Table of contents

```python
from sklearn.datasets import load_diabetes

# 1. Carrega os dados já separados em X (variáveis) e y (alvo)
# scaled=False garante que venham os números reais (idade, pressão, etc)
X_numpy, y_numpy = load_diabetes(return_X_y=True, scaled=False)

# 2. Converte de Array do Numpy para Listas nativas do Python
dataset_X = X_numpy.tolist()
dataset_y = y_numpy.tolist()
```

```python
class Value:
    def __init__(self, data, _children=(), _op=""):
        self.data = data
        self.grad = 0
        self._backward = lambda: None
        self._prev = set(_children)
        self._op = _op

    def __repr__(self):
        return f"Value(data={self.data})"

    def __add__(self, other):
        other = other if isinstance(other, Value) else Value(other)
        out =  Value(self.data + other.data, (self,other), '+')

        def _backward():
            self.grad += 1. * out.grad
            other.grad += 1. * out.grad

        out._backward = _backward

        return out
    def __radd__(self, other):
        return self + other

    def exp(self):
        x = self.data
```

```
        out = Value(math.exp(x), (self, ), "exp")

        def _backward():
            self.grad += out.data * out.grad
        out._backward = _backward
        return out

    def __rmul__(self, other):
        return self * other

    def __truediv__(self, other):
        return self * other**-1

    def __pow__(self, other):
        assert isinstance(other, (int,float))
        out = Value(self.data**other, (self,), (f"**{other}"))

        def _backward():
            self.grad += (other)*(self.data**(other-1))*out.grad
        out._backward = _backward
        return out

    def __neg__(self):
        return self*-1

    def __sub__(self, other):
        return self + (-other)

    def __mul__(self, other):
        other = other if isinstance(other, Value) else Value(other)
        out = Value(self.data * other.data, (self, other), "*")

        def _backward():
            self.grad += other.data * out.grad
            other.grad += self.data * out.grad
        out._backward = _backward

        return out

    def tahn(self):
        n = self.data
        t = (math.exp(2*n)-1) / (math.exp(2*n)+1)
        out = Value(t, (self, ), "tahn")

        def _backward():
```

```
            self.grad += (1 - t**2) * out.grad

        out._backward = _backward

        return out

    def backward(self):
        self.grad = 1
        # Montar ordem topologica
        topo = []
        visited = set()
        def build_topo(v):
            if v not in visited:
                visited.add(v)
                for child in v._prev:
                    build_topo(child)
                topo.append(v)
        build_topo(self)

        for node in reversed(topo):
            node._backward()



a = Value(2.0)
b = Value(-3.0)
c = Value(10.)
e = a*b
d = e + c
f = Value(-2)
L = d * f
L
```

```
Value(data=-8.0)
```

```
class Neuron:
    def __init__(self, input_num):
        self.w = [Value(random.uniform(-1,1)) for _ in range(input_num)]
        self.b = Value(random.uniform(-1,1))

    def __call__(self,x):
        act = sum((wi*xi for wi,xi in zip(self.w, x)), self.b)
        out = act.tahn()
        return out

```

```
    def paramerters(self):
        return self.w + [self.b]

class Layer:
    def __init__(self, input_num, output_num):
        self.neurons = [Neuron(input_num) for _ in range(output_num)]

    def __call__(self, x):
        outs = [n(x) for n in self.neurons]
        return outs[0] if len(outs)==1 else outs

    def parameters(self):
        out = []
        for neuron in self.neurons:
            out.extend(neuron.paramerters())
        return out

class MLP:
    def __init__(self, input_num, output_nums):
        sz = [input_num] + output_nums
        self.layers = [Layer(sz[i], sz[i+1]) for i in range(len(output_nums))]

    def __call__(self, x):
        for layer in self.layers:
            x = layer(x)
        return x

    def parameters(self):
        out = []
        for layer in self.layers:
            out.extend(layer.parameters())
        return out

```

```
y_pred
```

```
[Value(data=-0.8415627312965999),
 Value(data=-0.8416194345012991),
 Value(data=-0.05490668837286004),
 Value(data=-0.00322658100945949),
 Value(data=-0.8416194652573291),
 Value(data=0.0007289701558785097),
 Value(data=0.590527223921252),
 Value(data=-0.8219020753186649),
 Value(data=0.752305549190352),
```

```
Value(data=0.23634884911671442),
Value(data=0.2324369999890968),
Value(data=-0.3667103294682982),
Value(data=-0.0634004663894011),
Value(data=-0.8416194659720974),
Value(data=0.23260575920595203),
Value(data=0.5111901974753233),
Value(data=0.6789648937094968),
Value(data=-0.5820612124669666),
Value(data=0.7523055491842697),
Value(data=0.7523055481961325),
Value(data=0.7134230126114081),
Value(data=-0.4338715956237026),
Value(data=-0.6284457596711006),
Value(data=-0.3690893360127478),
Value(data=-0.09381265348574475),
Value(data=0.87271323627814),
Value(data=-0.8416194651298385),
Value(data=-0.9465516580056076),
Value(data=0.6439211564975865),
Value(data=-0.3132323456409167),
Value(data=-0.6248843771202373),
Value(data=-0.059336119312374674),
Value(data=0.044008110571278354),
Value(data=0.44431944527176626),
Value(data=-0.024021510427910762),
Value(data=-0.8389613990859471),
Value(data=-0.7017444187863259),
Value(data=-0.7523382655692067),
Value(data=0.752303815305045),
Value(data=0.23616106070457457),
Value(data=0.18313362989175042),
Value(data=-0.19748526834772273),
Value(data=-0.0840158967698916),
Value(data=0.7523055491899738),
Value(data=0.7523024221780781),
Value(data=0.09969676087824433),
Value(data=-0.7523383153911842),
Value(data=0.7523055491903367),
Value(data=-0.8416194648385672),
Value(data=-0.31319871037159847),
Value(data=-0.6743053204039691),
Value(data=-0.36038983638177446),
Value(data=0.7146308588140945),
Value(data=0.6924538940391899),
Value(data=0.23539476271963145),
```

```
Value(data=-0.8416194659719878),
Value(data=0.7091475440192085),
Value(data=0.752305549190352),
Value(data=0.749567585937358),
Value(data=0.6850527027384172),
Value(data=-0.0028793061189391336),
Value(data=-0.18004229028886187),
Value(data=-0.3603898579532061),
Value(data=0.5157441055952386),
Value(data=0.6703340716478448),
Value(data=0.7523055491881069),
Value(data=-0.8416194659720978),
Value(data=-0.093791992022148),
Value(data=0.18510086987575336),
Value(data=-0.6521854896346359),
Value(data=-0.004256640642177963),
Value(data=0.024684441678020343),
Value(data=0.18511176825466774),
Value(data=0.7523055490943796),
Value(data=-0.1976408483882895),
Value(data=-0.36962425337342913),
Value(data=-0.6737516431771209),
Value(data=-0.43469453775631567),
Value(data=-0.8416194651837271),
Value(data=-0.8427683133504643),
Value(data=-0.19747056174811134),
Value(data=-0.0788405819523186),
Value(data=-0.06551384837024385),
Value(data=-0.31419026668488),
Value(data=-0.6743022098606871),
Value(data=-0.003208504052582923),
Value(data=0.5661926082228022),
Value(data=0.5635018667693333),
Value(data=-0.6767795537425427),
Value(data=-0.8415986549635354),
Value(data=-0.841619465967908),
Value(data=0.7523055488024072),
Value(data=-0.23694711795071086),
Value(data=-0.5584860595760806),
Value(data=-0.09285084385372601),
Value(data=0.003024127224643731),
Value(data=0.5912616232171111),
Value(data=0.752305549190352),
Value(data=0.014517020279806161),
Value(data=0.7523055491903511),
Value(data=0.23558694368662905),
```

```
Value(data=0.7523031105656448),
Value(data=-0.025508027330395006),
Value(data=-0.05048733355456281),
Value(data=-0.0028240711731751644),
Value(data=-0.7513830942301281),
Value(data=0.29267648572453403),
Value(data=-0.7522903404138543),
Value(data=-0.8373894531624996),
Value(data=0.7466691994770996),
Value(data=-0.8416194659720978),
Value(data=-0.8029218382079596),
Value(data=-0.3133435335209921),
Value(data=0.509902420486061),
Value(data=0.18500698487244632),
Value(data=0.5113379799054405),
Value(data=0.583687280279636),
Value(data=-0.8416194659720978),
Value(data=-0.003208612058276315),
Value(data=0.1969726245227751),
Value(data=-0.474113301816531),
Value(data=-0.8416194658797662),
Value(data=0.8691837171743972),
Value(data=0.18757281543159693),
Value(data=0.6725515495111626),
Value(data=0.720881763004354),
Value(data=0.752305549190352),
Value(data=-0.4741132617990162),
Value(data=0.23615955854604265),
Value(data=-0.08444515269806618),
Value(data=0.7523054556296074),
Value(data=0.5097564822471878),
Value(data=-0.8416194637674294),
Value(data=-0.0593551394115986),
Value(data=0.7485459777087161),
Value(data=-0.09381114271535491),
Value(data=0.752305549190352),
Value(data=-0.15862722575148513),
Value(data=-0.8416194657442588),
Value(data=-0.7607994583074659),
Value(data=-0.4741115229153861),
Value(data=-0.22273378886756315),
Value(data=-0.8416194679359383),
Value(data=-0.8601828662825803),
Value(data=0.752305549190352),
Value(data=-0.7509124087377687),
Value(data=-0.16629515667842584),
```

```
Value(data=-0.7086078963308614),
Value(data=0.7523048542371348),
Value(data=0.34742185870513487),
Value(data=-0.3193211085607453),
Value(data=-0.36039493366096437),
Value(data=0.24977974799348493),
Value(data=-0.24709125210916782),
Value(data=-0.841619465962903),
Value(data=-0.8416194659720978),
Value(data=-0.5584355129009783),
Value(data=0.709863805850473),
Value(data=-0.8416194659720978),
Value(data=0.46973202614222553),
Value(data=-0.08900547101921032),
Value(data=0.7670456716332431),
Value(data=0.7515276671006267),
Value(data=0.7146241315324272),
Value(data=-0.8416194659719851),
Value(data=-0.0974290471886943),
Value(data=-0.8415996817577456),
Value(data=-0.6743022441884394),
Value(data=0.23654287597639276),
Value(data=-0.8416142482394962),
Value(data=0.23608454287752165),
Value(data=-0.8416194652693524),
Value(data=0.7523055491862759),
Value(data=-0.49769929274007324),
Value(data=-0.05052105471629754),
Value(data=0.8737993797450273),
Value(data=0.5113379717328306),
Value(data=-0.3603898369863178),
Value(data=0.5105110298613881),
Value(data=0.1784529165872902),
Value(data=-0.003208485355548673),
Value(data=-0.367503058354979),
Value(data=0.7273686645401503),
Value(data=0.7346547973173071),
Value(data=-0.0032084126852894965),
Value(data=-0.5584471691751651),
Value(data=0.7523055491903519),
Value(data=-0.8416194659718954),
Value(data=0.23408588765246047),
Value(data=0.7199877300953721),
Value(data=-0.5586151821859324),
Value(data=-0.003209461544153728),
Value(data=0.752305549190351),
```

```
Value(data=0.4241575912072729),
Value(data=-0.9499326872627337),
Value(data=0.5632522783444234),
Value(data=-0.8416194659720904),
Value(data=-0.5503305526023017),
Value(data=-0.5074614796773629),
Value(data=-0.3132303043756266),
Value(data=0.5496188285813405),
Value(data=0.23263946374591082),
Value(data=0.7523055457411794),
Value(data=0.7159129564402817),
Value(data=-0.8267128751522281),
Value(data=-0.5355832829842535),
Value(data=-0.8416194659720981),
Value(data=-0.8416194639932238),
Value(data=0.019937665655614032),
Value(data=-0.006744363230917061),
Value(data=-0.36039421706283054),
Value(data=-0.0032084906895803508),
Value(data=0.7513079966756456),
Value(data=-0.8416194659720978),
Value(data=0.5630762184358562),
Value(data=0.08561947076335483),
Value(data=-0.5584860585809881),
Value(data=0.6104858532595809),
Value(data=0.752305549190352),
Value(data=-0.31293002576889695),
Value(data=0.5632525450904063),
Value(data=-0.4228627180439916),
Value(data=0.23260734140768688),
Value(data=0.2369460786920258),
Value(data=-0.31176538621943306),
Value(data=-0.49569335724978925),
Value(data=0.6221369202591238),
Value(data=0.5656596768737575),
Value(data=-0.8416194659720978),
Value(data=-0.8416194659720502),
Value(data=-0.8411933204770886),
Value(data=0.18511166354523823),
Value(data=-0.8416185878942044),
Value(data=-0.8415418931677148),
Value(data=0.6206112780185206),
Value(data=-0.8416194659720978),
Value(data=0.18513186345033525),
Value(data=0.1892236997428502),
Value(data=-0.3605192634077059),
```

```
Value(data=-0.6021811647630487),
Value(data=-0.841618396782954),
Value(data=-0.6729886456646893),
Value(data=0.5647714271902672),
Value(data=0.7523055491903476),
Value(data=0.6787356784188849),
Value(data=-0.3725574548062828),
Value(data=-0.8411498997358485),
Value(data=0.7742502701844043),
Value(data=0.2326742426402757),
Value(data=-0.5584860616637264),
Value(data=-0.9429067064175995),
Value(data=-0.5584826837518017),
Value(data=0.23615983514194652),
Value(data=0.7516779472527761),
Value(data=0.5105891499633911),
Value(data=-0.6743010193676416),
Value(data=0.5632527693303969),
Value(data=0.034105614509376436),
Value(data=-0.05114026955503507),
Value(data=0.752316275694912),
Value(data=-0.7191224692779549),
Value(data=0.7523055486284501),
Value(data=-0.9370486838321045),
Value(data=-0.058582436805900674),
Value(data=0.4862353382319841),
Value(data=0.6821932430647532),
Value(data=-0.3605010533488893),
Value(data=-0.4538413125592747),
Value(data=-0.8416194659720978),
Value(data=0.46101563449175254),
Value(data=0.7146017960954045),
Value(data=-0.9400364733812405),
Value(data=0.7019553365951331),
Value(data=0.7522109563258358),
Value(data=0.1851117679104247),
Value(data=-0.8411156955114124),
Value(data=0.029660301875815785),
Value(data=0.2347886298415126),
Value(data=0.23362576319234912),
Value(data=0.3590145540058608),
Value(data=0.6176855000514068),
Value(data=0.029653689586272233),
Value(data=-0.19746838644820003),
Value(data=0.36409827903463715),
Value(data=-0.057647468059235256),
```

```
Value(data=0.7523055491903436),
Value(data=0.622168645001492),
Value(data=-0.21387125809352206),
Value(data=0.21988333112652295),
Value(data=0.7523055491879931),
Value(data=-0.7511737149303823),
Value(data=-0.3603903247121275),
Value(data=0.7523055491903518),
Value(data=-0.6563797387450829),
Value(data=0.23520627952852413),
Value(data=0.23260570229665228),
Value(data=0.7032756993398123),
Value(data=0.5066481831131525),
Value(data=-0.2404017901213023),
Value(data=0.5632540840024651),
Value(data=0.5664989819577885),
Value(data=0.7519268447824776),
Value(data=-0.8367209440642345),
Value(data=0.752305549190352),
Value(data=0.41524869726648794),
Value(data=-0.49922680397243036),
Value(data=-0.6686194083308274),
Value(data=-0.6742563437093902),
Value(data=-0.48551031801585187),
Value(data=0.6777394899537427),
Value(data=-0.6128440182158071),
Value(data=0.35939847371983774),
Value(data=0.5919714460080509),
Value(data=-0.00461969234204162),
Value(data=0.7522942521882868),
Value(data=-0.041658769425593434),
Value(data=0.752305549190352),
Value(data=0.752305549190352),
Value(data=0.6634346621592201),
Value(data=0.7522891046506945),
Value(data=0.5113345029362342),
Value(data=0.752305549190352),
Value(data=0.10958751060851984),
Value(data=0.8540632539687025),
Value(data=0.23615980326231642),
Value(data=0.7523008240893215),
Value(data=-0.8416192132579584),
Value(data=-0.8416194761155075),
Value(data=-0.4986479816526874),
Value(data=-0.8416194659720959),
Value(data=-0.5585078805960447),
```

```
Value(data=0.7673342447522788),
Value(data=-0.4985204323350121),
Value(data=0.7523054128367977),
Value(data=0.716350928127463),
Value(data=-0.45039977154037836),
Value(data=0.6215254352035466),
Value(data=0.7522167943023804),
Value(data=-0.4741131608806469),
Value(data=-0.9465029851737213),
Value(data=-0.47408927783700056),
Value(data=-0.5584952018882203),
Value(data=0.23260589843667775),
Value(data=-0.05048496976355426),
Value(data=-0.7523383154382636),
Value(data=-0.9465537677545641),
Value(data=-0.5573154083123765),
Value(data=0.2326034577814231),
Value(data=-0.3603900354816428),
Value(data=-0.36039411753335376),
Value(data=0.23586925502251227),
Value(data=-0.3132323487713915),
Value(data=0.5113379799047896),
Value(data=-0.6620153809729447),
Value(data=-0.674301800692371),
Value(data=0.5684033839912273),
Value(data=-0.6488311870577662),
Value(data=-0.8387879374129402),
Value(data=-0.36690210384676103),
Value(data=-0.6326840952371331),
Value(data=-0.5520782971414756),
Value(data=-0.003317043198650031),
Value(data=0.21270729868614527),
Value(data=-0.3165565306168389),
Value(data=-0.36038983625028764),
Value(data=-0.36039003211639725),
Value(data=-0.8346528350933287),
Value(data=-0.8052721511512086),
Value(data=0.1201802190955259),
Value(data=-0.10363564628486345),
Value(data=0.7742502723589525),
Value(data=0.23615979361871514),
Value(data=-0.016514756364055402),
Value(data=0.0988431983471681),
Value(data=-0.1974410811853431),
Value(data=-0.5108415433129084),
Value(data=0.23081492658983743),
```

```
Value(data=0.640490752005102),
Value(data=-0.6742796655069111),
Value(data=0.752305549190352),
Value(data=0.7146343383173848),
Value(data=0.005717646215201245),
Value(data=-0.09405003586902935),
Value(data=0.8297055666428749),
Value(data=0.5632533264877847),
Value(data=0.752305549190352),
Value(data=0.7085631722212521),
Value(data=-0.7523309403475649),
Value(data=0.5113379781882526),
Value(data=0.5113379704175836),
Value(data=0.08489760811997542),
Value(data=-0.0745143990115551),
Value(data=-0.8069478831969156),
Value(data=0.008301830314252726),
Value(data=-0.7523233512119889),
Value(data=-0.3578237664745138),
Value(data=0.7523051593887048),
Value(data=-0.9315255372557891),
Value(data=-0.8082132658152666),
Value(data=0.5113379797997366),
Value(data=-0.0032062806027131152),
Value(data=-0.08441484960807619),
Value(data=0.6400307244885958),
Value(data=-0.7944714485446122),
Value(data=-0.3132568721771691),
Value(data=-0.8416019240126361),
Value(data=0.1851128921994444),
Value(data=0.7523055491588815),
Value(data=0.16296506802657443),
Value(data=0.6157196745700914),
Value(data=-0.3132323492984473),
Value(data=-0.5902555181350494),
Value(data=-0.6742591108261897),
Value(data=0.7523055491901379),
Value(data=0.5827759579307711),
Value(data=-0.9429502100952012),
Value(data=-0.7522345404954394),
Value(data=-0.05919293098025813),
Value(data=0.5113368149082986),
Value(data=-0.05554372557101371),
Value(data=-0.7523011291803815),
Value(data=-0.5584860593800294),
Value(data=-0.7249402167984264),
```

```
Value(data=-0.8416194659410665),
Value(data=-0.6742615098097214),
Value(data=-0.9465521744386456),
Value(data=0.8716017131487845),
Value(data=0.7521137124382034),
Value(data=-0.8701220813035121),
Value(data=0.7522515317779896),
Value(data=-0.8100815697025662),
Value(data=0.5113379797503332),
Value(data=0.5112410060808564),
Value(data=-0.9213390635663748),
Value(data=0.6241387240767196),
Value(data=-0.7524917422457629),
Value(data=0.622136919175138),
Value(data=0.5113378117473063),
Value(data=0.7363234138142228),
Value(data=-0.7512528103473067),
Value(data=-0.36373076401436666),
Value(data=0.18481359208097703),
Value(data=0.5613798172030012),
Value(data=-0.752090104866937),
Value(data=-0.7839430548234217),
Value(data=-0.24374822156224446),
Value(data=-0.7523383154518689),
Value(data=-0.05192059771370694),
Value(data=0.7523055491899329),
Value(data=0.4710324973693049),
Value(data=-0.5482401684047341),
Value(data=-0.34932535559253),
Value(data=0.4564274849415448),
Value(data=-0.8415819328615802),
Value(data=-0.5584861499755054),
Value(data=0.510822026575943),
Value(data=-0.8416192146427229),
Value(data=0.8737956419722532),
Value(data=-0.15446190448285183),
Value(data=-0.8574994600651251),
Value(data=-0.47287047375278257),
Value(data=-0.03763798138832406),
Value(data=0.1593090927994636),
Value(data=-0.3612768235484027),
Value(data=-0.25320446597254237),
Value(data=-0.36578430476534707),
Value(data=0.5111870953911984),
Value(data=-0.8416194659695543),
Value(data=0.01757336127249222),
```

```
 Value(data=-0.48443827158601716),
 Value(data=-0.5584860458158947),
 Value(data=-0.4741110781057349),
 Value(data=0.6776781641918045),
 Value(data=-0.00614598892424131),
 Value(data=-0.8416123667306783),
 Value(data=-0.09422396507235047),
 Value(data=-0.44752168490506344),
 Value(data=0.1851116662608289),
 Value(data=0.7523055420485438),
 Value(data=-0.49864579804908604),
 Value(data=-0.9112952585453536),
 Value(data=-0.05057648718631836),
 Value(data=-0.8416182277238314),
 Value(data=-0.09478720964637219),
 Value(data=0.7523055481129937),
 Value(data=0.626757315790661),
 Value(data=-0.3133071007926418),
 Value(data=-0.5600983101893039),
 Value(data=0.752305549190352),
 Value(data=-0.8416194659720978),
 Value(data=0.23629757700617524),
 Value(data=0.7376631677073783),
 Value(data=-0.6743000730182208),
 Value(data=-0.697903018745952),
 Value(data=-0.09381303449557063),
 Value(data=-0.7380269540468758),
 Value(data=-0.5598514541428071),
 Value(data=-0.8416194659672857),
 Value(data=0.21173372021229414),
 Value(data=-0.30721995070056907)]
```

```python
import random
import math

def f(x,y,z):
    return 0.004*x**2 + 0.07*y*x - z + random.gauss(0, 1)

X = []
y = []
for i in range(500):
    X.append([random.uniform(-50,50) for _ in range(3)])
    y.append(f(X[i][0],X[i][1],X[i][2]))

print(y)
```

```
14
15 ann = MLP(3, [8,8,1])
16
17 for i in range(100): # Our code will do 100 epochs of training
18     y_pred = [ann(x) for x in X] # Our model only accept one prediction per time
19     print(y_pred[0], print(y[0]))
20     loss = sum((pred-origin)**2 for pred,origin in zip(y_pred, y))
21
22     loss.backward() # Calc of gradients
```

```
[-35.00006447265811, 14.452764707564574, -28.61126541419166, 16.42604207222155, 19.2540434
-35.00006447265811
Value(data=0.9476197044921958) None
-35.00006447265811
Value(data=0.9476197044921958) None
-35.00006447265811
Value(data=0.9476197044921958) None
-35.00006447265811
Value(data=0.9476197044921958) None
```