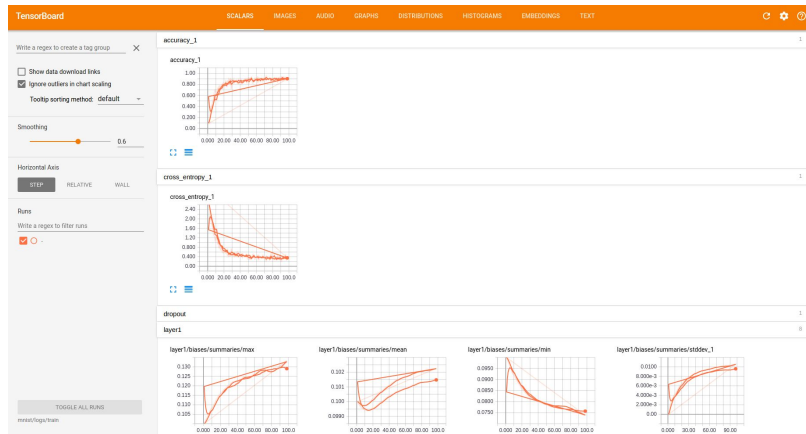


Tensor Board

TensorBoard

- Tensorflow에서 제공하는 매우 강력한 Debugging tool
 - 원하는 값이나 이미지, Matrix 값의 분포 등을 Graph, Histogram 등으로 확인할 수 있다.
 - 또한, Embedding을 3d space 를 통해 확인하거나 전체 network 를 그래프로 그려주는 등 편리한 기능을 다수 제공한다.
-
- tf.summary 에 있는 library를 이용한다.
 - https://www.tensorflow.org/api_docs/python/tf/summary



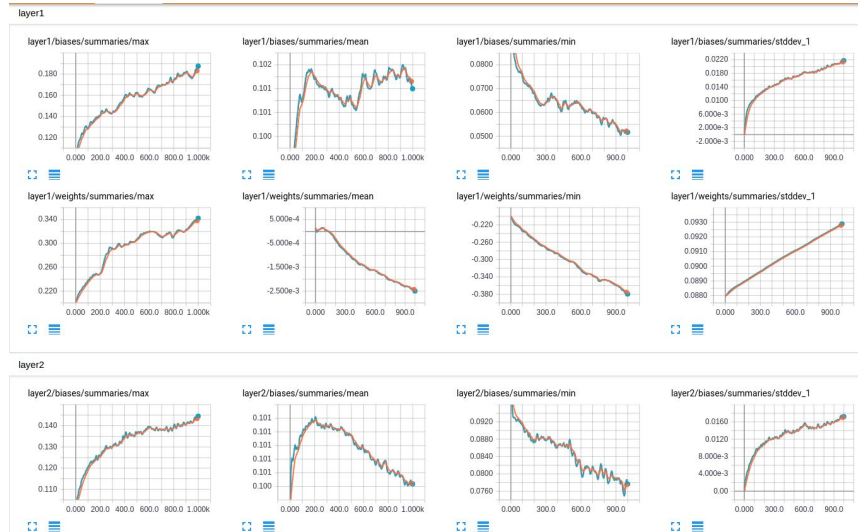
TensorBoard - Scalar

`tf.summary.scalar(name, variable)`

Scalar value, 즉 하나의 값을 추적해서 그래프로 나타낼 수 있다.

오른쪽과 같이 어떤 하나의 값을 epoch마다 저장해 그래프로 나타내 주기 때문에 내가 생각한 값이 어떻게 움직이는 지를 쉽게 파악할 수 있다.

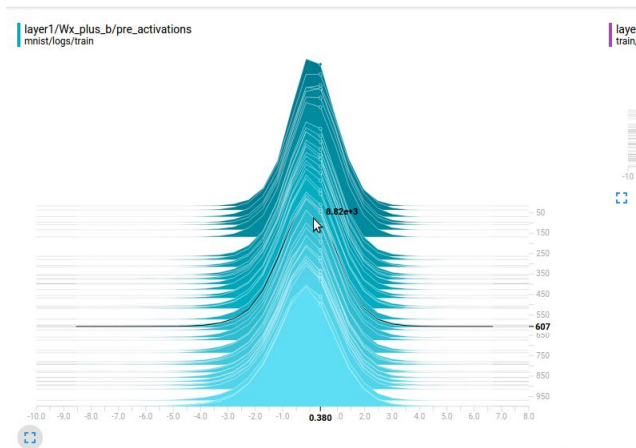
name은 그 위에 있는 name scope를 따라간다.



TensorBoard - Histogram

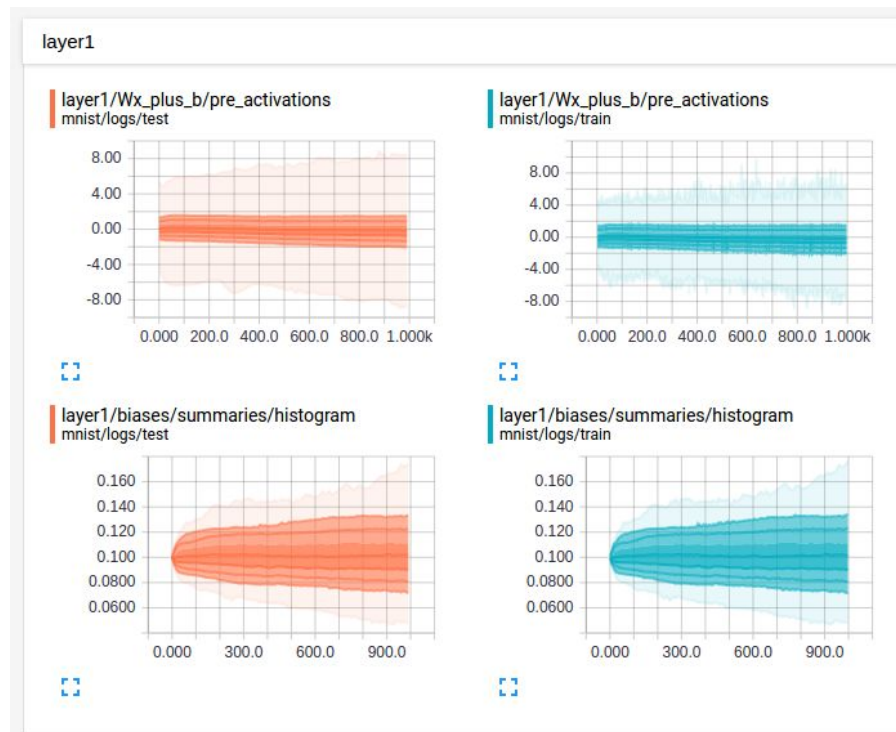
`tf.summary.histogram(name, tensor)`

tensor 안에 있는 값들의 분포를 표현한다.



TensorBoard - Distributions

Histogram을 distribution graph로 변경해서도
출력해준다.



TensorBoard - Image

`tf.summary.image(name, tensor, max_outputs)`

tensor: [batch_size, height, width, channels]

위 텐서에 있는 image 중 max_outputs 개
만큼의 image를 랜덤으로 뽑아 보여준다.

channels : 1-gray, 3-RGB, 4-RGBA

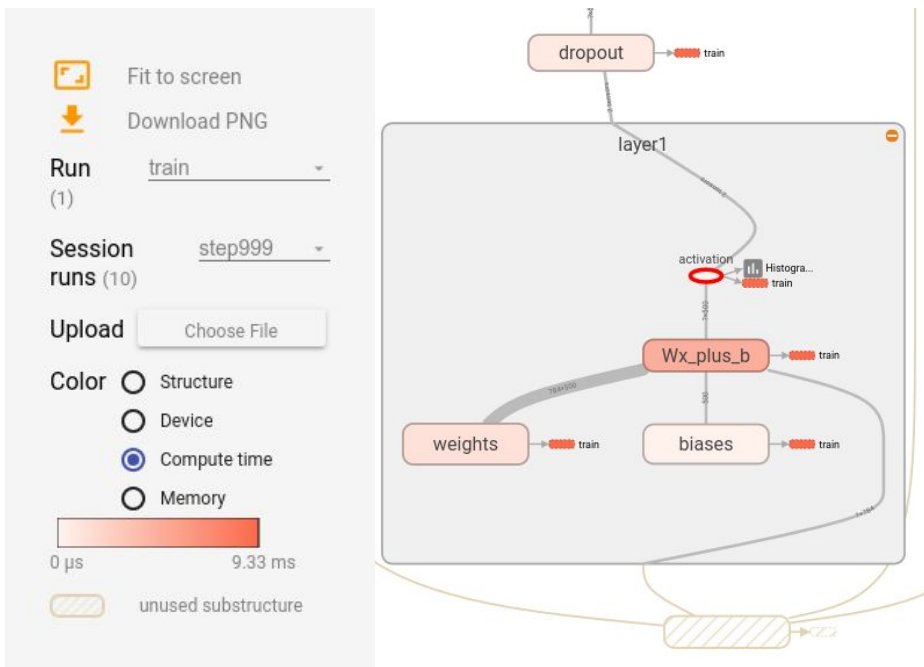


TensorBoard - Graph

```
train_writer =  
tf.summary.FileWriter(FLAGS.log_dir + '/train',  
sess.graph)
```

FileWrite를 만들 때 graph를 끼워넣어
네트워크의 전체 구조를 쉽게 파악할 수 있다.

Sess.run할 때 RunOptions, RunMetadata 등의
class를 이용하여 사용 device, memory, running
time 등을 추적할 수 있다.



TensorBoard - audio, text

Audio와 Text 타입 또한 나타낼 수 있다.

```
tf.summary.audio(name, tensor, sample_rate, max_outputs)
```

```
tf.summary.text(name, tensor)
```

자세한 내용은 다음 링크 참조

Audio - https://www.tensorflow.org/api_docs/python/tf/summary/audio

Text - https://www.tensorflow.org/api_docs/python/tf/summary/text

TensorBoard - Merge and Write

- ❖ `tf.summary.merge_all()`
- ❖ `tf.summary.merge(inptus)`
 - 지금까지 만든 `summary`를 합쳐 하나의 `node`로 만든다.
 - `merged = tf.summary.merge_all()`
`summary, acc = sess.run([merged, accuracy], feed_dict=feed_dict(False))`
- ❖ `tf.summary.FileWriter(log_path, graph)`
 - `log` 파일을 만들기 위한 `object`를 생성해준다.
 - `train_writer = tf.summary.FileWriter(FLAGS.log_dir + '/train', sess.graph)`
- ❖ `tf.summary.FileWriter().add_summary(summary, epoch)`
 - 만들어진 `summary`를 파일에 저장해 `tensorboard`에 출력해준다.
 - `train_writer.add_summary(summary, i)`

Exercises

1. CIFAR-10 classification 코드에 `summary`를 추가해 필요한 정보를 `tensorboard`에 출력해보세요.