TCN-Attention特征融合成本预测模型

由附件3的历史成本数据，不同季节时同一种蔬菜批发价格不同，考虑到蔬菜批发价格受季节变化影响较强（如下图所示），我们将在对7月1日到7月6日的各品类蔬菜批发价格预测中考虑季节的影响。我们将二十四节气和对应产品品类的时间序列，预测未来七天平类加权平均正本的变化。成本预测模式使用TCN-Attention特征融合神经网络模型，实现多输入单输出的训练和预测。

**数据预处理**

1. 二十四节气层次one-hot编码

本文选取中国传统文化中的二十四节气作为季节，时间对某一特定蔬菜品类加权平均成本的影响因子之一，本文将某一个特定日期编码为其相距时间最短的节气为进行神经网络模型训练，需对其进行编码。中国农历中有二十四个节气，如果使用简单的数字编码会导致编码结果缺乏周期信息，同时由于二十四节气在时间序列上是周期性的，如果采用数字编码会导致一年最后一个节气(大寒)和第一个节气(立春)编码结果相差巨大，与实际情况不符。

如果使用简单one-hot编码会导致维度稀疏问题，会影响模型学习数据集特征的能力。因此我们使用层次one-hot编码方法对二十四节气进行编码。

层次one-hot编码的特点在于利用了季节和节气之间的从属关系，一个季节存在六个节气，我们首先对季节进行one-hot编码，随后按照一个季节中的六个节气的先后顺序，对六个节气进行one-hot编码，将两个结果横向拼接得到编码结果。

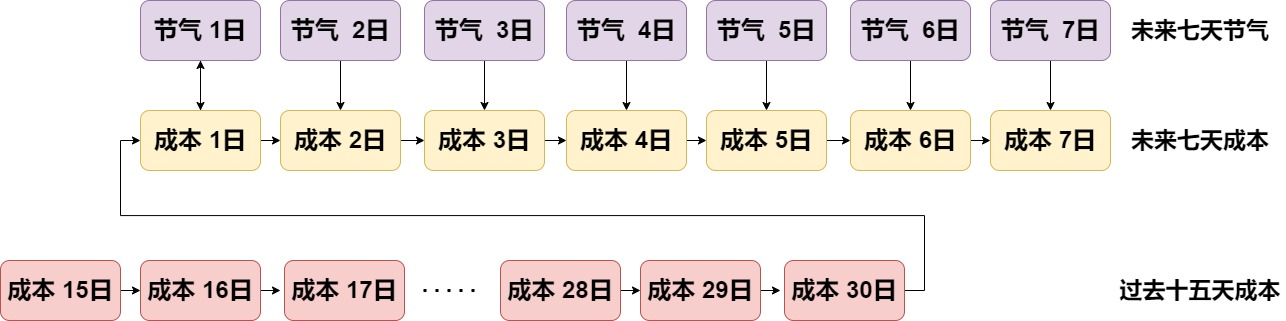
|  |  |
| --- | --- |
| **节气** | **二十四节气层次one-hot编码结果** |
| 立春 | [1,0,0,0,1,0,0,0,0,0] |
| 清明 | [1,0,0,0,0,0,0,0,1,0] |
| 大寒 | [0,0,0,1,0,0,0,0,0,1] |

采用层次one-hot编码使得节气信息训练数据相较one-hot编码维度收缩53.8%，显著提升编码结果信息表达能力。

1. 时间序列滑动窗口

本模型为实现对未来七天某一品类成本的预测，需要使用滑动窗口对数据进行预处理，即在模型训练前，将数据集分割为多个特定长度的窗口用于从历史数据中提取有用信息和模式。

1. **窗口设置**：滑动窗口被设置为15天，意味着模型将考虑过去15天的数据来预测未来7天的成本。
2. **节气信息**：除了过去15天的成本数据外，该模型还将使用未来7天的节气信息。节气信息作为一个附加的特征输入。
3. **窗口滑动**：窗口以一天为步长滑动，生成新的数据子集，用于训练。



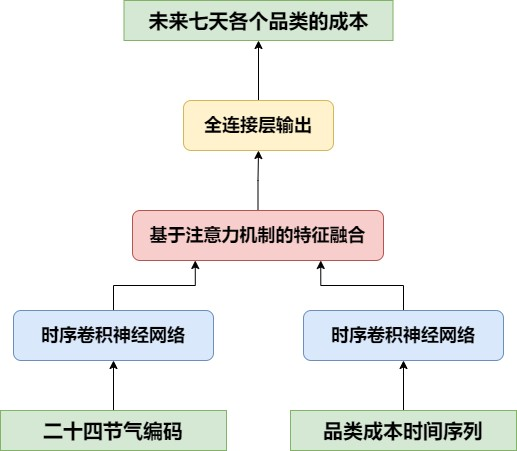
通过滑动窗口，你可以在一个小的时间段内观察变量，从而更好地捕捉其动态性或季节性。这使得模型能够学习这些短期依赖关系，并据此进行更准确的预测。

1. 数据归一化

当二十四节气编码和滑动窗口完成之后需要对所有数据进行归一化，本文使用的归一化算法为MinMaxScaler算法。

**模型结构**

本文构建的模型为时序卷积神经网络-注意力机制特征融合模型(TCN-Attention-feature-fusion Model)，由两个输入层和一个特征融合层以及一个全连接输出层构成。



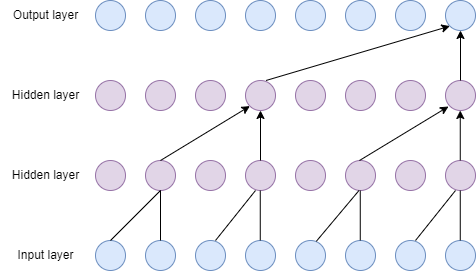
1. **时序卷积层**

时序卷积（TCN）层在本模型中是模型的输入层，本模型的两个时序卷积神经网络分别作为二十四节气编码和品类成本时间序列的输入层。时序卷积神经网络的特征在本模型中可以优化神经网络模型的拟合精度，并且增强了模型在时间序列上的推理能力。

1. **因果卷积**：TCN中的因果卷积是一种特殊的卷积操作，其主要特点是在计算当前时间步（或点）的输出时，仅使用该时间点及其之前的信息，而不使用任何未来信息。这种方式确保了输出序列与输入序列之间存在明确的因果关系，即输出仅由输入及其历史信息决定。数学上，给定一个输入序列，因果卷积可以表述为：

其中是给定的时间序列，是卷积核在时间的值，是卷积核的长度。因果卷积同时限制模型从未来数据获取信息的能力，提高了模型推理能力。

1. **空洞卷积(Dilated Convolution)**：相较于传统序列神经网络如LSTM等，本模型使用TCN作为输入层具有空洞卷积的特征，空洞卷积的重要特征是其卷积核并不是一个连续卷积核，而是在时间序列上以一个较大的间隔进行卷积，在数学上可以表示为：



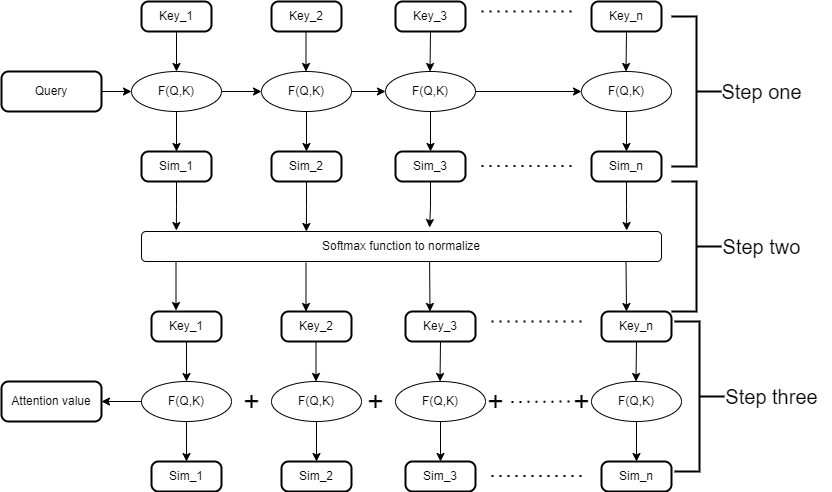
空洞卷积可以增强感受视野，由于空洞卷积具有较大视野有助于模型捕捉不同季节（节气）或长期趋势对蔬菜价格的影响，从而提高预测准确性。

1. **基于注意力机制的特征融合**

本模型的特点是多输入单输出，在训练和预测的过程中需要进行特征融合，即将两个输入模型输入的特征矩阵进行融合。

注意力机制自被发表依赖已经广泛地被用于各种神经网络模型中，尤其是处理如本模型一样的多信息源模型或者多模态模型。在外面的模型中，主要的输入有两个，一个是经过时序卷积网络（TCN）处理的蔬菜品类成本时间序列，另一个是经过时序卷积网络（TCN）二十四节气时间序列的层次one-hot编码。为了有效结合这两种信息进行准确预测，本文映入了基于注意力机制的特征融合层。

1. 注意力机制



在神经网络模型中，注意力机制是一种允许网络模型在处理不同输入同时自动关注重要信息的机制，可以帮助网络实现注入选择性注意力，动态调整，减少干扰和提高可解释性性能。在本文中，我们使用二十四节气特征作为键值，历史成本时间序列特征作为查询值，输出未来成本时间序列作为查询值，本模型注意力机制的运算流程如下：

第一步，横向拼接两个特征矩阵，计算每个query和每个Key，得到权重系数:

第二步，使用Softmax对权重进行归一化处理，得到每个特征的注意力权重：

第三步，有了注意力权重后，我们可以计算加权平均的融合特征：

融合特征后，我们通过一个全连接输出层进行最终预测。这一层的目的是将融合的特征映射到成本预测的具体值。

通过对六个品类的农产品分别进行模型训练得到六个预测模型，得到了未来七天六种农产品的具体单位成本。

**TCN-Attention Bootstrap抽样时间序列预测模型**

为了对未来一周的销售量区间进行预测，我们采用了基于TCN-Attention的Bootstrap抽样时间序列预测模型，通过对同一数据集不同切分方式得到的数据集分别训练神经网络模型，通过多个模型对未来七天销量的的预测，得到一个置信区间作为未来七天销售量的范围。

**数据预处理**

**时间窗口-归一化**

对于本模型的训练，我们使用过去七天的销售量预测未来一天的销售量，所以我们设置时间窗口大小为7，获得完整的待训练数据。完成滑动窗口处理之后使用MinMaxScaler算法对其实现归一化。

**Bootstrap sampling**

为了预测未来一段时间的六种品类的农产品的销售情况区间，我们需要对各个品类分别训练100个模型。为了更好地学习数据集中的依赖模式，同时增加模型训练过程中的噪音，我们对同一品类下不同模型分别训练在不同训练数据集，对于每一个品类的模型，我们在完整归一化之后的完整待训练数据集中切分出100个不同的训练集，为了保证时间序列的完整性，我们需要切分的是连续的数据集而非离散的。

我们在每个品类不同数据集上分别训练模型，得到每个品类的100个神经网络模型，预测出未来七天，六种特定品类的销售量区间，这个区间符合正态分布。

**最优化模型**

TCN-Attention Bootstrap抽样时间序列预测模型的预测得到了六种品类在未来七天内的销售量区间，我们通过线性回归得到了六种品类，价格和销售量之间的关系，本模型使用BFGS算法在限制条件内进行最优化。

**目标函数**

其中，是总收益、是品类数量、是第个品类的价格、是第个品类的c成本、是在价格下的预测销量。

**约束条件**

价格-成本约束：，保证产品的售价不能低于成本。

价格上限-下限约束：，，其中和分别是TCN-Attention Bootstrap抽样时间序列预测模型预测出六种品类的销售量的上限和下限。

销量约束：，我们需要预期销量不能为0。

**BFGS算法**

BFGS算法是一种拟牛顿方法，适用于可微分的目标函数，BFGS的基本思想是通过对Hessian矩阵（即目标函数的二阶导数矩阵）的近似来进行优化。

**基本步骤：**

1. 初始化：选择一个起始点，在本模型中起始点是随机生成的六个品类在未来七天.的定价和一个初始近似Hessian的矩阵。
2. 迭代：对于执行以下操作：
3. 搜索方向：计算搜索方向，作为解的。
4. 线搜索：找到一个步长，使得，其中。
5. 更新近似Hessian：计算和的差值，和，并且用这些更新为。
6. 收敛检查：如果满足某个收敛标准则停止迭代。

通过对其进行最优化得到了以下定价策略。

问题三

**TCN-Attention农产品单品成本预测模型模型**

本模型和前文预测品类成本的模型一致，修改了时间窗口大小为7-1，即使用过去7天的成本历史数据，预测未来一天一个特定农产品的成本。

使用MinMaxScaler进行数据归一化，通过训练和预测得到了33种农产品的成本价。

**TCN-Attention Bootstrap抽样时间序列售价范围预测模型**

遗传算法具有贪心属性，其会在一切可能的价格区间中寻求利润最大化，为了使得模型最优化结果更合理，我们需要对价格进行TCN-Attention Bootstrap抽样时间序列模型预测。

该模型与问题二的Bootstrap模型一样，使用滑动窗口大小为7-1，使用过去七天的历史售价预测未来一天的售价，33种农产品每一种切分100个子数据集训练100个模型，预测获得其价格区间。

**遗传算法最优化模型**

对于33种农产品，我们需要在限制条件下对齐实现收益最优化。我们使用遗传算法对其进行最优化处理。遗传算法是一种优化和选择算法，其与自然界中种群进化的过程类似。模型通过在不断迭代中启发式搜索，交叉，变异得到全局最优解。

**基本原理和组成部分**

种群：一组可能的解（个体）构成一个种群。种群的大小是预先设定的。

染色体（Chromosome）：每一个解都可以被表示为一个“染色体”——通常是一个位串（bit string）、数组或其他数据结构。

适应度函数（Fitness Function）：该函数用于评估每个个体（染色体）的质量或优势，即它们的“适应度”。

选择（Selection）：种群中的个体按照他们的适应度被选择以产生下一代。

交叉（Crossover）/重组（Recombination）：选出的个体通过交叉或重组生成新的后代。这一过程模仿了生物的性繁殖。

变异（Mutation）：某个小概率下，新生成的后代会发生变异，增加种群的多样性。

**限制条件**

1. 定价在Bootstrap sampling预测出的价格区间内。

2. 定价和销量必须大于0。

3. 如果补货量不为0，至少等于2.5。

**运算过程**

1. 模型首先初始化100组配货量矩阵，矩阵中的所有元素均为随机生成。
2. 根据前面线性回归模型得到的33个产品价格和销量的关系以及TCN-Attention预测出每一个产品当天的进货成本，适应函数通过计算100组配货量的总利润作为适应度。
3. 本模型设置了选择数量为3个，模型根据第二步计算的适应度选择适应度最优的三个配货计划作为下一迭代的亲本。
4. 对于选择出来的三个配货计划（染色体），本模型混合因子设置为0.5即意味着新的后代将“均匀”地从其两个亲本那里继承特质。这种交叉方式旨在找到一个介于两个亲本之间的解，这通常有助于维持种群的多样性，同时还能在一定程度上加速收敛。
5. 本模型自定义了高斯变异（Gaussian Mutation）作为变异函数。高斯变异（Gaussian Mutation）是一种用于连续或实数编码的基因（例如，浮点数）的变异方法。这种变异方法的核心思想是对选定的基因值添加一个从高斯分布（也称为正态分布）抽取的随机数。

数学上高斯分布的概率密度定义如下：

在高斯变异中，我们通常得到一个原始值，然后从一个均值为和标准差为的高斯分布中取随机数，将其添加到原始值上，以获得新的基因型。

1. 通过不断迭代得到最优化的结果。

