1. ネットワークアーキテクチャの変更

初期値に対して、チャンネル数・層数・カーネルサイズを変更し、比較を行った。

チャンネル数

|  |  |
| --- | --- |
| **Accuracy** | **Loss** |
| 初期値 | |
|  |  |
|  |  |
| channel：１⇒５⇒１０ | |
|  |  |

考察：

* チャンネル数を変更してもAccuracy・Lossの変化に大きな違いは現れなかった。
* チャンネル数を増やすことで扱える特徴が増えるので、色々なデータにたいする汎化性能が向上すると考えられる。結果として、少しではあるがvalidのAccuracyが増加し、Lossが小さくなっている。
* どのように関係しているのか他のパラメータとともに詳しく調べる必要がある。

層数

|  |  |
| --- | --- |
| **Accuracy** | **Loss** |
| 初期値 | |
|  |  |
|  |  |
| 層を追加  ３層：　１⇒３⇒６⇒１０ | |
|  |  |

* 層数を変更すると、Accuracyに関してはTrainとValidがほぼ同じ動きをすることが見られた。
* 学習が足りていないと考え、同じ条件でエポック数を増やした。

|  |  |
| --- | --- |
| Accuracy | Loss |
|  |  |

* この結果から学習が足りていないというよりモデルの汎化性能が高まっていると考えられる。

カーネルサイズ

|  |  |
| --- | --- |
| **Accuracy** | **Loss** |
| 初期値 | |
|  |  |
|  |  |
| カーネルサイズ変更：  ３×３ | |
|  |  |

・カーネルサイズを変更することによって、全体のAccuracyは下がり、Lossは大きくなっった。

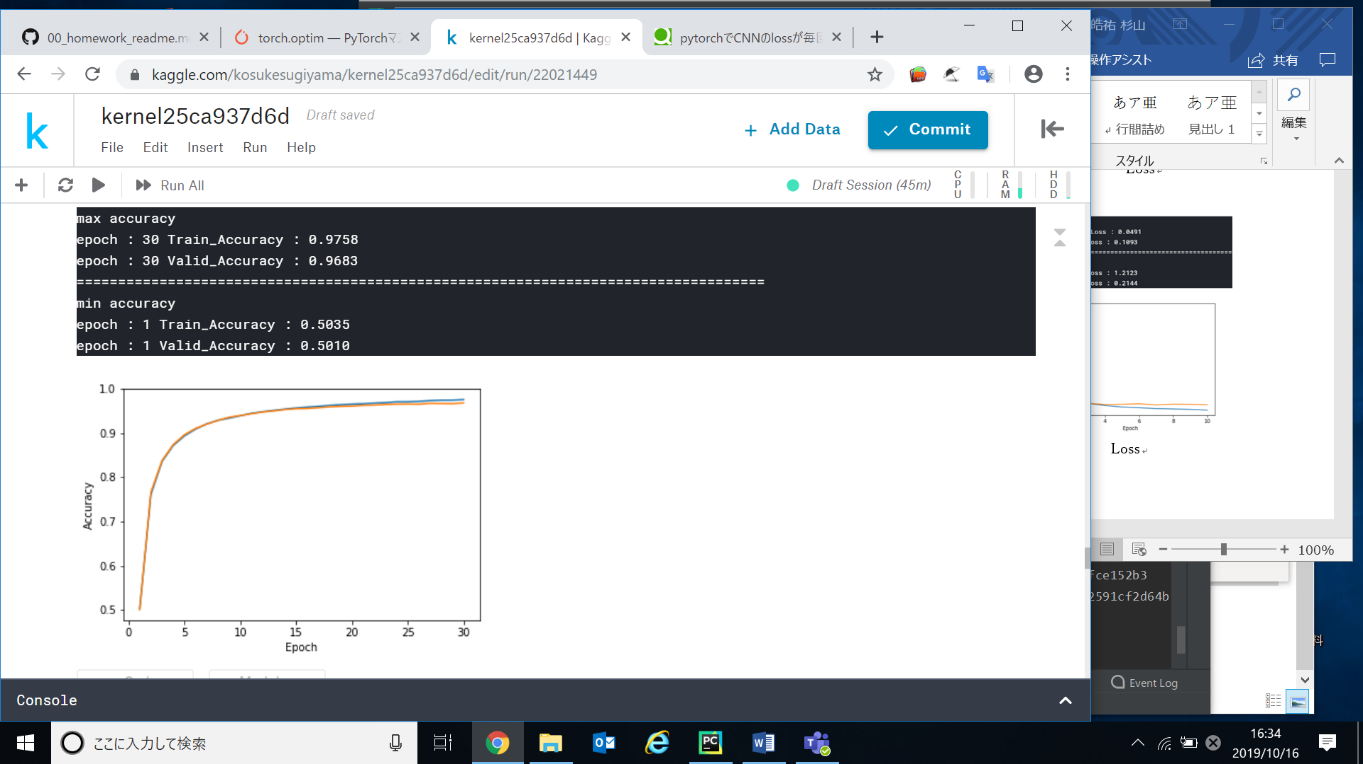
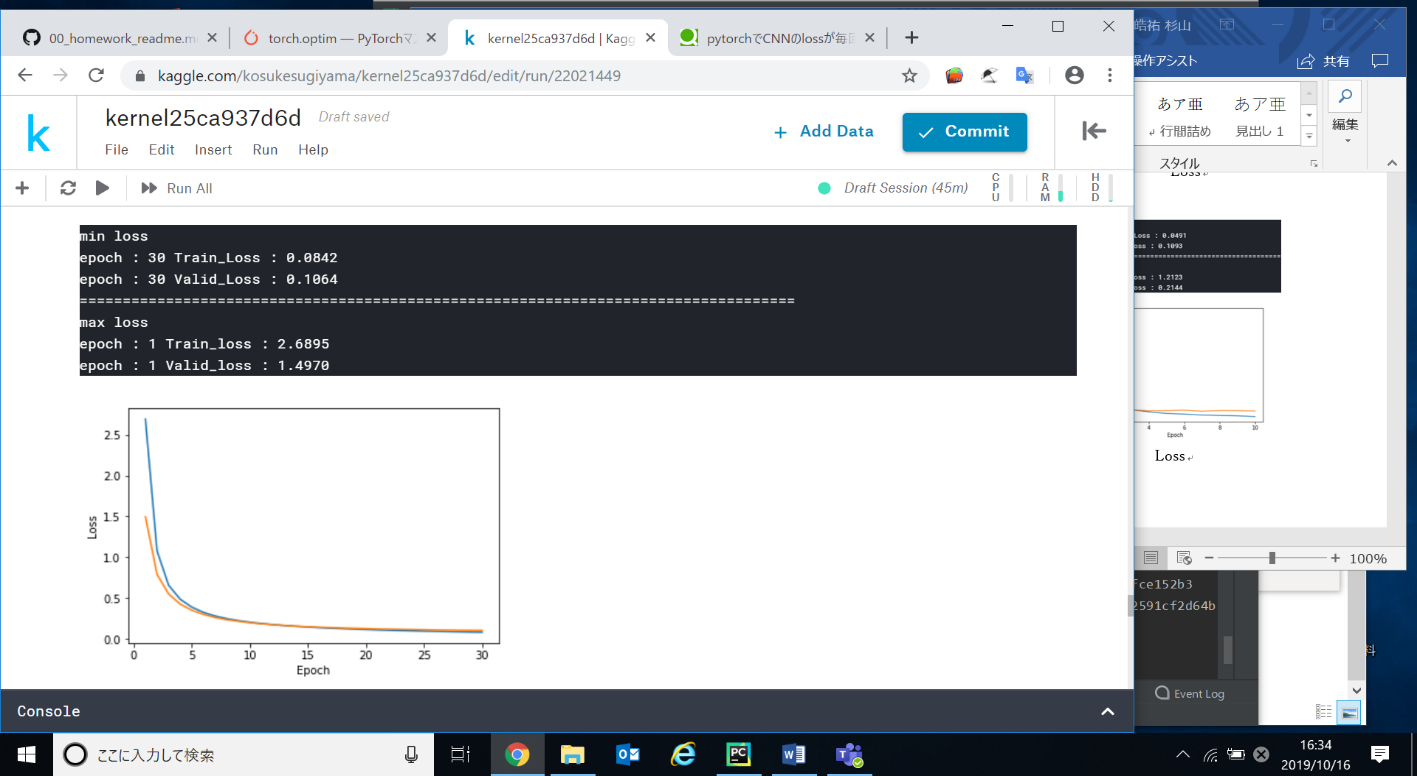
・初期値に比べ、AccuracyとLossが収束しきっていないように見られる。カーネルサイズを２×２から３×３に変えたことによって、画像の特徴を大きく計算することになるのでその分、データの特徴を学習しきれなかったのだと考えられる。

・加えて、単純に２８×２８の画像なので、３×３で畳み込みとプーリングを行うとつぶれるデータの割合が大きくなってしまうという可能性もある。

1. **最適化関数を変更(Adam)**

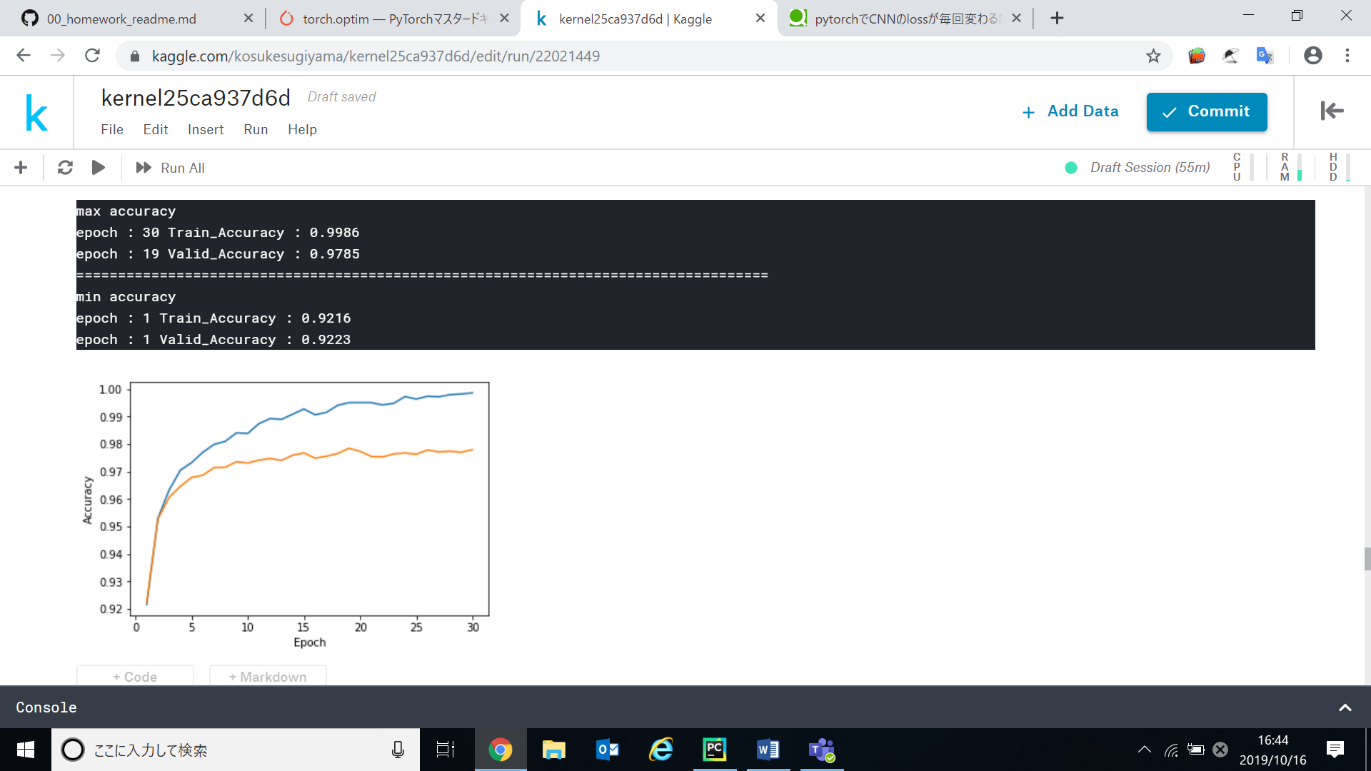
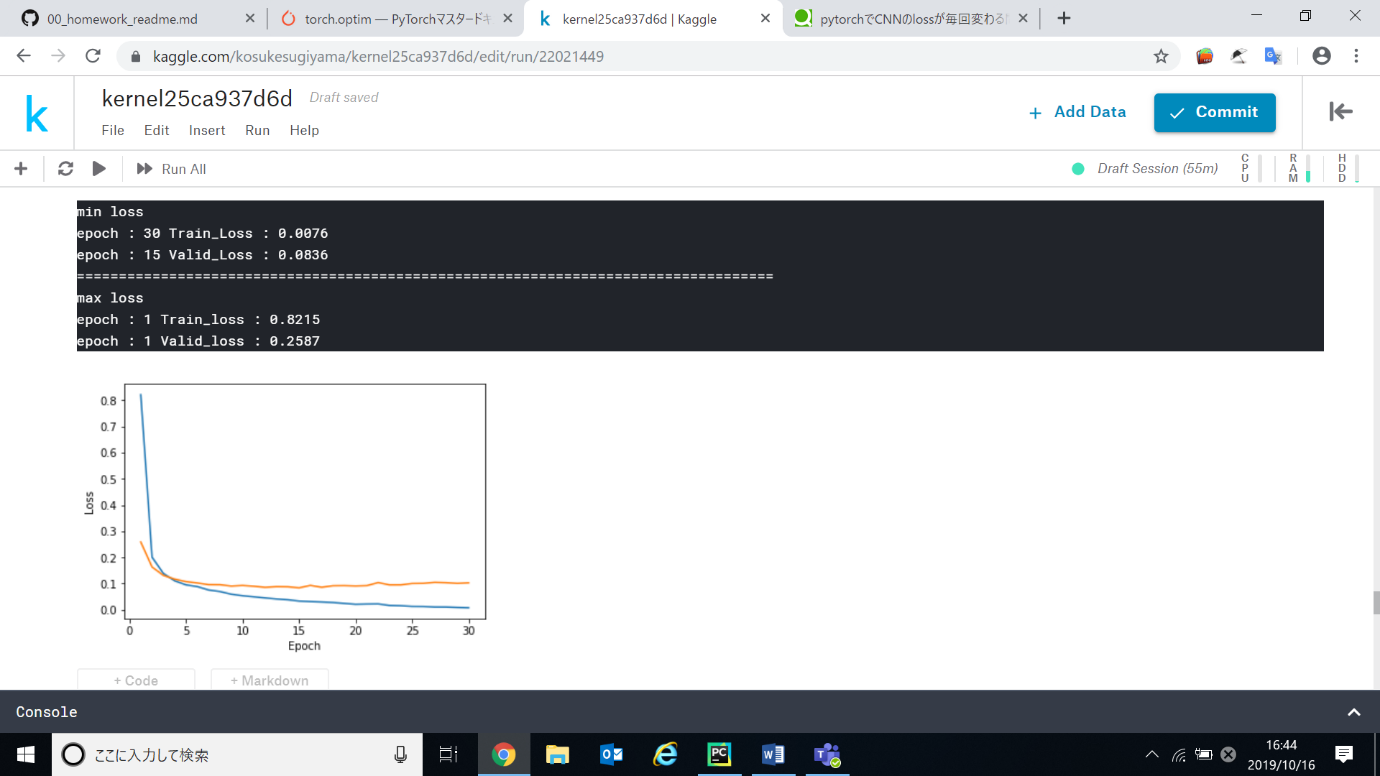
**グラフの青線が学習データ、オレンジ線が検証データ**

* lr = 0.0001 実行時間 : 1.45 s/epoch

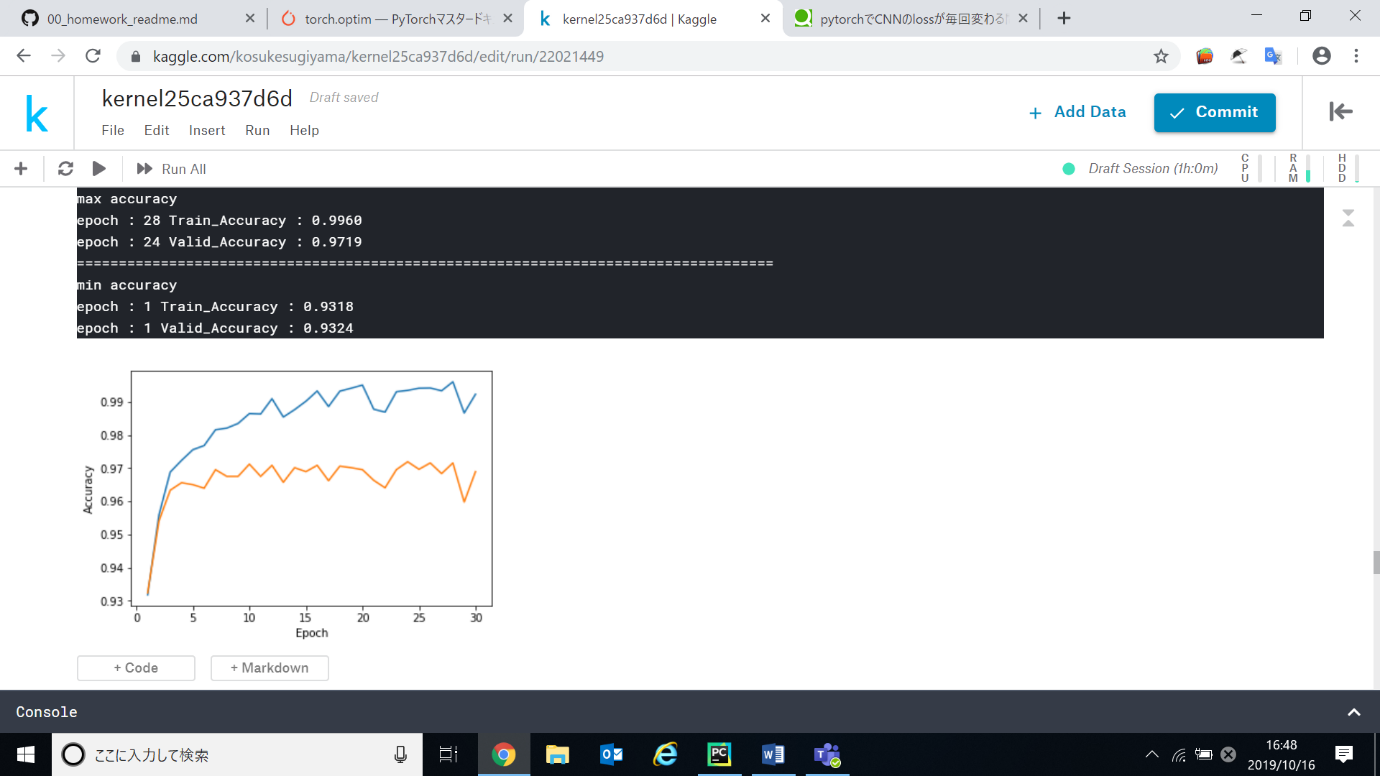
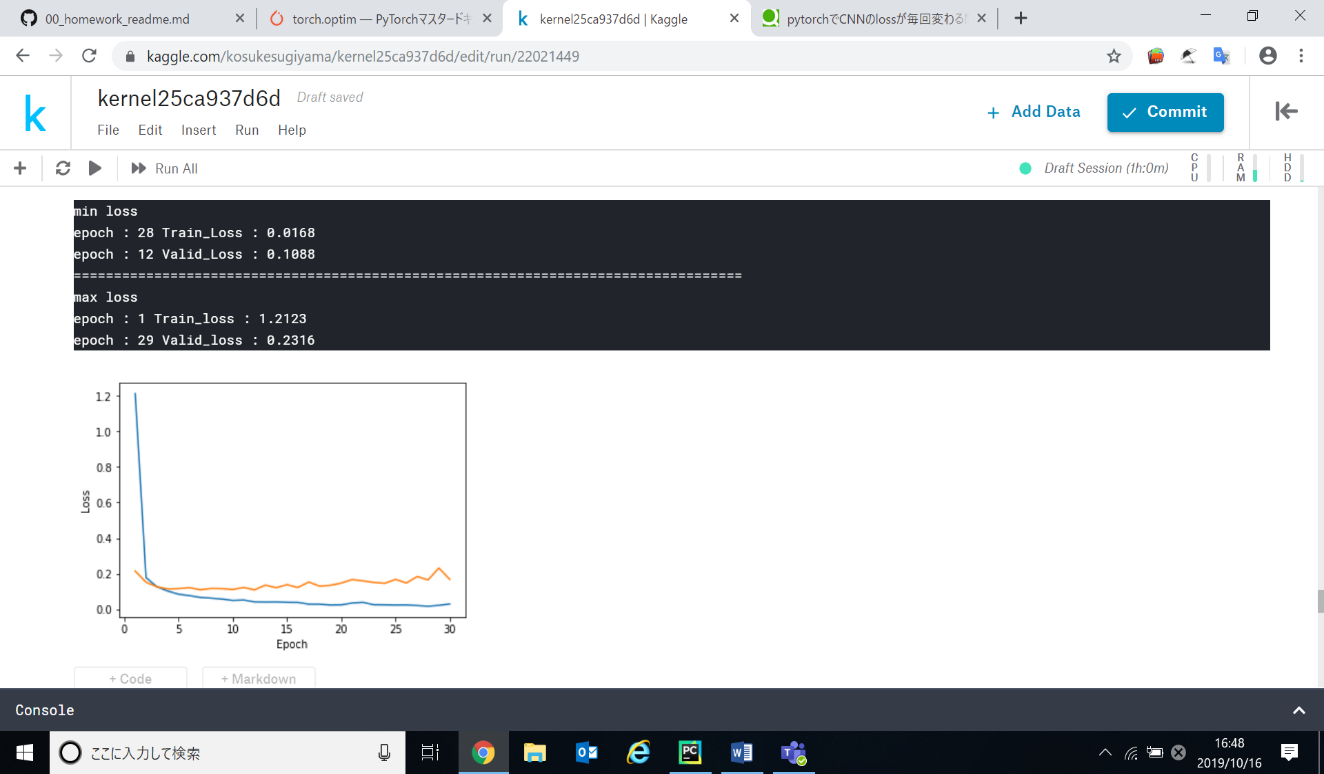
Accuracy Loss

* lr = 0.001 実行時間 : 1.45 s/epoch

Accuracy Loss

* lr = 0.01実行時間 : 1.40 s/epoch

Accuracy Loss

* 学習率が高いほど実行時間が短くなることが分かった

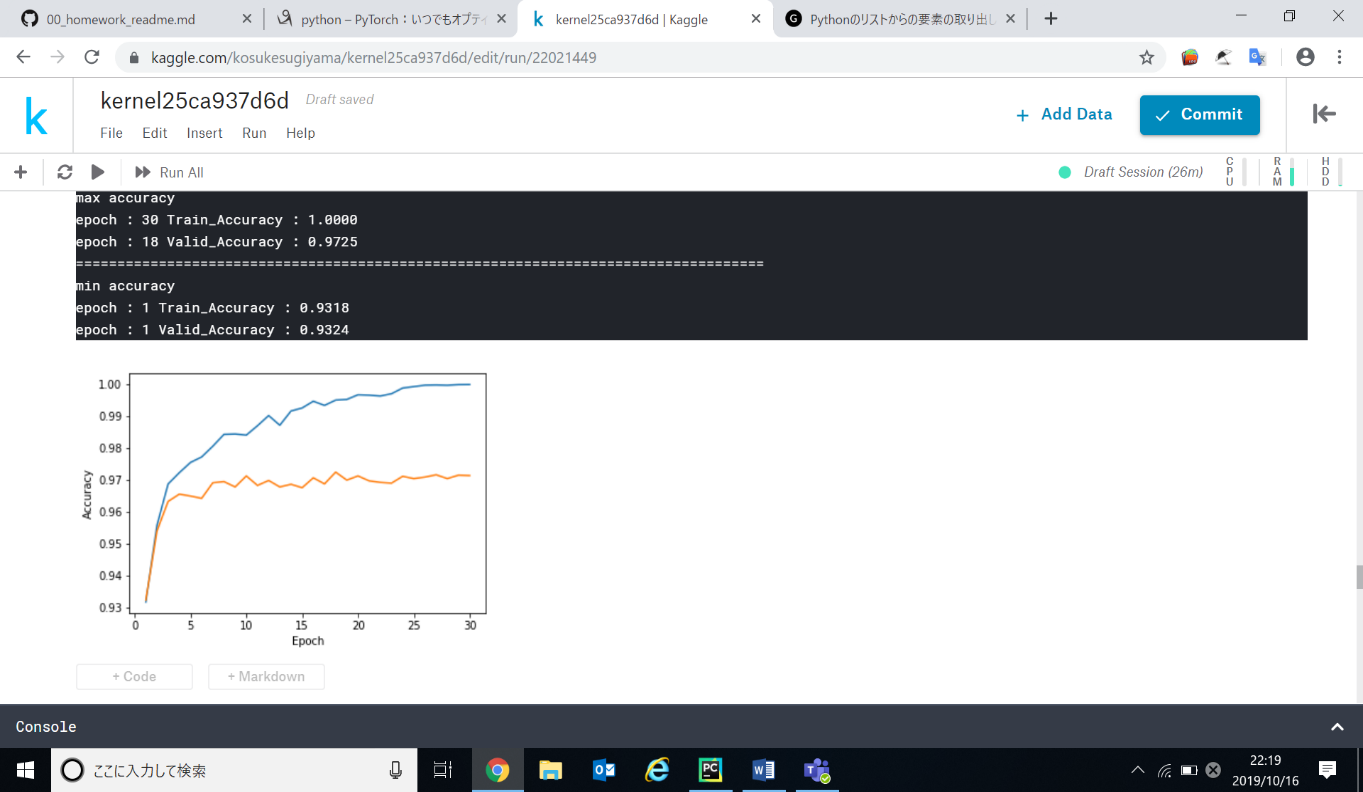
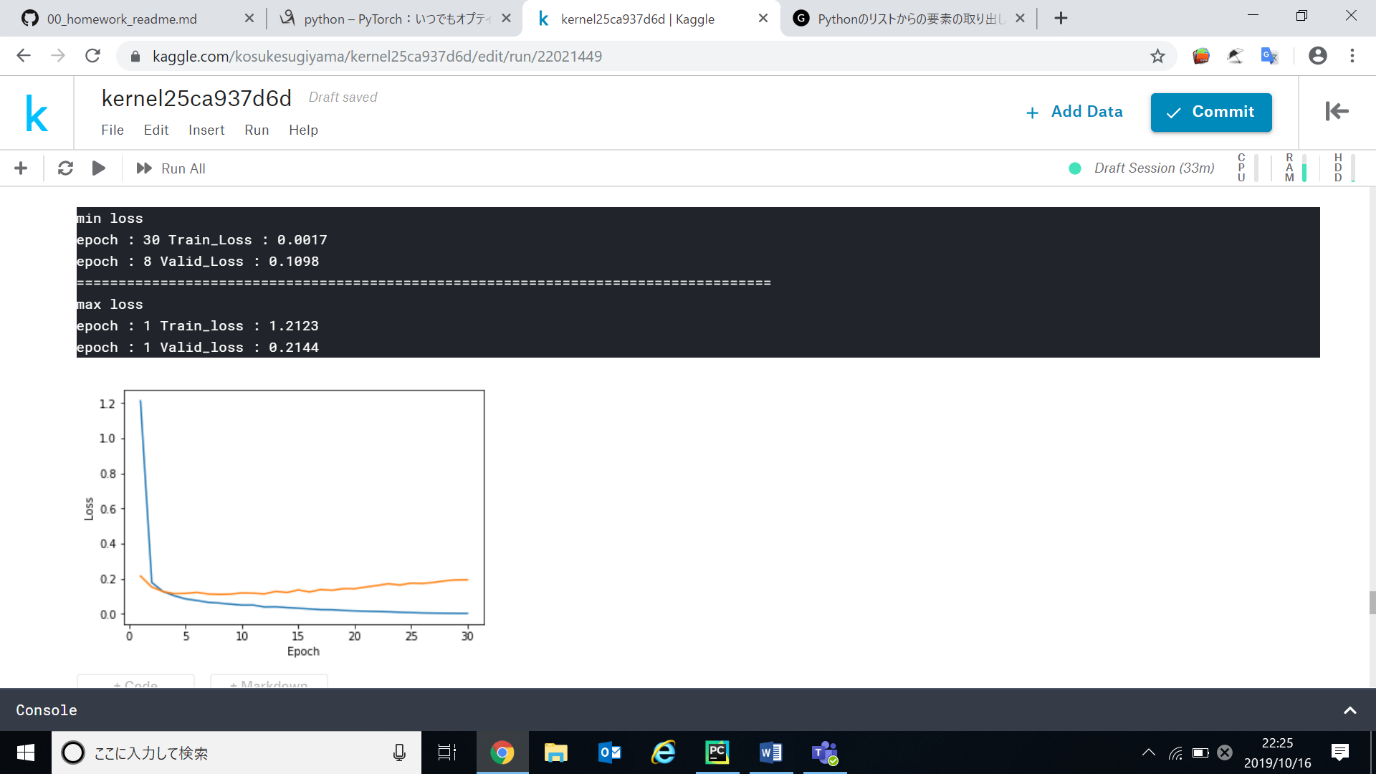
→　学習率が低いと収縮までに時間がかかるから

* 学習率が高いほど1 epoch目のAccuracyが高い
* 学習率が高いと早い段階でAccuracy、Lossが収束してしまうため、十分な学習ができていないと考えられる、しかし学習率が低いと学習が完了するために多くのepochが必要となってしまう

そのため学習率が高い状態で学習を始め、エポックが進むにつれ学習率を下げることで、実行時間の減少と、十分な学習が期待できる。

そこで、直前のエポックの検証データの正答率を今エポックの正答率が下回ったときに、学習率が減少するプログラムを作成する。

初期の学習率は0.001 下げ幅は0.0005とする。

検証データの正答率は、素直に学習率0.001を使用する手法でいい結果が出ているが、

訓練に関しては100%の正答率が出るなど結果につながっていた。

また、これはlossでも同じことがいえる。

さらに、検証用データでは早い段階で収束していることからも、検証用データでより良い評価を出すためには、別のモデルを使用する必要があると考えられる。

1. 前処理を実施(正規化)

― Train　― valid

|  |  |
| --- | --- |
| 元データのまま | |
|  |  |
| X/255 | |
|  |  |

<考察>

画像データを255で割って正規化を行ったところ、今回の設定の場合、精度が低下した。表を見る限り、正規化したデータを用いたモデルは、精度、損失共に、ValidとTrainデータに大きな差がなく、学習が不足しているものと考えられる。そこで、エポック数を30にして学習を行った結果は以下の様になった。

|  |  |
| --- | --- |
| X/255(epoch=30) | |
|  |  |

エポックを30にすることで、Trainの値がValidよりAccuracy、Lossともによくなっている事から学習は十分行われたものと思われる。

正規化した結果と正規化していない結果を比較すると元のモデルと比べてあまり精度はよくなっていない事がわかる。このことから今回の場合（optimizerがAdamで学習率0.001の場合）正規化することは精度に大きな成果は得られていないように感じる。

一方で、TrainとValidのAccuracy、Lossの差は少なくなっているため、過学習が起きにくくなっているように感じられる。

正規化を画像データに対して行うメリットについて参考文献で確認したところ、観測したような画像データの場合ですべての画像データの濃淡の値が0~255以内の値に収まっていない場合は、正規化を行うことで、データの輝度値が同じ範囲に集約され、最適化（学習）がうまく行うことができるとのことでした。

今回の場合に当てはめて考えるとMNISTのデータはすべて0~255までのデータが入っており、正規化した事によるメリットが大きく出なかったものと考えられる。

<参考URL>

ニューラルネットワークの学習の工夫

<https://www.hellocybernetics.tech/entry/2017/01/31/041148>

（付録）元データのパラメータ

1. ニューラルネットのアーキテクチャ

入力時の特徴量: (C=1, w=28, h=28)

↓

畳み込み層、ReLu、ブーリング層

↓

特徴量マップ①(3, 13, 13)

↓

畳み込み層、Relu、ブーリング層

↓

特徴量マップ②(6, 6, 6)

↓

次元変換(216)

↓

全結合、Relu

↓

特徴量(100)

↓

全結合

↓

出力(10)

1. エポック数、バッチ数

バッチ数: 700

エポック数: 10

1. Optimizer

Adam(学習率: 0.001)