**作业3：基于计算图的自动求导**

**Tips:**

1. 本次作业难度较大，持续时间为2-3周，分为两个部分：基础部分、进阶部分

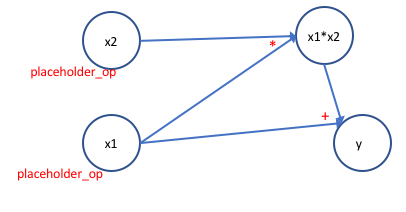
2. **不能使用numpy以外的第三方库**

**计算图**

计算图是Pytorch等深度学习框架实现自动求导的基本工具。在本次作业中我们需要基于numpy手动实现基础函数的前向和反向自动求导运算。

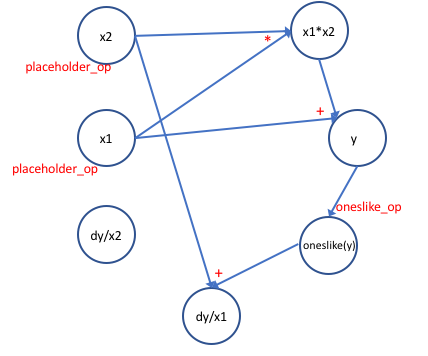
基础的计算图：

例如 ，它的计算图如下所示：



图中使用了一个有向无环图来表示表达式计算的流程，边代表计算的方向。图中左边红色的placeholder\_op表示计算图中的占位符，它是参与运算的基本变量单元；右边红色的运算符是我们计算过程中的基本运算，也是进行反向求导的基本单元。这里给出的例子是两个变量相连的，但实际上也有变量和常数之间的计算(常数不作为节点出现在图中)。

它的导数计算图如下所示：



构建出计算图后，首先进行前向运算求出目标函数的值，然后通过简单的拓扑排序并根据每个计算单元（如乘法、加法）的求导法则我们可以反向求出目标函数相对于每一个输入变量的导数。

总结来说在计算图中，我们有如下要素：

- 节点（计算的基本单元）

- 操作符 (如Add, Sub, Multi, Div, Placeholder, Oneslike，运算的基本单元)

- 执行器Executor（前向运算）

- 求导器gradients（用于反向求导）

给出的代码文件有两个：autodiff.py和autodiff\_test.py；前者是需要同学们去完善的("TODO"部分需要完成)，后者是给同学们的测试文件，提供了十五个测试样例供大家测试（请不要改动这个文件）。我们可以运行如下脚本对我们的代码进行测试：

# pip install nose

nosetests -v autodiff\_test.py

**任务**

**基础部分**（6pts）

请根据autodiff.py中给出的加减的样例

1. 实现乘法 (2pts) 和除法 (2pts) 运算

2. 实现指数exp (1pts) 和对数log (1pts) 运算

**进阶部分**（4pts）

实现矩阵运算MatMulOp（具体请参考autodiff.py的提示，需要用到线性代数的知识）(2pts)

请构造一个简单的二分类问题数据集（如两个高斯分布的分类），使用你搭建的autodiff进行logistic regression，补全”logistic\_regression\_todo.py”中缺失部分，让文件中梯度下降算法进行求解能正常运行。(2pts)

**提交要求：**

将完成后的所有.py文件压缩后提交至canvas。

**参考资料：**

计算图： [1.2.5 计算图 - 哔哩哔哩 (bilibili.com)](https://www.bilibili.com/read/cv4270886)。

导数计算：<https://www.derivative-calculator.net/>

矩阵求导：<https://www.math.uwaterloo.ca/~hwolkowi/matrixcookbook.pdf>

激活函数：<https://www.jeremyjordan.me/neural-networks-activation-functions/>

**Reference：**

<https://github.com/dlsys-course/assignment1>