27/03/22

Mare with ML

- np.eye → identity matrix
- torch.index_select

Foundations

* Linear Regression

Linear model ŷ that models y given X using weights W and bias b

$$\hat{y} = XW + b$$

Objective:
Line of best fit that minimizes the distance between the predicted y (model's output) and target (ground truth) values.

x The model only performs well for classification when the data is linearly separable

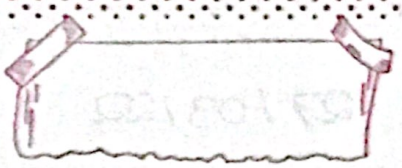
Loss: MSE

* Logistic regression

$$\hat{y} = \frac{e^{X \cdot W_y}}{\sum_j e^{XW}}$$ * bias excluded

np.bincount = pd.value_counts

My playlist today. #




✓ Can predict class probabilities
✗ sensitive to outliers, since its objective is to minimize cross-entropy loss

sigmoid → binary
softmax → multiclass

Loss: Cross-entropy

* Neural Networks
✓ Can models non linear patterns very well
✗ Overfits easily
- predict the probability of class y given inputs X
✗ not easily interpretable

* Convolutional Neural Networks (CNN)
Ct the core of CNNs are filters (aka. weights, Kernels, etc) which convolve (slide) across an input to extract relevant features
The filters are randomly initiated.

Objetive: extract ~~relevant features~~ meaningful spatial substruct from encoded data.

✓ small numbers of weights (shared)
✓ parallelizable
✓ detects spatial substructures (feature extractors)
✓ interpretability via filters

✓ can be used in images, text, time-series
✗ many hyperparameters (Kernel size, stride, etc to tune)

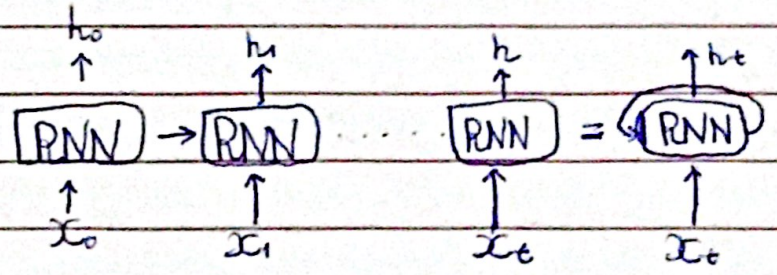
* a foundation
* constantly updated

Poding
The resulting of convolving filters on an input is a feature map. Poding is a way to summarize to a low dimension through max, average, etc.

Batch Normalization
Operation that will standarize an input through mean and std so our model can optimize with larger learning rates

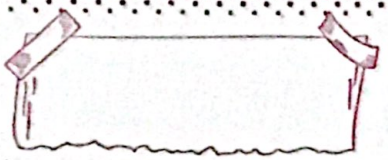
*Recurrent Neural Network (RNN)
We can also process an input sequentially. We can think of each token in an text as an event in time (timestep)

$$h_t = \tanh(W_{hh} h^{t-1} + W_{xh} x_t + b_h)$$



RNN foward for a single time step X_t

My playlist today: #




Objective

Process sequential data by accounting for current input and also what has been learned from previous inputs.

✓ Account for order and previous inputs in a meaningful way
✓ Conditioned generation for generating sequences
× difficult to parallelize, depends on previous steps
× Processing long sequence can yield memory and computation issue
× interpretability is difficult

Gated RNNs — LSTM
Gated RNNs — GRU