基本参数

T为取样周期

t_0 为开始时间点，t_{-1}为结束时间点

K为样本总数，t_K为样本总时长，有$t_K = t_{-1} - t_0 = KT$

L为段的样本长度，t_L为段的时间长度，$t_L = LT$

F为需要预测的长度，t_F为需要预测的时间长度，$t_F = FT$

τ为主周期，ϕ为主频率，计算有以下几种方式：

1. 取振幅最大：通过$\phi = \mathrm{FreqWithMaxAmp}(\mathrm{FFT}(x))$得出，有$\tau = 1/\phi$
2. 有阈值取振幅最大：通过$\zeta = \mathrm{Freq}(\mathrm{top}_k(\mathrm{FFT}(x))) \in \mathrm{R}^{k\times 1}$得到$k$个振幅最大的周期，$\theta$是筛选阈值，有$\phi = \mathrm{FirstMoreThan}_\theta(\zeta)$和$\tau = 1/\phi$
3. 对 topk 的频率和周期做多尺度混合？？？？？？？？？？？
4. 通过 FFT 的 topk 的频率振幅加权平均得到：

$$\alpha = \mathrm{Amp}(\mathrm{top}_k(\mathrm{FFT}(x))) \in \mathrm{R}^{k\times 1}, k \in [1, K] \tag{1}$$

$$\zeta = \mathrm{Freq}(\mathrm{top}_k(\mathrm{FFT}(x))) \in \mathrm{R}^{k\times 1} \tag{2}$$

$$\phi = \mathrm{WeightedMean}_\alpha(\zeta) = \frac{\alpha\zeta^T}{\alpha\alpha^T} \tag{3}$$

$$\tau = 1/\phi \tag{4}$$

b为 PF 间断点，t_b为 PF 间断时间点，$t_b = t_0 + bT$

段定义

$s_0 = x[t_b - t_L : t_b]$

$s_1 = x[t_b - t_L - \tau : t_b - \tau]$

$s_m = x[t_b - t_L - m\tau : t_b - m\tau]$，最多$m+1$段

$s_i = x[t_b - t_L - i\tau : t_b - i\tau]$，其中$i \in [0, m]$

想要预测F个数据，需要涉及的过去的数据的时间长度$t_P = t_L + m\tau = PT$，过去的数据长度$P = t_P/T$

训练参数

a为过去数据的开始点，t_a为过去数据的开始时间点，有$t_a = t_0 + aT$，为了训练时可以评估与验证，需要使得$t_a \in [t_0, t_{-1} - t_P - t_F]$，即$\Delta t_a = t_{-1} - t_0 - t_F - t_P = t_K - t_F - t_P$

总行数为$N = \lfloor \frac{\Delta t_a}{t_p} \rfloor$

n为每批行数

B为批数，有$B = \lfloor \frac{N}{n} \rfloor$

设$i \in [0, B-1], j \in [0, n-1], k \in [0, m]$，则第$i$批中的第$j$个行的行号为$a = in + j$，其中的第$k$段，起始为

线性模型参数

$$Y = XA + b$$

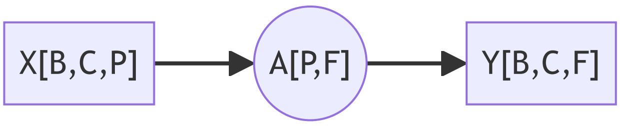
X为线性模型的输入特征数，$X \in \mathrm{R}^{n\times(m+1)L}$，$X$的行由$s_0, s_1, \cdots, s_m$共$m+1$个段依次拼接而成，其中$s_i \in \mathrm{R}^{1\times L}$

$A \in \mathrm{R}^{(m+1)L\times F}$ 为权重矩阵

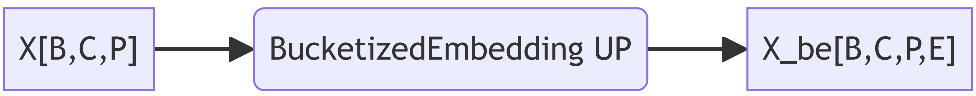
```
nn.Linear(in_features=(m+1)*L, outfeatures=F)
```

Patch Token Attention 结构

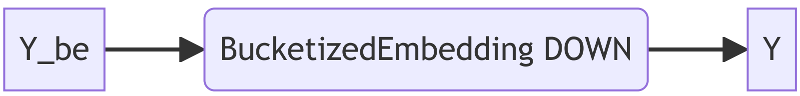
线性层



Bucketize Embedding 方案



将"C 个变量 P 个过去的数据之间的相互影响"变为"C 个变量每个过去的数据之间的相互影响"（粒度变细）



Patch Self-Attention 层

$Y \in \mathrm{R}^{B\times C\times F}$

每个 Channel/变量为一个 Token，故 Token 数为C

变量一段时间F内的值为 Token 分量，故向量维数或Y的特征数为F

一个"句子"矩阵$S_i = Y[i,:,:] \in \mathrm{R}^{C\times F}$

d_K是 Key 维度，$W_K \in \mathrm{R}^{F\times d_K}$是 Key 权重，$K = YW_K \in \mathrm{R}^{B\times C\times d_K}$是 Key

$d_Q = d_K$是 Query 维度，$W_Q \in \mathrm{R}^{F\times d_Q}$是 Query 权重，$Q = YW_Q \in \mathrm{R}^{B\times C\times d_Q}$是 Query

d_V =是 Value 维度，$W_V \in \mathrm{R}^{F\times d_K}$是 Value 权重，$V = YW_V \in \mathrm{R}^{B\times C\times d_V}$是 Value

```
WK = nn.Linear(F,d_k)
K = WK(Y)
```

或者

```
WK = torch.rand(size=(F,d_k))
K = Y@WK
```

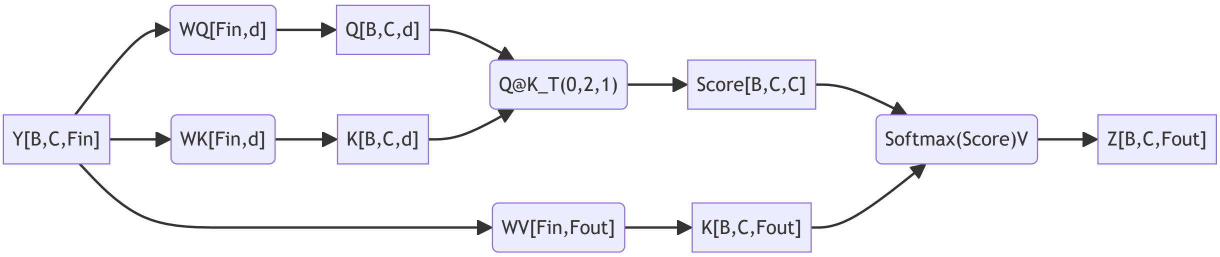
$$\mathrm{Score} = QK^T = Q@K.\mathrm{permute}(0,2,1) \in \mathrm{R}^{B\times C\times C} \tag{5}$$
$$\mathrm{Score}^* = \mathrm{Score}/\sqrt{\delta} \tag{6}$$
$$W = \mathrm{Softmax}(\mathrm{Score}*) \tag{7}$$
$$Z = \mathrm{Attention} = WV \tag{8}$$

其代码实现采用`nn.MultiheadAttention`实现

```
attn = nn.MultiheadAttention(
    embed_dim=d_q,
    num_heads=1,
    kdim=d_k,
    vdim=d_v,
)
```

其中，`embed_dim`是指的注意力机制的输入，即Q，的特征维数d_Q，而不是自注意力的输入Y特征维数F

输出Z是注意力机制的输出，即修正后的 V 的值，与 Q 的尺寸一致$Z \in \mathrm{R}^{B\times C\times d_Q}$，$Z, W = \mathrm{attn}(Q, K, V)$



Patch Self-Attention 策略

1

$$
\begin{aligned}
Y &= XA + b \\
Z &= \mathrm{SelfAttention}(Y)
\end{aligned}
$$

```
linear = nn.Linear(P,P1)
attn = PatchSelfAttention(P1,F)
z=attn(linear(x))
```

2

$$
\begin{aligned}
Y_1 &= XA + b \\
Y_2 &= \mathrm{SelfAttention}(X) \\
Z &= \mathrm{Mix}(Y_1, Y_2)
\end{aligned}
$$

```
linear = nn.Linear(P,F)
attn = PatchSelfAttention(P,F)
mix = nn.Linear(2*C,C)
z = mix(cat([linear(x),attn(x)],dim=1).permute(0,2,1)).permute(0,2,1)
```

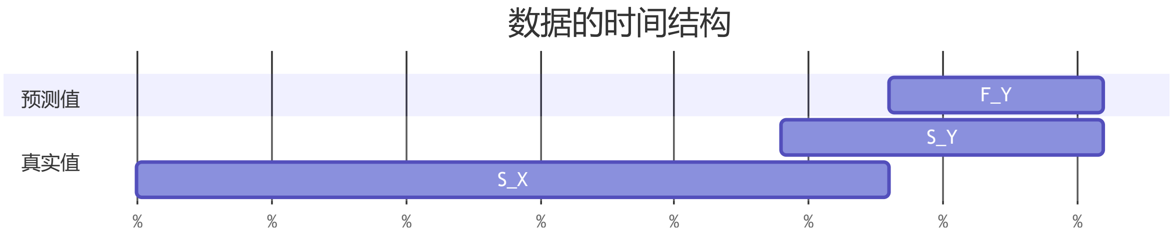
Exp

总流程

1. get item 获取单个样本点 Sample
2. next 组成 Batch

Dataset

```
P = seq_len = 336
F = pred_len = 96
R = label_len = 48
```



$\eta_{\text{train}} = 0.7$

$\eta_{\text{test}} = 0.2$

$\eta_{\text{vali}} = 1 - \eta_{\text{train}} - \eta_{\text{test}}$

$n_{\text{train}} = \eta_{\text{train}} L(D_0)$

$D_{\text{train}} = D_0[0 : n_{\text{train}}]$

$S_X(i) = D_{\text{train}}[i : i + P] \in \mathrm{R}^{P \times C}$

$S_Y(i) = D_{\text{train}}[i + P - R : i + P + F] \in \mathrm{R}^{(R+F) \times C}$???返回R+F个，多R个，为什么

$S(i) = S_X(i), S_Y(i)$

$L(D) = L(D_{\text{train}}) - P - F + 1$

DataLoader