MLP on MNIST

In [1]:

```
from keras.utils import np_utils
from keras.datasets import mnist
from keras.initializers import RandomNormal
```

```
Using TensorFlow backend.
```

This MNIST dataset contains 70,000 hand-written digits in 28*28 pixels image. It has 10 class labels from 0 to 9

In [2]:

```
(x_train,y_train),(x_test,y_test)=mnist.load_data() #Loading the train and tes
t data of mnist dataset
print("total number of data points in train data",len(x_train))
print("total number of data points in test data",len(x_test))
```

```
total number of data points in train data 60000
total number of data points in test data 10000
```

In [3]:

```
print(x_train.shape)
print(x_test.shape)
```

```
(60000, 28, 28)
(10000, 28, 28)
```

we have 6000 image of 28*28

In [0]:

```
x_train=x_train.reshape(x_train.shape[0],x_train.shape[1]*x_train.shape[2])
x_test=x_test.reshape(x_test.shape[0],x_test.shape[1]*x_test.shape[2])
```

In [5]:

```
print(x_train.shape)
print(x_test.shape)
```

```
(60000, 784)
(10000, 784)
```

now we convert 3-dimention(60000,28,28) matrix into 2-dimention(60000,784) matrix

Data Normalization

here we normalize the data using X => (X - Xmin)/(Xmax-Xmin) = X/256

In [0]:

```
x_train=x_train/255
x_test=x_test/255
```

```
In [0]: y_train=np_utils.to_categorical(y_train,10)
        y_test=np_utils.to_categorical(y_test,10)
```

Building model

1. Two layers MNP with activation- ReLU and optimizer-Adam

Case 1-using with only BatchNormalization

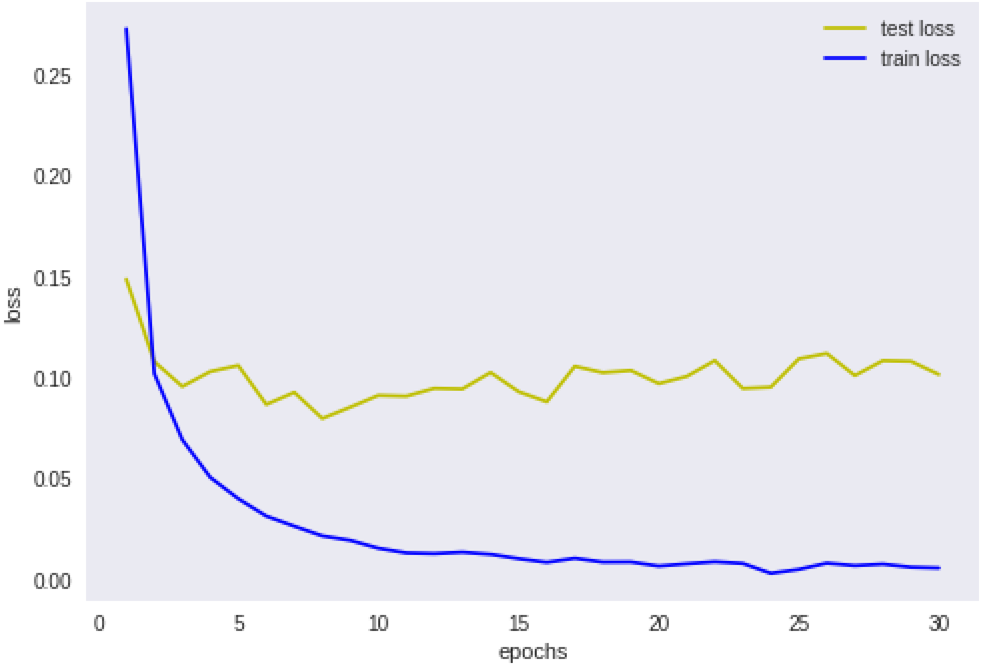
In [8]:
```
from keras.models import Sequential
from keras.layers import Activation
from keras.layers import Dense
from keras.layers.normalization import BatchNormalization

model_1=Sequential()
model_1.add(Dense(128,input_dim=784,activation="relu",kernel_initializer='he_n
ormal'))
model_1.add(BatchNormalization()) #batchNormalization
model_1.add(Dense(64,activation='relu',kernel_initializer='he_normal'))
model_1.add(Dense(10,activation='softmax'))
model_1.compile(optimizer='adam',loss='categorical_crossentropy',metrics=['acc
uracy'])
history1=model_1.fit(x_train,y_train,epochs=30,verbose=1,batch_size=128,valida
tion_data=(x_test,y_test))
```

```
Train on 60000 samples, validate on 10000 samples
Epoch 1/30
60000/60000 [==============================] - 3s 57us/step - loss: 0.2731 -
acc: 0.9195 - val_loss: 0.1487 - val_acc: 0.9536
Epoch 2/30
60000/60000 [==============================] - 3s 53us/step - loss: 0.1021 -
acc: 0.9695 - val_loss: 0.1081 - val_acc: 0.9676
Epoch 3/30
60000/60000 [==============================] - 3s 53us/step - loss: 0.0693 -
acc: 0.9792 - val_loss: 0.0957 - val_acc: 0.9694
Epoch 4/30
60000/60000 [==============================] - 3s 53us/step - loss: 0.0506 -
acc: 0.9843 - val_loss: 0.1031 - val_acc: 0.9668
Epoch 5/30
60000/60000 [==============================] - 3s 52us/step - loss: 0.0400 -
acc: 0.9870 - val_loss: 0.1061 - val_acc: 0.9679
Epoch 6/30
60000/60000 [==============================] - 3s 51us/step - loss: 0.0313 -
acc: 0.9902 - val_loss: 0.0868 - val_acc: 0.9759
Epoch 7/30
60000/60000 [==============================] - 3s 51us/step - loss: 0.0263 -
acc: 0.9916 - val_loss: 0.0928 - val_acc: 0.9734
Epoch 8/30
60000/60000 [==============================] - 3s 53us/step - loss: 0.0215 -
acc: 0.9930 - val_loss: 0.0798 - val_acc: 0.9771
Epoch 9/30
60000/60000 [==============================] - 3s 51us/step - loss: 0.0193 -
acc: 0.9937 - val_loss: 0.0854 - val_acc: 0.9761
Epoch 10/30
60000/60000 [==============================] - 3s 52us/step - loss: 0.0154 -
acc: 0.9950 - val_loss: 0.0913 - val_acc: 0.9760
Epoch 11/30
60000/60000 [==============================] - 3s 53us/step - loss: 0.0131 -
acc: 0.9958 - val_loss: 0.0909 - val_acc: 0.9763
Epoch 12/30
60000/60000 [==============================] - 3s 51us/step - loss: 0.0127 -
acc: 0.9958 - val_loss: 0.0948 - val_acc: 0.9759
Epoch 13/30
60000/60000 [==============================] - 3s 51us/step - loss: 0.0134 -
acc: 0.9954 - val_loss: 0.0945 - val_acc: 0.9755
Epoch 14/30
60000/60000 [==============================] - 3s 53us/step - loss: 0.0124 -
acc: 0.9960 - val_loss: 0.1026 - val_acc: 0.9740
Epoch 15/30
60000/60000 [==============================] - 3s 52us/step - loss: 0.0102 -
acc: 0.9967 - val_loss: 0.0931 - val_acc: 0.9774
Epoch 16/30
60000/60000 [==============================] - 3s 54us/step - loss: 0.0084 -
acc: 0.9974 - val_loss: 0.0881 - val_acc: 0.9785
Epoch 17/30
60000/60000 [==============================] - 3s 53us/step - loss: 0.0104 -
acc: 0.9965 - val_loss: 0.1058 - val_acc: 0.9764
Epoch 18/30
60000/60000 [==============================] - 3s 54us/step - loss: 0.0085 -
acc: 0.9971 - val_loss: 0.1026 - val_acc: 0.9760
Epoch 19/30
60000/60000 [==============================] - 3s 52us/step - loss: 0.0086 -
```

```
acc: 0.9973 - val_loss: 0.1037 - val_acc: 0.9785
Epoch 20/30
60000/60000 [==============================] - 3s 52us/step - loss: 0.0065 -
acc: 0.9980 - val_loss: 0.0971 - val_acc: 0.9796
Epoch 21/30
60000/60000 [==============================] - 3s 52us/step - loss: 0.0077 -
acc: 0.9972 - val_loss: 0.1007 - val_acc: 0.9770
Epoch 22/30
60000/60000 [==============================] - 3s 52us/step - loss: 0.0087 -
acc: 0.9969 - val_loss: 0.1086 - val_acc: 0.9788
Epoch 23/30
60000/60000 [==============================] - 3s 52us/step - loss: 0.0078 -
acc: 0.9976 - val_loss: 0.0947 - val_acc: 0.9790
Epoch 24/30
60000/60000 [==============================] - 3s 52us/step - loss: 0.0029 -
acc: 0.9992 - val_loss: 0.0954 - val_acc: 0.9799
Epoch 25/30
60000/60000 [==============================] - 3s 53us/step - loss: 0.0049 -
acc: 0.9983 - val_loss: 0.1095 - val_acc: 0.9765
Epoch 26/30
60000/60000 [==============================] - 3s 50us/step - loss: 0.0080 -
acc: 0.9972 - val_loss: 0.1121 - val_acc: 0.9778
Epoch 27/30
60000/60000 [==============================] - 3s 50us/step - loss: 0.0068 -
acc: 0.9977 - val_loss: 0.1011 - val_acc: 0.9807
Epoch 28/30
60000/60000 [==============================] - 3s 50us/step - loss: 0.0075 -
acc: 0.9974 - val_loss: 0.1085 - val_acc: 0.9774
Epoch 29/30
60000/60000 [==============================] - 3s 50us/step - loss: 0.0060 -
acc: 0.9979 - val_loss: 0.1082 - val_acc: 0.9776
Epoch 30/30
60000/60000 [==============================] - 3s 51us/step - loss: 0.0056 -
acc: 0.9980 - val_loss: 0.1017 - val_acc: 0.9793
```

```
In [9]: import matplotlib.pyplot as plt

        score = model_1.evaluate(x_test, y_test, verbose=1)
        print('Test error:', score[0])
        print('Test accuracy:', score[1])

        x=list(range(1,31))

        vl1=history1.history['val_loss']
        l1=history1.history['loss']
        plt.plot(x,vl1,color='y',label='test loss')
        plt.plot(x,l1,color='b',label='train loss')
        plt.legend()
        plt.xlabel('epochs')
        plt.ylabel('loss')


        plt.grid()
        plt.show()
```

```
10000/10000 [==============================] - 0s 29us/step
Test error: 0.1016737612550397
Test accuracy: 0.9793
```



case 2-with Dropout only

In [10]:

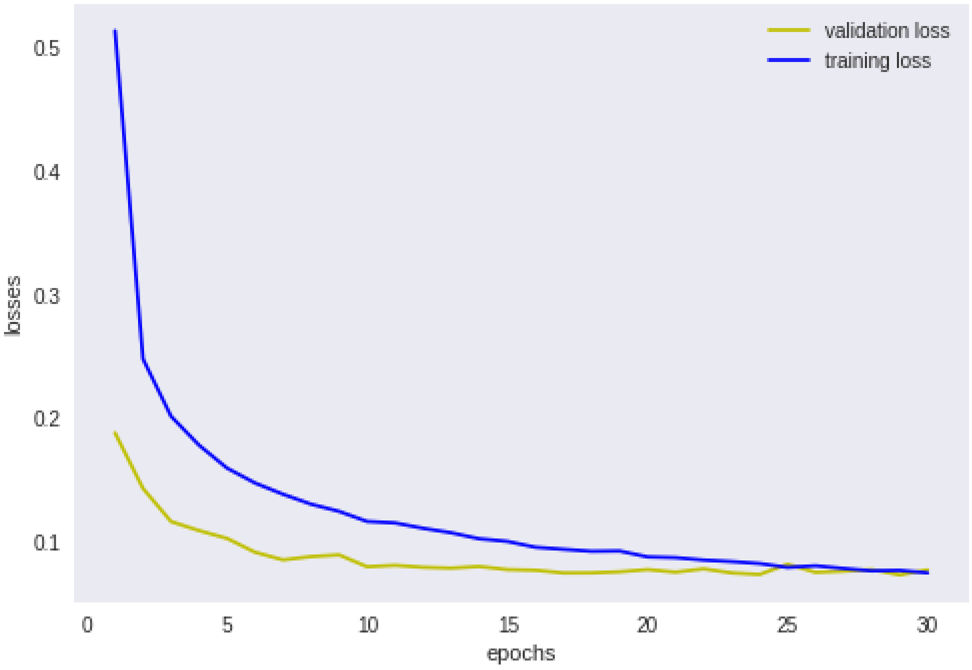
```
from keras.layers import Dropout
model_2=Sequential()
model_2.add(Dense(128,input_dim=784,activation="relu",kernel_initializer='he_normal'))
model_2.add(Dropout(0.5))
model_2.add(Dense(64,activation='relu',kernel_initializer='he_normal'))
model_2.add(Dense(10,activation='softmax'))
model_2.compile(optimizer='adam',loss='categorical_crossentropy',metrics=['accuracy'])
history2=model_2.fit(x_train,y_train,epochs=30,verbose=1,batch_size=128,validation_data=(x_test,y_test))
```

```
Train on 60000 samples, validate on 10000 samples
Epoch 1/30
60000/60000 [==============================] - 3s 50us/step - loss: 0.5139 -
acc: 0.8447 - val_loss: 0.1883 - val_acc: 0.9442
Epoch 2/30
60000/60000 [==============================] - 3s 45us/step - loss: 0.2482 -
acc: 0.9263 - val_loss: 0.1432 - val_acc: 0.9566
Epoch 3/30
60000/60000 [==============================] - 3s 45us/step - loss: 0.2018 -
acc: 0.9400 - val_loss: 0.1164 - val_acc: 0.9636
Epoch 4/30
60000/60000 [==============================] - 3s 45us/step - loss: 0.1783 -
acc: 0.9462 - val_loss: 0.1091 - val_acc: 0.9685
Epoch 5/30
60000/60000 [==============================] - 3s 44us/step - loss: 0.1597 -
acc: 0.9504 - val_loss: 0.1028 - val_acc: 0.9696
Epoch 6/30
60000/60000 [==============================] - 3s 44us/step - loss: 0.1477 -
acc: 0.9549 - val_loss: 0.0915 - val_acc: 0.9729
Epoch 7/30
60000/60000 [==============================] - 3s 44us/step - loss: 0.1385 -
acc: 0.9567 - val_loss: 0.0853 - val_acc: 0.9735
Epoch 8/30
60000/60000 [==============================] - 3s 44us/step - loss: 0.1305 -
acc: 0.9599 - val_loss: 0.0880 - val_acc: 0.9745
Epoch 9/30
60000/60000 [==============================] - 3s 44us/step - loss: 0.1247 -
acc: 0.9607 - val_loss: 0.0894 - val_acc: 0.9738
Epoch 10/30
60000/60000 [==============================] - 3s 44us/step - loss: 0.1165 -
acc: 0.9634 - val_loss: 0.0798 - val_acc: 0.9768
Epoch 11/30
60000/60000 [==============================] - 3s 44us/step - loss: 0.1153 -
acc: 0.9640 - val_loss: 0.0809 - val_acc: 0.9763
Epoch 12/30
60000/60000 [==============================] - 3s 44us/step - loss: 0.1109 -
acc: 0.9654 - val_loss: 0.0793 - val_acc: 0.9776
Epoch 13/30
60000/60000 [==============================] - 3s 44us/step - loss: 0.1073 -
acc: 0.9661 - val_loss: 0.0787 - val_acc: 0.9762
Epoch 14/30
60000/60000 [==============================] - 3s 44us/step - loss: 0.1025 -
acc: 0.9671 - val_loss: 0.0800 - val_acc: 0.9776
Epoch 15/30
60000/60000 [==============================] - 3s 44us/step - loss: 0.1003 -
acc: 0.9673 - val_loss: 0.0775 - val_acc: 0.9775
Epoch 16/30
60000/60000 [==============================] - 3s 44us/step - loss: 0.0956 -
acc: 0.9690 - val_loss: 0.0769 - val_acc: 0.9777
Epoch 17/30
60000/60000 [==============================] - 3s 44us/step - loss: 0.0939 -
acc: 0.9697 - val_loss: 0.0747 - val_acc: 0.9789
Epoch 18/30
60000/60000 [==============================] - 3s 44us/step - loss: 0.0923 -
acc: 0.9700 - val_loss: 0.0748 - val_acc: 0.9799
Epoch 19/30
60000/60000 [==============================] - 3s 44us/step - loss: 0.0925 -
```

```
acc: 0.9701 - val_loss: 0.0756 - val_acc: 0.9786
Epoch 20/30
60000/60000 [==============================] - 3s 44us/step - loss: 0.0878 -
acc: 0.9711 - val_loss: 0.0775 - val_acc: 0.9779
Epoch 21/30
60000/60000 [==============================] - 3s 44us/step - loss: 0.0872 -
acc: 0.9708 - val_loss: 0.0754 - val_acc: 0.9792
Epoch 22/30
60000/60000 [==============================] - 3s 44us/step - loss: 0.0852 -
acc: 0.9718 - val_loss: 0.0781 - val_acc: 0.9786
Epoch 23/30
60000/60000 [==============================] - 3s 45us/step - loss: 0.0840 -
acc: 0.9726 - val_loss: 0.0748 - val_acc: 0.9798
Epoch 24/30
60000/60000 [==============================] - 3s 45us/step - loss: 0.0824 -
acc: 0.9734 - val_loss: 0.0735 - val_acc: 0.9804
Epoch 25/30
60000/60000 [==============================] - 3s 45us/step - loss: 0.0792 -
acc: 0.9740 - val_loss: 0.0818 - val_acc: 0.9788
Epoch 26/30
60000/60000 [==============================] - 3s 45us/step - loss: 0.0806 -
acc: 0.9726 - val_loss: 0.0752 - val_acc: 0.9786
Epoch 27/30
60000/60000 [==============================] - 3s 46us/step - loss: 0.0784 -
acc: 0.9737 - val_loss: 0.0761 - val_acc: 0.9794
Epoch 28/30
60000/60000 [==============================] - 3s 46us/step - loss: 0.0764 -
acc: 0.9748 - val_loss: 0.0775 - val_acc: 0.9800
Epoch 29/30
60000/60000 [==============================] - 3s 45us/step - loss: 0.0768 -
acc: 0.9743 - val_loss: 0.0731 - val_acc: 0.9805
Epoch 30/30
60000/60000 [==============================] - 3s 45us/step - loss: 0.0748 -
acc: 0.9754 - val_loss: 0.0773 - val_acc: 0.9788
```

In [11]:
```
score=model_2.evaluate(x_test,y_test,verbose=1)
print("test error",score[0])
print("test accuracy",score[1])

x=list(range(1,31))
vl2=history2.history['val_loss']
l2=history2.history['loss']
plt.plot(x,vl2,'y',label='validation loss')
plt.plot(x,l2,'b',label='training loss')
plt.legend()
plt.xlabel("epochs")
plt.ylabel("losses")
plt.grid()
plt.show()
```

```
10000/10000 [==============================] - 0s 26us/step
test error 0.07725040358993283
test accuracy 0.9788
```



case 3- with both Dropout and BatchNormalization

In [12]:

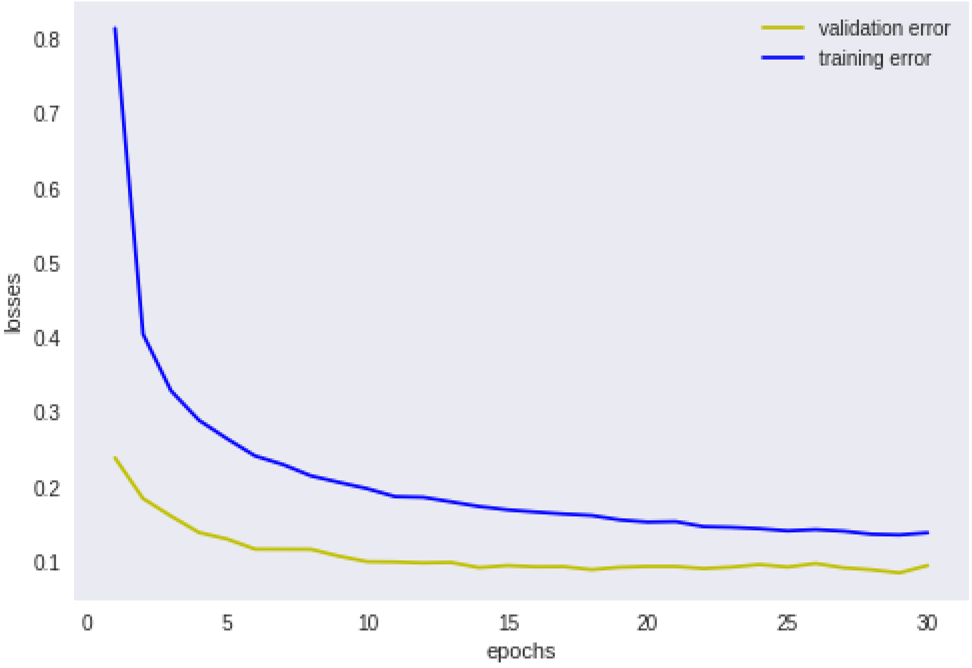
```
model_3=Sequential()
model_3.add(Dense(128,input_dim=784,activation='relu',kernel_initializer='he_n
ormal'))
model_3.add(BatchNormalization())
model_3.add(Dropout(0.5))
model_3.add(Dense(64,activation='relu',kernel_initializer='he_normal'))
model_3.add(Dropout(0.5))
model_3.add(Dense(10,activation='softmax'))
model_3.compile(loss='categorical_crossentropy',metrics=['accuracy'],optimizer
='adam')
history3=model_3.fit(x_train,y_train,epochs=30,batch_size=128,verbose=1,valida
tion_data=(x_test,y_test))
```

```
Train on 60000 samples, validate on 10000 samples
Epoch 1/30
60000/60000 [==============================] - 4s 60us/step - loss: 0.8147 -
acc: 0.7492 - val_loss: 0.2388 - val_acc: 0.9277
Epoch 2/30
60000/60000 [==============================] - 3s 51us/step - loss: 0.4049 -
acc: 0.8805 - val_loss: 0.1842 - val_acc: 0.9441
Epoch 3/30
60000/60000 [==============================] - 3s 51us/step - loss: 0.3288 -
acc: 0.9053 - val_loss: 0.1606 - val_acc: 0.9517
Epoch 4/30
60000/60000 [==============================] - 3s 51us/step - loss: 0.2892 -
acc: 0.9182 - val_loss: 0.1385 - val_acc: 0.9578
Epoch 5/30
60000/60000 [==============================] - 3s 52us/step - loss: 0.2645 -
acc: 0.9236 - val_loss: 0.1298 - val_acc: 0.9605
Epoch 6/30
60000/60000 [==============================] - 3s 52us/step - loss: 0.2414 -
acc: 0.9320 - val_loss: 0.1162 - val_acc: 0.9644
Epoch 7/30
60000/60000 [==============================] - 3s 52us/step - loss: 0.2298 -
acc: 0.9351 - val_loss: 0.1161 - val_acc: 0.9641
Epoch 8/30
60000/60000 [==============================] - 3s 51us/step - loss: 0.2146 -
acc: 0.9392 - val_loss: 0.1160 - val_acc: 0.9668
Epoch 9/30
60000/60000 [==============================] - 3s 52us/step - loss: 0.2059 -
acc: 0.9417 - val_loss: 0.1068 - val_acc: 0.9680
Epoch 10/30
60000/60000 [==============================] - 3s 52us/step - loss: 0.1975 -
acc: 0.9450 - val_loss: 0.0994 - val_acc: 0.9716
Epoch 11/30
60000/60000 [==============================] - 3s 52us/step - loss: 0.1867 -
acc: 0.9459 - val_loss: 0.0990 - val_acc: 0.9710
Epoch 12/30
60000/60000 [==============================] - 3s 52us/step - loss: 0.1859 -
acc: 0.9464 - val_loss: 0.0978 - val_acc: 0.9698
Epoch 13/30
60000/60000 [==============================] - 3s 58us/step - loss: 0.1797 -
acc: 0.9490 - val_loss: 0.0984 - val_acc: 0.9713
Epoch 14/30
60000/60000 [==============================] - 3s 58us/step - loss: 0.1735 -
acc: 0.9507 - val_loss: 0.0915 - val_acc: 0.9727
Epoch 15/30
60000/60000 [==============================] - 4s 60us/step - loss: 0.1690 -
acc: 0.9514 - val_loss: 0.0942 - val_acc: 0.9739
Epoch 16/30
60000/60000 [==============================] - 3s 57us/step - loss: 0.1661 -
acc: 0.9525 - val_loss: 0.0925 - val_acc: 0.9734
Epoch 17/30
60000/60000 [==============================] - 3s 53us/step - loss: 0.1635 -
acc: 0.9521 - val_loss: 0.0928 - val_acc: 0.9732
Epoch 18/30
60000/60000 [==============================] - 3s 53us/step - loss: 0.1614 -
acc: 0.9541 - val_loss: 0.0887 - val_acc: 0.9741
Epoch 19/30
60000/60000 [==============================] - 3s 52us/step - loss: 0.1555 -
```

```
acc: 0.9553 - val_loss: 0.0919 - val_acc: 0.9732
Epoch 20/30
60000/60000 [==============================] - 3s 52us/step - loss: 0.1526 -
acc: 0.9565 - val_loss: 0.0930 - val_acc: 0.9733
Epoch 21/30
60000/60000 [==============================] - 3s 52us/step - loss: 0.1532 -
acc: 0.9559 - val_loss: 0.0930 - val_acc: 0.9732
Epoch 22/30
60000/60000 [==============================] - 3s 52us/step - loss: 0.1465 -
acc: 0.9573 - val_loss: 0.0902 - val_acc: 0.9739
Epoch 23/30
60000/60000 [==============================] - 3s 51us/step - loss: 0.1455 -
acc: 0.9584 - val_loss: 0.0921 - val_acc: 0.9727
Epoch 24/30
60000/60000 [==============================] - 3s 51us/step - loss: 0.1437 -
acc: 0.9579 - val_loss: 0.0956 - val_acc: 0.9727
Epoch 25/30
60000/60000 [==============================] - 3s 53us/step - loss: 0.1408 -
acc: 0.9578 - val_loss: 0.0922 - val_acc: 0.9734
Epoch 26/30
60000/60000 [==============================] - 3s 53us/step - loss: 0.1425 -
acc: 0.9592 - val_loss: 0.0970 - val_acc: 0.9722
Epoch 27/30
60000/60000 [==============================] - 3s 52us/step - loss: 0.1404 -
acc: 0.9597 - val_loss: 0.0912 - val_acc: 0.9729
Epoch 28/30
60000/60000 [==============================] - 3s 52us/step - loss: 0.1363 -
acc: 0.9601 - val_loss: 0.0885 - val_acc: 0.9744
Epoch 29/30
60000/60000 [==============================] - 3s 54us/step - loss: 0.1353 -
acc: 0.9604 - val_loss: 0.0844 - val_acc: 0.9747
Epoch 30/30
60000/60000 [==============================] - 3s 53us/step - loss: 0.1382 -
acc: 0.9595 - val_loss: 0.0942 - val_acc: 0.9740
```

In [13]:
```
score = model_3.evaluate(x_test, y_test, verbose=1)
print('Test error:', score[0])
print('Test accuracy:', score[1])


x=list(range(1,31))
vl3=history3.history['val_loss']
l3=history3.history['loss']
plt.plot(x,vl3,'y',label='validation error')
plt.plot(x,l3,'b',label='training error')
plt.legend()
plt.grid()
plt.xlabel('epochs')
plt.ylabel('losses')
plt.show()
```

```
10000/10000 [==============================] - 0s 31us/step
Test error: 0.09424658504130203
Test accuracy: 0.974
```



case 4- without Dropout and BatchNormalization

In [14]:

```
model_4=Sequential()
model_4.add(Dense(128,input_dim=784,activation='relu',kernel_initializer='he_normal'))
model_4.add(Dense(64,activation='relu',kernel_initializer='he_normal'))
model_4.add(Dropout(0.5))
model_4.add(Dense(10,activation='softmax'))
model_4.compile(loss='categorical_crossentropy',metrics=['accuracy'],optimizer='adam')
history4=model_4.fit(x_train,y_train,epochs=50,batch_size=128,verbose=1,validation_data=(x_test,y_test))
```

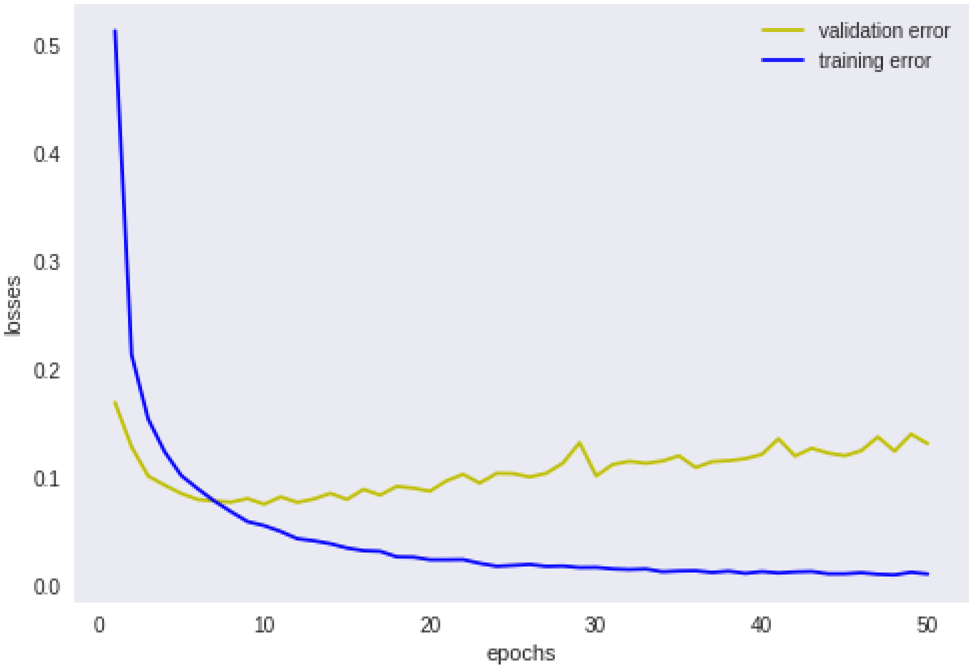
```
Train on 60000 samples, validate on 10000 samples
Epoch 1/50
60000/60000 [==============================] - 3s 48us/step - loss: 0.5135 -
acc: 0.8470 - val_loss: 0.1695 - val_acc: 0.9464
Epoch 2/50
60000/60000 [==============================] - 2s 42us/step - loss: 0.2138 -
acc: 0.9395 - val_loss: 0.1285 - val_acc: 0.9594
Epoch 3/50
60000/60000 [==============================] - 3s 42us/step - loss: 0.1542 -
acc: 0.9567 - val_loss: 0.1014 - val_acc: 0.9687
Epoch 4/50
60000/60000 [==============================] - 2s 41us/step - loss: 0.1237 -
acc: 0.9651 - val_loss: 0.0929 - val_acc: 0.9720
Epoch 5/50
60000/60000 [==============================] - 3s 42us/step - loss: 0.1018 -
acc: 0.9710 - val_loss: 0.0852 - val_acc: 0.9754
Epoch 6/50
60000/60000 [==============================] - 3s 42us/step - loss: 0.0895 -
acc: 0.9745 - val_loss: 0.0796 - val_acc: 0.9757
Epoch 7/50
60000/60000 [==============================] - 3s 42us/step - loss: 0.0782 -
acc: 0.9773 - val_loss: 0.0783 - val_acc: 0.9770
Epoch 8/50
60000/60000 [==============================] - 2s 41us/step - loss: 0.0683 -
acc: 0.9796 - val_loss: 0.0772 - val_acc: 0.9789
Epoch 9/50
60000/60000 [==============================] - 3s 42us/step - loss: 0.0591 -
acc: 0.9820 - val_loss: 0.0805 - val_acc: 0.9758
Epoch 10/50
60000/60000 [==============================] - 3s 42us/step - loss: 0.0554 -
acc: 0.9836 - val_loss: 0.0753 - val_acc: 0.9787
Epoch 11/50
60000/60000 [==============================] - 3s 42us/step - loss: 0.0500 -
acc: 0.9847 - val_loss: 0.0821 - val_acc: 0.9779
Epoch 12/50
60000/60000 [==============================] - 3s 43us/step - loss: 0.0433 -
acc: 0.9868 - val_loss: 0.0770 - val_acc: 0.9798
Epoch 13/50
60000/60000 [==============================] - 2s 41us/step - loss: 0.0413 -
acc: 0.9873 - val_loss: 0.0801 - val_acc: 0.9794
Epoch 14/50
60000/60000 [==============================] - 2s 41us/step - loss: 0.0387 -
acc: 0.9883 - val_loss: 0.0853 - val_acc: 0.9788
Epoch 15/50
60000/60000 [==============================] - 2s 41us/step - loss: 0.0347 -
acc: 0.9892 - val_loss: 0.0797 - val_acc: 0.9799
Epoch 16/50
60000/60000 [==============================] - 2s 41us/step - loss: 0.0322 -
acc: 0.9897 - val_loss: 0.0890 - val_acc: 0.9797
Epoch 17/50
60000/60000 [==============================] - 2s 41us/step - loss: 0.0316 -
acc: 0.9896 - val_loss: 0.0837 - val_acc: 0.9814
Epoch 18/50
60000/60000 [==============================] - 2s 41us/step - loss: 0.0265 -
acc: 0.9914 - val_loss: 0.0918 - val_acc: 0.9807
Epoch 19/50
60000/60000 [==============================] - 2s 41us/step - loss: 0.0263 -
```

```
acc: 0.9917 - val_loss: 0.0902 - val_acc: 0.9810
Epoch 20/50
60000/60000 [==============================] - 2s 42us/step - loss: 0.0238 -
acc: 0.9921 - val_loss: 0.0873 - val_acc: 0.9808
Epoch 21/50
60000/60000 [==============================] - 2s 41us/step - loss: 0.0238 -
acc: 0.9922 - val_loss: 0.0969 - val_acc: 0.9797
Epoch 22/50
60000/60000 [==============================] - 2s 41us/step - loss: 0.0240 -
acc: 0.9922 - val_loss: 0.1030 - val_acc: 0.9789
Epoch 23/50
60000/60000 [==============================] - 3s 42us/step - loss: 0.0205 -
acc: 0.9935 - val_loss: 0.0948 - val_acc: 0.9804
Epoch 24/50
60000/60000 [==============================] - 2s 42us/step - loss: 0.0177 -
acc: 0.9941 - val_loss: 0.1039 - val_acc: 0.9794
Epoch 25/50
60000/60000 [==============================] - 2s 41us/step - loss: 0.0186 -
acc: 0.9938 - val_loss: 0.1036 - val_acc: 0.9813
Epoch 26/50
60000/60000 [==============================] - 2s 41us/step - loss: 0.0195 -
acc: 0.9936 - val_loss: 0.1005 - val_acc: 0.9803
Epoch 27/50
60000/60000 [==============================] - 2s 41us/step - loss: 0.0176 -
acc: 0.9941 - val_loss: 0.1040 - val_acc: 0.9788
Epoch 28/50
60000/60000 [==============================] - 2s 41us/step - loss: 0.0180 -
acc: 0.9940 - val_loss: 0.1134 - val_acc: 0.9796
Epoch 29/50
60000/60000 [==============================] - 3s 42us/step - loss: 0.0166 -
acc: 0.9945 - val_loss: 0.1324 - val_acc: 0.9758
Epoch 30/50
60000/60000 [==============================] - 3s 42us/step - loss: 0.0167 -
acc: 0.9945 - val_loss: 0.1015 - val_acc: 0.9809
Epoch 31/50
60000/60000 [==============================] - 3s 42us/step - loss: 0.0153 -
acc: 0.9949 - val_loss: 0.1121 - val_acc: 0.9799
Epoch 32/50
60000/60000 [==============================] - 2s 41us/step - loss: 0.0147 -
acc: 0.9954 - val_loss: 0.1150 - val_acc: 0.9793
Epoch 33/50
60000/60000 [==============================] - 2s 41us/step - loss: 0.0153 -
acc: 0.9948 - val_loss: 0.1131 - val_acc: 0.9790
Epoch 34/50
60000/60000 [==============================] - 2s 41us/step - loss: 0.0127 -
acc: 0.9959 - val_loss: 0.1153 - val_acc: 0.9794
Epoch 35/50
60000/60000 [==============================] - 2s 42us/step - loss: 0.0134 -
acc: 0.9957 - val_loss: 0.1201 - val_acc: 0.9786
Epoch 36/50
60000/60000 [==============================] - 2s 42us/step - loss: 0.0136 -
acc: 0.9953 - val_loss: 0.1092 - val_acc: 0.9804
Epoch 37/50
60000/60000 [==============================] - 2s 41us/step - loss: 0.0120 -
acc: 0.9959 - val_loss: 0.1147 - val_acc: 0.9828
Epoch 38/50
60000/60000 [==============================] - 2s 41us/step - loss: 0.0132 -
```

```
acc: 0.9957 - val_loss: 0.1156 - val_acc: 0.9796
Epoch 39/50
60000/60000 [==============================] - 2s 41us/step - loss: 0.0112 -
acc: 0.9961 - val_loss: 0.1174 - val_acc: 0.9814
Epoch 40/50
60000/60000 [==============================] - 2s 41us/step - loss: 0.0128 -
acc: 0.9958 - val_loss: 0.1214 - val_acc: 0.9796
Epoch 41/50
60000/60000 [==============================] - 2s 41us/step - loss: 0.0115 -
acc: 0.9959 - val_loss: 0.1359 - val_acc: 0.9784
Epoch 42/50
60000/60000 [==============================] - 2s 41us/step - loss: 0.0125 -
acc: 0.9959 - val_loss: 0.1199 - val_acc: 0.9814
Epoch 43/50
60000/60000 [==============================] - 2s 41us/step - loss: 0.0129 -
acc: 0.9957 - val_loss: 0.1271 - val_acc: 0.9788
Epoch 44/50
60000/60000 [==============================] - 2s 41us/step - loss: 0.0108 -
acc: 0.9962 - val_loss: 0.1226 - val_acc: 0.9803
Epoch 45/50
60000/60000 [==============================] - 2s 41us/step - loss: 0.0108 -
acc: 0.9967 - val_loss: 0.1202 - val_acc: 0.9801
Epoch 46/50
60000/60000 [==============================] - 2s 41us/step - loss: 0.0118 -
acc: 0.9962 - val_loss: 0.1249 - val_acc: 0.9814
Epoch 47/50
60000/60000 [==============================] - 2s 41us/step - loss: 0.0104 -
acc: 0.9966 - val_loss: 0.1377 - val_acc: 0.9784
Epoch 48/50
60000/60000 [==============================] - 2s 41us/step - loss: 0.0097 -
acc: 0.9967 - val_loss: 0.1244 - val_acc: 0.9802
Epoch 49/50
60000/60000 [==============================] - 3s 42us/step - loss: 0.0122 -
acc: 0.9961 - val_loss: 0.1401 - val_acc: 0.9789
Epoch 50/50
60000/60000 [==============================] - 3s 43us/step - loss: 0.0106 -
acc: 0.9968 - val_loss: 0.1314 - val_acc: 0.9802
```

```
In [15]: score=model_4.evaluate(x_test,y_test)
         print("test error",score[0])
         print("test accuracy",score[1])

         x1=list(range(1,51))
         vl4=history4.history['val_loss']
         l4=history4.history['loss']
         plt.plot(x1,vl4,'y',label='validation error')
         plt.plot(x1,l4,'b',label='training error')
         plt.legend()
         plt.grid()
         plt.xlabel('epochs')
         plt.ylabel('losses')
         plt.show()
```

```
10000/10000 [==============================] - 0s 27us/step
test error 0.13143735942197668
test accuracy 0.9802
```



2. Three layers MNP with activation- ReLU and optimizer-Adam

case 1-with only BatchNormalization

In [16]:

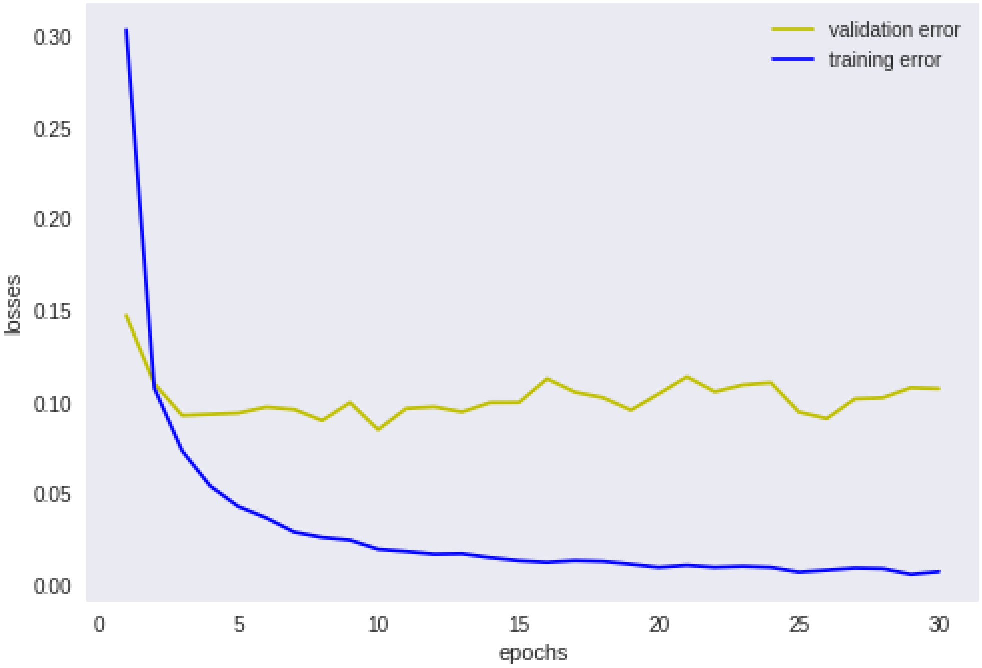
```
model_5=Sequential()
model_5.add(Dense(128,input_dim=784,activation='relu',kernel_initializer='he_normal'))
model_5.add(BatchNormalization())
model_5.add(Dense(64,activation='relu',kernel_initializer='he_normal'))
model_5.add(BatchNormalization())
model_5.add(Dense(32,activation='relu',kernel_initializer='he_normal'))
model_5.add(Dense(10,activation='softmax'))
model_5.compile(optimizer='adam',loss='categorical_crossentropy',metrics=['accuracy'])
history5=model_5.fit(x_train,y_train,epochs=30,batch_size=128,verbose=1,validation_data=(x_test,y_test))
```

```
Train on 60000 samples, validate on 10000 samples
Epoch 1/30
60000/60000 [==============================] - 4s 69us/step - loss: 0.3039 -
acc: 0.9120 - val_loss: 0.1472 - val_acc: 0.9565
Epoch 2/30
60000/60000 [==============================] - 3s 55us/step - loss: 0.1079 -
acc: 0.9666 - val_loss: 0.1103 - val_acc: 0.9647
Epoch 3/30
60000/60000 [==============================] - 3s 56us/step - loss: 0.0733 -
acc: 0.9767 - val_loss: 0.0928 - val_acc: 0.9708
Epoch 4/30
60000/60000 [==============================] - 3s 55us/step - loss: 0.0541 -
acc: 0.9835 - val_loss: 0.0935 - val_acc: 0.9717
Epoch 5/30
60000/60000 [==============================] - 3s 55us/step - loss: 0.0429 -
acc: 0.9860 - val_loss: 0.0941 - val_acc: 0.9727
Epoch 6/30
60000/60000 [==============================] - 3s 54us/step - loss: 0.0366 -
acc: 0.9885 - val_loss: 0.0974 - val_acc: 0.9715
Epoch 7/30
60000/60000 [==============================] - 3s 55us/step - loss: 0.0288 -
acc: 0.9906 - val_loss: 0.0961 - val_acc: 0.9735
Epoch 8/30
60000/60000 [==============================] - 3s 55us/step - loss: 0.0258 -
acc: 0.9919 - val_loss: 0.0900 - val_acc: 0.9737
Epoch 9/30
60000/60000 [==============================] - 3s 55us/step - loss: 0.0244 -
acc: 0.9917 - val_loss: 0.0999 - val_acc: 0.9730
Epoch 10/30
60000/60000 [==============================] - 3s 55us/step - loss: 0.0193 -
acc: 0.9937 - val_loss: 0.0850 - val_acc: 0.9771
Epoch 11/30
60000/60000 [==============================] - 3s 54us/step - loss: 0.0181 -
acc: 0.9941 - val_loss: 0.0966 - val_acc: 0.9742
Epoch 12/30
60000/60000 [==============================] - 3s 55us/step - loss: 0.0166 -
acc: 0.9944 - val_loss: 0.0975 - val_acc: 0.9736
Epoch 13/30
60000/60000 [==============================] - 3s 55us/step - loss: 0.0169 -
acc: 0.9943 - val_loss: 0.0947 - val_acc: 0.9758
Epoch 14/30
60000/60000 [==============================] - 3s 55us/step - loss: 0.0147 -
acc: 0.9949 - val_loss: 0.0998 - val_acc: 0.9768
Epoch 15/30
60000/60000 [==============================] - 3s 54us/step - loss: 0.0131 -
acc: 0.9958 - val_loss: 0.0999 - val_acc: 0.9764
Epoch 16/30
60000/60000 [==============================] - 3s 54us/step - loss: 0.0122 -
acc: 0.9956 - val_loss: 0.1129 - val_acc: 0.9734
Epoch 17/30
60000/60000 [==============================] - 3s 53us/step - loss: 0.0132 -
acc: 0.9955 - val_loss: 0.1056 - val_acc: 0.9745
Epoch 18/30
60000/60000 [==============================] - 3s 55us/step - loss: 0.0127 -
acc: 0.9952 - val_loss: 0.1026 - val_acc: 0.9754
Epoch 19/30
60000/60000 [==============================] - 3s 56us/step - loss: 0.0111 -
```

```
acc: 0.9962 - val_loss: 0.0957 - val_acc: 0.9771
Epoch 20/30
60000/60000 [==============================] - 3s 57us/step - loss: 0.0094 -
acc: 0.9966 - val_loss: 0.1047 - val_acc: 0.9757
Epoch 21/30
60000/60000 [==============================] - 3s 55us/step - loss: 0.0105 -
acc: 0.9965 - val_loss: 0.1140 - val_acc: 0.9742
Epoch 22/30
60000/60000 [==============================] - 3s 55us/step - loss: 0.0094 -
acc: 0.9966 - val_loss: 0.1058 - val_acc: 0.9771
Epoch 23/30
60000/60000 [==============================] - 3s 54us/step - loss: 0.0100 -
acc: 0.9966 - val_loss: 0.1095 - val_acc: 0.9748
Epoch 24/30
60000/60000 [==============================] - 3s 53us/step - loss: 0.0094 -
acc: 0.9967 - val_loss: 0.1107 - val_acc: 0.9757
Epoch 25/30
60000/60000 [==============================] - 3s 54us/step - loss: 0.0068 -
acc: 0.9975 - val_loss: 0.0947 - val_acc: 0.9785
Epoch 26/30
60000/60000 [==============================] - 3s 54us/step - loss: 0.0079 -
acc: 0.9974 - val_loss: 0.0911 - val_acc: 0.9793
Epoch 27/30
60000/60000 [==============================] - 3s 54us/step - loss: 0.0090 -
acc: 0.9969 - val_loss: 0.1019 - val_acc: 0.9778
Epoch 28/30
60000/60000 [==============================] - 3s 54us/step - loss: 0.0087 -
acc: 0.9971 - val_loss: 0.1025 - val_acc: 0.9784
Epoch 29/30
60000/60000 [==============================] - 3s 55us/step - loss: 0.0055 -
acc: 0.9981 - val_loss: 0.1080 - val_acc: 0.9775
Epoch 30/30
60000/60000 [==============================] - 3s 54us/step - loss: 0.0070 -
acc: 0.9977 - val_loss: 0.1075 - val_acc: 0.9781
```

In [17]:
```
score=model_5.evaluate(x_test,y_test,verbose=0)
print('test error',score[0])
print('test accuracy',score[1])

x=list(range(1,31))
vl5=history5.history['val_loss']
l5=history5.history['loss']
plt.plot(x,vl5,'y',label='validation error')
plt.plot(x,l5,'b',label='training error')
plt.legend()
plt.xlabel('epochs')
plt.ylabel('losses')
plt.grid()
plt.show()
```

```
test error 0.10749240423332085
test accuracy 0.9781
```



case 2-with Dropout

In [18]:

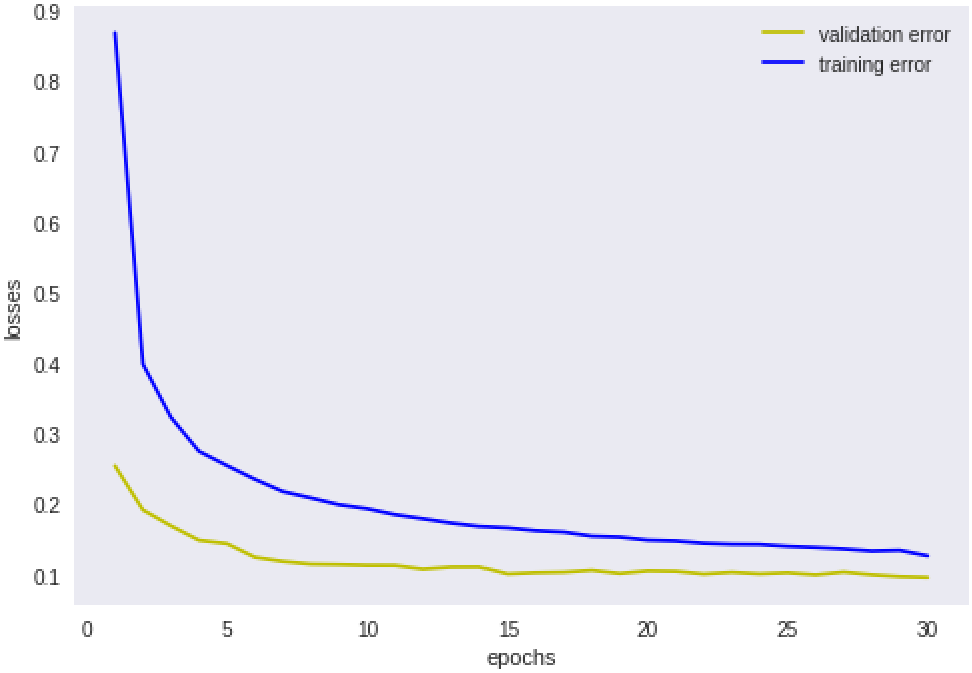
```
model_6=Sequential()
model_6.add(Dense(128,input_dim=784,activation='relu',kernel_initializer='he_normal'))
model_6.add(Dropout(0.5))
model_6.add(Dense(64,activation='relu',kernel_initializer='he_normal'))
model_6.add(Dropout(0.5))
model_6.add(Dense(32,activation='relu',kernel_initializer='he_normal'))
model_6.add(Dense(10,activation='softmax'))
model_6.compile(loss='categorical_crossentropy',optimizer='adam',metrics=['accuracy'])
history6=model_6.fit(x_train,y_train,epochs=30,batch_size=128,verbose=1,validation_data=(x_test,y_test))
```

```
Train on 60000 samples, validate on 10000 samples
Epoch 1/30
60000/60000 [==============================] - 3s 56us/step - loss: 0.8688 -
acc: 0.7177 - val_loss: 0.2545 - val_acc: 0.9264
Epoch 2/30
60000/60000 [==============================] - 3s 46us/step - loss: 0.3992 -
acc: 0.8862 - val_loss: 0.1918 - val_acc: 0.9437
Epoch 3/30
60000/60000 [==============================] - 3s 46us/step - loss: 0.3235 -
acc: 0.9096 - val_loss: 0.1693 - val_acc: 0.9517
Epoch 4/30
60000/60000 [==============================] - 3s 46us/step - loss: 0.2753 -
acc: 0.9242 - val_loss: 0.1488 - val_acc: 0.9565
Epoch 5/30
60000/60000 [==============================] - 3s 46us/step - loss: 0.2550 -
acc: 0.9282 - val_loss: 0.1440 - val_acc: 0.9589
Epoch 6/30
60000/60000 [==============================] - 3s 46us/step - loss: 0.2354 -
acc: 0.9335 - val_loss: 0.1246 - val_acc: 0.9639
Epoch 7/30
60000/60000 [==============================] - 3s 48us/step - loss: 0.2180 -
acc: 0.9389 - val_loss: 0.1187 - val_acc: 0.9663
Epoch 8/30
60000/60000 [==============================] - 3s 47us/step - loss: 0.2090 -
acc: 0.9422 - val_loss: 0.1149 - val_acc: 0.9649
Epoch 9/30
60000/60000 [==============================] - 3s 50us/step - loss: 0.1994 -
acc: 0.9439 - val_loss: 0.1143 - val_acc: 0.9676
Epoch 10/30
60000/60000 [==============================] - 3s 55us/step - loss: 0.1937 -
acc: 0.9451 - val_loss: 0.1133 - val_acc: 0.9671
Epoch 11/30
60000/60000 [==============================] - 3s 56us/step - loss: 0.1852 -
acc: 0.9472 - val_loss: 0.1131 - val_acc: 0.9686
Epoch 12/30
60000/60000 [==============================] - 3s 52us/step - loss: 0.1793 -
acc: 0.9493 - val_loss: 0.1078 - val_acc: 0.9693
Epoch 13/30
60000/60000 [==============================] - 3s 47us/step - loss: 0.1733 -
acc: 0.9501 - val_loss: 0.1108 - val_acc: 0.9685
Epoch 14/30
60000/60000 [==============================] - 3s 46us/step - loss: 0.1685 -
acc: 0.9516 - val_loss: 0.1109 - val_acc: 0.9696
Epoch 15/30
60000/60000 [==============================] - 3s 47us/step - loss: 0.1664 -
acc: 0.9533 - val_loss: 0.1009 - val_acc: 0.9715
Epoch 16/30
60000/60000 [==============================] - 3s 46us/step - loss: 0.1624 -
acc: 0.9537 - val_loss: 0.1026 - val_acc: 0.9713
Epoch 17/30
60000/60000 [==============================] - 3s 47us/step - loss: 0.1604 -
acc: 0.9537 - val_loss: 0.1033 - val_acc: 0.9718
Epoch 18/30
60000/60000 [==============================] - 3s 46us/step - loss: 0.1546 -
acc: 0.9548 - val_loss: 0.1061 - val_acc: 0.9707
Epoch 19/30
60000/60000 [==============================] - 3s 46us/step - loss: 0.1533 -
```

```
acc: 0.9568 - val_loss: 0.1015 - val_acc: 0.9723
Epoch 20/30
60000/60000 [==============================] - 3s 46us/step - loss: 0.1489 -
acc: 0.9567 - val_loss: 0.1053 - val_acc: 0.9698
Epoch 21/30
60000/60000 [==============================] - 3s 46us/step - loss: 0.1478 -
acc: 0.9578 - val_loss: 0.1047 - val_acc: 0.9725
Epoch 22/30
60000/60000 [==============================] - 3s 46us/step - loss: 0.1445 -
acc: 0.9583 - val_loss: 0.1007 - val_acc: 0.9725
Epoch 23/30
60000/60000 [==============================] - 3s 46us/step - loss: 0.1431 -
acc: 0.9589 - val_loss: 0.1032 - val_acc: 0.9719
Epoch 24/30
60000/60000 [==============================] - 3s 46us/step - loss: 0.1429 -
acc: 0.9588 - val_loss: 0.1010 - val_acc: 0.9734
Epoch 25/30
60000/60000 [==============================] - 3s 46us/step - loss: 0.1401 -
acc: 0.9596 - val_loss: 0.1026 - val_acc: 0.9720
Epoch 26/30
60000/60000 [==============================] - 3s 47us/step - loss: 0.1385 -
acc: 0.9602 - val_loss: 0.0995 - val_acc: 0.9719
Epoch 27/30
60000/60000 [==============================] - 3s 47us/step - loss: 0.1364 -
acc: 0.9607 - val_loss: 0.1036 - val_acc: 0.9724
Epoch 28/30
60000/60000 [==============================] - 3s 48us/step - loss: 0.1334 -
acc: 0.9611 - val_loss: 0.0997 - val_acc: 0.9720
Epoch 29/30
60000/60000 [==============================] - 3s 46us/step - loss: 0.1344 -
acc: 0.9613 - val_loss: 0.0970 - val_acc: 0.9738
Epoch 30/30
60000/60000 [==============================] - 3s 46us/step - loss: 0.1267 -
acc: 0.9636 - val_loss: 0.0959 - val_acc: 0.9741
```

In [19]:
```
score=model_6.evaluate(x_test,y_test,verbose=0)
print('test error',score[0])
print('test accuracy',score[1])

x=list(range(1,31))
vl6=history6.history['val_loss']
l6=history6.history['loss']
plt.plot(x,vl6,'y',label='validation error')
plt.plot(x,l6,'b',label='training error')
plt.legend()
plt.xlabel('epochs')
plt.ylabel('losses')
plt.grid()
plt.show()
```

```
test error 0.095910078465670082
test accuracy 0.9741
```



case 3-with both Dropout and BatchNormalization

In [20]:

```
model_7=Sequential()
model_7.add(Dense(128,input_dim=784,activation='relu',kernel_initializer='he_normal'))
model_7.add(Dropout(0.5))
model_7.add(Dense(64,activation='relu',kernel_initializer='he_normal'))
model_7.add(BatchNormalization())
model_7.add(Dropout(0.5))
model_7.add(Dense(32,activation='relu',kernel_initializer='he_normal'))
model_7.add(Dense(10,activation='softmax'))
model_7.compile(loss='categorical_crossentropy',optimizer='adam',metrics=['accuracy'])
history7=model_7.fit(x_train,y_train,epochs=30,batch_size=127,verbose=1,validation_data=(x_test,y_test))
```

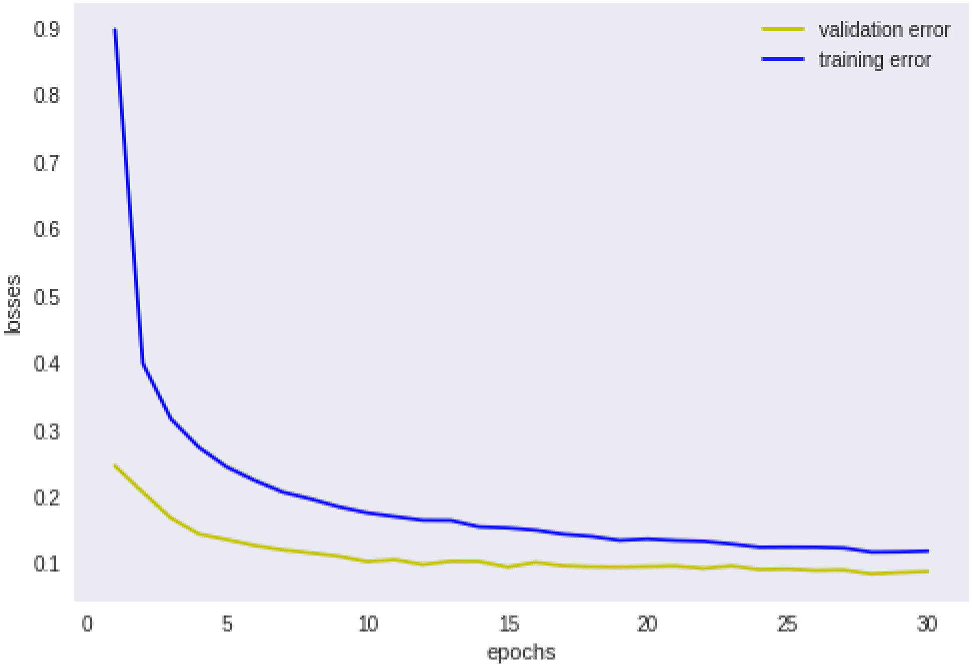
```
Train on 60000 samples, validate on 10000 samples
Epoch 1/30
60000/60000 [==============================] - 4s 67us/step - loss: 0.8978 -
acc: 0.7159 - val_loss: 0.2460 - val_acc: 0.9272
Epoch 2/30
60000/60000 [==============================] - 3s 52us/step - loss: 0.3989 -
acc: 0.8838 - val_loss: 0.2064 - val_acc: 0.9396
Epoch 3/30
60000/60000 [==============================] - 3s 52us/step - loss: 0.3165 -
acc: 0.9104 - val_loss: 0.1678 - val_acc: 0.9512
Epoch 4/30
60000/60000 [==============================] - 3s 53us/step - loss: 0.2739 -
acc: 0.9217 - val_loss: 0.1440 - val_acc: 0.9579
Epoch 5/30
60000/60000 [==============================] - 3s 54us/step - loss: 0.2443 -
acc: 0.9308 - val_loss: 0.1357 - val_acc: 0.9625
Epoch 6/30
60000/60000 [==============================] - 3s 53us/step - loss: 0.2240 -
acc: 0.9355 - val_loss: 0.1267 - val_acc: 0.9629
Epoch 7/30
60000/60000 [==============================] - 3s 53us/step - loss: 0.2067 -
acc: 0.9419 - val_loss: 0.1202 - val_acc: 0.9656
Epoch 8/30
60000/60000 [==============================] - 3s 54us/step - loss: 0.1964 -
acc: 0.9449 - val_loss: 0.1157 - val_acc: 0.9636
Epoch 9/30
60000/60000 [==============================] - 3s 52us/step - loss: 0.1846 -
acc: 0.9470 - val_loss: 0.1106 - val_acc: 0.9686
Epoch 10/30
60000/60000 [==============================] - 3s 51us/step - loss: 0.1757 -
acc: 0.9502 - val_loss: 0.1029 - val_acc: 0.9705
Epoch 11/30
60000/60000 [==============================] - 3s 52us/step - loss: 0.1700 -
acc: 0.9514 - val_loss: 0.1056 - val_acc: 0.9699
Epoch 12/30
60000/60000 [==============================] - 3s 52us/step - loss: 0.1646 -
acc: 0.9535 - val_loss: 0.0983 - val_acc: 0.9714
Epoch 13/30
60000/60000 [==============================] - 3s 52us/step - loss: 0.1643 -
acc: 0.9526 - val_loss: 0.1033 - val_acc: 0.9708
Epoch 14/30
60000/60000 [==============================] - 3s 52us/step - loss: 0.1548 -
acc: 0.9561 - val_loss: 0.1029 - val_acc: 0.9700
Epoch 15/30
60000/60000 [==============================] - 3s 52us/step - loss: 0.1532 -
acc: 0.9573 - val_loss: 0.0946 - val_acc: 0.9720
Epoch 16/30
60000/60000 [==============================] - 3s 52us/step - loss: 0.1497 -
acc: 0.9574 - val_loss: 0.1015 - val_acc: 0.9703
Epoch 17/30
60000/60000 [==============================] - 3s 53us/step - loss: 0.1438 -
acc: 0.9586 - val_loss: 0.0964 - val_acc: 0.9724
Epoch 18/30
60000/60000 [==============================] - 3s 52us/step - loss: 0.1405 -
acc: 0.9598 - val_loss: 0.0950 - val_acc: 0.9735
Epoch 19/30
60000/60000 [==============================] - 3s 53us/step - loss: 0.1345 -
```

```
acc: 0.9616 - val_loss: 0.0945 - val_acc: 0.9736
Epoch 20/30
60000/60000 [==============================] - 3s 53us/step - loss: 0.1366 -
acc: 0.9603 - val_loss: 0.0953 - val_acc: 0.9747
Epoch 21/30
60000/60000 [==============================] - 3s 52us/step - loss: 0.1343 -
acc: 0.9607 - val_loss: 0.0961 - val_acc: 0.9722
Epoch 22/30
60000/60000 [==============================] - 3s 52us/step - loss: 0.1331 -
acc: 0.9618 - val_loss: 0.0926 - val_acc: 0.9744
Epoch 23/30
60000/60000 [==============================] - 3s 52us/step - loss: 0.1291 -
acc: 0.9627 - val_loss: 0.0963 - val_acc: 0.9722
Epoch 24/30
60000/60000 [==============================] - 3s 52us/step - loss: 0.1242 -
acc: 0.9637 - val_loss: 0.0909 - val_acc: 0.9729
Epoch 25/30
60000/60000 [==============================] - 3s 51us/step - loss: 0.1244 -
acc: 0.9640 - val_loss: 0.0916 - val_acc: 0.9732
Epoch 26/30
60000/60000 [==============================] - 3s 51us/step - loss: 0.1242 -
acc: 0.9638 - val_loss: 0.0895 - val_acc: 0.9738
Epoch 27/30
60000/60000 [==============================] - 3s 51us/step - loss: 0.1231 -
acc: 0.9649 - val_loss: 0.0902 - val_acc: 0.9750
Epoch 28/30
60000/60000 [==============================] - 3s 51us/step - loss: 0.1169 -
acc: 0.9661 - val_loss: 0.0843 - val_acc: 0.9767
Epoch 29/30
60000/60000 [==============================] - 3s 52us/step - loss: 0.1173 -
acc: 0.9667 - val_loss: 0.0864 - val_acc: 0.9758
Epoch 30/30
60000/60000 [==============================] - 3s 52us/step - loss: 0.1184 -
acc: 0.9657 - val_loss: 0.0878 - val_acc: 0.9751
```

In [21]:

```
score=model_7.evaluate(x_test,y_test,verbose=0)
print("test error",score[0])
print("test accuracy",score[1])

x = list(range(1,31))
vl7=history7.history['val_loss']
l7=history7.history['loss']
plt.plot(x,vl7,'y',label='validation error')
plt.plot(x,l7,'b',label='training error')
plt.legend()
plt.xlabel('epochs')
plt.ylabel('losses')
plt.grid()
plt.show()
```

```
test error 0.08784910060963594
test accuracy 0.9751
```



case 4-without Dropout and BatchNormalization

In [22]:

```
model_8=Sequential()
model_8.add(Dense(128,input_dim=784,activation='relu',kernel_initializer='he_normal'))
model_8.add(Dense(64,activation='relu',kernel_initializer='he_normal'))
model_8.add(Dense(32,activation='relu',kernel_initializer='he_normal'))
model_8.add(Dense(10,activation='softmax'))
model_8.compile(loss='categorical_crossentropy',optimizer='adam',metrics=['accuracy'])
history8=model_8.fit(x_train,y_train,epochs=50,batch_size=128,verbose=1,validation_data=(x_test,y_test))
```

```
Train on 60000 samples, validate on 10000 samples
Epoch 1/50
60000/60000 [==============================] - 3s 54us/step - loss: 0.3483 -
acc: 0.8962 - val_loss: 0.1468 - val_acc: 0.9577
Epoch 2/50
60000/60000 [==============================] - 2s 42us/step - loss: 0.1301 -
acc: 0.9610 - val_loss: 0.1308 - val_acc: 0.9605
Epoch 3/50
60000/60000 [==============================] - 3s 42us/step - loss: 0.0914 -
acc: 0.9724 - val_loss: 0.1100 - val_acc: 0.9692
Epoch 4/50
60000/60000 [==============================] - 2s 41us/step - loss: 0.0704 -
acc: 0.9788 - val_loss: 0.0895 - val_acc: 0.9716
Epoch 5/50
60000/60000 [==============================] - 2s 41us/step - loss: 0.0552 -
acc: 0.9829 - val_loss: 0.0933 - val_acc: 0.9703
Epoch 6/50
60000/60000 [==============================] - 3s 42us/step - loss: 0.0441 -
acc: 0.9863 - val_loss: 0.0885 - val_acc: 0.9724
Epoch 7/50
60000/60000 [==============================] - 3s 43us/step - loss: 0.0358 -
acc: 0.9889 - val_loss: 0.0843 - val_acc: 0.9756
Epoch 8/50
60000/60000 [==============================] - 3s 42us/step - loss: 0.0311 -
acc: 0.9902 - val_loss: 0.0934 - val_acc: 0.9748
Epoch 9/50
60000/60000 [==============================] - 3s 42us/step - loss: 0.0257 -
acc: 0.9918 - val_loss: 0.0843 - val_acc: 0.9769
Epoch 10/50
60000/60000 [==============================] - 3s 42us/step - loss: 0.0224 -
acc: 0.9931 - val_loss: 0.0829 - val_acc: 0.9766
Epoch 11/50
60000/60000 [==============================] - 2s 42us/step - loss: 0.0199 -
acc: 0.9933 - val_loss: 0.0963 - val_acc: 0.9741
Epoch 12/50
60000/60000 [==============================] - 3s 42us/step - loss: 0.0166 -
acc: 0.9946 - val_loss: 0.0954 - val_acc: 0.9758
Epoch 13/50
60000/60000 [==============================] - 2s 42us/step - loss: 0.0163 -
acc: 0.9946 - val_loss: 0.1005 - val_acc: 0.9768
Epoch 14/50
60000/60000 [==============================] - 2s 41us/step - loss: 0.0137 -
acc: 0.9956 - val_loss: 0.0941 - val_acc: 0.9771
Epoch 15/50
60000/60000 [==============================] - 2s 41us/step - loss: 0.0120 -
acc: 0.9959 - val_loss: 0.1115 - val_acc: 0.9758
Epoch 16/50
60000/60000 [==============================] - 2s 42us/step - loss: 0.0121 -
acc: 0.9959 - val_loss: 0.1346 - val_acc: 0.9717
Epoch 17/50
60000/60000 [==============================] - 2s 41us/step - loss: 0.0152 -
acc: 0.9950 - val_loss: 0.1149 - val_acc: 0.9749
Epoch 18/50
60000/60000 [==============================] - 2s 41us/step - loss: 0.0128 -
acc: 0.9956 - val_loss: 0.1119 - val_acc: 0.9772
Epoch 19/50
60000/60000 [==============================] - 2s 41us/step - loss: 0.0052 -
```

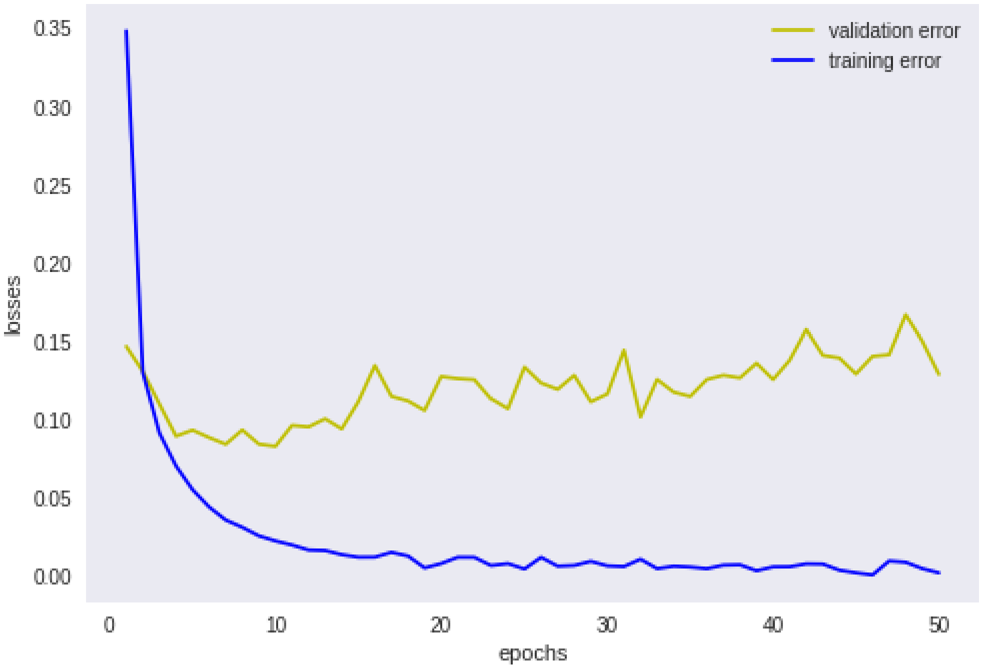
```
acc: 0.9983 - val_loss: 0.1059 - val_acc: 0.9782
Epoch 20/50
60000/60000 [==============================] - 2s 41us/step - loss: 0.0080 -
acc: 0.9973 - val_loss: 0.1276 - val_acc: 0.9744
Epoch 21/50
60000/60000 [==============================] - 2s 41us/step - loss: 0.0120 -
acc: 0.9959 - val_loss: 0.1262 - val_acc: 0.9747
Epoch 22/50
60000/60000 [==============================] - 2s 41us/step - loss: 0.0120 -
acc: 0.9959 - val_loss: 0.1256 - val_acc: 0.9733
Epoch 23/50
60000/60000 [==============================] - 2s 41us/step - loss: 0.0067 -
acc: 0.9975 - val_loss: 0.1133 - val_acc: 0.9779
Epoch 24/50
60000/60000 [==============================] - 3s 42us/step - loss: 0.0080 -
acc: 0.9973 - val_loss: 0.1070 - val_acc: 0.9795
Epoch 25/50
60000/60000 [==============================] - 2s 41us/step - loss: 0.0045 -
acc: 0.9986 - val_loss: 0.1336 - val_acc: 0.9754
Epoch 26/50
60000/60000 [==============================] - 2s 41us/step - loss: 0.0119 -
acc: 0.9960 - val_loss: 0.1234 - val_acc: 0.9778
Epoch 27/50
60000/60000 [==============================] - 2s 42us/step - loss: 0.0063 -
acc: 0.9979 - val_loss: 0.1194 - val_acc: 0.9791
Epoch 28/50
60000/60000 [==============================] - 2s 42us/step - loss: 0.0068 -
acc: 0.9979 - val_loss: 0.1283 - val_acc: 0.9777
Epoch 29/50
60000/60000 [==============================] - 2s 41us/step - loss: 0.0094 -
acc: 0.9972 - val_loss: 0.1114 - val_acc: 0.9789
Epoch 30/50
60000/60000 [==============================] - 3s 43us/step - loss: 0.0065 -
acc: 0.9980 - val_loss: 0.1164 - val_acc: 0.9785
Epoch 31/50
60000/60000 [==============================] - 3s 43us/step - loss: 0.0061 -
acc: 0.9980 - val_loss: 0.1444 - val_acc: 0.9763
Epoch 32/50
60000/60000 [==============================] - 3s 42us/step - loss: 0.0107 -
acc: 0.9965 - val_loss: 0.1017 - val_acc: 0.9812
Epoch 33/50
60000/60000 [==============================] - 3s 42us/step - loss: 0.0047 -
acc: 0.9985 - val_loss: 0.1257 - val_acc: 0.9792
Epoch 34/50
60000/60000 [==============================] - 3s 42us/step - loss: 0.0063 -
acc: 0.9978 - val_loss: 0.1176 - val_acc: 0.9799
Epoch 35/50
60000/60000 [==============================] - 3s 42us/step - loss: 0.0058 -
acc: 0.9982 - val_loss: 0.1149 - val_acc: 0.9798
Epoch 36/50
60000/60000 [==============================] - 3s 42us/step - loss: 0.0047 -
acc: 0.9984 - val_loss: 0.1257 - val_acc: 0.9787
Epoch 37/50
60000/60000 [==============================] - 2s 42us/step - loss: 0.0070 -
acc: 0.9978 - val_loss: 0.1284 - val_acc: 0.9776
Epoch 38/50
60000/60000 [==============================] - 3s 42us/step - loss: 0.0072 -
```

```
acc: 0.9978 - val_loss: 0.1267 - val_acc: 0.9801
Epoch 39/50
60000/60000 [==============================] - 2s 42us/step - loss: 0.0034 -
acc: 0.9989 - val_loss: 0.1361 - val_acc: 0.9788
Epoch 40/50
60000/60000 [==============================] - 3s 42us/step - loss: 0.0059 -
acc: 0.9981 - val_loss: 0.1256 - val_acc: 0.9798
Epoch 41/50
60000/60000 [==============================] - 3s 42us/step - loss: 0.0059 -
acc: 0.9981 - val_loss: 0.1380 - val_acc: 0.9794
Epoch 42/50
60000/60000 [==============================] - 3s 42us/step - loss: 0.0078 -
acc: 0.9974 - val_loss: 0.1575 - val_acc: 0.9753
Epoch 43/50
60000/60000 [==============================] - 3s 42us/step - loss: 0.0077 -
acc: 0.9976 - val_loss: 0.1411 - val_acc: 0.9773
Epoch 44/50
60000/60000 [==============================] - 3s 42us/step - loss: 0.0037 -
acc: 0.9987 - val_loss: 0.1394 - val_acc: 0.9760
Epoch 45/50
60000/60000 [==============================] - 3s 42us/step - loss: 0.0021 -
acc: 0.9994 - val_loss: 0.1292 - val_acc: 0.9807
Epoch 46/50
60000/60000 [==============================] - 3s 42us/step - loss: 6.9593e-0
4 - acc: 0.9999 - val_loss: 0.1405 - val_acc: 0.9793
Epoch 47/50
60000/60000 [==============================] - 2s 41us/step - loss: 0.0096 -
acc: 0.9967 - val_loss: 0.1415 - val_acc: 0.9771
Epoch 48/50
60000/60000 [==============================] - 2s 41us/step - loss: 0.0088 -
acc: 0.9974 - val_loss: 0.1668 - val_acc: 0.9743
Epoch 49/50
60000/60000 [==============================] - 2s 41us/step - loss: 0.0048 -
acc: 0.9983 - val_loss: 0.1501 - val_acc: 0.9762
Epoch 50/50
60000/60000 [==============================] - 2s 41us/step - loss: 0.0020 -
acc: 0.9995 - val_loss: 0.1290 - val_acc: 0.9816
```

In [23]:

```
score=model_8.evaluate(x_test,y_test,verbose=0)
print('test error',score[0])
print('test accuracy',score[1])


x=list(range(1,51))
vl8=history8.history['val_loss']
l8=history8.history['loss']
plt.plot(x1,vl8,'y',label='validation error')
plt.plot(x1,l8,'b',label='training error')
plt.legend()
plt.xlabel('epochs')
plt.ylabel('losses')
plt.grid()
plt.show()
```

```
test error 0.12899938047406814
test accuracy 0.9816
```



3. Five layers MNPs with activation 'relu' and optimizer 'adam'

case 1-with only BatchNormalization

In [24]:

```
model_9=Sequential()
model_9.add(Dense(512,input_dim=784,activation='relu',kernel_initializer='he_normal'))
model_9.add(Dense(256,activation='relu',kernel_initializer='he_normal'))
model_9.add(Dense(128,activation='relu',kernel_initializer='he_normal'))
model_9.add(Dense(64,activation='relu',kernel_initializer='he_normal'))
model_9.add(BatchNormalization())
model_9.add(Dense(32,activation='relu',kernel_initializer='he_normal'))
model_9.add(Dense(10,activation='softmax'))
model_9.compile(loss='categorical_crossentropy',optimizer='adam',metrics=['accuracy'])
history9=model_9.fit(x_train,y_train,epochs=30,batch_size=128,verbose=1,validation_data=(x_test,y_test))
```

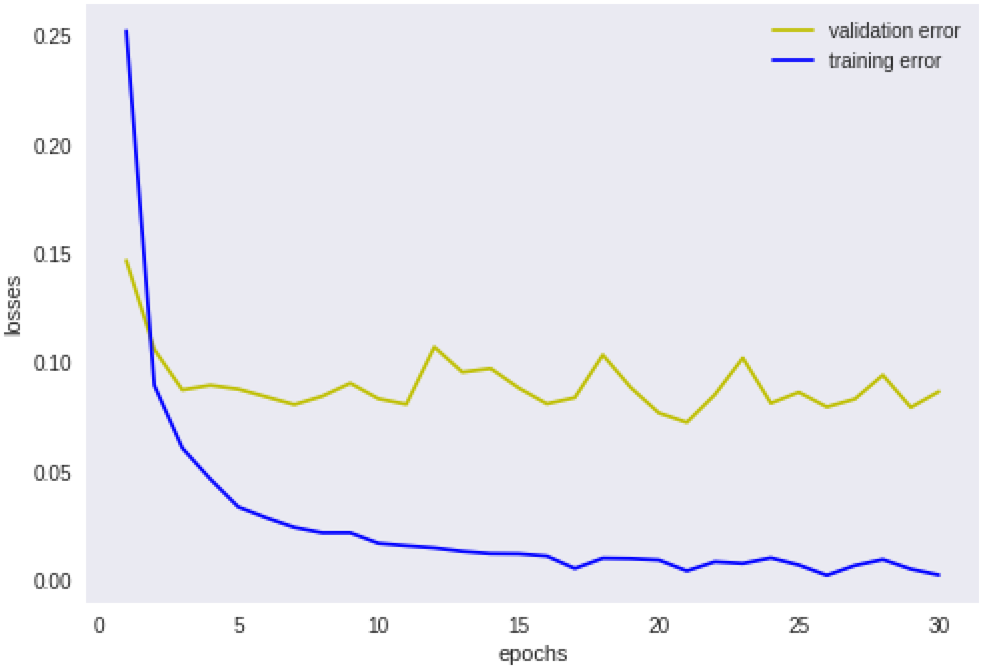
```
Train on 60000 samples, validate on 10000 samples
Epoch 1/30
60000/60000 [==============================] - 9s 158us/step - loss: 0.2523 -
acc: 0.9286 - val_loss: 0.1468 - val_acc: 0.9559
Epoch 2/30
60000/60000 [==============================] - 8s 139us/step - loss: 0.0898 -
acc: 0.9725 - val_loss: 0.1062 - val_acc: 0.9673
Epoch 3/30
60000/60000 [==============================] - 9s 143us/step - loss: 0.0610 -
acc: 0.9810 - val_loss: 0.0877 - val_acc: 0.9715
Epoch 4/30
60000/60000 [==============================] - 9s 150us/step - loss: 0.0467 -
acc: 0.9852 - val_loss: 0.0898 - val_acc: 0.9724
Epoch 5/30
60000/60000 [==============================] - 8s 136us/step - loss: 0.0339 -
acc: 0.9887 - val_loss: 0.0880 - val_acc: 0.9742
Epoch 6/30
60000/60000 [==============================] - 8s 136us/step - loss: 0.0289 -
acc: 0.9904 - val_loss: 0.0845 - val_acc: 0.9768
Epoch 7/30
60000/60000 [==============================] - 8s 134us/step - loss: 0.0245 -
acc: 0.9922 - val_loss: 0.0809 - val_acc: 0.9777
Epoch 8/30
60000/60000 [==============================] - 8s 136us/step - loss: 0.0220 -
acc: 0.9929 - val_loss: 0.0848 - val_acc: 0.9773
Epoch 9/30
60000/60000 [==============================] - 8s 139us/step - loss: 0.0221 -
acc: 0.9927 - val_loss: 0.0907 - val_acc: 0.9761
Epoch 10/30
60000/60000 [==============================] - 8s 137us/step - loss: 0.0172 -
acc: 0.9943 - val_loss: 0.0836 - val_acc: 0.9776
Epoch 11/30
60000/60000 [==============================] - 8s 137us/step - loss: 0.0161 -
acc: 0.9948 - val_loss: 0.0811 - val_acc: 0.9780
Epoch 12/30
60000/60000 [==============================] - 8s 138us/step - loss: 0.0151 -
acc: 0.9948 - val_loss: 0.1074 - val_acc: 0.9752
Epoch 13/30
60000/60000 [==============================] - 8s 137us/step - loss: 0.0135 -
acc: 0.9957 - val_loss: 0.0958 - val_acc: 0.9783
Epoch 14/30
60000/60000 [==============================] - 8s 138us/step - loss: 0.0125 -
acc: 0.9959 - val_loss: 0.0975 - val_acc: 0.9778
Epoch 15/30
60000/60000 [==============================] - 8s 137us/step - loss: 0.0124 -
acc: 0.9957 - val_loss: 0.0885 - val_acc: 0.9794
Epoch 16/30
60000/60000 [==============================] - 8s 142us/step - loss: 0.0113 -
acc: 0.9963 - val_loss: 0.0813 - val_acc: 0.9806
Epoch 17/30
60000/60000 [==============================] - 8s 139us/step - loss: 0.0056 -
acc: 0.9980 - val_loss: 0.0841 - val_acc: 0.9798
Epoch 18/30
60000/60000 [==============================] - 8s 138us/step - loss: 0.0103 -
acc: 0.9965 - val_loss: 0.1037 - val_acc: 0.9783
Epoch 19/30
60000/60000 [==============================] - 8s 139us/step - loss: 0.0102 -
```

```
acc: 0.9965 - val_loss: 0.0887 - val_acc: 0.9793
Epoch 20/30
60000/60000 [==============================] - 8s 141us/step - loss: 0.0095 -
acc: 0.9971 - val_loss: 0.0771 - val_acc: 0.9811
Epoch 21/30
60000/60000 [==============================] - 8s 139us/step - loss: 0.0045 -
acc: 0.9988 - val_loss: 0.0728 - val_acc: 0.9836
Epoch 22/30
60000/60000 [==============================] - 8s 137us/step - loss: 0.0087 -
acc: 0.9971 - val_loss: 0.0854 - val_acc: 0.9793
Epoch 23/30
60000/60000 [==============================] - 8s 141us/step - loss: 0.0079 -
acc: 0.9975 - val_loss: 0.1023 - val_acc: 0.9780
Epoch 24/30
60000/60000 [==============================] - 8s 139us/step - loss: 0.0105 -
acc: 0.9966 - val_loss: 0.0816 - val_acc: 0.9812
Epoch 25/30
60000/60000 [==============================] - 8s 134us/step - loss: 0.0072 -
acc: 0.9978 - val_loss: 0.0866 - val_acc: 0.9797
Epoch 26/30
60000/60000 [==============================] - 8s 135us/step - loss: 0.0025 -
acc: 0.9992 - val_loss: 0.0798 - val_acc: 0.9803
Epoch 27/30
60000/60000 [==============================] - 8s 138us/step - loss: 0.0070 -
acc: 0.9979 - val_loss: 0.0835 - val_acc: 0.9820
Epoch 28/30
60000/60000 [==============================] - 8s 135us/step - loss: 0.0098 -
acc: 0.9969 - val_loss: 0.0944 - val_acc: 0.9807
Epoch 29/30
60000/60000 [==============================] - 8s 137us/step - loss: 0.0054 -
acc: 0.9982 - val_loss: 0.0796 - val_acc: 0.9829
Epoch 30/30
60000/60000 [==============================] - 8s 140us/step - loss: 0.0026 -
acc: 0.9992 - val_loss: 0.0869 - val_acc: 0.9833
```

In [32]:

```
score=model_9.evaluate(x_test,y_test,verbose=1)
print('test error',score[0])
print('test accuracy',score[1])

x=list(range(1,31))
vl9=history9.history['val_loss']
l9=history9.history['loss']
plt.plot(x,vl9,'y',label='validation error')
plt.plot(x,l9,'b',label='training error')
plt.legend()
plt.xlabel('epochs')
plt.ylabel('losses')
plt.grid()
plt.show()
```

```
10000/10000 [==============================] - 1s 84us/step
test error 0.08685781101842004
test accuracy 0.9833
```



case 2-with Dropout only

In [26]:

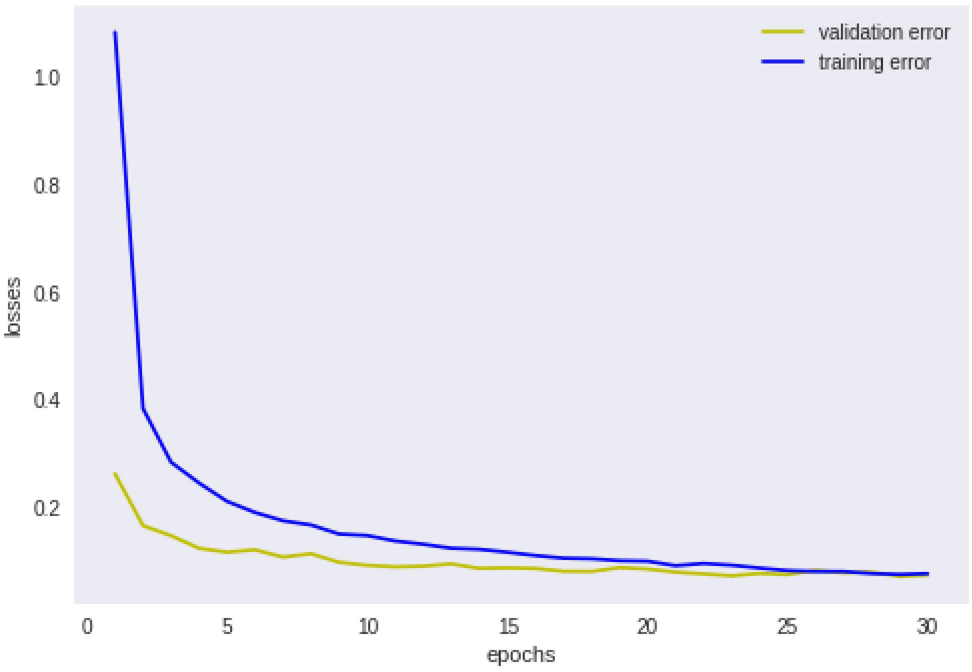
```
model_10=Sequential()
model_10.add(Dense(512,input_dim=784,activation='relu',kernel_initializer='he_
normal'))
model_10.add(Dropout(0.5))
model_10.add(Dense(256,activation='relu',kernel_initializer='he_normal'))
model_10.add(Dropout(0.5))
model_10.add(Dense(128,activation='relu',kernel_initializer='he_normal'))
model_10.add(Dropout(0.5))
model_10.add(Dense(64,activation='relu',kernel_initializer='he_normal'))
model_10.add(Dropout(0.5))
model_10.add(Dense(32,activation='relu',kernel_initializer='he_normal'))
model_10.add(Dense(10,activation='softmax'))
model_10.compile(loss='categorical_crossentropy',optimizer='adam',metrics=['ac
curacy'])
history10=model_10.fit(x_train,y_train,epochs=30,batch_size=108,verbose=1,vali
dation_data=(x_test,y_test))
```

```
Train on 60000 samples, validate on 10000 samples
Epoch 1/30
60000/60000 [==============================] - 11s 190us/step - loss: 1.0826
- acc: 0.6271 - val_loss: 0.2629 - val_acc: 0.9330
Epoch 2/30
60000/60000 [==============================] - 10s 166us/step - loss: 0.3844
- acc: 0.8969 - val_loss: 0.1667 - val_acc: 0.9544
Epoch 3/30
60000/60000 [==============================] - 10s 166us/step - loss: 0.2847
- acc: 0.9281 - val_loss: 0.1480 - val_acc: 0.9615
Epoch 4/30
60000/60000 [==============================] - 10s 163us/step - loss: 0.2460
- acc: 0.9408 - val_loss: 0.1246 - val_acc: 0.9679
Epoch 5/30
60000/60000 [==============================] - 10s 165us/step - loss: 0.2118
- acc: 0.9477 - val_loss: 0.1174 - val_acc: 0.9681
Epoch 6/30
60000/60000 [==============================] - 10s 169us/step - loss: 0.1911
- acc: 0.9535 - val_loss: 0.1217 - val_acc: 0.9701
Epoch 7/30
60000/60000 [==============================] - 10s 170us/step - loss: 0.1757
- acc: 0.9573 - val_loss: 0.1082 - val_acc: 0.9742
Epoch 8/30
60000/60000 [==============================] - 10s 172us/step - loss: 0.1680
- acc: 0.9588 - val_loss: 0.1144 - val_acc: 0.9714
Epoch 9/30
60000/60000 [==============================] - 11s 180us/step - loss: 0.1511
- acc: 0.9629 - val_loss: 0.0985 - val_acc: 0.9760
Epoch 10/30
60000/60000 [==============================] - 10s 169us/step - loss: 0.1483
- acc: 0.9637 - val_loss: 0.0930 - val_acc: 0.9757
Epoch 11/30
60000/60000 [==============================] - 10s 164us/step - loss: 0.1382
- acc: 0.9659 - val_loss: 0.0901 - val_acc: 0.9780
Epoch 12/30
60000/60000 [==============================] - 10s 170us/step - loss: 0.1322
- acc: 0.9678 - val_loss: 0.0913 - val_acc: 0.9781
Epoch 13/30
60000/60000 [==============================] - 10s 170us/step - loss: 0.1247
- acc: 0.9685 - val_loss: 0.0957 - val_acc: 0.9777
Epoch 14/30
60000/60000 [==============================] - 10s 168us/step - loss: 0.1228
- acc: 0.9698 - val_loss: 0.0873 - val_acc: 0.9787
Epoch 15/30
60000/60000 [==============================] - 10s 166us/step - loss: 0.1172
- acc: 0.9711 - val_loss: 0.0881 - val_acc: 0.9792
Epoch 16/30
60000/60000 [==============================] - 10s 166us/step - loss: 0.1111
- acc: 0.9726 - val_loss: 0.0871 - val_acc: 0.9803
Epoch 17/30
60000/60000 [==============================] - 10s 168us/step - loss: 0.1062
- acc: 0.9734 - val_loss: 0.0819 - val_acc: 0.9818
Epoch 18/30
60000/60000 [==============================] - 10s 171us/step - loss: 0.1051
- acc: 0.9739 - val_loss: 0.0815 - val_acc: 0.9820
Epoch 19/30
60000/60000 [==============================] - 10s 171us/step - loss: 0.1016
```

```
- acc: 0.9753 - val_loss: 0.0886 - val_acc: 0.9802
Epoch 20/30
60000/60000 [==============================] - 10s 168us/step - loss: 0.1006
- acc: 0.9754 - val_loss: 0.0860 - val_acc: 0.9798
Epoch 21/30
60000/60000 [==============================] - 10s 168us/step - loss: 0.0923
- acc: 0.9771 - val_loss: 0.0803 - val_acc: 0.9811
Epoch 22/30
60000/60000 [==============================] - 10s 167us/step - loss: 0.0964
- acc: 0.9759 - val_loss: 0.0770 - val_acc: 0.9815
Epoch 23/30
60000/60000 [==============================] - 10s 168us/step - loss: 0.0932
- acc: 0.9765 - val_loss: 0.0734 - val_acc: 0.9830
Epoch 24/30
60000/60000 [==============================] - 10s 172us/step - loss: 0.0879
- acc: 0.9781 - val_loss: 0.0779 - val_acc: 0.9820
Epoch 25/30
60000/60000 [==============================] - 10s 169us/step - loss: 0.0831
- acc: 0.9797 - val_loss: 0.0758 - val_acc: 0.9837
Epoch 26/30
60000/60000 [==============================] - 10s 167us/step - loss: 0.0815
- acc: 0.9795 - val_loss: 0.0836 - val_acc: 0.9797
Epoch 27/30
60000/60000 [==============================] - 10s 165us/step - loss: 0.0814
- acc: 0.9797 - val_loss: 0.0789 - val_acc: 0.9824
Epoch 28/30
60000/60000 [==============================] - 10s 167us/step - loss: 0.0777
- acc: 0.9804 - val_loss: 0.0802 - val_acc: 0.9814
Epoch 29/30
60000/60000 [==============================] - 10s 168us/step - loss: 0.0760
- acc: 0.9814 - val_loss: 0.0723 - val_acc: 0.9828
Epoch 30/30
60000/60000 [==============================] - 10s 173us/step - loss: 0.0775
- acc: 0.9810 - val_loss: 0.0747 - val_acc: 0.9830
```

In [27]:
```
score=model_10.evaluate(x_test,y_test,verbose=0)
print('test error',score[0])
print('test accuracy',score[1])

vl10=history10.history['val_loss']
l10=history10.history['loss']
plt.plot(x,vl10,'y',label='validation error')
plt.plot(x,l10,'b',label='training error')
plt.legend()
plt.xlabel('epochs')
plt.ylabel('losses')
plt.grid()
plt.show()
```

```
test error 0.0746986644620818
test accuracy 0.983
```



case 3-with both Dropout and BatchNormalization

In [28]:

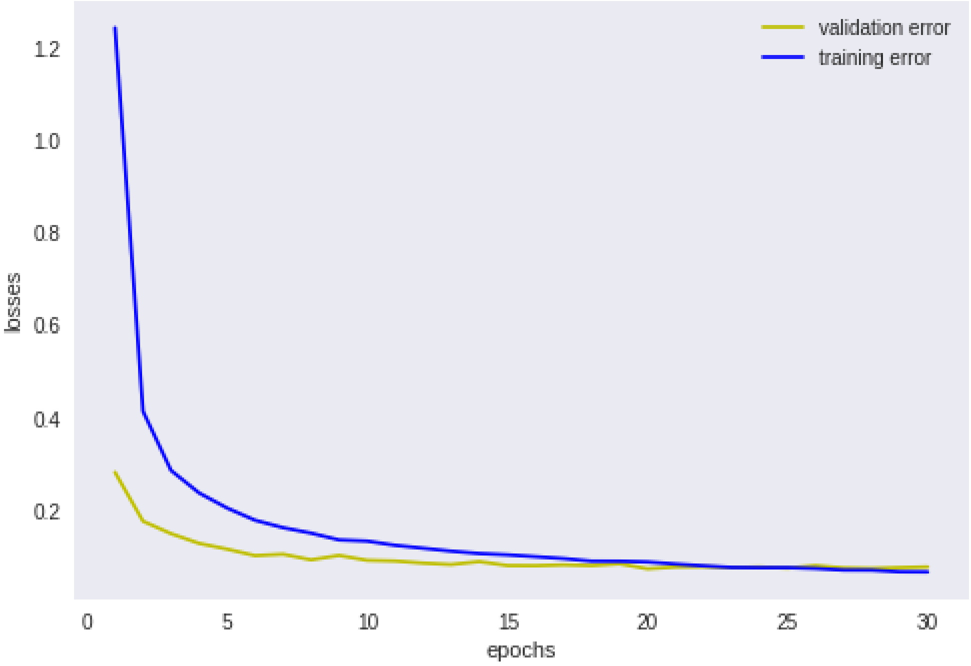
```
model_11=Sequential()
model_11.add(Dense(512,input_dim=784,activation='relu',kernel_initializer='he_normal'))
model_11.add(Dropout(0.5))
model_11.add(Dense(256,activation='relu',kernel_initializer='he_normal'))
model_11.add(Dropout(0.5))
model_11.add(Dense(128,activation='relu',kernel_initializer='he_normal'))
model_11.add(Dropout(0.5))
model_11.add(Dense(64,activation='relu',kernel_initializer='he_normal'))
model_11.add(BatchNormalization())
model_11.add(Dropout(0.5))

model_11.add(Dense(32,activation='relu',kernel_initializer='he_normal'))
model_11.add(Dense(10,activation='softmax'))
model_11.compile(loss='categorical_crossentropy',optimizer='adam',metrics=['accuracy'])
history11=model_11.fit(x_train,y_train,epochs=30,batch_size=128,verbose=1,validation_data=(x_test,y_test))
```

```
Train on 60000 samples, validate on 10000 samples
Epoch 1/30
60000/60000 [==============================] - 11s 186us/step - loss: 1.2445
- acc: 0.5913 - val_loss: 0.2819 - val_acc: 0.9219
Epoch 2/30
60000/60000 [==============================] - 10s 160us/step - loss: 0.4142
- acc: 0.8829 - val_loss: 0.1760 - val_acc: 0.9506
Epoch 3/30
60000/60000 [==============================] - 10s 160us/step - loss: 0.2860
- acc: 0.9228 - val_loss: 0.1490 - val_acc: 0.9606
Epoch 4/30
60000/60000 [==============================] - 10s 160us/step - loss: 0.2373
- acc: 0.9393 - val_loss: 0.1280 - val_acc: 0.9651
Epoch 5/30
60000/60000 [==============================] - 10s 160us/step - loss: 0.2046
- acc: 0.9468 - val_loss: 0.1155 - val_acc: 0.9694
Epoch 6/30
60000/60000 [==============================] - 10s 160us/step - loss: 0.1781
- acc: 0.9542 - val_loss: 0.1014 - val_acc: 0.9735
Epoch 7/30
60000/60000 [==============================] - 9s 158us/step - loss: 0.1620 -
acc: 0.9591 - val_loss: 0.1045 - val_acc: 0.9718
Epoch 8/30
60000/60000 [==============================] - 10s 162us/step - loss: 0.1501
- acc: 0.9626 - val_loss: 0.0922 - val_acc: 0.9754
Epoch 9/30
60000/60000 [==============================] - 10s 161us/step - loss: 0.1354
- acc: 0.9656 - val_loss: 0.1019 - val_acc: 0.9732
Epoch 10/30
60000/60000 [==============================] - 10s 166us/step - loss: 0.1329
- acc: 0.9670 - val_loss: 0.0912 - val_acc: 0.9769
Epoch 11/30
60000/60000 [==============================] - 9s 156us/step - loss: 0.1238 -
acc: 0.9688 - val_loss: 0.0897 - val_acc: 0.9775
Epoch 12/30
60000/60000 [==============================] - 10s 162us/step - loss: 0.1174
- acc: 0.9696 - val_loss: 0.0851 - val_acc: 0.9777
Epoch 13/30
60000/60000 [==============================] - 10s 159us/step - loss: 0.1111
- acc: 0.9716 - val_loss: 0.0820 - val_acc: 0.9784
Epoch 14/30
60000/60000 [==============================] - 9s 158us/step - loss: 0.1059 -
acc: 0.9734 - val_loss: 0.0884 - val_acc: 0.9782
Epoch 15/30
60000/60000 [==============================] - 9s 156us/step - loss: 0.1031 -
acc: 0.9739 - val_loss: 0.0798 - val_acc: 0.9800
Epoch 16/30
60000/60000 [==============================] - 9s 154us/step - loss: 0.0991 -
acc: 0.9749 - val_loss: 0.0795 - val_acc: 0.9794
Epoch 17/30
60000/60000 [==============================] - 9s 154us/step - loss: 0.0950 -
acc: 0.9762 - val_loss: 0.0812 - val_acc: 0.9798
Epoch 18/30
60000/60000 [==============================] - 10s 161us/step - loss: 0.0894
- acc: 0.9775 - val_loss: 0.0798 - val_acc: 0.9800
Epoch 19/30
60000/60000 [==============================] - 10s 160us/step - loss: 0.0892
```

```
- acc: 0.9766 - val_loss: 0.0835 - val_acc: 0.9797
Epoch 20/30
60000/60000 [==============================] - 10s 160us/step - loss: 0.0876
- acc: 0.9780 - val_loss: 0.0728 - val_acc: 0.9823
Epoch 21/30
60000/60000 [==============================] - 10s 162us/step - loss: 0.0831
- acc: 0.9784 - val_loss: 0.0766 - val_acc: 0.9816
Epoch 22/30
60000/60000 [==============================] - 10s 162us/step - loss: 0.0792
- acc: 0.9798 - val_loss: 0.0772 - val_acc: 0.9824
Epoch 23/30
60000/60000 [==============================] - 9s 158us/step - loss: 0.0757 -
acc: 0.9816 - val_loss: 0.0755 - val_acc: 0.9817
Epoch 24/30
60000/60000 [==============================] - 10s 161us/step - loss: 0.0752
- acc: 0.9808 - val_loss: 0.0768 - val_acc: 0.9825
Epoch 25/30
60000/60000 [==============================] - 10s 162us/step - loss: 0.0753
- acc: 0.9812 - val_loss: 0.0743 - val_acc: 0.9828
Epoch 26/30
60000/60000 [==============================] - 10s 162us/step - loss: 0.0734
- acc: 0.9813 - val_loss: 0.0796 - val_acc: 0.9811
Epoch 27/30
60000/60000 [==============================] - 10s 161us/step - loss: 0.0700
- acc: 0.9822 - val_loss: 0.0747 - val_acc: 0.9813
Epoch 28/30
60000/60000 [==============================] - 10s 160us/step - loss: 0.0698
- acc: 0.9829 - val_loss: 0.0736 - val_acc: 0.9829
Epoch 29/30
60000/60000 [==============================] - 9s 158us/step - loss: 0.0663 -
acc: 0.9835 - val_loss: 0.0754 - val_acc: 0.9819
Epoch 30/30
60000/60000 [==============================] - 10s 159us/step - loss: 0.0660
- acc: 0.9831 - val_loss: 0.0769 - val_acc: 0.9814
```

```
In [29]: score=model_11.evaluate(x_test,y_test,verbose=0)
         print('test error',score[0])
         print('test accuracy',score[1])


         x=list(range(1,31))
         vl11=history11.history['val_loss']
         l11=history11.history['loss']
         plt.plot(x,vl11,'y',label='validation error')
         plt.plot(x,l11,'b',label='training error')
         plt.legend()
         plt.xlabel('epochs')
         plt.ylabel('losses')
         plt.grid()
         plt.show()
```

```
test error 0.07691529653224861
test accuracy 0.9814
```



case 4-without Dropout and BatchNormalization

In [30]:

```
model_12=Sequential()
model_12.add(Dense(512,input_dim=784,activation='relu',kernel_initializer='he_normal'))
model_12.add(Dense(256,activation='relu',kernel_initializer='he_normal'))
model_12.add(Dense(128,activation='relu',kernel_initializer='he_normal'))
model_12.add(Dense(64,activation='relu',kernel_initializer='he_normal'))


model_12.add(Dense(32,activation='relu',kernel_initializer='he_normal'))
model_12.add(Dense(10,activation='softmax'))
model_12.compile(loss='categorical_crossentropy',optimizer='adam',metrics=['accuracy'])
history12=model_12.fit(x_train,y_train,epochs=50,batch_size=128,verbose=1,validation_data=(x_test,y_test))
```

```
Train on 60000 samples, validate on 10000 samples
Epoch 1/50
60000/60000 [==============================] - 10s 161us/step - loss: 0.2462
- acc: 0.9261 - val_loss: 0.1293 - val_acc: 0.9584
Epoch 2/50
60000/60000 [==============================] - 8s 139us/step - loss: 0.0880 -
acc: 0.9728 - val_loss: 0.0848 - val_acc: 0.9760
Epoch 3/50
60000/60000 [==============================] - 8s 134us/step - loss: 0.0609 -
acc: 0.9805 - val_loss: 0.0846 - val_acc: 0.9726
Epoch 4/50
60000/60000 [==============================] - 8s 132us/step - loss: 0.0449 -
acc: 0.9857 - val_loss: 0.0952 - val_acc: 0.9730
Epoch 5/50
60000/60000 [==============================] - 8s 138us/step - loss: 0.0332 -
acc: 0.9891 - val_loss: 0.0833 - val_acc: 0.9778
Epoch 6/50
60000/60000 [==============================] - 8s 136us/step - loss: 0.0326 -
acc: 0.9897 - val_loss: 0.0902 - val_acc: 0.9739
Epoch 7/50
60000/60000 [==============================] - 8s 136us/step - loss: 0.0241 -
acc: 0.9923 - val_loss: 0.0857 - val_acc: 0.9777
Epoch 8/50
60000/60000 [==============================] - 8s 137us/step - loss: 0.0236 -
acc: 0.9923 - val_loss: 0.0883 - val_acc: 0.9771
Epoch 9/50
60000/60000 [==============================] - 8s 138us/step - loss: 0.0199 -
acc: 0.9936 - val_loss: 0.0757 - val_acc: 0.9820
Epoch 10/50
60000/60000 [==============================] - 8s 136us/step - loss: 0.0143 -
acc: 0.9955 - val_loss: 0.0914 - val_acc: 0.9804
Epoch 11/50
60000/60000 [==============================] - 8s 140us/step - loss: 0.0208 -
acc: 0.9939 - val_loss: 0.0866 - val_acc: 0.9807
Epoch 12/50
60000/60000 [==============================] - 8s 142us/step - loss: 0.0159 -
acc: 0.9946 - val_loss: 0.0857 - val_acc: 0.9809
Epoch 13/50
60000/60000 [==============================] - 9s 143us/step - loss: 0.0151 -
acc: 0.9952 - val_loss: 0.0848 - val_acc: 0.9799
Epoch 14/50
60000/60000 [==============================] - 9s 146us/step - loss: 0.0125 -
acc: 0.9961 - val_loss: 0.0834 - val_acc: 0.9787
Epoch 15/50
60000/60000 [==============================] - 8s 135us/step - loss: 0.0138 -
acc: 0.9959 - val_loss: 0.0763 - val_acc: 0.9823
Epoch 16/50
60000/60000 [==============================] - 8s 132us/step - loss: 0.0106 -
acc: 0.9966 - val_loss: 0.0950 - val_acc: 0.9787
Epoch 17/50
60000/60000 [==============================] - 8s 133us/step - loss: 0.0147 -
acc: 0.9956 - val_loss: 0.0870 - val_acc: 0.9803
Epoch 18/50
60000/60000 [==============================] - 8s 133us/step - loss: 0.0093 -
acc: 0.9970 - val_loss: 0.0816 - val_acc: 0.9835
Epoch 19/50
60000/60000 [==============================] - 8s 130us/step - loss: 0.0102 -
```

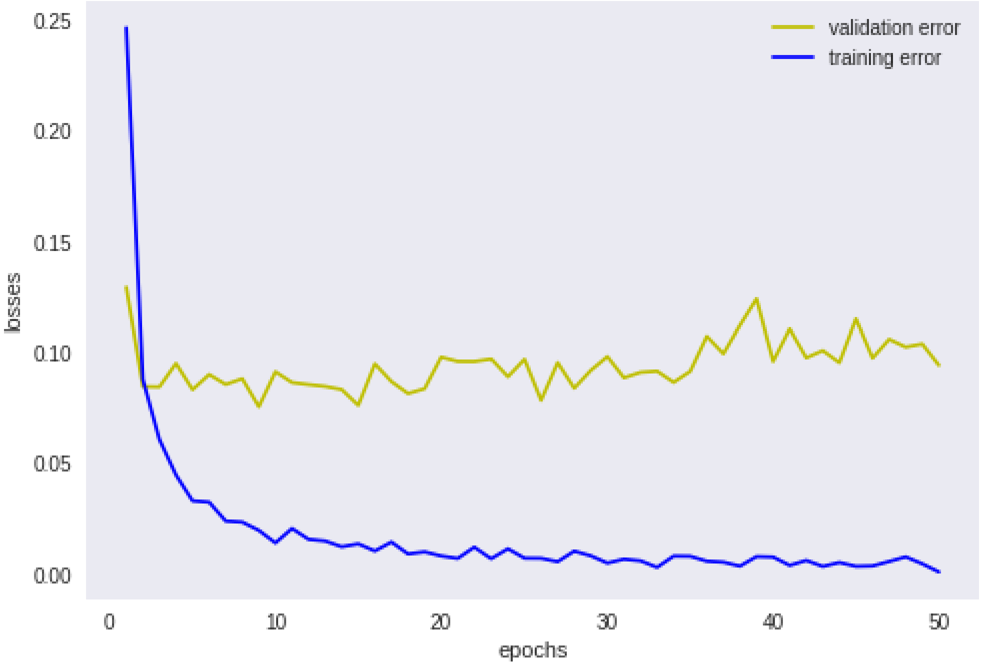
```
acc: 0.9969 - val_loss: 0.0838 - val_acc: 0.9819
Epoch 20/50
60000/60000 [==============================] - 8s 132us/step - loss: 0.0084 -
acc: 0.9973 - val_loss: 0.0980 - val_acc: 0.9797
Epoch 21/50
60000/60000 [==============================] - 8s 134us/step - loss: 0.0073 -
acc: 0.9979 - val_loss: 0.0961 - val_acc: 0.9831
Epoch 22/50
60000/60000 [==============================] - 8s 135us/step - loss: 0.0124 -
acc: 0.9962 - val_loss: 0.0960 - val_acc: 0.9811
Epoch 23/50
60000/60000 [==============================] - 8s 138us/step - loss: 0.0072 -
acc: 0.9979 - val_loss: 0.0971 - val_acc: 0.9814
Epoch 24/50
60000/60000 [==============================] - 8s 140us/step - loss: 0.0116 -
acc: 0.9968 - val_loss: 0.0892 - val_acc: 0.9813
Epoch 25/50
60000/60000 [==============================] - 8s 132us/step - loss: 0.0074 -
acc: 0.9980 - val_loss: 0.0970 - val_acc: 0.9814
Epoch 26/50
60000/60000 [==============================] - 8s 140us/step - loss: 0.0073 -
acc: 0.9981 - val_loss: 0.0786 - val_acc: 0.9843
Epoch 27/50
60000/60000 [==============================] - 8s 135us/step - loss: 0.0058 -
acc: 0.9983 - val_loss: 0.0955 - val_acc: 0.9813
Epoch 28/50
60000/60000 [==============================] - 8s 135us/step - loss: 0.0106 -
acc: 0.9971 - val_loss: 0.0840 - val_acc: 0.9832
Epoch 29/50
60000/60000 [==============================] - 8s 136us/step - loss: 0.0085 -
acc: 0.9975 - val_loss: 0.0920 - val_acc: 0.9816
Epoch 30/50
60000/60000 [==============================] - 8s 140us/step - loss: 0.0052 -
acc: 0.9988 - val_loss: 0.0982 - val_acc: 0.9839
Epoch 31/50
60000/60000 [==============================] - 8s 139us/step - loss: 0.0069 -
acc: 0.9982 - val_loss: 0.0887 - val_acc: 0.9825
Epoch 32/50
60000/60000 [==============================] - 8s 135us/step - loss: 0.0062 -
acc: 0.9981 - val_loss: 0.0911 - val_acc: 0.9839
Epoch 33/50
60000/60000 [==============================] - 8s 136us/step - loss: 0.0033 -
acc: 0.9992 - val_loss: 0.0917 - val_acc: 0.9851
Epoch 34/50
60000/60000 [==============================] - 8s 132us/step - loss: 0.0083 -
acc: 0.9977 - val_loss: 0.0866 - val_acc: 0.9839
Epoch 35/50
60000/60000 [==============================] - 8s 132us/step - loss: 0.0082 -
acc: 0.9977 - val_loss: 0.0916 - val_acc: 0.9835
Epoch 36/50
60000/60000 [==============================] - 8s 133us/step - loss: 0.0060 -
acc: 0.9984 - val_loss: 0.1073 - val_acc: 0.9789
Epoch 37/50
60000/60000 [==============================] - 8s 140us/step - loss: 0.0056 -
acc: 0.9986 - val_loss: 0.0995 - val_acc: 0.9816
Epoch 38/50
60000/60000 [==============================] - 8s 138us/step - loss: 0.0038 -
```

```
acc: 0.9990 - val_loss: 0.1122 - val_acc: 0.9811
Epoch 39/50
60000/60000 [==============================] - 8s 138us/step - loss: 0.0081 -
acc: 0.9978 - val_loss: 0.1241 - val_acc: 0.9784
Epoch 40/50
60000/60000 [==============================] - 8s 135us/step - loss: 0.0079 -
acc: 0.9981 - val_loss: 0.0961 - val_acc: 0.9842
Epoch 41/50
60000/60000 [==============================] - 8s 134us/step - loss: 0.0041 -
acc: 0.9990 - val_loss: 0.1106 - val_acc: 0.9817
Epoch 42/50
60000/60000 [==============================] - 8s 133us/step - loss: 0.0064 -
acc: 0.9984 - val_loss: 0.0977 - val_acc: 0.9838
Epoch 43/50
60000/60000 [==============================] - 8s 134us/step - loss: 0.0037 -
acc: 0.9992 - val_loss: 0.1009 - val_acc: 0.9828
Epoch 44/50
60000/60000 [==============================] - 8s 136us/step - loss: 0.0054 -
acc: 0.9985 - val_loss: 0.0955 - val_acc: 0.9826
Epoch 45/50
60000/60000 [==============================] - 8s 137us/step - loss: 0.0037 -
acc: 0.9991 - val_loss: 0.1151 - val_acc: 0.9801
Epoch 46/50
60000/60000 [==============================] - 8s 134us/step - loss: 0.0039 -
acc: 0.9989 - val_loss: 0.0975 - val_acc: 0.9848
Epoch 47/50
60000/60000 [==============================] - 8s 135us/step - loss: 0.0059 -
acc: 0.9983 - val_loss: 0.1060 - val_acc: 0.9810
Epoch 48/50
60000/60000 [==============================] - 8s 138us/step - loss: 0.0080 -
acc: 0.9980 - val_loss: 0.1025 - val_acc: 0.9829
Epoch 49/50
60000/60000 [==============================] - 9s 142us/step - loss: 0.0048 -
acc: 0.9988 - val_loss: 0.1039 - val_acc: 0.9826
Epoch 50/50
60000/60000 [==============================] - 8s 138us/step - loss: 0.0012 -
acc: 0.9997 - val_loss: 0.0944 - val_acc: 0.9851
```

In [31]:

```
score=model_12.evaluate(x_test,y_test,verbose=0)
print('test error',score[0])
print('test accuracy',score[1])

x1=list(range(1,51))
vl12=history12.history['val_loss']
l12=history12.history['loss']
plt.plot(x1,vl12,'y',label='validation error')
plt.plot(x1,l12,'b',label='training error')
plt.legend()
plt.xlabel('epochs')
plt.ylabel('losses')
plt.grid()
plt.show()
```

```
test error 0.09436505556378368
test accuracy 0.9851
```



2-layer	**Test Error**	**Test Accuracy**
Batch Normal	0.1016	0.9793
DropOut	0.7725	0.9788
with Batch-Normal & DropOut	0.0942	0.974
without Batch-Normal & DropOut	0.1314	0.9802

3-layer	**Test Error**	**Test Accuracy**
Batch Normal	0.1074	0.9781
DropOut	0.0959	0.9741
with Batch-Normal & DropOut	0.0878	0.9751
without Batch-Normal & DropOut	0.1289	0.9816

5-layer	**Test Error**	**Test Accuracy**
Batch Normal	0.0868	0.9833
DropOut	0.0746	0.983
with Batch-Normal & DropOut	0.0769	0.98114
without Batch-Normal & DropOut	0.0943	0.9851

Conclusion-

```
1.As we increase number of layer it may increase accuracy.
2.As we increase number of neurons in layer it may increase accuracy.
3.without Dropout and BatchNormalization model goes overfit.
4.with Dropout or BatchNormalization or both it give appox same accuracy.
5.The model with five layers, activation-relu and optimizer-adam, with dropout and
 Batch normalization gives the best with accuracy 0.9814
```