Name: Venkata Sai Ram Gunda

I'd: 1002073073

## Code Screen Shots:

```
class cnnModel(nn.Module):
    def __init__(self):
        nn.Module.__init__(self)
        self.network=nn.Sequential(
        nn.Conv2d(3,16,kernel_size=3,padding=1),
        nn.ReLU(),
        nn.Conv2d(16,32,kernel_size=3,stride=1,padding=1),
        nn.MaxPool2d(2,2),
        nn.Conv2d(32,64,kernel_size=3,stride=1,padding=1),
        nn.ReLU(),
        nn.Conv2d(64,64,kernel_size=3,stride=1,padding=1),
        nn.MaxPool2d(2,2),
        nn.Conv2d(64,128,kernel_size=3,stride=1,padding=1),
        nn.ReLU(),
        nn.Conv2d(128,128,kernel_size=3,stride=1,padding=1),
        nn.MaxPool2d(2,2),

        nn.Flatten(),
        nn.Linear(128*6*6,1024),
        nn.ReLU(),
        nn.Linear(1024,512),
        nn.ReLU(),
        nn.Linear(512,7)
        )
```

```
a=evaluate(model,val_batches)
print("val_loss:{},val_acc:{}".format(a[0],a[1]))
```

```
val_loss:1.9363374263048172,val_acc:0.17236328125
```

```
num_epoch=6
opt_function=torch.optim.SGD
learning_rate=0.003
history=fit(num_epoch,learning_rate,model,train_batches,val_batches,opt_function)
```

```
Epoch [0],train_loss:1.9181,val_loss:1.9003,val_acc:0.2512
Epoch [1],train_loss:1.8892,val_loss:1.8748,val_acc:0.2563
Epoch [2],train_loss:1.8695,val_loss:1.8572,val_acc:0.2563
Epoch [3],train_loss:1.8553,val_loss:1.8448,val_acc:0.2534
Epoch [4],train_loss:1.8446,val_loss:1.8354,val_acc:0.2527
Epoch [5],train_loss:1.8365,val_loss:1.8229,val_acc:0.2607
```

```
valid_data=DataLoader(df_valid,batch_size=32,shuffle=True)
result=evaluate(model,valid_data)
result
```

```
[1.8013283661587745, 0.2542203272359943]
```

```
print("val_loss:{},val_acc:{}".format(result[0],result[1]))
```

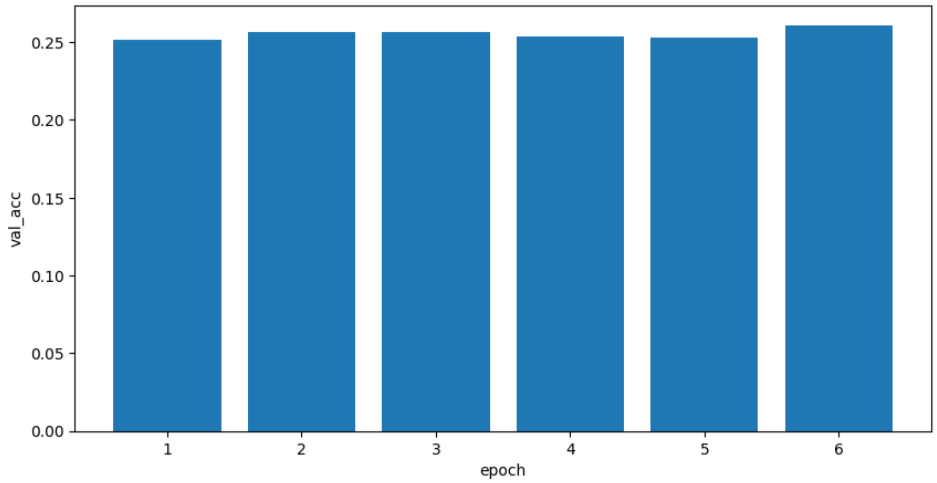
```
val_loss:1.8013283661587745,val_acc:0.2542203272359943
```

```
history
```

```
[[1.900278352200985, 0.251220703125, 1.918100067327932],
 [1.8747502639889717, 0.25634765625, 1.8892290211215461],
 [1.857187308371067, 0.25634765625, 1.8694848940544522],
 [1.8447943776845932, 0.25341796875, 1.8552547496618683],
 [1.8354331739246845, 0.252685546875, 1.844592745771113],
 [1.822874329984188, 0.2607421875, 1.8364811948279745]]
```

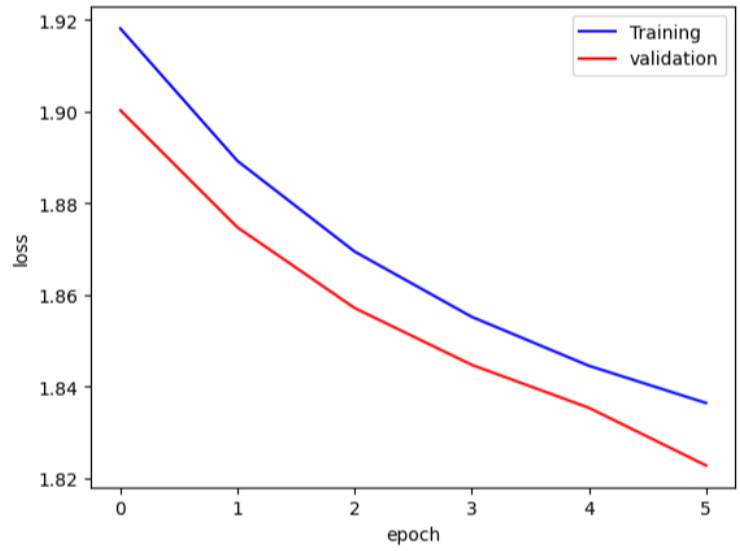
In [17]:
```
epoch=[i for i in range(1,7)]
acc=[x[1] for x in history]
fig = plt.figure(figsize = (10, 5))
plt.bar(epoch[:],acc[:])
plt.xlabel('epoch')
plt.ylabel('val_acc')
```

Out[17]: Text(0, 0.5, 'val_acc')




In [18]:
```
train_loss=[x[2] for x in history]
val=[x[0] for x in history]
plt.plot(train_loss,'-b')
plt.plot(val,'-r')
plt.legend(["Training","validation"])
plt.xlabel('epoch')
plt.ylabel('loss')
```

Out[18]: Text(0, 0.5, 'loss')



In [19]:
```
class cnnModel1(cnnModel):
    def __init__(self):
        super().__init__()
        self.network=nn.Sequential(
        nn.Conv2d(3,32,kernel_size=3,padding=1),
        nn.ReLU(),
        nn.Conv2d(32,64,kernel_size=3,stride=1,padding=1),
        nn.MaxPool2d(2,2),
        nn.Conv2d(64,128,kernel_size=3,stride=1,padding=1),
        nn.Conv2d(128,128,kernel_size=3,stride=1,padding=1),
        nn.MaxPool2d(2,2),
        nn.Conv2d(128,256,kernel_size=3,stride=1,padding=1),
        nn.ReLU(),
        nn.Conv2d(256,256,kernel_size=3,stride=1,padding=1),
        nn.MaxPool2d(2,2),

        nn.Flatten(),
        nn.Linear(256*6*6,1024),
        nn.ReLU(),
        nn.Linear(1024,512),
        nn.ReLU(),
        nn.Linear(512,7)
        )
```

```
a=evaluate(model,val_batches)
print("val_loss:{},val_acc:{}".format(a[0],a[1]))
```

```
val_loss:1.9405790753662586,val_acc:0.251953125
```

```
num_epoch=10
opt_function=torch.optim.Adam
learning_rate=0.001
history=fit(num_epoch,learning_rate,model,train_batches,val_batches,opt_function)
```

```
Epoch [0],train_loss:1.8454,val_loss:1.8017,val_acc:0.2534
Epoch [1],train_loss:2.4405,val_loss:1.8253,val_acc:0.2539
Epoch [2],train_loss:1.8143,val_loss:1.7980,val_acc:0.2551
Epoch [3],train_loss:1.8049,val_loss:1.7906,val_acc:0.2598
Epoch [4],train_loss:1.7878,val_loss:1.7760,val_acc:0.2737
Epoch [5],train_loss:1.7304,val_loss:1.7021,val_acc:0.3066
Epoch [6],train_loss:1.6856,val_loss:1.6815,val_acc:0.3252
Epoch [7],train_loss:1.6375,val_loss:1.6194,val_acc:0.3647
Epoch [8],train_loss:1.7005,val_loss:1.8015,val_acc:0.2537
Epoch [9],train_loss:1.8007,val_loss:1.8015,val_acc:0.2415
```

```
valid_data=DataLoader(df_valid,batch_size=128,shuffle=True)
result=evaluate(model,valid_data)
```

```
print("val_loss:{},val_acc:{}".format(result[0],result[1]))
```

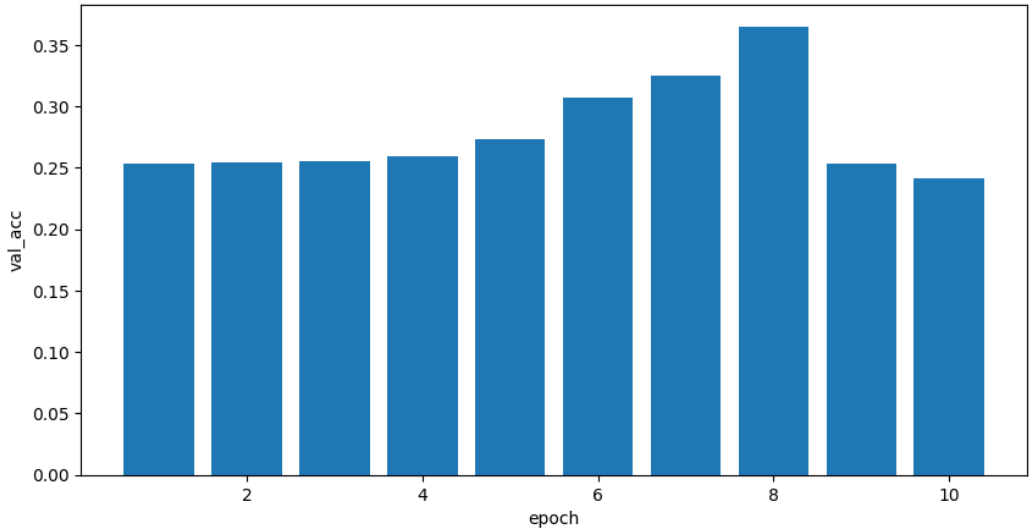
```
val_loss:1.8017291660819734,val_acc:0.25603107841951506
```

```
history
```

```
[[1.8017187006771564, 0.25341796875, 1.845390591424765],
 [1.825336031615734, 0.25390625, 2.4405429688925597],
 [1.7979875020682812, 0.255126953125, 1.8142768448775577],
 [1.7905513383448124, 0.259765625, 1.804943179654092],
 [1.7760060466825962, 0.273681640625, 1.7877776134259922],
 [1.7020943760871887, 0.306640625, 1.7303666229715053],
 [1.6814815811812878, 0.3251953125, 1.6855558009491753],
 [1.619397658854723, 0.36474609375, 1.6375403379656606],
 [1.8014682903885841, 0.253662109375, 1.7005142016509145],
 [1.8015021495521069, 0.241455078125, 1.8007008737510013]]
```

```
epoch=[i for i in range(1,11)]
acc=[x[1] for x in history]
fig = plt.figure(figsize = (10, 5))
plt.bar(epoch[:],acc[:])
plt.xlabel('epoch')
plt.ylabel('val_acc')
```

```
Text(0, 0.5, 'val_acc')
```




```
train_loss=[x[2] for x in history]
val=[x[0] for x in history]
plt.plot(train_loss,'-b')
plt.plot(val,'-r')
plt.legend(["Training","validation"])
plt.xlabel('epoch')
plt.ylabel('loss')
```

```
Text(0, 0.5, 'loss')
```