1. 对齐，点乘，加 bias：卷起来！

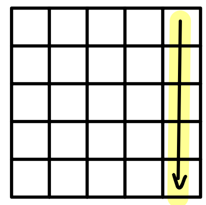
5.1	2.1	4.1
2.1	4.1	2.1
4.1	2.1	3.1

2. Normalization 是解决训练过程中不同 layer 的形状、norm 等"不一致"而提出的规范化层输入方法。

它能提高训练速度，增强不同 initialization 的 robustness.

本来最直觉的角度是 layer normalization，但是考虑到有时候一层的形状、norm 等也是重要 feature（想象 MNIST 的笔迹粗细等），所以采用"同类型"正则化，即 batch normalization (BN).

其实现方法是，加入一层：$\hat{z}_{i+1} = \sigma_i(z_i W_i + b_i^T)$ 正则化矩阵的 column，



这可能带来不必要的 dependency on the entire batch，因此考虑对所有层 feature 计算实时 mean, variance 为 $\hat{\mu}_{i+1}$, $\hat{\sigma}^2_{i+1}$ 并在 test-time 正则化：$(z_{i+1})_j = \dfrac{(\hat{z}_{i+1})_j - (\hat{\mu}_{i+1})_j}{((\hat{\sigma}^2_{i+1})_j + \varepsilon)^{\frac{1}{2}}}$

附上我曾经的实现：

```python
class BatchNorm1d(Module):
    def __init__(self, dim, eps=1e-5, momentum=0.1, device=None, dtype="float32"):
        super().__init__()
        self.dim = dim
        self.eps = eps
        self.momentum = momentum
        self.weight = Parameter(init.ones(dim, requires_grad=True))
        self.bias = Parameter(init.zeros(dim, requires_grad=True))
        self.running_mean = init.zeros(dim)
        self.running_var = init.ones(dim)


    def forward(self, x: Tensor) -> Tensor:
        if self.training:
            batch_mean = (x.sum((0,)) / x.shape[0])
            batch_var = ((x - batch_mean.broadcast_to(x.shape)) ** 2).sum((0,)) / x.shape[0]
            # Update the running statistics using momentum technic.
            self.running_mean = (1 - self.momentum) * self.running_mean + self.momentum * batch_me (function) data: Any
            self.running_var = (1 - self.momentum) * self.running_var + self.momentum * batch_var.data
            norm = (x - batch_mean.broadcast_to(x.shape)) / (batch_var.broadcast_to(x.shape) + self.eps) ** 0.5
            return self.weight.broadcast_to(x.shape) * norm + self.bias.broadcast_to(x.shape)
        else:
            norm = (x - self.running_mean.broadcast_to(x.shape)) \
                        / (self.running_var.broadcast_to(x.shape) + self.eps) ** 0.5
            return self.weight.broadcast_to(x.shape) * norm + self.bias.broadcast_to(x.shape)
```

$$y = w \circ \frac{z_i - E[x]}{(\mathrm{Var}(x) + \varepsilon)^{\frac{1}{2}}} + b$$

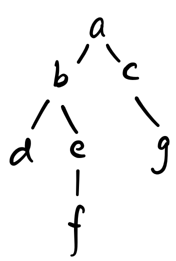
$$y = \frac{x - \hat{\mu}}{((\hat{\sigma}^2_{i+1})_j + \varepsilon)^{\frac{1}{2}}}$$

$$\hat{x} := (1 - m)\hat{x} + m x_{obs}$$

3. 前序 nlr：abdefcg

中序 lnr：dbfeagc

后序 lrn



=> d, f, e, b, g, c, a