In [0]:

```
!curl https://course-v3.fast.ai/setup/colab | bash
```

```
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  
Current
                                 Dload  Upload   Total   Spent    Left  
Speed
100   665  100   665    0     0   2427      0 --:--:-- --:--:-- --:--:-
-  2427
Collecting pillow==4.1.1
  Downloading https://files.pythonhosted.org/packages/36/e5/88b3d60924a
3f8476fa74ec086f5fbaba56dd6cee0d82845f883b6b6dd18/Pillow-4.1.1-cp36-cp3
6m-manylinux1_x86_64.whl (5.7MB)
    100% |████████████████████████████████| 5.7MB 7.8MB/s 
Requirement already satisfied, skipping upgrade: olefile in /usr/local/
lib/python3.6/dist-packages (from pillow==4.1.1) (0.46)
Installing collected packages: pillow
  Found existing installation: Pillow 4.0.0
    Uninstalling Pillow-4.0.0:
      Successfully uninstalled Pillow-4.0.0
Successfully installed pillow-4.1.1
Looking in links: https://download.pytorch.org/whl/nightly/cu92/torch_n
ightly.html
Collecting torch_nightly
  Downloading https://download.pytorch.org/whl/nightly/cu92/torch_night
ly-1.0.0.dev20181206-cp36-cp36m-linux_x86_64.whl (576.2MB)
    100% |████████████████████████████████| 576.2MB 31kB/s 
tcmalloc: large alloc 1073750016 bytes == 0x62c60000 @  0x7f31251682a4 
0x591a07 0x5b5d56 0x502e9a 0x506859 0x502209 0x502f3d 0x506859 0x504c28 
0x502540 0x502f3d 0x506859 0x504c28 0x502540 0x502f3d 0x506859 0x504c28 
0x502540 0x502f3d 0x507641 0x502209 0x502f3d 0x506859 0x504c28 0x502540 
0x502f3d 0x507641 0x504c28 0x502540 0x502f3d 0x507641
Installing collected packages: torch-nightly
Successfully installed torch-nightly-1.0.0.dev20181206
Cloning into 'course-v3'...
remote: Enumerating objects: 2583, done.
remote: Total 2583 (delta 0), reused 0 (delta 0), pack-reused 2583
Receiving objects: 100% (2583/2583), 74.87 MiB | 30.03 MiB/s, done.
Resolving deltas: 100% (1406/1406), done.
Collecting fastai
  Downloading https://files.pythonhosted.org/packages/78/7c/67ada54ff19
f296af81c3bba65ac519c0564b176d867c3903b5ad605a233/fastai-1.0.34-py3-non
e-any.whl (139kB)
    100% |████████████████████████████████| 143kB 13.1MB/s 
Collecting fastprogress>=0.1.18 (from fastai)
  Downloading https://files.pythonhosted.org/packages/78/57/24a5e20f4a3
57f7f1c90dd5250071951c832b2480fd4fefd7be48edf4180/fastprogress-0.1.18-p
y3-none-any.whl
Requirement already satisfied, skipping upgrade: numpy>=1.12 in /usr/lo
cal/lib/python3.6/dist-packages (from fastai) (1.14.6)
Collecting numexpr (from fastai)
  Downloading https://files.pythonhosted.org/packages/db/ea/efd9e162836
37eb5b6c0042b6cc3521f1b9a5b47767ac463c88bbd37670c/numexpr-2.6.8-cp36-cp
36m-manylinux1_x86_64.whl (162kB)
    100% |████████████████████████████████| 163kB 18.3MB/s 
Collecting thinc==6.12.0 (from fastai)
  Downloading https://files.pythonhosted.org/packages/1f/18/e320bfc57c2
0df39cc5ffa1915c7b5402a9038f290ddd85b5b72689bd57a/thinc-6.12.0-cp36-cp3
6m-manylinux1_x86_64.whl (1.9MB)
    100% |████████████████████████████████| 1.9MB 10.4MB/s 
```

```
Requirement already satisfied, skipping upgrade: pandas in /usr/local/l
ib/python3.6/dist-packages (from fastai) (0.22.0)
Requirement already satisfied, skipping upgrade: requests in /usr/loca
l/lib/python3.6/dist-packages (from fastai) (2.18.4)
Requirement already satisfied, skipping upgrade: scipy in /usr/local/li
b/python3.6/dist-packages (from fastai) (1.1.0)
Requirement already satisfied, skipping upgrade: typing in /usr/local/l
ib/python3.6/dist-packages (from fastai) (3.6.6)
Collecting torchvision-nightly (from fastai)
  Downloading https://files.pythonhosted.org/packages/ca/bd/d0f9a33c81c
79710eb7ee428b66869b49a8be16c7f1e446c211a7fbfb7be/torchvision_nightly-
0.2.1-py2.py3-none-any.whl (54kB)
    100% |████████████████████████████████| 61kB 24.1MB/s
Collecting spacy==2.0.16 (from fastai)
  Downloading https://files.pythonhosted.org/packages/ed/39/288640f591b
29aac6996c97ddfafc3262ae0be7513e06bc560921b112d7c/spacy-2.0.16-cp36-cp3
6m-manylinux1_x86_64.whl (23.3MB)
    100% |████████████████████████████████| 23.3MB 912kB/s
Requirement already satisfied, skipping upgrade: regex in /usr/local/li
b/python3.6/dist-packages (from fastai) (2018.1.10)
Requirement already satisfied, skipping upgrade: pyyaml in /usr/local/l
ib/python3.6/dist-packages (from fastai) (3.13)
Requirement already satisfied, skipping upgrade: matplotlib in /usr/loc
al/lib/python3.6/dist-packages (from fastai) (2.1.2)
Requirement already satisfied, skipping upgrade: cymem==2.0.2 in /usr/l
ocal/lib/python3.6/dist-packages (from fastai) (2.0.2)
Collecting bottleneck (from fastai)
  Downloading https://files.pythonhosted.org/packages/05/ae/cedf5323f39
8ab4e4ff92d6c431a3e1c6a186f9b41ab3e8258dff786a290/Bottleneck-1.2.1.tar.
gz (105kB)
    100% |████████████████████████████████| 112kB 32.3MB/s
Collecting dataclasses; python_version < "3.7" (from fastai)
  Downloading https://files.pythonhosted.org/packages/26/2f/1095cdc2868
052dd1e64520f7c0d5c8c550ad297e944e641dbf1ffbb9a5d/dataclasses-0.6-py3-n
one-any.whl
Requirement already satisfied, skipping upgrade: Pillow in /usr/local/l
ib/python3.6/dist-packages (from fastai) (4.1.1)
Requirement already satisfied, skipping upgrade: msgpack<1.0.0,>=0.5.6
in /usr/local/lib/python3.6/dist-packages (from thinc==6.12.0->fastai)
(0.5.6)
Requirement already satisfied, skipping upgrade: wrapt<1.11.0,>=1.10.0
in /usr/local/lib/python3.6/dist-packages (from thinc==6.12.0->fastai)
(1.10.11)
Requirement already satisfied, skipping upgrade: dill<0.3.0,>=0.2.7 in
/usr/local/lib/python3.6/dist-packages (from thinc==6.12.0->fastai) (0.
2.8.2)
Requirement already satisfied, skipping upgrade: six<2.0.0,>=1.10.0 in
/usr/local/lib/python3.6/dist-packages (from thinc==6.12.0->fastai) (1.
11.0)
Requirement already satisfied, skipping upgrade: cytoolz<0.10,>=0.9.0 i
n /usr/local/lib/python3.6/dist-packages (from thinc==6.12.0->fastai)
(0.9.0.1)
Requirement already satisfied, skipping upgrade: preshed<3.0.0,>=2.0.1
in /usr/local/lib/python3.6/dist-packages (from thinc==6.12.0->fastai)
(2.0.1)
Requirement already satisfied, skipping upgrade: tqdm<5.0.0,>=4.10.0 in
/usr/local/lib/python3.6/dist-packages (from thinc==6.12.0->fastai) (4.
28.1)
Requirement already satisfied, skipping upgrade: murmurhash<1.1.0,>=0.2
8.0 in /usr/local/lib/python3.6/dist-packages (from thinc==6.12.0->fast
ai) (1.0.1)
Requirement already satisfied, skipping upgrade: plac<1.0.0,>=0.9.6 in
/usr/local/lib/python3.6/dist-packages (from thinc==6.12.0->fastai) (0.
9.6)
Requirement already satisfied, skipping upgrade: msgpack-numpy<0.4.4.0
in /usr/local/lib/python3.6/dist-packages (from thinc==6.12.0->fastai)
(0.4.3.2)
Requirement already satisfied, skipping upgrade: python-dateutil>=2 in
/usr/local/lib/python3.6/dist-packages (from pandas->fastai) (2.5.3)
Requirement already satisfied, skipping upgrade: pytz>=2011k in /usr/lo
cal/lib/python3.6/dist-packages (from pandas->fastai) (2018.7)
Requirement already satisfied, skipping upgrade: idna<2.7,>=2.5 in /us
r/local/lib/python3.6/dist-packages (from requests->fastai) (2.6)
Requirement already satisfied, skipping upgrade: urllib3<1.23,>=1.21.1
in /usr/local/lib/python3.6/dist-packages (from requests->fastai) (1.2
2)
Requirement already satisfied, skipping upgrade: chardet<3.1.0,>=3.0.2
in /usr/local/lib/python3.6/dist-packages (from requests->fastai) (3.0.
4)
Requirement already satisfied, skipping upgrade: certifi>=2017.4.17 in
/usr/local/lib/python3.6/dist-packages (from requests->fastai) (2018.1
1.29)
Requirement already satisfied, skipping upgrade: ujson>=1.35 in /usr/lo
cal/lib/python3.6/dist-packages (from spacy==2.0.16->fastai) (1.35)
Requirement already satisfied, skipping upgrade: cycler>=0.10 in /usr/l
ocal/lib/python3.6/dist-packages (from matplotlib->fastai) (0.10.0)
Requirement already satisfied, skipping upgrade: pyparsing!=2.0.4,!=2.
1.2,!=2.1.6,>=2.0.1 in /usr/local/lib/python3.6/dist-packages (from mat
plotlib->fastai) (2.3.0)
Requirement already satisfied, skipping upgrade: olefile in /usr/local/
lib/python3.6/dist-packages (from Pillow->fastai) (0.46)
Requirement already satisfied, skipping upgrade: toolz>=0.8.0 in /usr/l
ocal/lib/python3.6/dist-packages (from cytoolz<0.10,>=0.9.0->thinc==6.1
2.0->fastai) (0.9.0)
Building wheels for collected packages: bottleneck
  Running setup.py bdist_wheel for bottleneck ... - \ | / - \ | / done
  Stored in directory: /root/.cache/pip/wheels/f2/bf/ec/e0f39aa27001525
ad455139ee57ec7d0776fe074dfd78c97e4
Successfully built bottleneck
spacy 2.0.16 has requirement numpy>=1.15.0, but you'll have numpy 1.14.
6 which is incompatible.
Installing collected packages: fastprogress, numexpr, thinc, torchvisio
n-nightly, spacy, bottleneck, dataclasses, fastai
  Found existing installation: thinc 6.12.1
    Uninstalling thinc-6.12.1:
      Successfully uninstalled thinc-6.12.1
  Found existing installation: spacy 2.0.18
    Uninstalling spacy-2.0.18:
      Successfully uninstalled spacy-2.0.18
Successfully installed bottleneck-1.2.1 dataclasses-0.6 fastai-1.0.34 f
astprogress-0.1.18 numexpr-2.6.8 spacy-2.0.16 thinc-6.12.0 torchvision-
nightly-0.2.1
Already up to date.
```

In [0]:

```
!pip install kaggle
```

```
Collecting kaggle
  Downloading https://files.pythonhosted.org/packages/9e/94/5370052b9cb
c63a927bda08c4f7473a35d3bb27cc071baa1a83b7f783352/kaggle-1.5.1.1.tar.gz
(53kB)
    100% |████████████████████████████████| 61kB 5.3MB/s
Requirement already satisfied: urllib3<1.23.0,>=1.15 in /usr/local/lib/
python3.6/dist-packages (from kaggle) (1.22)
Requirement already satisfied: six>=1.10 in /usr/local/lib/python3.6/di
st-packages (from kaggle) (1.11.0)
Requirement already satisfied: certifi in /usr/local/lib/python3.6/dist
-packages (from kaggle) (2018.11.29)
Requirement already satisfied: python-dateutil in /usr/local/lib/python
3.6/dist-packages (from kaggle) (2.5.3)
Requirement already satisfied: requests in /usr/local/lib/python3.6/dis
t-packages (from kaggle) (2.18.4)
Requirement already satisfied: tqdm in /usr/local/lib/python3.6/dist-pa
ckages (from kaggle) (4.28.1)
Collecting python-slugify (from kaggle)
  Downloading https://files.pythonhosted.org/packages/00/ad/c778a6df614
b6217c30fe80045b365bfa08b5dd3cb02e8b37a6d25126781/python-slugify-1.2.6.
tar.gz
Requirement already satisfied: chardet<3.1.0,>=3.0.2 in /usr/local/lib/
python3.6/dist-packages (from requests->kaggle) (3.0.4)
Requirement already satisfied: idna<2.7,>=2.5 in /usr/local/lib/python
3.6/dist-packages (from requests->kaggle) (2.6)
Collecting Unidecode>=0.04.16 (from python-slugify->kaggle)
  Downloading https://files.pythonhosted.org/packages/31/39/53096f9217b
057cb049fe872b7fc7ce799a1a89b76cf917d9639e7a558b5/Unidecode-1.0.23-py2.
py3-none-any.whl (237kB)
    100% |████████████████████████████████| 245kB 14.5MB/s
Building wheels for collected packages: kaggle, python-slugify
  Running setup.py bdist_wheel for kaggle ... - \ done
  Stored in directory: /root/.cache/pip/wheels/5a/2d/0c/9fc539e558586b9
ed9127916a7f4e620163c24cc97460b1188
  Running setup.py bdist_wheel for python-slugify ... - done
  Stored in directory: /root/.cache/pip/wheels/e3/65/da/2045deea3098ed7
471eca0e2460cfbd3fdfe8c1d6fa6fcac92
Successfully built kaggle python-slugify
Installing collected packages: Unidecode, python-slugify, kaggle
Successfully installed Unidecode-1.0.23 kaggle-1.5.1.1 python-slugify-
1.2.6
```

In [0]:

```
!mkdir -p /root/.kaggle
import json
with open('/root/.kaggle/kaggle.json', 'w') as outfile:
    json.dump({"username":"yokolet","key":"93e6215ff8545c2c157ab645a3134
fe4"}, outfile)

!chmod 600 /root/.kaggle/kaggle.json
```

In [0]:

```
!mkdir inputs
!kaggle competitions download -p inputs -c digit-recognizer
```

```
Downloading train.csv to inputs
 78% 57.0M/73.2M [00:01<00:00, 33.0MB/s]
100% 73.2M/73.2M [00:01<00:00, 59.8MB/s]
Downloading test.csv to inputs
100% 48.8M/48.8M [00:01<00:00, 21.3MB/s]

Downloading sample_submission.csv to inputs
  0% 0.00/235k [00:00<?, ?B/s]
100% 235k/235k [00:00<00:00, 134MB/s]
```

In [0]:

```
!ls -latr inputs
```

```
total 125148
drwxr-xr-x 1 root root     4096 Dec  7 19:13 ..
-rw-r--r-- 1 root root 76775041 Dec  7 19:13 train.csv
-rw-r--r-- 1 root root 51118296 Dec  7 19:13 test.csv
drwxr-xr-x 2 root root     4096 Dec  7 19:13 .
-rw-r--r-- 1 root root   240909 Dec  7 19:13 sample_submission.csv
```

In [0]:

```
%reload_ext autoreload
%autoreload 2
%matplotlib inline
```

In [0]:

```
from fastai import *
from fastai.vision import *
```

In [0]:

```
from sklearn.model_selection import train_test_split
```

In [0]:

```
input_df = pd.read_csv('inputs/train.csv')
print('train size', input_df.shape)
input_df.head()
```

```
train size (42000, 785)
```

Out[0]:

	label	pixel0	pixel1	pixel2	pixel3	pixel4	pixel5	pixel6	pixel7	pixel8	...	pixel774	pixel775
0	1	0	0	0	0	0	0	0	0	0	...	0	0
1	0	0	0	0	0	0	0	0	0	0	...	0	0
2	1	0	0	0	0	0	0	0	0	0	...	0	0
3	4	0	0	0	0	0	0	0	0	0	...	0	0
4	0	0	0	0	0	0	0	0	0	0	...	0	0

5 rows × 785 columns

In [0]:

```
test_df = pd.read_csv('inputs/test.csv')
print('test size', test_df.shape)
test_df.head()
```

```
test size (28000, 784)
```

Out[0]:

	pixel0	pixel1	pixel2	pixel3	pixel4	pixel5	pixel6	pixel7	pixel8	pixel9	...	pixel774	pixel77
0	0	0	0	0	0	0	0	0	0	0	...	0	
1	0	0	0	0	0	0	0	0	0	0	...	0	
2	0	0	0	0	0	0	0	0	0	0	...	0	
3	0	0	0	0	0	0	0	0	0	0	...	0	
4	0	0	0	0	0	0	0	0	0	0	...	0	

5 rows × 784 columns

In [0]:

```
sample_df = pd.read_csv('inputs/sample_submission.csv')
print('sample size', sample_df.shape[0])
sample_df.head()
```

```
sample size 28000
```

Out[0]:

	ImageId	Label
0	1	0
1	2	0
2	3	0
3	4	0
4	5	0

In [0]:

```
# data statistics
data = input_df.drop('label', axis=1)
print('max {}, min {}'.format(data.max().max(), data.min().min()))
```

```
max 255, min 0
```

In [0]:

```
TEST_SIZE=0.1
train, val = train_test_split(input_df, test_size=TEST_SIZE)
train.shape[0], val.shape[0]
```

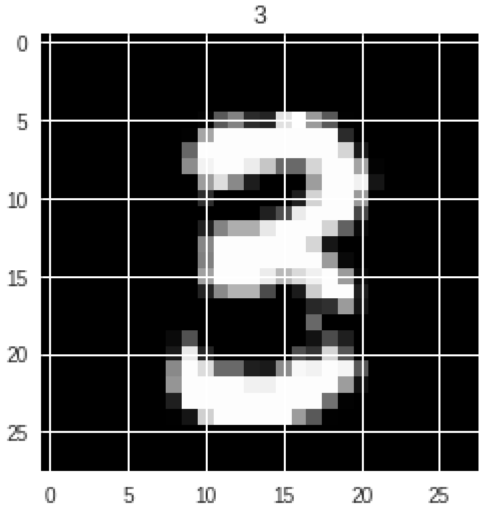
Out[0]:

```
(37800, 4200)
```

In [0]:
```
train_X = train.drop('label', axis=1).values.reshape(-1, 1, 28, 28) / 25
5 # reshape and normalize
train_y = train.label.values
print('X: {} {}, y {} {}'.format(
    type(train_X[0]), train_X[0].shape, type(train_y[0]), train_y[0]))
plt.imshow(train_X[0][0], cmap="gray")
plt.title(train_y[0])
```

```
X: <class 'numpy.ndarray'> (1, 28, 28), y <class 'numpy.int64'> 3
```

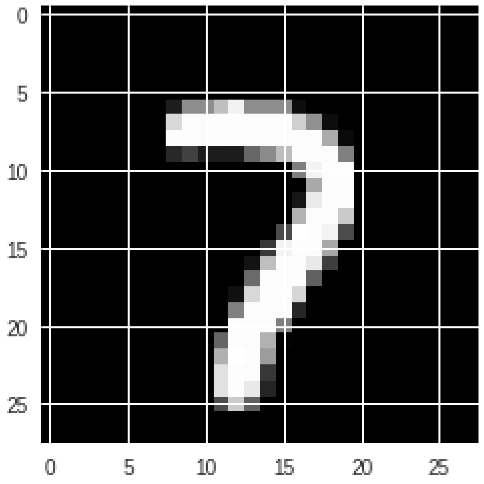
Out[0]:
```
Text(0.5,1,'3')
```



In [0]:
```
val_X = val.drop('label', axis=1).values.reshape(-1, 1, 28, 28) / 255 #
 reshape and normalize
val_y = val.label.values
print('X: {} {}, y {} {}'.format(
    type(val_X[0]), val_X[0].shape, type(val_y[0]), val_y[0]))
plt.imshow(val_X[0][0], cmap="gray")
```

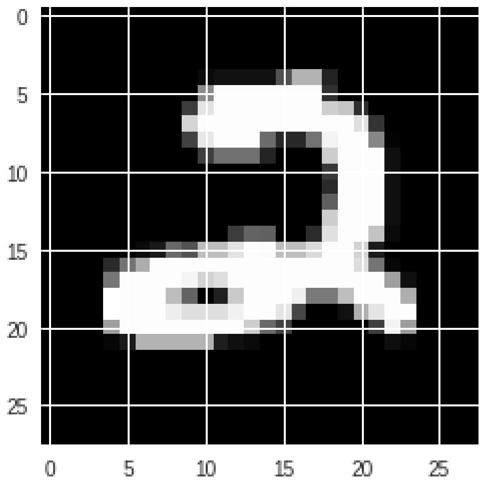
```
X: <class 'numpy.ndarray'> (1, 28, 28), y <class 'numpy.int64'> 7
```

Out[0]:
```
<matplotlib.image.AxesImage at 0x7f96148f90b8>
```



```
In [0]: test_X = test_df.values.reshape(-1, 1, 28, 28) / 255 # reshape and normalize
        test_id = np.arange(1, test_X.shape[0]+1)
        plt.imshow(test_X[0][0], cmap="gray")
```

```
Out[0]: <matplotlib.image.AxesImage at 0x7f961c4bdb38>
```



```
In [0]: train_X.shape, val_X.shape, test_X.shape
```

```
Out[0]: ((37800, 1, 28, 28), (4200, 1, 28, 28), (28000, 1, 28, 28))
```

```
In [0]: # convert to torch tensor
        train_X_, train_y_ = map(torch.tensor, (train_X, train_y))
        val_X_, val_y_ = map(torch.tensor, (val_X, val_y))
        test_X_, test_id_ = map(torch.tensor, (test_X, test_id))
        train_X_.shape, train_y_.min(), train_y_.max()
```

```
Out[0]: (torch.Size([37800, 1, 28, 28]), tensor(0), tensor(9))
```

```
In [0]: train_ds = TensorDataset(train_X_, train_y_)
        setattr(train_ds, 'classes', [0, 1, 2, 3, 4, 5, 6, 7, 8, 9])
        setattr(train_ds, 'c', 10)
        train_ds.classes, train_ds.c
```

```
Out[0]: ([0, 1, 2, 3, 4, 5, 6, 7, 8, 9], 10)
```

```
In [0]: val_ds = TensorDataset(val_X_, val_y_)
        setattr(val_ds, 'classes', [0, 1, 2, 3, 4, 5, 6, 7, 8, 9])
        setattr(val_ds, 'c', 10)
        val_ds.classes, val_ds.c
```

```
Out[0]: ([0, 1, 2, 3, 4, 5, 6, 7, 8, 9], 10)
```

```
In [0]: test_size = test_X_.shape[0]
        #test_ds = TensorDataset(test_X, train_y[:test_size])
        test_ds = TensorDataset(test_X_, test_id_)
```

```
In [0]: BATCH_SIZE = 100
```

In [0]:

```
train_dl = DataLoader(train_ds, batch_size=BATCH_SIZE, shuffle=True)
val_dl = DataLoader(val_ds, batch_size=BATCH_SIZE, shuffle=False)
test_dl = DataLoader(test_ds, batch_size=BATCH_SIZE, shuffle=False)
```

In [0]:

```
data_bunch = DataBunch.create(train_ds, val_ds, test_ds=test_ds, bs=BATC
H_SIZE)
```

In [0]:

```
#data_bunch = ImageDataBunch(train_dl, val_dl, test_dl)
```

In [0]:

```
# test data
x, y = next(iter(data_bunch.train_dl))
x.shape, y.shape
```

Out[0]:

```
(torch.Size([100, 1, 28, 28]), torch.Size([100]))
```

In [0]:

```
# define model
class Net(nn.Module):
    def __init__(self):
        super(Net, self).__init__()
        self.conv1 = nn.Conv2d(1, 10, kernel_size=5)
        self.conv2 = nn.Conv2d(10, 20, kernel_size=5)
        self.conv2_drop = nn.Dropout2d(p=0.25)
        self.fc1 = nn.Linear(320, 50)
        self.fc2 = nn.Linear(50, 10)

    def forward(self, x):
        x = x.float()
        x = F.relu(F.max_pool2d(self.conv1(x), 2))
        x = F.relu(F.max_pool2d(self.conv2_drop(self.conv2(x)), 2))
        x = x.view(-1, 320)
        x = F.relu(self.fc1(x))
        x = F.dropout(x, training=self.training)
        x = self.fc2(x)
        return F.log_softmax(x, dim=1)
```

In [0]:

```
model = Net().cuda()
model
```

Out[0]:

```
Net(
  (conv1): Conv2d(1, 10, kernel_size=(5, 5), stride=(1, 1))
  (conv2): Conv2d(10, 20, kernel_size=(5, 5), stride=(1, 1))
  (conv2_drop): Dropout2d(p=0.25)
  (fc1): Linear(in_features=320, out_features=50, bias=True)
  (fc2): Linear(in_features=50, out_features=10, bias=True)
)
```

In [0]:

```
[p.shape for p in model.parameters()]
```

Out[0]:

```
[torch.Size([10, 1, 5, 5]),
 torch.Size([10]),
 torch.Size([20, 10, 5, 5]),
 torch.Size([20]),
 torch.Size([50, 320]),
 torch.Size([50]),
 torch.Size([10, 50]),
 torch.Size([10])]
```

In [0]:

```
model = Sequential()

model.add(Conv2D(filters = 16, kernel_size = (3, 3), activation='relu',
                 input_shape = (28, 28, 1)))
model.add(BatchNormalization())
model.add(Conv2D(filters = 16, kernel_size = (3, 3), activation='relu'))
model.add(BatchNormalization())
#model.add(Conv2D(filters = 16, kernel_size = (3, 3), activation='rel
u'))
#model.add(BatchNormalization())
model.add(MaxPool2D(strides=(2,2)))
model.add(Dropout(0.25))

model.add(Conv2D(filters = 32, kernel_size = (3, 3), activation='relu'))
model.add(BatchNormalization())
model.add(Conv2D(filters = 32, kernel_size = (3, 3), activation='relu'))
model.add(BatchNormalization())
#model.add(Conv2D(filters = 32, kernel_size = (3, 3), activation='rel
u'))
#model.add(BatchNormalization())
model.add(MaxPool2D(strides=(2,2)))
model.add(Dropout(0.25))

model.add(Flatten())
model.add(Dense(512, activation='relu'))
model.add(Dropout(0.25))
model.add(Dense(1024, activation='relu'))
model.add(Dropout(0.5))
model.add(Dense(10, activation='softmax'))
```

In [0]:

```
# torch.nn.BatchNorm2d(num_features, eps=1e-05, momentum=0.1,
#                      affine=True, track_running_stats=True)
```

In [0]:

```
# define model
class Net2(nn.Module):
    def __init__(self):
        super(Net2, self).__init__()
        self.conv1 = nn.Conv2d(1, 16, kernel_size=3)
        self.norm1 = nn.BatchNorm2d(16)
        self.conv2 = nn.Conv2d(16, 16, kernel_size=3)
        self.drop1 = nn.Dropout2d(p=0.25)
        self.conv3 = nn.Conv2d(16, 32, kernel_size=3)
        self.norm2 = nn.BatchNorm2d(32)
        self.conv4 = nn.Conv2d(32, 32, kernel_size=3)
        self.fc1 = nn.Linear(512, 256)
        self.fc2 = nn.Linear(256, 128)
        self.drop2 = nn.Dropout2d(p=0.5)
        self.fc3 = nn.Linear(128, 10)

    def forward(self, x):
        x = x.float()
        x = F.relu(self.conv1(x))
        x = self.norm1(x)
        x = F.relu(self.conv2(x))
        x = F.max_pool2d(self.norm1(x), 2)
        x = self.drop1(x)

        x = F.relu(F.max_pool2d(self.conv1(x), 2))
        x = F.relu(F.max_pool2d(self.drop1(self.conv2(x)), 2))

        x = F.relu(self.conv3(x))
        x = self.norm2(x)
        x = F.relu(self.conv4(x))
        x = F.max_pool2d(self.norm2(x), 2)
        x = self.drop1(x)

        x = x.view(-1, 512)
        x = F.relu(self.fc1(x))
        x = drop1(x)
        x = F.relu(self.fc2(x))
        x = self.drop2(x)
        x = self.fc3(x)
        return F.log_softmax(x, dim=1)
```

In [0]:
```
model2 = Net2().cuda()
model2
```

Out[0]:
```
Net2(
  (conv1): Conv2d(1, 16, kernel_size=(3, 3), stride=(1, 1))
  (norm1): BatchNorm2d(16, eps=1e-05, momentum=0.1, affine=True, track_
running_stats=True)
  (conv2): Conv2d(16, 16, kernel_size=(3, 3), stride=(1, 1))
  (drop1): Dropout2d(p=0.25)
  (conv3): Conv2d(16, 32, kernel_size=(3, 3), stride=(1, 1))
  (norm2): BatchNorm2d(32, eps=1e-05, momentum=0.1, affine=True, track_
running_stats=True)
  (conv4): Conv2d(32, 32, kernel_size=(3, 3), stride=(1, 1))
  (fc1): Linear(in_features=512, out_features=256, bias=True)
  (fc2): Linear(in_features=256, out_features=128, bias=True)
  (drop2): Dropout2d(p=0.5)
  (fc3): Linear(in_features=128, out_features=10, bias=True)
)
```

In [0]:
```
[p.shape for p in model2.parameters()]
```

Out[0]:
```
[torch.Size([16, 1, 3, 3]),
 torch.Size([16]),
 torch.Size([16]),
 torch.Size([16]),
 torch.Size([16, 16, 3, 3]),
 torch.Size([16]),
 torch.Size([32, 16, 3, 3]),
 torch.Size([32]),
 torch.Size([32]),
 torch.Size([32]),
 torch.Size([32, 32, 3, 3]),
 torch.Size([32]),
 torch.Size([256, 512]),
 torch.Size([256]),
 torch.Size([128, 256]),
 torch.Size([128]),
 torch.Size([10, 128]),
 torch.Size([10])]
```

In [0]:

```
x = torch.randn(3, 1, 28, 28)
conv1 = nn.Conv2d(1, 16, kernel_size=3, padding=0)
x = conv1(x)
print('conv1', x.size())
norm1 = nn.BatchNorm2d(16)
x = norm1(x)
print('norm1', x.size())
conv2 = nn.Conv2d(16, 16, kernel_size=3, padding=0)
x = conv2(x)
print('conv2', x.size())
x = norm1(x)
print('norm1', x.size())
x = F.max_pool2d(x, 2)
print('pool1', x.size())
drop1 = nn.Dropout2d(p=0.25)
x = drop1(x)
print('drop1', x.size())

conv3 = nn.Conv2d(16, 32, kernel_size=3, padding=0)
x = conv3(x)
print('conv3', x.size())
norm2 = nn.BatchNorm2d(32)
x = norm2(x)
print('norm2', x.size())
conv4 = nn.Conv2d(32, 32, kernel_size=3, padding=0)
x = conv4(x)
print('conv4', x.size())
x = norm2(x)
print('norm2', x.size())
x = F.max_pool2d(x, 2)
print('pool2', x.size())
x = drop1(x)
print('drop1', x.size())

x = x.view(-1, 512)
print('view', x.size())
fc1 = nn.Linear(512, 256)
x = fc1(x)
print('fc1', x.size())
x = drop1(x)
print('drop1', x.size())
fc2 = nn.Linear(256, 128)
x = fc2(x)
print('fc2', x.size())
drop2 = nn.Dropout2d(p=0.5)
x = drop2(x)
print('drop2', x.size())
fc3 = nn.Linear(128, 10)
x = fc3(x)
print('fc3', x.size())
x = F.log_softmax(x, dim=1)
print('softmax', x.size())
```

```
conv1 torch.Size([3, 16, 26, 26])
norm1 torch.Size([3, 16, 26, 26])
conv2 torch.Size([3, 16, 24, 24])
norm1 torch.Size([3, 16, 24, 24])
pool1 torch.Size([3, 16, 12, 12])
drop1 torch.Size([3, 16, 12, 12])
conv3 torch.Size([3, 32, 10, 10])
norm2 torch.Size([3, 32, 10, 10])
conv4 torch.Size([3, 32, 8, 8])
norm2 torch.Size([3, 32, 8, 8])
pool2 torch.Size([3, 32, 4, 4])
drop1 torch.Size([3, 32, 4, 4])
view torch.Size([3, 512])
fc1 torch.Size([3, 256])
drop1 torch.Size([3, 256])
fc2 torch.Size([3, 128])
drop2 torch.Size([3, 128])
fc3 torch.Size([3, 10])
softmax torch.Size([3, 10])
```

In [0]:

```
conv1 = nn.Conv2d(1, 16, kernel_size=5, padding=0)
conv2 = nn.Conv2d(16, 16, kernel_size=5, padding=0)
conv3 = nn.Conv2d(16, 32, kernel_size=5, padding=0)
drop1 = nn.Dropout2d(p=0.25)
drop2 = nn.Dropout2d(p=0.5)
fc1 = nn.Linear(512, 256)
fc2 = nn.Linear(256, 10)

x = torch.randn(3, 1, 28, 28)
print('input', x.size())
x = F.relu(F.max_pool2d(conv1(x), 2))
print('conv1, pool, relu', x.size())
#x = F.relu(F.max_pool2d(conv2(x), 2))
#print('conv2, drop, pool, relu', x.size())
x = F.relu(F.max_pool2d(drop1(conv3(x)), 2))
print('conv3, drop, pool, relu', x.size())
x = x.view(-1, 512)
print('view', x.size())
x = F.relu(fc1(x))
print('fc1, relu', x.size())
x = drop2(x)
print('drop2', x.size())
x = fc2(x)
print('fc2', x.size())
x = F.log_softmax(x, dim=1)
print('softmax', x.size())
```

```
input torch.Size([3, 1, 28, 28])
conv1, pool, relu torch.Size([3, 16, 12, 12])
conv3, drop, pool, relu torch.Size([3, 32, 4, 4])
view torch.Size([3, 512])
fc1, relu torch.Size([3, 256])
drop2 torch.Size([3, 256])
fc2 torch.Size([3, 10])
softmax torch.Size([3, 10])
```

In [0]:
```
class Net3(nn.Module):
    def __init__(self):
        super(Net3, self).__init__()
        self.conv1 = nn.Conv2d(1, 16, kernel_size=5, padding=0)
        self.conv2 = nn.Conv2d(16, 16, kernel_size=5, padding=0)
        self.conv3 = nn.Conv2d(16, 32, kernel_size=5, padding=0)
        self.drop1 = nn.Dropout2d(p=0.25)
        self.drop2 = nn.Dropout2d(p=0.5)
        self.fc1 = nn.Linear(512, 256)
        self.fc2 = nn.Linear(256, 10)

    def forward(self, x):
        x = x.float()
        x = F.relu(F.max_pool2d(self.conv1(x), 2))
        x = F.relu(F.max_pool2d(self.drop1(self.conv3(x)), 2))
        x = x.view(-1, 512)
        x = F.relu(self.fc1(x))
        x = self.drop2(x)
        x = self.fc2(x)
        return F.log_softmax(x, dim=1)
```

In [0]:
```
model3 = Net3().cuda()
model3
```

Out[0]:
```
Net3(
  (conv1): Conv2d(1, 16, kernel_size=(5, 5), stride=(1, 1))
  (conv2): Conv2d(16, 16, kernel_size=(5, 5), stride=(1, 1))
  (conv3): Conv2d(16, 32, kernel_size=(5, 5), stride=(1, 1))
  (drop1): Dropout2d(p=0.25)
  (drop2): Dropout2d(p=0.5)
  (fc1): Linear(in_features=512, out_features=256, bias=True)
  (fc2): Linear(in_features=256, out_features=10, bias=True)
)
```

In [0]:
```
lr=2e-2
```

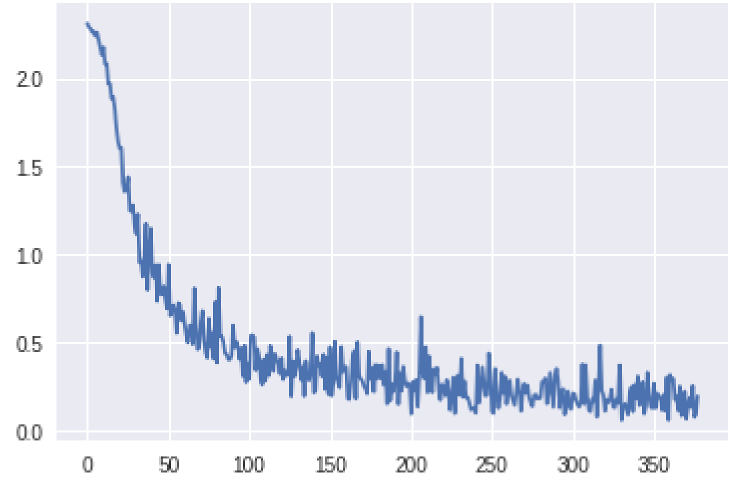
In [0]:
```
loss_func = nn.CrossEntropyLoss()
```

In [0]:
```
def update(m, x, y, lr):
    opt = optim.Adam(m.parameters(), lr)
    y_hat = m(x)
    loss = loss_func(y_hat, y)
    loss.backward()
    opt.step()
    opt.zero_grad()
    return loss.item()
```

In [0]:
```
losses = [update(model3, x, y, 1e-3) for x, y in data_bunch.train_dl]
```

In [0]:
```
plt.plot(losses)
```

Out[0]:
```
[<matplotlib.lines.Line2D at 0x7f96171fc4e0>]
```



In [0]:
```
learn = Learner(data_bunch, Net3(), loss_func=loss_func, metrics=[accuracy])
```

In [0]:
```
learn.lr_find()
```

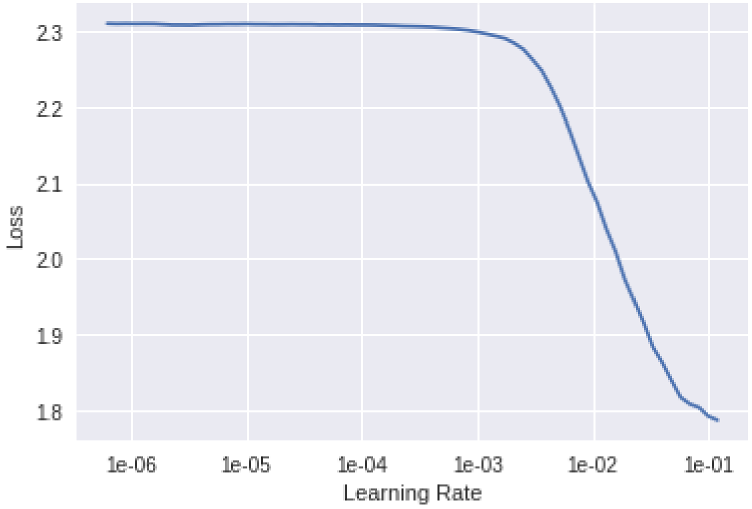
```
0.00% [0/1 00:00<00:00]

epoch   train_loss   valid_loss   accuracy

Interrupted

LR Finder is complete, type {learner_name}.recorder.plot() to see the graph.
```

In [0]:
```
learn.recorder.plot()
```



In [0]:

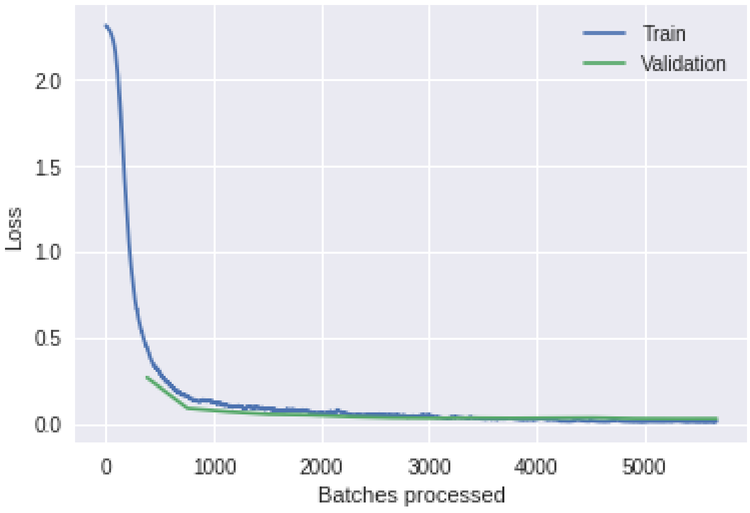
```
learn.fit_one_cycle(15)
```

Total time: 01:17

epoch	train_loss	valid_loss	accuracy
1	0.439003	0.264777	0.922381
2	0.159091	0.088650	0.972619
3	0.101303	0.068729	0.979286
4	0.089500	0.054357	0.984762
5	0.068262	0.050309	0.983810
6	0.053931	0.039999	0.988095
7	0.050181	0.033803	0.988810
8	0.044288	0.031376	0.990714
9	0.034064	0.032796	0.990000
10	0.028213	0.031178	0.990238
11	0.021225	0.032805	0.991190
12	0.015365	0.034120	0.990952
13	0.012217	0.028995	0.992857
14	0.014344	0.028273	0.992857
15	0.012521	0.028325	0.992857

In [0]:

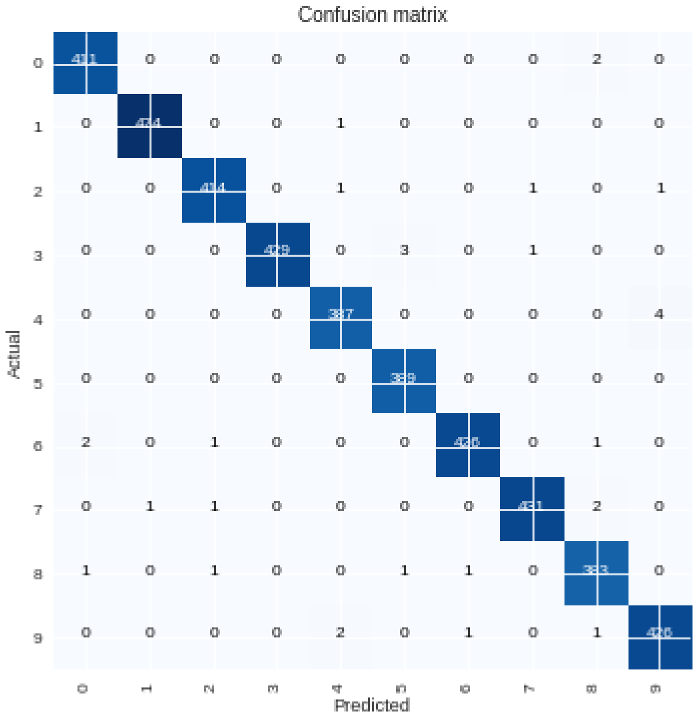
```
learn.recorder.plot_losses()
```



In [0]:

```
interp = ClassificationInterpretation.from_learner(learn)
```

In [0]:
```
interp.plot_confusion_matrix(figsize=(6, 6), dpi=60)
```



In [0]:
```
interp.most_confused(min_val=2)
```

Out[0]:
```
[(4, 9, 4), (3, 5, 3)]
```

In [0]:
```
preds, out = learn.get_preds(ds_type=DatasetType.Test)
```

In [0]:
```
preds.size()
```

Out[0]:
```
torch.Size([28000, 10])
```

In [0]:
```
preds[:5]
```

Out[0]:
```
tensor([[4.9538e-12, 2.1620e-13, 1.0000e+00, 1.0625e-09, 9.9504e-13, 5.
4501e-18,
         3.7914e-16, 2.6048e-11, 1.9172e-12, 8.4144e-15],
        [1.0000e+00, 3.7047e-13, 2.5669e-09, 2.6698e-12, 4.3469e-12, 1.
9986e-10,
         4.2957e-08, 5.4159e-11, 2.3332e-10, 1.1511e-09],
        [2.7344e-15, 3.3137e-14, 1.4718e-13, 6.3924e-12, 7.0355e-07, 3.
7977e-13,
         1.5634e-19, 1.6721e-09, 2.4343e-09, 1.0000e+00],
        [8.1703e-02, 7.8189e-11, 9.6000e-05, 6.2761e-06, 1.3285e-05, 2.
2289e-07,
         1.6469e-08, 1.0042e-04, 4.4099e-06, 9.1808e-01],
        [2.7905e-13, 4.3743e-11, 3.6177e-09, 1.0000e+00, 2.1209e-15, 1.
4122e-09,
         8.6600e-14, 4.8329e-11, 1.6029e-08, 3.0042e-12]])
```

In [0]:
```
out[:5], out[-5:]
```

Out[0]:
```
(tensor([1, 2, 3, 4, 5]), tensor([27996, 27997, 27998, 27999, 28000]))
```

In [0]:
```
y_pred = np.argmax(preds.cpu().numpy(), axis=1)
y_pred.shape
```

Out[0]:
```
(28000,)
```

In [0]:
```
y_pred[:5]
```

Out[0]:
```
array([2, 0, 9, 9, 3])
```

In [0]:
```
submit_df = pd.DataFrame({'ImageId': out.numpy(), 'Label': y_pred})
submit_df.head()
```

Out[0]:

	ImageId	Label
0	1	2
1	2	0
2	3	9
3	4	9
4	5	3

In [0]:
```
submit_df.to_csv("submission.csv", index=False)
```

In [0]:
```
!kaggle competitions submit -c digit-recognizer -f submission.csv -m "first submission"
```

```
100% 208k/208k [00:04<00:00, 43.2kB/s]
Successfully submitted to Digit Recognizer
```

In [0]:
```

```