# Preliminaries

**Linear algebra**

Symmetric matrix:

Positive definite: for all non-zero and A symmetric

If is a matrix and is a vector, then the i-th element of is given by

这特么我需要什么例子啊，就是直观的方程组。代表的就是第几个方程式。

**Vector calculus**

Let and be a vector, and a scalar

Derivatives with respect to a vector are defined by

scalar-vector

vector-vector

**Normal distribution**

**µ** is a D-dimensional mean vector

**∑** is a D×D covariance matrix (symmetric, positive definite)

|∑| is the determinant of ∑

# Linear models

Root mean square (RMS) error

Regularization

Linear basis function models

Maximum likelihood

with regularization,

Bias-Variance Decomposition

**Bayesian linear regression**

**Exercise 3.7**

From Bayes’ theorem we have

The first exponential corresponds to the posterior, unnormalized Gaussian distribution over w, while the second exponential is independent of w and hence can be absorbed into the normalization factor.

**Exercise 3.8**

Combining the prior

And the likelihood

Where , we obtain a posterior of the form

We can expand the argument of the exponential, omitting the -1/2 factors, as follows

Where const denotes remaining terms independent of w.

From this we can read off the desire result directly.

With

And

# Classification

Activation function

logistic sigmoid function

Logit function

SoftMax function

Cross entropy

# Neural Networks

ReLu

Leaky ReLu

Invariances

# Deep neural networks

**Bias and variance**

**Overfitting**

Regularization

Dropout regularization

Intuition

Early stopping

Stop at smallest error with validation data

**Vanishing / Exploding gradients**

Use linear activation function

**Optimization algorithms**

Mini-batch gradient descent

Batch gradient descent

Stochastic gradient descent

Adam optimization

# Convolutional Neural Networks

Edge detection

Filter

Padding

Input:

Filter:

Output:

Input:

Filter:

Padding:

Output:

**Convolutions**

No padding

Output is

Same convolution

Padding

Output

Image:

Filter:

Padding:

Stride:

Output:

Filter sizes smallest possible filter to capture 4 directions

1 larger filter can be replaced by a deeper stack of successive smaller filters.

Parameters

#of parameters of a filter: 27+1 bias = 28

Independent of the size of the picture

Notation for layer l

**Pooling**

f=2, s=2

Max pooling

Average pooling

Why do convolutions work?

Parameter sharing

A feature detector that is useful is in one part of the image is probably useful in another part of the image.

Sparsity of connections

In each layer, each output value depends only on a small number of inputs.

ResNet

Residual block

* AlexNet 是现代深度学习的先锋，证明了深度神经网络在图像分类任务上的有效性。
* VGG 证明了深度和简单性可以并存，通过叠加小的卷积核可以实现很深的网络。
* ResNet 解决了深度学习中的退化问题，使得我们可以训练上百层的网络。
* Inception Networks 关注于如何有效地增加网络的宽度。

**Objection detection**

Bounding box

Object localization ,0,1,0

Unique detections: Non-max suppression algorithm

Overlapping objects: Anchor box algorithm

**Face verification vs recognition**

**Verification**

Input image, name/ID

Output whether the input image is that of the claimed person

**Recognition**

Has a database of K persons

Get an input image

Output ID if the image is any of the K persons (or not recognized)

One-short learning

Learn a similarity function

**Siamese Network**

Parameters of NN define an encoding

Triplet loss function

**Neural style transfer**

Cost function

# **Recurrent Neural Networks**

Name entity recognition

输入输出长度不同

不同位置不共享特征

Recurrent neural networks

One to one:

One to many: Speech recognition, Music generation

Many to one: Sentiment classification

Many to many:

1. DNA sequence analysis, machine translation, name entity recognition
2. video recognition

language modeling

A, X, Y

Basic units in RNNs

RNN unit

Gated recurrent unit (GRU)

Fully-gated recurrent unit



Long Short Term Memory (LSTM) unit



**Word embeddings**

Similarity function

**Word2vec**

**skip gram model: word2vec variant**

**negative sampling**

# Hidden Markov Models

# Kernel methods

Memory-based methods

Fast to train, slow in predictions

**Kernel trick**

Any algorithm in which the examples only appear as dot products can be kernelized by replacing by

If we use algorithms that only depend on the Gram-matrix, then we never have to know (compute) the actual features.

**Kernels**: map data to higher dimensions where it exhibits linear patterns

We did not have to define the mapping

Not any function can be used

**Mercer’s condition**: The Gram matrix K must be a positive semidefinite matrix

**Sparse kernel machines**

Distance of a point to decision surface:

Optimization problem

最小化所有点到**w**的距离

For the closest point to the decision surface, set

Optimization problem now becomes to maximize , or minimize

SVM后面不看了，太复杂了，没时间看了。

Reinforcement Learning 不考暂时也不看了。

复习就先到这里了。