitters	3
X-mines	4
SheenJavan	5
Rajnish Deo	6
swissknife	7
Recency Benchmark	8
RandomForecast	8.5
Yike Li	8.5
peaky-finders	10
...	
Tao's Vanilla Benchmark	16

Team	最终排名
Amperon	1
KIT-IAI	2
Overfitters	3
peaky-finders	4
X-mines	5
swissknife	6
Rajnish Deo	7
Team SGEM KIT	9
Recency Benchmark	10
...	
Tao's Vanilla Benchmark	14

献是定义了一组庞大的解释变量，其中一些借鉴自信号处理领域的文献，并提出了一种新颖的特征选择策略。

我们将这种方法称为聚类置换特征重要性（CPFI）。选择基于梯度提升（GB）的回归模型，使特征工程成为必要。在不久前，尤其是 XGBoost 和 LightGBM 的实现，已证明其有效，而且在某种程度上优于竞争性的神经网络。随后，我们在传统的表格应用中，验证了 LightGBM 作为负荷预测回归器的潜力。我们指出，DART 提升器相较于传统的树梯度提升（GB）所带来的改进。本文的第二项重要贡献，是实现了 LightGBM 最新的概率扩展版本，它返回预测分布的各阶矩，而不只是点预测。这影响重大，因为决策者也可以利用预测中固有的不确定性。即使在能源预测领域，这些模型仍未广泛普及。借助分布预测，我们得以应用时间层级，进一步提升结果。竞赛并未评估预测不确定性，但我们的概率方法也证明了其区间具有良好的校准性。

数据和材料的可用性

截至提交日期，竞赛数据集尚未公开。组织者计划最终将其发布。

利益冲突

作者声明，据其所知不存在可能被认为影响本文所报告工作的竞争性财务利益或个人关系。

参考文献

[1]T. Akiba、S. Sano、T. Yanase、T. Ohta 和 M. Koyama。Optuna：新一代超参数优化框架。载于第25届 ACM SIGKDD 知识发现与数据挖掘国际会议论文集，第2623–2631页，2019年。[2] J. S. Armstrong 和 F. Collopy。用于概括预测方法的误差度量：实证比较。《国际预测学杂志》，8(1):69–80，1992年。[3] G. Athanasopoulos、R. J. Hyndman、N. Kourentzes 和 F. Petropoulos。基于时间层级的预测。《欧洲运筹学杂志》，262(1):60–74，2017年。

[4] G. Bontempi、S. Ben Taieb 和 Y.-A. Le Borgne。时间序列预测的机器学习策略。《商业智能：第二届欧洲暑期学校》，eBISS 2012，比利时布鲁塞尔，2012年7月15日至21日，教程讲义，第62–77页，2013年。[5] L. Breiman。随机森林。《机器学习》，45：5–32，2001年。[6] B. Butcher 和 B. J. Smith。《特征工程与选择：预测模型的实用方法》，2020年。[7] H. Carlens。2022年竞争性机器学习的现状。mlcontests.com/state-of-competitive-machine-learning-2022/，2022年。访问日期：2023年4月1日。[8] N. Charlton 和 C. Singleton。用于短期负荷预测的改进参数模型。《国际预测学杂志》，30(2)：364–368，2014年。[9]T. Chen 和 C. Guestrin。Xgboost：一种可扩展的树提升系统。载于《第22届ACM SIGKDD知识发现与数据挖掘国际会议论文集》，第785–794页，2016年。[10]G. Corani、D. Azzimonti、J. P. Augusto 和 M. Zaffalon。通过贝叶斯规则对分层预测进行概率协调。载于《数据库中的机器学习与知识发现：欧洲会议，ECML PKDD 2020》，比利时根特，2020年9月14日至18日，论文集，第三部分，第211–226页。施普林格，2021年。[11] M. L. De Prado。《金融机器学习进展》，2018年。[12] A. V. Dorogush、V. Ershov 和 A. Gulin。Catboost：支持类别特征的梯度提升。arXiv预印本 arXiv:1810.11363，2018年。[13] T. Duan、A. Anand、D. Y. Ding、K. K. Thai、S. Basu、A. Ng 和 A. Schuler。NGBoost：用于概率预测的自然梯度提升。载于《国际机器学习会议》，第2690–2700页。PMLR，2020年。[14] H. Eristi、A. Ucar 和 Y. Demir。使用支持向量机对电力系统扰动进行分类的基于小波的特征提取与选择。《电力系统研究》，80(7)：743–752，2010年。[15] S. Fan 和 R. J. Hyndman。基于半参数加性模型的短期负荷预测。《IEEE电力系统汇刊》，27(1)：134–141，2011年。[16] J. H. Friedman。随机梯度提升。《计算统计与数据分析》，38(4)：367–378，2002年。

[17] P. Gaillard、Y. Goude 和 R. Nedellec。用于 gefcom2014 概率电力负荷和电价预测的加性模型与稳健聚合。《国际预测学杂志》，32(3):1038–1050，2016年。[18] T. Gneiting。作为最优点预测的分位数。《国际预测学杂志》，27(2):197–207，2011年。[19] T. Gneiting、A. E. Raftery、A. H. Westveld 和 T. Goldman。使用集合模型输出统计和最小连续秩概率得分估计进行校准概率预测。《月度天气评论》，133(5):1098–1118，2005年。[20] B. Gregorutti、B. Michel 和 P. Saint-Pierre。随机森林中的相关性与变量重要性。Statistics and Computing，27:659–678，2017年。[21] T. Hong。BigDeal 挑战赛 2022。blog.drhongtao.com/2022/10/ bigdeal-challenge-2022.html。访问日期：2023-04-09。[22] T. Hong。BigDeal 挑战赛 2022，决赛。blog.drhongtao.com/2022/12/bigdeal-challenge-2022-final-leaderboard.html。访问日期：2023-04-09。[23] T. Hong。BigDeal 挑战赛 2022，资格赛。blog.drhongtao.com/2022/11/bigdeal-challenge-2022-qualifying-match.html。访问日期：2023-04-09。[24] T. Hong、P. Pinson 和 S. Fan。全球能源预测竞赛 2012、2014。[25] T. Hong、P. Pinson、S. Fan、H. Zareipour、A. Troccoli 和 R. J. Hyndman。概率能源预测：全球能源预测竞赛2014及其后续发展，2016年。[26] T. Hong、J. Xie 和 J. Black。全球能源预测竞赛 2017：分层概率负荷预测。《国际预测学杂志》，35(4):1389–1399，2019年。[27] R. J. Hyndman 和 S. Fan。长期峰值电力需求的密度预测。《IEEE电力系统汇刊》，25(2):1142–1153，2009年。[28] G. James、D. Witten、T. Hastie 和 R. Tibshirani。《统计学习导论》，2013年。[29] T. Januschowski、Y. Wang、K. Torkkola、T. Erkkil¨a、H. Hasson 和 J. Gasthaus。基于树的预测。《国际预测学杂志》，38(4):1473–1481，2022年。

[30] G. Ke、Q. Meng、T. Finley、T. Wang、W. Chen、W. Ma、Q. Ye 和 T.-Y. Liu。LightGBM：一种高效的梯度提升决策树。神经信息处理系统进展，30，2017年。[31] N. Kourentzes 和 G. Athanasopoulos。阐明间歇性需求序列的结构。《欧洲运筹学杂志》，288(1):141–152，2021年。[32] J. Li、K. Cheng、S. Wang、F. Morstatter、R. P. Trevino、J. Tang 和 H. Liu。特征选择：数据视角。ACM计算调查（CSUR），50(6):1–45，2017年。[33] H. Lu、F. Cheng、X. Ma 和 G. Hu。采用改进的极端梯度提升模型进行建筑能耗短期预测：进水塔案例研究。《能源》，203:117756，2020年。[34] S. M. Lundberg 和 S.-I. Lee。一种解释模型预测的统一方法。神经信息处理系统进展，30，2017年。[35] S. Makridakis、E. Spiliotis 和 V. Assimakopoulos。M5准确度竞赛：结果、发现与结论。《国际预测学杂志》，38(4):1346–1364，2022年。[36] G. Maroni、L. Pallante、G. Di Benedetto、M. A. Deriu、D. Piga 和 G. Grasso。通过可解释机器学习对甜味剂/苦味剂化合物进行有依据的分类。《食品科学当前研究》，5:2270–2280，2022年。[37] A. M¨arz。Xgboostlss——将 XGBoost 扩展至概率预测。arXiv预印本 arXiv:1907.03178，2019年。[38] A. M¨arz 和 T. Kneib。分布梯度提升机。arXiv预印本 arXiv:2204.00778，2022年。[39] N. Meinshausen 和 G. Ridgeway。分位数回归森林。《机器学习研究杂志》，7(6)，2006年。[40] C. Molnar。《可解释机器学习》。第2版，2022年。[41]L. Nespoli 和 V. Medici。多变量提升树及其在预测与控制中的应用。《机器学习研究杂志》，23(246):1–47，2022年。[42] R. Shwartz-Ziv 和 A. Armon。表格数据：深度学习并非你所需的一切。《信息融合》，81:84–90，2022年。

[43] S. Smyl 和 N. G. Hua。用于 gefcom2017 概率负荷预测的机器学习方法。《国际预测学杂志》，35(4):1424–1431，2019年。[44]S. B. Taieb 和 R. J. Hyndman。面向 kaggle 负荷预测竞赛的梯度提升方法。《国际预测学杂志》，30(2):382–394，2014年。[45] R. K. Vinayak 和 R. Gilad-Bachrach。DART：随机失活与多重加性回归树相遇。载于《人工智能与统计学》，第489–497页。PMLR，2015年。[46] P. Wang、B. Liu 和 T. Hong。考虑近期效应的电力负荷预测：一种大数据方法。《国际预测学杂志》，32(3):585–597，2016年。[47] Y. Wang、S. Sun、X. Chen、X. Zeng、Y. Kong、J. Chen、Y. Guo 和 T. Wang。基于 SVMD 和 XGBoost 的工业客户短期负荷预测。《国际电力与能源系统杂志》，129:106830，2021年。[48] S. L. Wickramasuriya、G. Athanasopoulos 和 R. J. Hyndman。通过迹最小化实现层级和分组时间序列的最优预测协调。《美国统计学会杂志》，114(526):804–819，2019年。[49] J. Yu。利用局部保持投影和高斯混合模型评估轴承性能退化。《机械系统与信号处理》，25(7):2573–2588，2011年。[50] F. Ziel。gefcom2017 概率负荷预测资格赛的分位数回归。《国际预测学杂志》，35(4):1400–1408，2019年。