Towards understanding the importance of time-series features in automated algorithm performance prediction

Abstract:

对于特定 TS 很难选择 forecasting algo

本文提供了方法和分析，旨在理解预测方法的重要性。

对于不同方法 可以关联上不同特征

量化重要性，有两个变量（meta-learning models & forecasting algo）

运用了不同重要性方法

Methodology：

Data collection:

数据集：M4 数据集 + 61 个 forecasting algo 与数据集的预测结果

提取特征：在提取前， 需要 **transform** TS，来稳定方差，避免例如来自季节的影响， 此处使用 logarithmic transformation。

使用 tsfresh + catch22

评估 forecasting algo: sMAPE，越小越好

Building a diverse portfolio of performance prediction models：

建造一个预测模型，用于提供对于特征重要性的解释。

训练后，可以得到用了哪个特征来预测+不同数据用了不同特征

Feature importance analysis：

用不同方法来计算重要性：Permutation, SHAP, XGBoost

Experiment:

Feature space analysis:

计算 correlation 来选择特征，结论是：方法相同，使用不同参数的 cor 很大；tsfresh 和 catch22 之间关系不大；对于 catch22， 未处理数据和 transformed 数据的 cor 较高，而 tsfresh 或高或低。

Performance space analysis:

整体来说，除了 algo225， 预测算法之间呈现高相关度。

Feature importance for explainability of forecasting algorithms：

展示了 top20 的特征重要性，基于 theta 和 arima 算法

Limitations:

Selection of the forecasting algo 过老（2020）， 可能有更先进的方法，因此本文侧重于选择预测算法时，决定特征重要性的一个方法。

当我们把 TS 分成训练集和测试集时，有些预测算法由于可能获取过测试集中的信息，结果可能会受到影响。

选用 sMAPE 的原因：M4 使用了 sMAPE；是相对 error 而不是绝对 error

结论：

大多数特征有强相关性

Tsfresh 表现更好

算法表现也有强相关性

未来可做：

Hyperparameter, multivariate, applying a similar pipeline

问题：

Transform 的必要性？

logarithmic transformation 只有在数据都为正数， 数据变化量随着时间增加指数级增加时可以使用

是否需要对比一下除了 sMAPE 以外的方法

Pearson correlation 的缺点：

1. 只对重叠的记录进行计算，举例来说，2 位评分相同，看了 200 部电影的用户，相关性会小于只看了一部电影且打分相同的观众。
2. 对于绝对数值不敏感，只关注变化趋势。

引出思考：使用 correlation 选出特征，是否可以对比一下别的计算方法？

特征选择的目的：减少特征数量、降维，使模型泛化能力更强，减少过拟合
增强对特征和特征值之间的理解

特征选择的方法：去掉取值变化小的特征
单变量特征选择
Pearson 相关系数
互信息和最大信息系数
距离相关系数
基于学习模型的特征排序
线性模型和正则化
随机森林
稳定性选择
递归特征消除

Do Feature Attribution Methods Correctly Attribute Features?

Abstract:

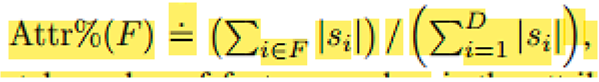
对于 feature 的贡献没有共识，也就是说没有一个系统性的评估。

本文提供了一个 procedure 来评估 3 种方法：saliency maps, attentions & rationales.

Desiderata for Attribution Values：

把 feature 分为两类：fundamental F_c, non-informative F_n

定义了 Attr%(F)，希望值 Attr%(F_c) ≈ 1, Attr%(F_n) ≈ 0



$$\mathrm{Attr}\%(F) \doteq \left(\sum_{i\in F} |s_i|\right) / \left(\sum_{i=1}^{D} |s_i|\right),$$

Dataset Modifcation with Ground Truth

分为两步：label reassignment 和 input manipulation

Label reassignment: 当有新的 feature 加入时，模型可能会无视新的 feature 来获得高性能，即使新特相关性更高，因此，这个方法可以削弱原先的特征和 label 的相关性。

Input manipulation:

实验：

对于 saliency maps, text attentions & text rationales 进行了实验

结论：

根据实验结果，没有一种方法获得了满意的表现

Feature Importance Explanations for Temporal Black-Box Models

Abstract：用一个叫做 TIME 的方法来解释模型

1. 用 model-agnostic permutation-based approach 来分析特征重要性
2. 用时间顺序和局部影响来辨别显著特征
3.

TIME 方法的用途：

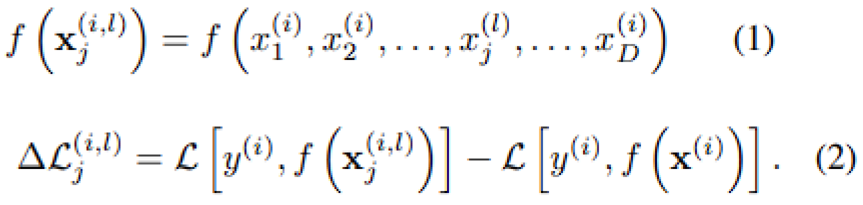
1. 辨别了特征们在不同分布上的重要性
2. 对于每个特征，辨别了其最重要的区域
3. 决定了是否预测与值的顺序有关
4. 用 false discovery rate control methodology 来辨别重要特征
5. 适用于黑盒

方法：

Identifying Important Features/Timesteps：

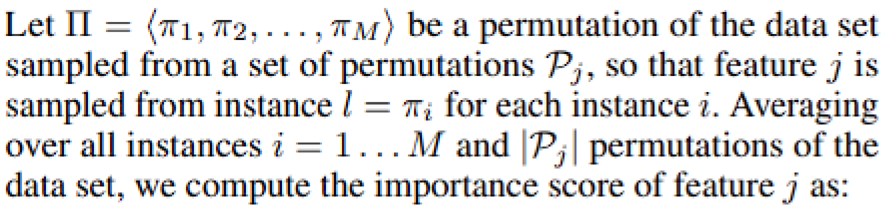
1. Non-temporal models:

   对于一对（x_i, y_i）x_i 是 i_th_instance，y 是 target，用 l_th_instance 来替换 l，得到被扰动的输出，计算替换后，loss 函数的变化，从而计算对于 feature_j 的重要性得分，f(x)为 model output



$$f\left(\mathbf{x}_j^{(i,l)}\right) = f\left(x_1^{(i)}, x_2^{(i)}, \ldots, x_j^{(l)}, \ldots, x_D^{(i)}\right) \qquad (1)$$

$$\Delta\mathcal{L}_j^{(i,l)} = \mathcal{L}\left[y^{(i)}, f\left(\mathbf{x}_j^{(i,l)}\right)\right] - \mathcal{L}\left[y^{(i)}, f\left(\mathbf{x}^{(i)}\right)\right]. \qquad (2)$$



Let $\Pi = \langle \pi_1, \pi_2, \ldots, \pi_M \rangle$ be a permutation of the data set sampled from a set of permutations $\mathcal{P}_j$, so that feature j is sampled from instance $l = \pi_i$ for each instance i. Averaging over all instances $i = 1 \ldots M$ and $|\mathcal{P}_j|$ permutations of the data set, we compute the importance score of feature j as:

$$I\left(f,j\right)=\frac{1}{\left|\mathcal{P}_j\right|}\sum_{\Pi\in\mathcal{P}_j}\left[\frac{1}{M}\sum_{i=1}^{M}\Delta\mathcal{L}_j^{(i,\pi_i)}\right]. \tag{3}$$

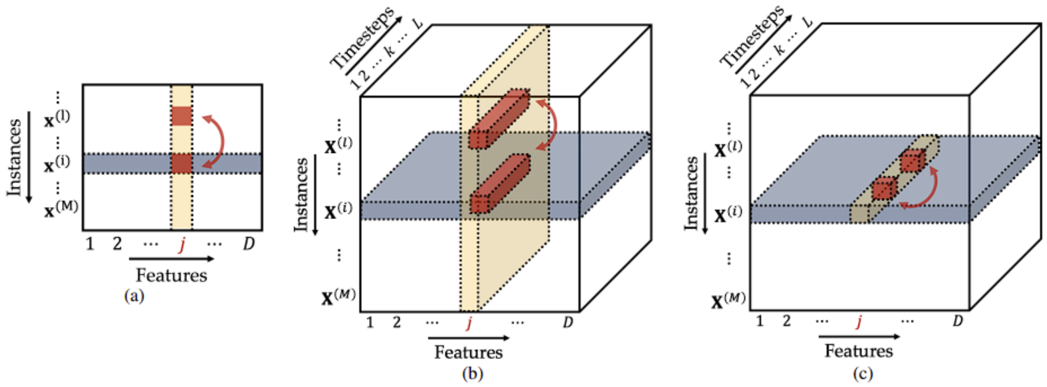


Figure 2: Perturbation for instance i and feature j to compute feature importance. (a) Data matrix showing the replacement of the value of feature j in instance i from instance l. (b) Data tensor showing the replacement of a window of feature j in instance i from the corresponding window of instance l . (c) Time series $\mathbf{x}_j^{(i)}$ showing the exchange of feature values at two timesteps.

2. Temporal models.

   如图 b 增加了时间轴

$$f\left(\mathbf{X}_{j,[k_1,k_2]}^{(i,l)}\right)=f\left(\mathbf{x}_1^{(i)},\mathbf{x}_2^{(i)},\ldots,\mathbf{x}_{j,[k_1,k_2]}^{(i,l)},\ldots,\mathbf{x}_D^{(i)}\right) \tag{4}$$

[k1,k2]是时间区间

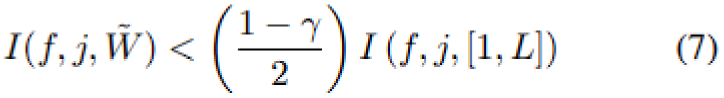
$$\mathbf{x}_{j,[k_1,k_2]}^{(i,l)}=\left\langle x_{j,1}^{(i)},x_{j,2}^{(i)},\ldots,x_{j,k_1}^{(l)},\ldots,x_{j,k_2}^{(l)},\ldots,x_{j,L}^{(i)}\right\rangle. \tag{5}$$

We compute the perturbed loss $\mathcal{L}\left[y^{(i)},f\left(\mathbf{X}_{j,[k_1,k_2]}^{(i,l)}\right)\right]$ and the change in loss (Equation 2) for instance i. We average this over all instances $i=1\ldots M$ and $|\mathcal{P}_j|$ permutations of the data set to compute the importance score corresponding to the window $[k_1,k_2]$ for feature j:

$$I\left(f,j,[k_1,k_2]\right)=\frac{1}{\left|\mathcal{P}_j\right|}\sum_{\Pi\in\mathcal{P}_j}\left[\frac{1}{M}\sum_{i=1}^{M}\Delta\mathcal{L}_{j,[k_1,k_2]}^{(i,\pi_i)}\right]. \tag{6}$$

The overall importance $I\left(f,j,[1,L]\right)$ of feature j is computed by selecting $k_1=1$ and $k_2=L$.

Identifying Important Windows:



$$I(f,j,\tilde{W}) < \left(\frac{1-\gamma}{2}\right) I\left(f,j,[1,L]\right) \qquad (7)$$

W = W_p + W* + W_s, W* = [k1,k2], 1 <=k1<k2<=L

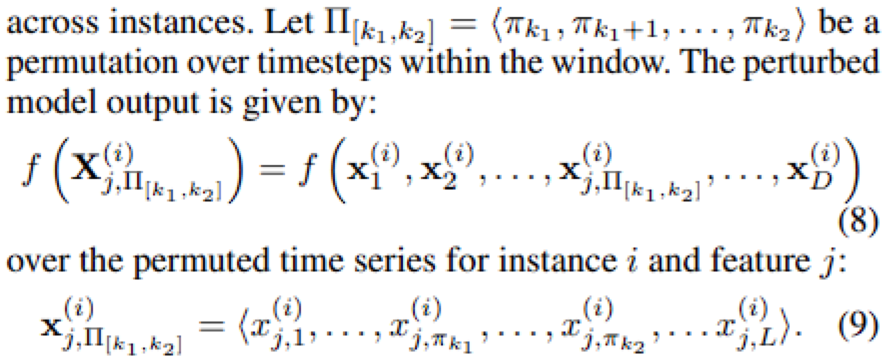
γ : 0 < γ < 1 controls the degree to which the model focuses on W* and affects the size of the identified windows.

W_p = prior Window, W_s = subsequent Window, 寻找这样两个区间,范围大但是重要性小

通过二分法找到这个区间

疑问：是否应该切割 time series 来获得更高性能

Identifying the Importance of Feature Ordering



across instances. Let $\Pi_{[k_1,k_2]} = \langle \pi_{k_1}, \pi_{k_1+1}, \ldots, \pi_{k_2} \rangle$ be a permutation over timesteps within the window. The perturbed model output is given by:

$$f\left(\mathbf{X}^{(i)}_{j,\Pi_{[k_1,k_2]}}\right) = f\left(\mathbf{x}^{(i)}_1, \mathbf{x}^{(i)}_2, \ldots, \mathbf{x}^{(i)}_{j,\Pi_{[k_1,k_2]}}, \ldots, \mathbf{x}^{(i)}_D\right) \qquad (8)$$

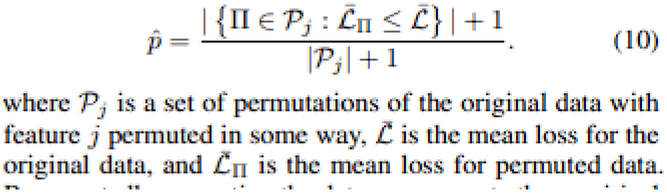
over the permuted time series for instance i and feature j:

$$\mathbf{x}^{(i)}_{j,\Pi_{[k_1,k_2]}} = \langle x^{(i)}_{j,1}, \ldots, x^{(i)}_{j,\pi_{k_1}}, \ldots, x^{(i)}_{j,\pi_{k_2}}, \ldots x^{(i)}_{j,L} \rangle. \qquad (9)$$

计算根据时间排序，feature 重要性

Hypothesis Testing and False Discovery Rate Control

用 hypothesis test 来测试统计学上的 因改变 feature 排列顺序而使性能下降的 显著性（就是计算 P-value）



$$\hat{p} = \frac{\left|\left\{\Pi \in \mathcal{P}_j : \bar{\mathcal{L}}_\Pi \le \bar{\mathcal{L}}\right\}\right| + 1}{|\mathcal{P}_j| + 1}. \qquad (10)$$

where $\mathcal{P}_j$ is a set of permutations of the original data with feature j permuted in some way, $\bar{\mathcal{L}}$ is the mean loss for the original data, and $\bar{\mathcal{L}}_\Pi$ is the mean loss for permuted data.

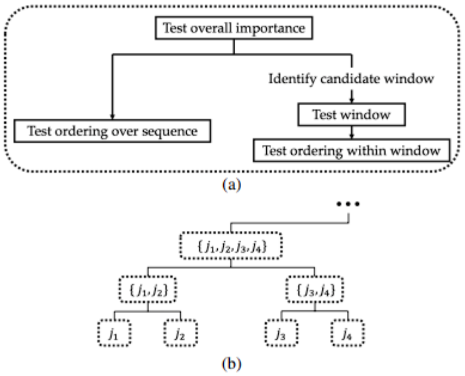


Figure 3: (a) A hierarchy of tests used to check a given feature for its (i) overall importance, (ii) important window and (iii) the importance of ordering within the window. (b) A hierarchy over the features, where each node is tested using the hierarchy shown in (a). Feature groups are tested via joint permutations of their constituent features. Hierarchical FDR control is used for multiple testing correction, and subtrees rooted at nodes with p-values above a threshold are pruned.

结论：TIME 能鉴别一组的特征重要性，且能找出在哪个时间区间，它有很显著的影响，实验表明他的性能强于 baseline methods（LIME, Feature Occlusion, CXPlain, SAGE, PERM）

Future works: 增加 limitations of TIME，目前方法只适用于 regularly sampled, time-aligned and fixed length sequences，未来可能会跳出这个框架。TIME 能鉴别一组的特征重要性，且能找出在哪个时间区间，它有很显著的影响，实验表明他的性能强于 baseline methods（LIME, Feature Occlusion, CXPlain, SAGE, PERM）

Future works: 增加 limitations of TIME，目前方法只适用于 regularly sampled, time-aligned and fixed length sequences，未来可能会跳出这个框架。

Feature selection based on mutual information with correlation coefficient

Abstract:

提供了一种基于 correlation coefficient (CCMI 算法)的 feature selection

介绍了 correlation coefficient，并将其与相互的信息结合来测量不同特征之间的关系

用不同特征的 correlation coefficient 的绝对值来作为冗余信息的权重

Related works:

MIM 算法只考虑 feature 和 label 的关系，用于排除不相关的特征，而不会考虑特征与特征之间的关系

$$J(X_m) = I(X_m; C)$$

Mutual Information based on Feature Selection (MIFS) 可以辨别特征之间的重复度

$$J(X_m) = I(X_m; C) - \beta \sum_{X_s \in S} I(X_m; X_s)$$

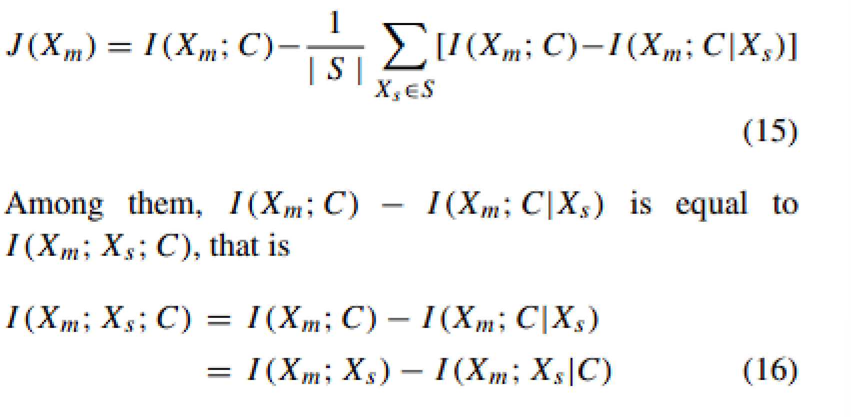
Minimal-redundancy-maximal-relevance (mRMR) 是 MIFS 的变种，将 beta 改为 1/S，来避免当选择的特征变多，越来越难挑选出重复的特征的情况

$$J(X_m) = I(X_m; C) - \frac{1}{|S|} \sum_{X_s \in S} I(X_m; X_s)$$

Conditional Informative Feature Extraction (CIFE) 提供了更准确的重复信息

$$J(X_m) = I(X_m; C) - \sum_{X_s \in S} [I(X_m; X_s) - I(X_m; X_s|C)]$$

Joint Mutual Information (JMI) 通过 joint mutual information 来测量特征与特征，label 间的关系，是 CIFE 的变种，用于获取 redundancy item



$$J(X_m) = I(X_m; C) - \frac{1}{|S|} \sum_{X_s \in S} [I(X_m; C) - I(X_m; C|X_s)] \tag{15}$$

Among them, $I(X_m; C) - I(X_m; C|X_s)$ is equal to $I(X_m; X_s; C)$, that is

$$\begin{aligned} I(X_m; X_s; C) &= I(X_m; C) - I(X_m; C|X_s) \\ &= I(X_m; X_s) - I(X_m; X_s|C) \end{aligned} \tag{16}$$

RelaxFS 一种测量 redundancy 的新方法，包含了更多冗余信息

$$J(X_m) = I(X_m;C) - \frac{1}{|S|}\sum_{X_j \in S} I(X_m;X_j) + \frac{1}{|S|}\sum_{X_j \in S} I(X_m;X_j|C) - \frac{1}{|S||S-1|}\sum_{X_j \in S}\sum_{X_i \in S} I(X_m;X_i|X_j) \qquad (17)$$

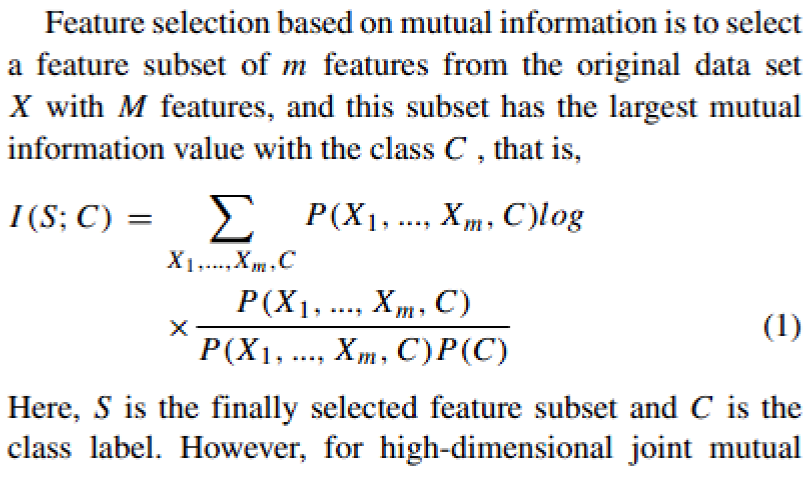
Max-Relevance and Max-Independence (MRI) algorithm 可以选择低重复，有区别的特征

$$J(X_m) = I(X_m;C) + \sum_{X_s \in S} ICI(C;X_s,X_m) = I(X_m;C) + \sum_{X_s \in S} [I(C;X_m|X_s) + I(C;X_s|X_m)$$

Composition of Feature Relevancy (CFR) 用 new classification information 和 redundant information 来得到 feature relevancy

$$J(X_m) = \sum_{X_s \in S} I(C;X_m|X_s) - I(C;X_m;X_s) \qquad (19)$$

Method:



Feature selection based on mutual information is to select a feature subset of m features from the original data set X with M features, and this subset has the largest mutual information value with the class C , that is,

$$I(S;C) = \sum_{X_1,\ldots,X_m,C} P(X_1,\ldots,X_m,C) log \times \frac{P(X_1,\ldots,X_m,C)}{P(X_1,\ldots,X_m,C)P(C)} \qquad (1)$$

Here, S is the finally selected feature subset and C is the class label. However, for high-dimensional joint mutual

Problem:

此方法很难直接计算，通常都是用低维信息来估计高维信息

$$J(X_m) = I(X_m;C) - \beta \sum_{X_s \in S} I(X_m;X_s) + \gamma \sum_{X_s \in S} I(X_m;X_s|C)$$

基于这个公式，对于 beta 和 gamma 进行改进（beta 影响特征之间，gamma 影响特征和 label 之间）

运用了 correlation， 0-1 线性关系递增，cov=covariance, d = variance

$$\rho_{xy} = \frac{Cov(X,Y)}{\sqrt{D(X)} \bullet \sqrt{D(Y)}} \tag{21}$$

$$Cov(X,Y) = E\{[X - E(X)][Y - E(Y)]\} \tag{22}$$

$$J_{CMIM}(X_m) = \min_{X_s \in S} I(X_m; C|X_s) \tag{25}$$

$$J_{JMI}(X_m) = \frac{1}{|S|} \sum_{X_s \in S} I(X_m; C|X_s) \tag{26}$$

，相结合

$$J_{CCMI}(X_m) = \min_{X_s \in S} I(X_m; C|X_s) - \min_{X_s \in S} |\rho_{X_m X_s}| \cdot I(X_m; X_s) \tag{27}$$

$$\rho_{X_m X_s} = \frac{Cov(X_m, X_s)}{\sqrt{D(X_m)} \bullet \sqrt{D(X_s)}} \tag{28}$$

Complexity analysis:

K 是选择的特征，N 是特征总数，M 是 sample 数量，那么 MIM 方法是 O（MN），RelaxFS 是 O（K^3MN），CCMI 是 O(K^2MN)

Conclusion:

本文的重点是研究了一种算法，用于选择高相关低重复的特征。Mutual information 用于测量 label 和特征或特征与特征之间的关系，correlation coefficient 用于测量特征之间的重复度，其绝对值用于测量重复值的重要性。