実装演習レポート

科目：深層学習

[Day 4]

【強化学習】

要点

・強化学習は長期的に報酬を最大化できるように環境の中で行動を選択できるエージェントを作ることを目標とする機械学習の一分野である。

・キャンペーンのコストという負の報酬と売上という正の報酬を最大化するために、行動を改善していく。

・過去のデータでベストとされる行動のみを常に取り続ければ他にもっとベストな行動を見つけることはできない（探索が足りない）。未知の行動のみを常に取り続ければ過去の経験が活かせない（利用が足りない）。探索と利用はトレードオフの関係性にある。

・教師あり教師なし学習ではデータに含まれるパターンを見つけ出して予測することが目標だが、強化学習では優れた方策を見つけることが目標である。

・Q学習は行動価値関数を行動するごとに更新することにより学習を進める方法である。

・関数近似法は価値関数や方策関数を関数近似する手法のことである。

・価値関数には、ある状態の価値に注目する状態価値関数と状態と価値を組み合わせた価値に注目する行動価値関数の二種類がある。

・方策関数は方策ベースの強化学習手法においてある状態でどのような行動を取るのかという確率を与える関数のことである。

・方策反復法は方策をモデル化して最適化する手法である。

【AlphaGo】

要点

・AlphaGo LeeはValueNet（19x19、49チャンネル）という価値関数とPolicyNet（盤面特徴入力19x19、48チャンネル）という方策関数から成る。

・二次元のデータなので畳み込みしてpoolingする。ReLU関数を用い、PolicyNetでは活性化関数としてSoftMaxを通して出力する。

・ValueNetではConvolutionの後に全結合をかけてTanH Layer（-∞〜∞の値を-1~1までに変換するlayer）を通して出力する。

・線形の方策関数であるRollOutPolicyやPolicyNetを教師あり学習で学習させる。

・モンテカルロ木探索でPlayOutと呼ばれるランダムシミュレーションを多数回行い、その勝敗を集計して着手の優劣を決定する。

・AlphaGo Zeroは教師あり学習を一切行わず、強化学習のみで作成する。Residual Netを導入している。ResidualNetworkはネットワークにショートカットを設ける。

【データ並列化】

要点

・分散深層学習とは、複数の計算資源（ワーカー）を使用し、並列的にニューラルネットを構成することでモデルを高速に学習させ、効率の良い学習を行うことである。

・深層学習は多くのデータを使用したり、パラメータ調整のために多くの時間を使用したりするため、データ並列化、モデル並列化、GPUによる高速技術が不可欠である。

・データ並列化では親モデルを各ワーカーに子モデルとしてコピーし、データを分割しワーカーごとに計算させる。

・各モデルのパラメータの合わせ方で同期型か非同期型か決まる。同期型は各ワーカーの計算終了を待ち、全ワーカーの勾配が出たところで勾配の平均を計算して親モデルのパラメータを更新する。非同期型は各ワーカーの計算を待たず、子モデルごとに更新する。非同期型の方が処理スピードは早いが同期型の方が精度は良いことが多い。

【モデル並列化】

要点

・親モデルを各ワーカーに分割し、それぞれのモデルを学習させ、全てのデータで学習が終わった後で一つのモデルに復元化する。モデルを分割して並列に処理する。

・モデルが大きい時はモデル並列化を、データが大きい時はデータ並列化をすると良い。

・モデルのパラメータ数が多いほど、スピードアップの効率も向上する。

・Large Scale Distributed Deep NetworksというGoogle社が2016年に出した論文で紹介され、TensorFlowの前身といわれている。

・GPGPU (General-purpose of GPU)は元々の使用目的であるグラフィック以外の用途で使用されるGPUの総称である。

・CPUは高性能なコアが少数あり、複雑で連続的な処理が得意である。

・GPUは比較的低性能なコアが多数あり、簡単な並列処理が得意である。ニューラルネットの学習は単純な行列演算が多いので高速化が可能である。

・CUDAはGPU上で並列コンピューティングを行うためのプラットフォームであり、N V I D I A社が開発しているGPUのみで使用可能である。Deep Learning用に提供されている。

・OpenCLはオープンな並列コンピューティングのプラットフォームでありIntel, AMD, ARMなどNVIDIA社以外の会社のGPUからでも使用可能である。

【量子化】

要点

・ネットワークが大きくなると大量のパラメータが必要となり学習や推論に多くのメモリと演算処理が必要となる。通常のパラメータの64bit浮動小数点を32bitなど下位の精度に落とすことでメモリと演算処理の削減を行う。少数の数字をコンピューターで表すことを量子化（Quantization）という。

・ニューロンの重みを浮動小数点のbit数を少なくし有効桁数を下げることでニューロンのメモリサイズを小さくすることができ、多くのメモリを消費するモデルのメモリ使用量を抑え、省メモリ化することができる。

・32bitを用いて小数点を表すことを単精度、64bitを倍精度、16bitを半精度と呼ぶ。

・FLOPSはfloating operationsで、GPUの精度を表す単位である。

・モデルの精度が多少悪くなっても16bit（半精度）の計算スピードで学習する方が現実的である。

【蒸留】

要点

・精度の高いモデルはニューロンの規模が大きなモデルになっているため、推論に多くのメモリと演算処理が必要である。規模の大きなモデルの知識を使い軽量なモデルの作成を行うことを蒸留という。

・学習済みの精度の高いモデルの知識を軽量なモデルへ継承させることをモデルの簡約化という。

・蒸留は教師モデル（予測精度の高い複雑なモデルやアンサンブルされたモデル）と生徒モデル（軽量化されたモデル）の２つで構成され、教師モデルの重みを固定し生徒モデルの重みを更新していく。誤差は教師モデルと生徒モデルのそれぞれの誤差を使い重みを更新していく。

・蒸留により少ない学習回数でより精度の良いモデルを作成することができる。

【プルーニング】

要点

・モデルの精度に寄与が少ないニューロンを削減することでモデルの軽量化、高速化を行うことをプルーニングという。重みが閾値以下のニューロンを削減する。

・重みの閾値を高くするとニューロンは大きく削減できるが精度も減少する。

【MobileNet】（画像認識ネットワーク）

要点

・MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applicationsという論文でディープラーニングモデルの軽量化・高速化・高精度化を実現した。Depthwise Separable Convolutionという手法を用いて計算量を軽減する。

・画像認識タスクに用いられるネットワークの軽量化を試みる。

・一般的な畳み込みレイヤーは入力特徴マップ（チャネル数）H x W x C、畳み込みカーエルサイズK x K x C、出力チャネル数（フィルタ数）M、出力マップはH x W x Mとなる。ストライド1でパディングを適用した場合の畳み込み計算の計算量はH x W x K x K x C x Mとなる。MobileNetsはDepthwise ConvolutionとPointwise Convolutionの組み合わせで軽量化を実現した。

・Depthwise Convolutionは入力マップのチャネルごとに畳み込み（フィルタ数は1固定）を実施し、出力マップをそれらと結合する。出力マップは入力マップのチャネル数と同じになり、計算量はH x W x C x K x Kとなる。

・Pointwise Convolutionは1 x 1 x Cのカーネルで畳み込みを入力マップのポイントごとに実施する。出力マップのチャネル数はフィルタ数分だけ作成可能で、計算量はH x W x C x Mとなる。

・Depthwise Convolutionはチャネル毎に空間方向へ畳み込む。すなわちDk x Dk x 1のサイズのフィルターをそれぞれ用いて計算を行うため、その計算量はDk x Dk x C x H x W

・Depthwise Convolutionの出力をPointwise Convolutionによってチャネル方向に畳み込む。すなわち、出力チャネル毎に1 x 1 x Mサイズのフィルターをそれぞれ用いて計算を行うため、その計算量は H x W x C x Mとなる。

【DenseNet】（画像認識ネットワーク）

要点

・Densely Connected Convolutional Networksという論文でDense Convolutional Networkという畳み込みニューラルネットワークアーキテクチャの一種が紹介されている。

・Dense Blockと呼ばれるモジュールを用いて前ブロックで計算した出力に入力特徴マップを足し合わせる。

・増加チャネル数kをgrowth rateと呼び、kが大きくなるほどネットワークが大きくなるため、小さな整数に設定するのが良い。

・Dense blockの間をconvolutionとpoolingを行うtransition layerでつなぎ、チャネル数を減らす。

・DenseNetのDenseBlockでは前方の各層からの出力全てが後方の層への入力として用いられる。ResNetのResidualBlockでは前1層の入力のみ後方の層へ入力される。

・DenseBlockでは成長率（Growth Rate）と呼ばれるハイパーパラメータが存在する。ブロック毎にk個ずつ特徴マップのチャネル数が増加していく時kを成長率と呼ぶ。

【BatchNorm】（画像認識ネットワーク）

要点

・Bath Normはミニバッチに含まれるsampleの同一チャネルが同一分布に従うよう正規化し、Layer Normはそれぞれのsampleの全てのpixelsが同一分布に従うよう正規化し、Instance Normはさらにchannelも同一分布に従うよう正規化する。

・Batch Sizeが小さい条件下では、学習が収束しないことがあり、代わりにLayer Normalizationなどの正規化手法が使われることが多い。

・H x W x CのサンプルがN個あった場合に、N個の同一チャネルが正規化の単位となる。ミニバッチのサイズを大きく取れない場合には効果が薄くなってします。

【LayerNorm】（画像認識ネットワーク）

要点

・Layer NormはN個のサンプルのうち一つに注目する。H x W x Cの全てのpixelが正規化の単位となる。ミニバッチの数に依存しない。

・Layer Normは入力データや重み行列に対して、以下の操作を施しても出力が変わらないことが知られている。

・Instance Normは各サンプルのチャネル毎に正規化し、コントラストの正規化に寄与する。画像のスタイル転送やテクスチャ合成タスクなどで利用される。

【WaveNet】（音声生成モデル）

要点

・時系列データに対して畳み込み（Dilated convolution）を適用する。

・Dilated convolutionは層が深くなるにつれて畳み込むリンクを離す。

・深層学習を用いて結合確率を学習する際に、効率的に学習できるアーキテクチャを提案したことがWaveNetの大きな貢献の一つである。提案された新しいConvolution型アーキテクチャはDilated causal convolutionと呼ばれ、結合確率を効率的に学習できるようになっている。

・Dilated causal convolutionを用いた際の大きな利点は、単純なConvolution layerと比べてパラメータ数に対する受容野が広い。

【Seq2seq】

要点

・Seq2seqは系列(Sequence)を入力として系列を出力するもので、Encoder-Decoderモデルとも呼ばれる。入力系列がEncode（内部状態に変換）され、内部状態からDecode（系列に変換）する。

・RNN（系列データを読み込むために再帰的に動作するNN）と言語モデル（単語の並びに確率を与えるモデル）の理解が重要である。

・RNNの目標は事後確率を求めることである。

【Transformer】

要点

・Attentionは辞書オブジェクトであり、queryに一致するkeyを索引し、対応するvalueを取り出す操作であると見なすことができる。

・Attentionは文長が大きくなっても精度が落ちない。

・2017年6月にAttention is all you needという論文でTransformerモデルが発表された。RNNを使わず、必要なのはAttentionのみである。

・単語ベクトルに単語の位置を追加し、複数のヘッドでDot Product Attentionを行い、単語の位置ごとに独立処理する全結合をし、未来の単語を見ないようにマスクする。

・入力を全て同じにして学習的に注意箇所を決めていくのがSelf-Attentionの特徴である。

・Multi-Head attentionは重みパラメタの異なる8個のヘッドを使用する。

・DecoderはEncoderと同じく6層だが、各層で二種類の注意機構がある。

・入出力の差分を学習させるAdd (Residual Connection)（学習・テストエラーの低減）と各層においてバイアスを除く活性化関数への入力を平均0、分散1に正則化するNorm (Layer Normalization)（学習の高速化）を実装してある。

【物体検知】

要点

・代表的なデータセットとしてVOC12、ILSVRC17、MS COCO18、OICOD18などがある。BOX/画像が大きい方がより実生活に則する。

・分類問題における評価指標はConfusion Matrix（混同行列）からPrecision TP/(TP+FP)やRecall TP/(TP+FN)として読み取る。Confidenceの閾値を変化させてもconfusion matrixに入るサンプル数の合計数は変化しないが、物体検出の場合はサンプル数自体が異なってくる。

・物体検出においてはクラスラベルだけでなく物体位置の予測精度の指標としてIoU (Intersection over Union)が導入された。別名Jaccard係数とも呼ばれ、IoU = TP / (TP + FP + FN)で表される。

・AP (Average Precision)はconfidenceの閾値をβとするとき、Recall = R (β), Precision = P(β)となるのでP = f(R) [Precision-Recall curve]であることから、（PR曲線の下側面積）となる。クラス数がCの時、となる。

・MS COCOで導入された指標は0.5で固定していたIoUを0.5から0.95まで0.05刻みでAP&mAPを計算し算術平均を計算する。

・最近はFPS (Flames per Second)という1秒間あたりの処理フレーム数（検出速度）も指標として利用される。

・inference time (ms)は1推論にかかる時間を示す指標である。

・2012年のAlexNetの登場によりSIFTからDCNNに移行した。

・2段階検出器(Two-stage detector)は候補領域の検出とクラス推定を別々に行うので相対的に精度が高いが計算量が大きく推論も遅い傾向がある。

・１段階検出器(One-stage detector)は候補領域の検出とクラス推定を同時に行うので相対的に精度が低いが計算量が小さく推論も早い傾向にある。

・SSD (Single Shot Detector)はVGG16を改良したネットワークアーキテクチャーである。

【セグメンテーション】

要点

・Semantic SegmentationではConvolutionやPoolingにより解像度が落ちていく。どのようにして元の解像度に戻すのかが課題であり、元の解像度に戻す手法をup-samplingという。

・Deconvolution/Transposed convolutionでは特徴マップのpixel間隔をstrideだけ空け、特徴マップの周りに(kernel size -1) -paddingだけ余白を作り畳み込み演算を行う。

・poolingによりローカルな情報（輪郭等）が失われていくので、底レイヤーpooling層の出力をelement-wise additionすることでローカルな情報を補完してからup-samplingする。

・pooling時の位置情報を保持しておき(switch variables)、unpoolingを行い、解像度を上げることができる。

・poolingは受容野を広げるために必要だが、Convolutionの段階で受容野を広げる工夫としてdilated convolutionという手法が用いられる。

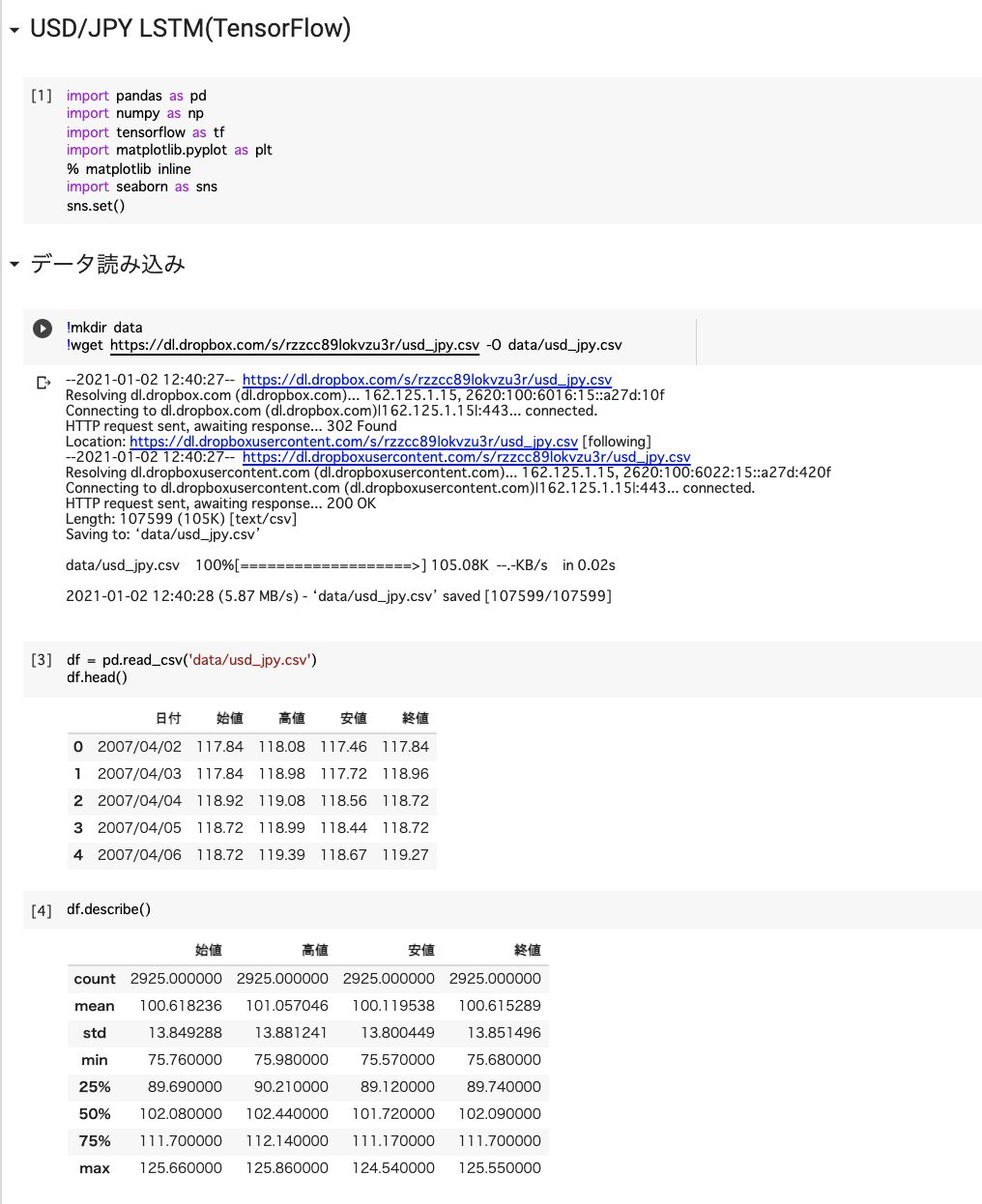
【為替データの扱い】

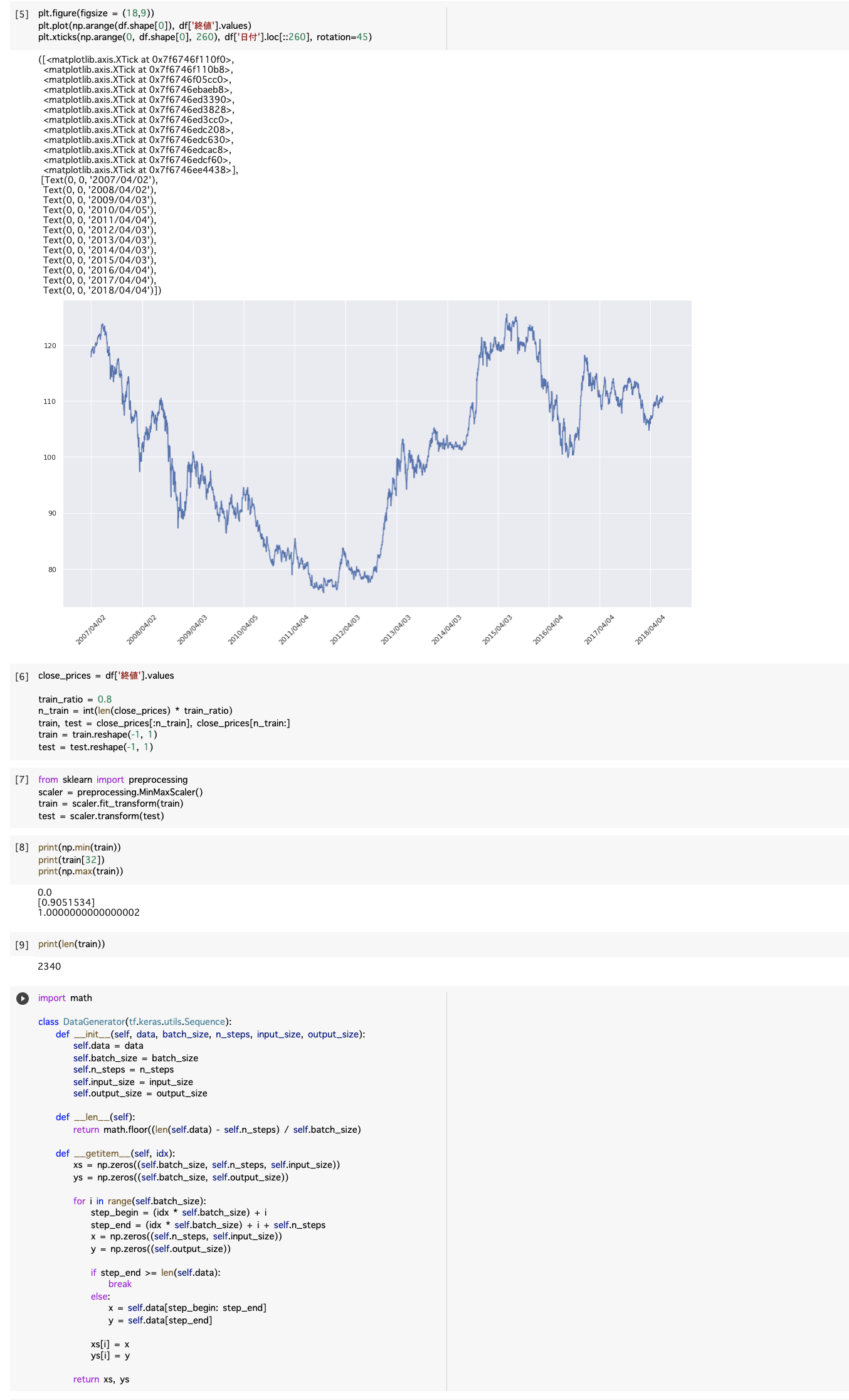
要点

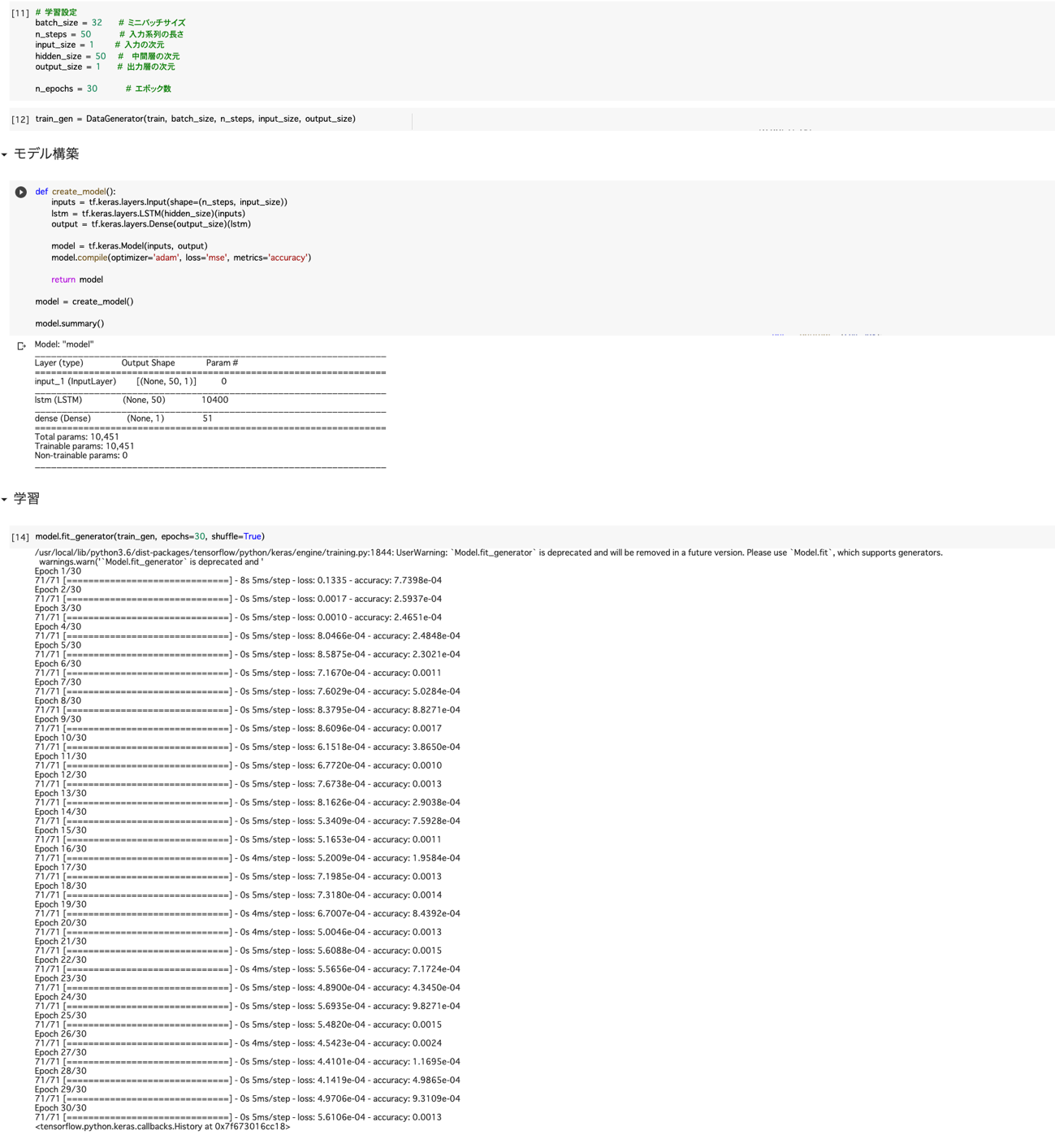
・為替の始値、高値、安値、終値のデータを読み込ませ、ミニバッチ処理（ミニバッチ数32）をして終値を予測するモデルを作成する。

・モデルはLSTMを利用し、optimizerとしてAdamを使用し、エポック数30の学習をさせている。

演習問題レポート









為替データを読み込んで予測できるようになった。

<https://study-ai.com/jdla/>

