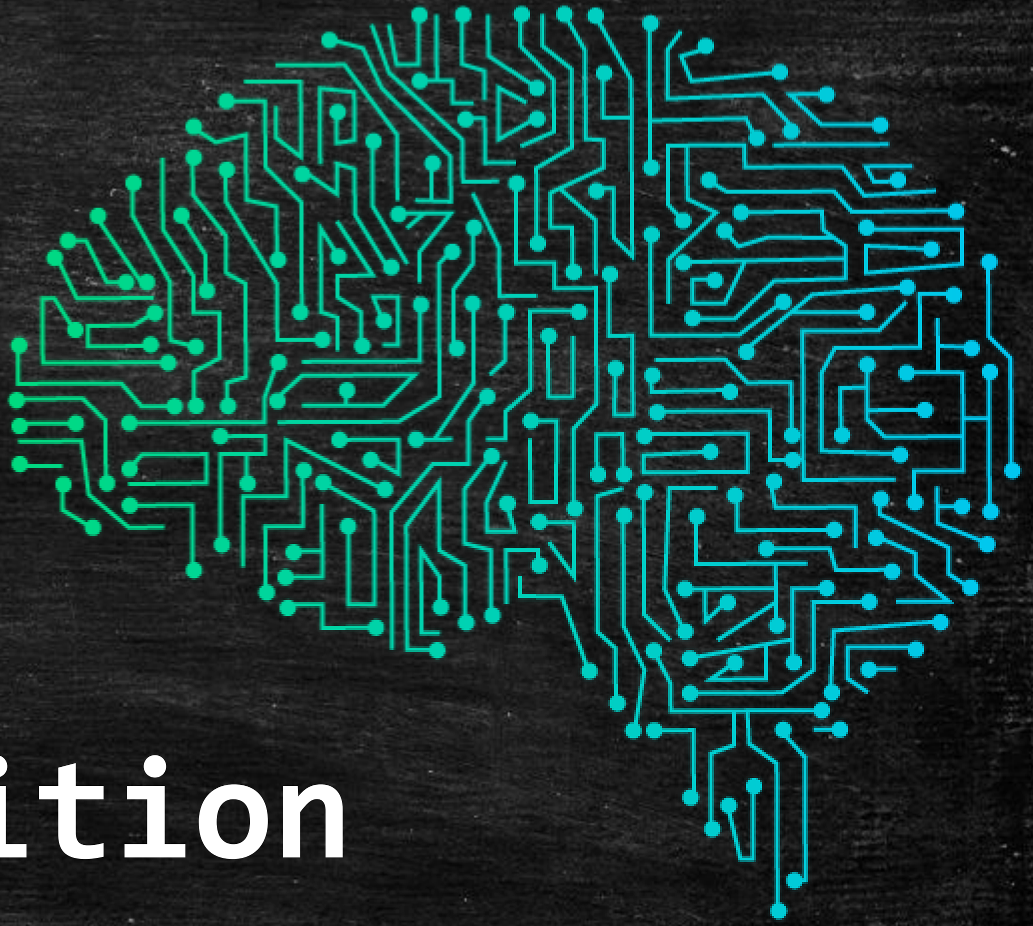


Speech Recognition

Sajad Dadgar

Topics

- Introduction
- Audio Signal
- Feature Extraction
- Convolutional Neural Network
- Implementation

Introduction

Introduction

- History

1950

Speech Recognition were limited to a single speaker and had limited vocabularies of about a dozen words.

1980

understanding that speech is accompanied by noise and distractors.

2010

Revolution in voice recognition with google's DNN-based voice search, apple's siri, microsoft's Cortana. Usable voice recognition running on powerful hardware.

Audio Signal

Audio Signal

- What is an Audio Signal?

- Parameters of an Audio Signal

Amplitude

Crest & Trough

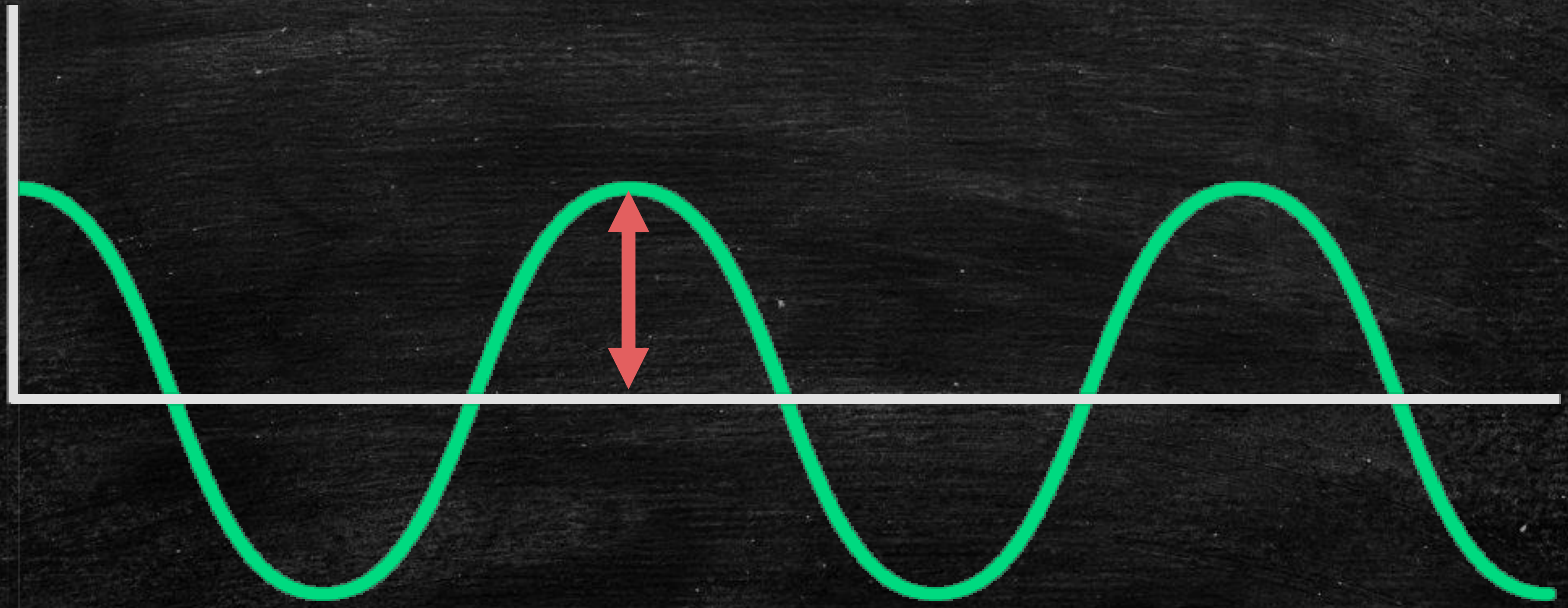
Wave Length

Frequency

Cycle

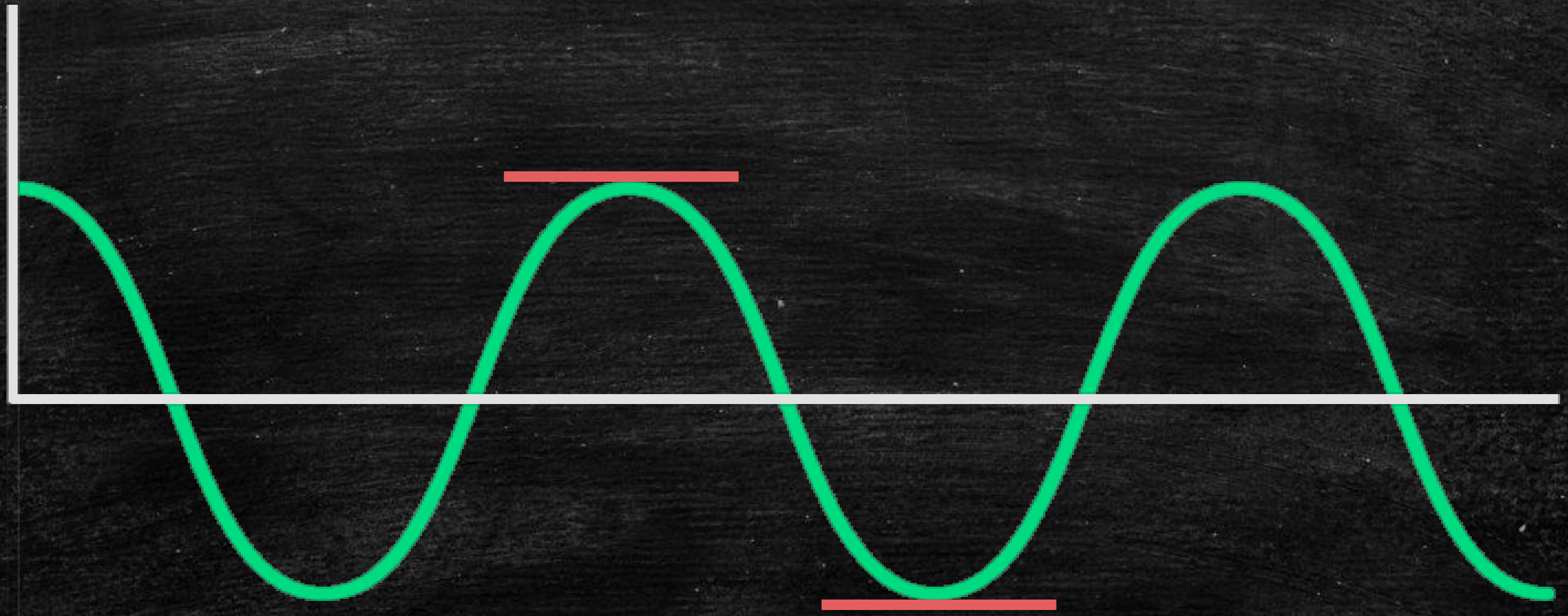
Audio Signal

- Amplitude



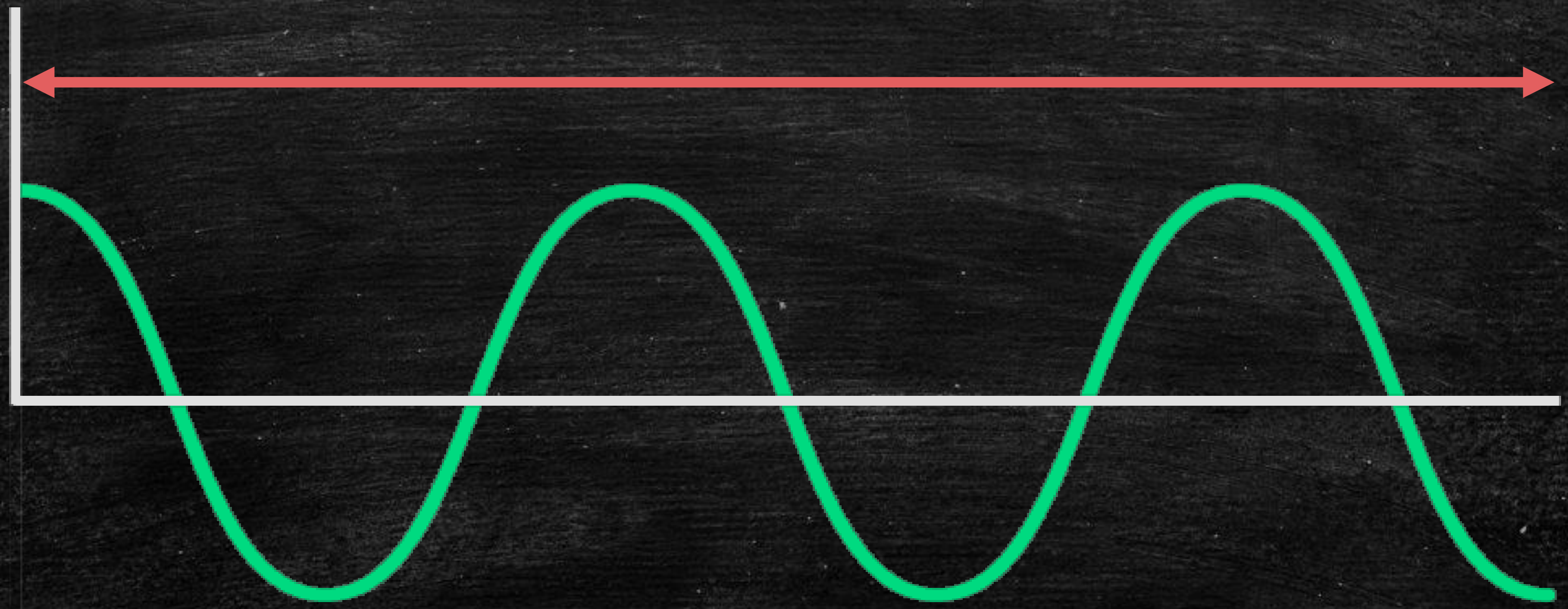
Audio Signal

- Crest & Trough



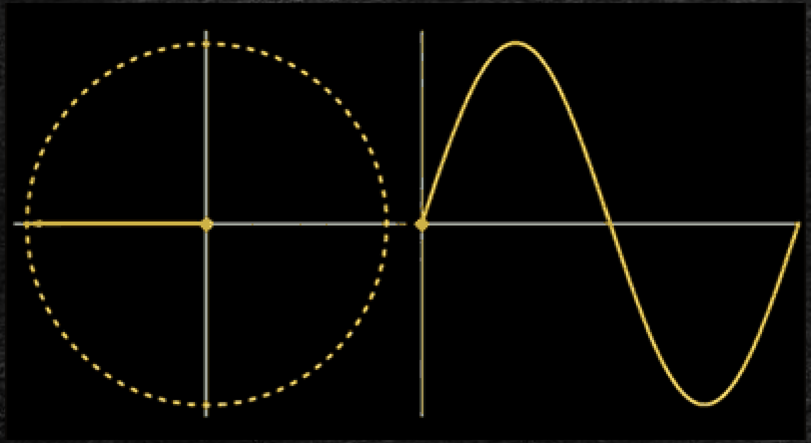
Audio Signal

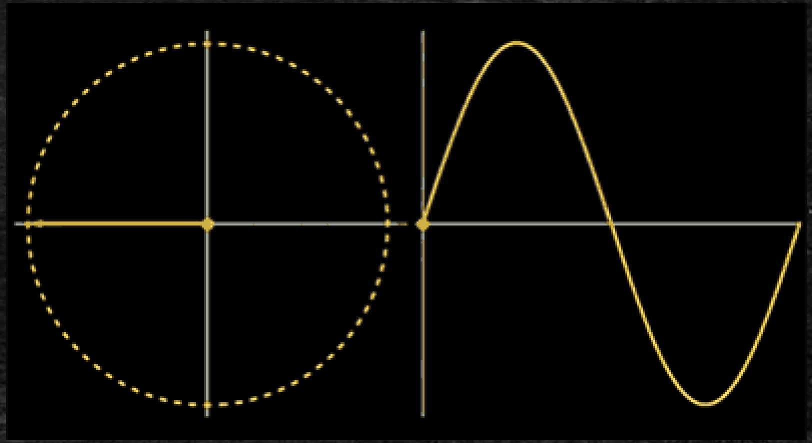
- Wave Length



Audio Signal

- Frequency & Cycle

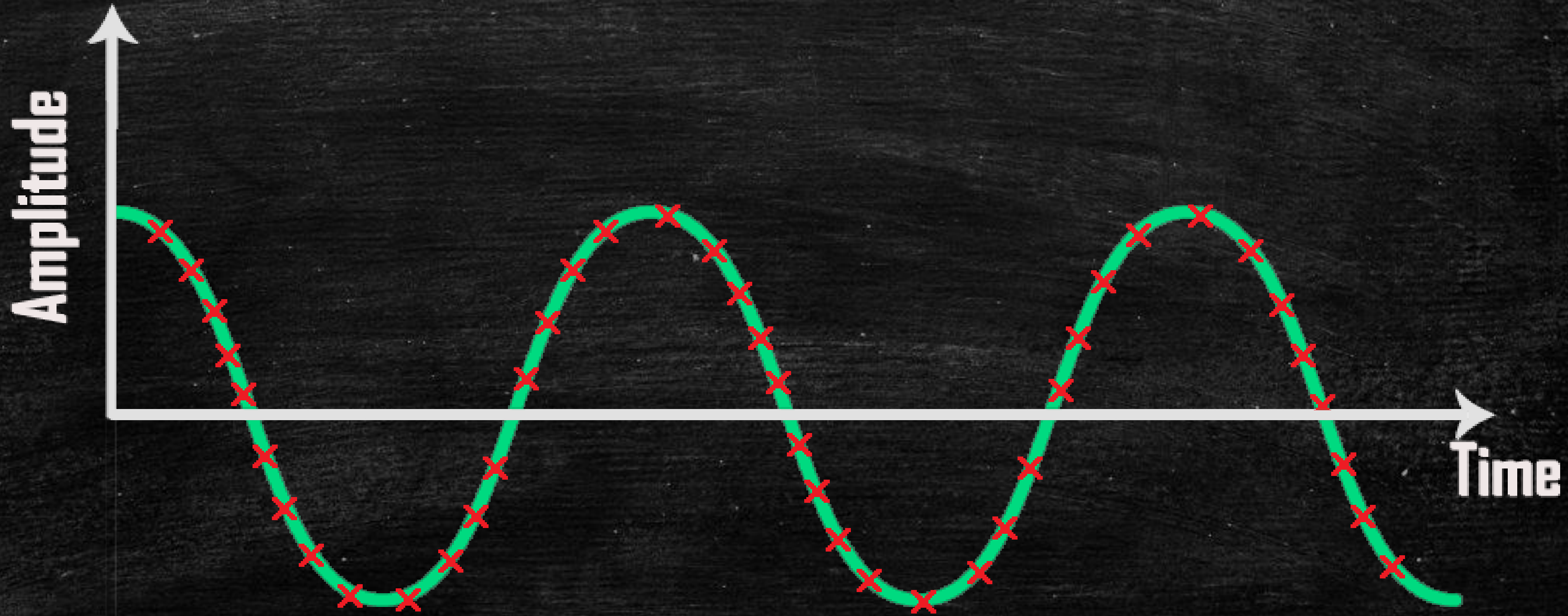




Audio Signal

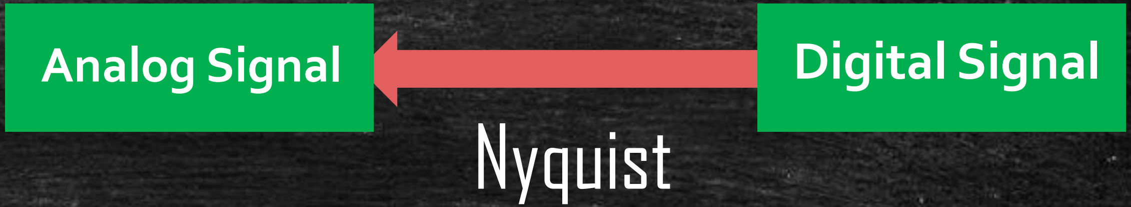
- Types of Signal

Analog Signal → Digital Signal



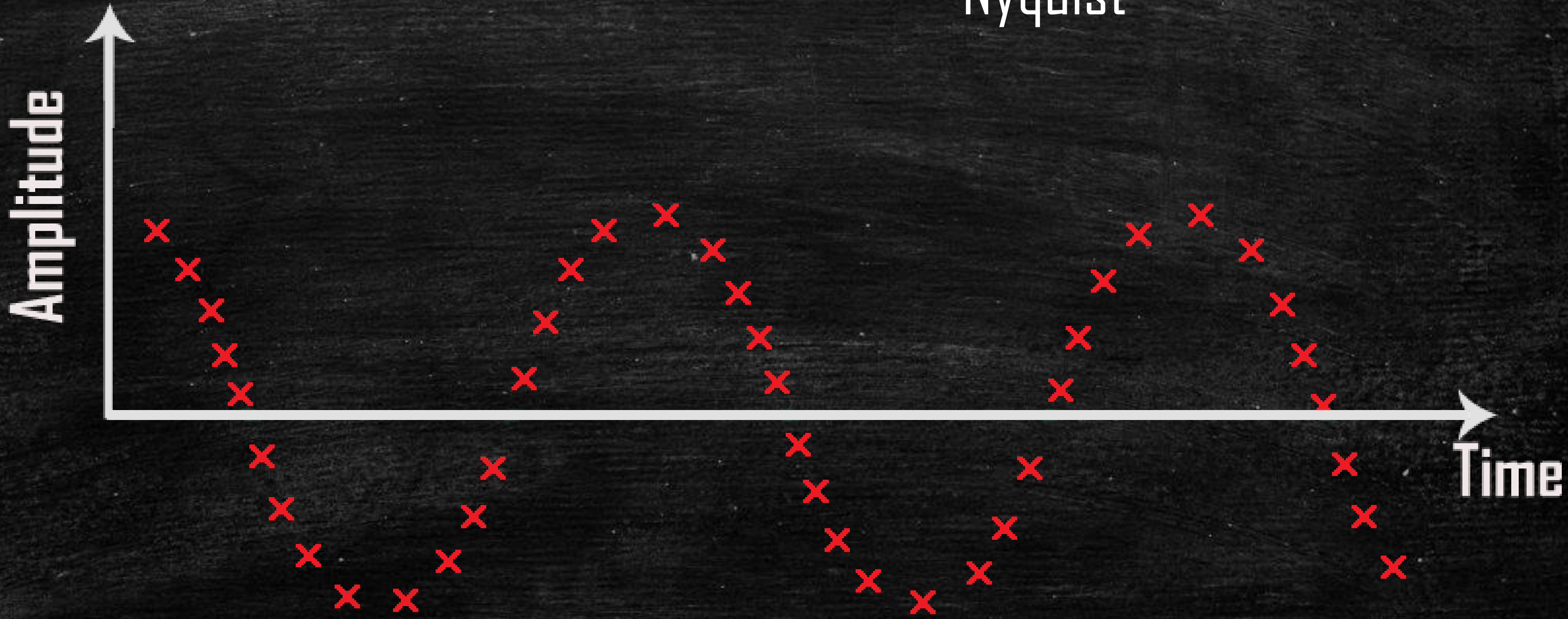
Audio Signal

- Types of Signal


Analog Signal

Digital Signal

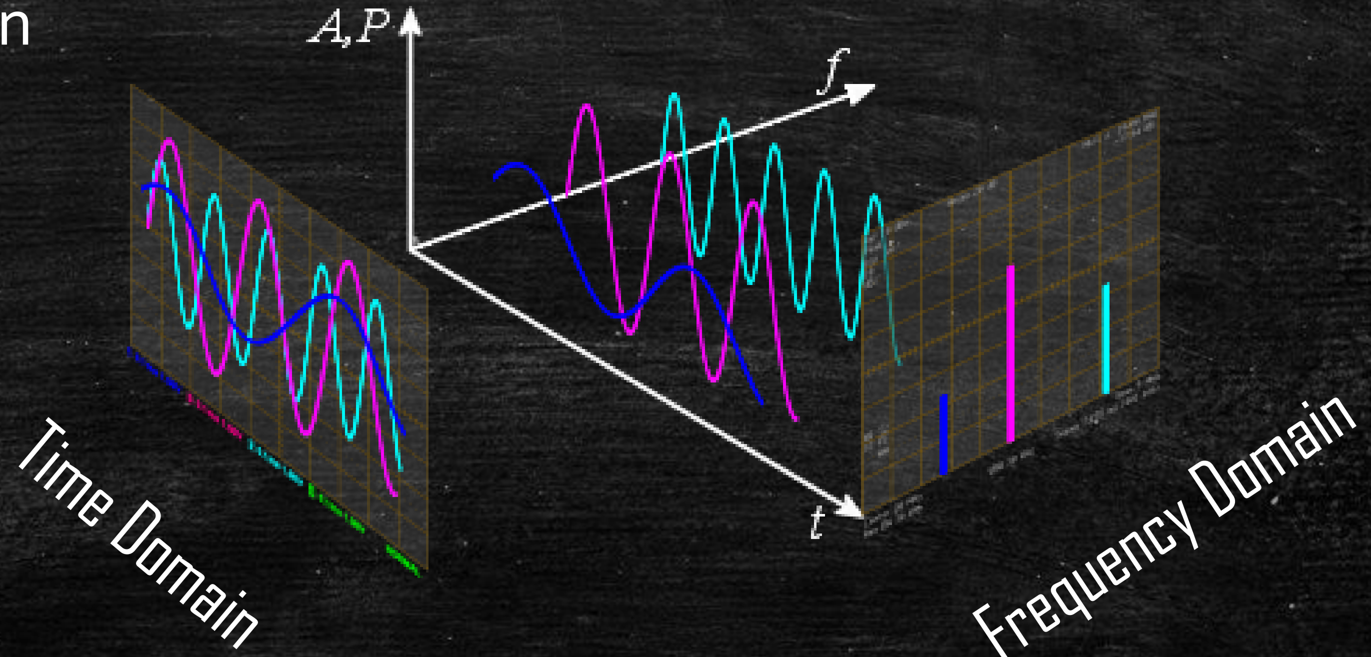
Nyquist



Feature Extraction

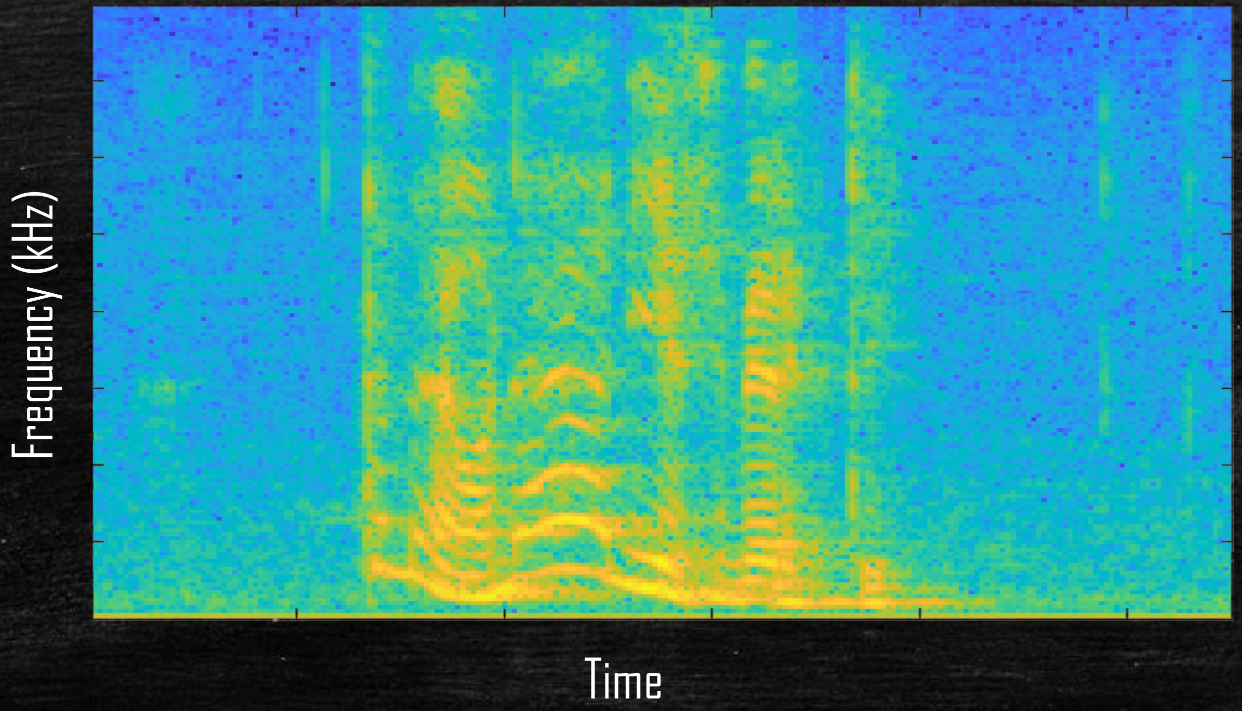
Feature Extraction

- Time Domain
- Frequency Domain



Feature Extraction

- Spectrogram



Feature Extraction

- Mel Frequency Cepstrum Coefficient (MFCC)

- Frame the signal into short frames.

- For each frame calculate the periodogram estimate of the power spectrum.

- Apply the Mel Filterbank to the power spectra

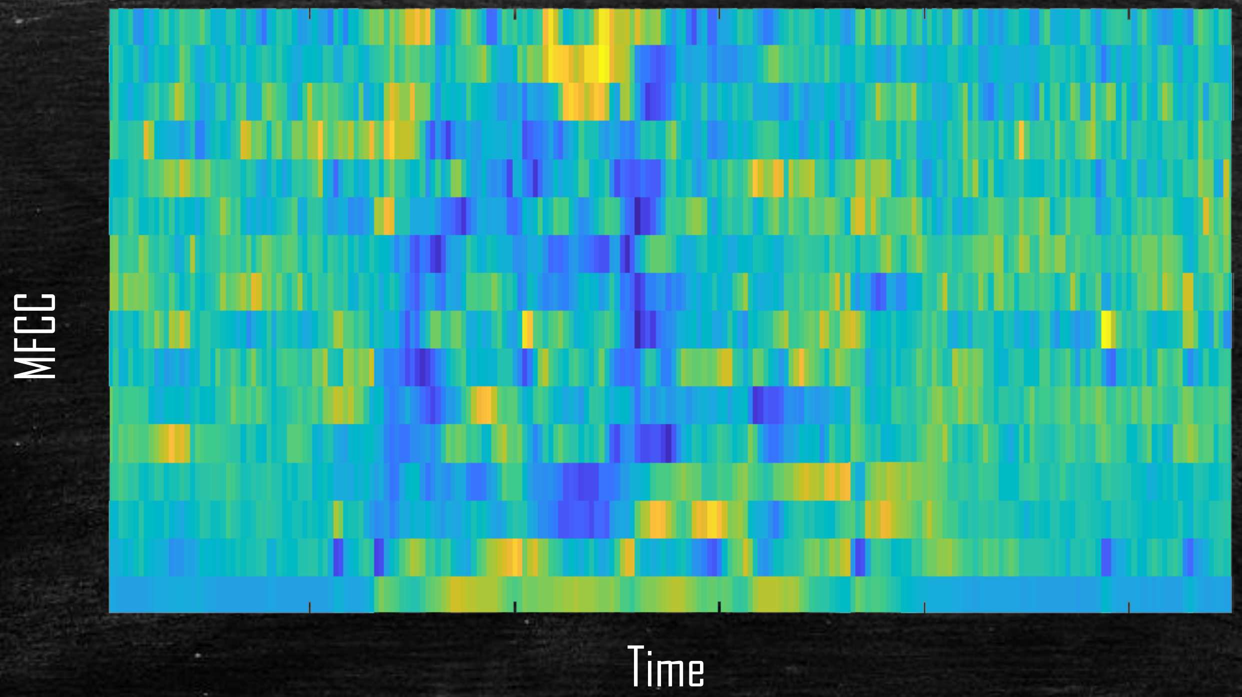
$$M = 2595log_{10}\left(1 + \frac{f}{700}\right)$$

$$f = 700\left(10^{\frac{m}{2595}} - 1\right)$$

- Take the DCT of the filterbank energies.

Feature Extraction

- Mel Frequency Cepstrum Coefficient (MFCC)



Feature Extraction

- MFFC

Speech Recognition using MFCC

https://pdfs.semanticscholar.org/3439/454a00ef811b3a244f2b0ce770e80f7bc3b6.pdf

Website:

https://wiki.aalto.fi/display/ITSP/Cepstrum+and+MFCC

Convolutional Neural Network

Convolutional Neural Network

- Convolution
- Pooling
- Flattening
- Full Connection

Convolutional Neural Network

- Convolution

0	0	0	0	0	0	0
0	1	0	0	0	1	0
0	0	0	0	0	0	0
0	0	0	1	0	0	0
0	1	0	0	0	1	0
0	0	1	1	1	0	0
0	0	0	0	0	0	0

Input image

⊗

0	0	1
1	0	0
0	1	1

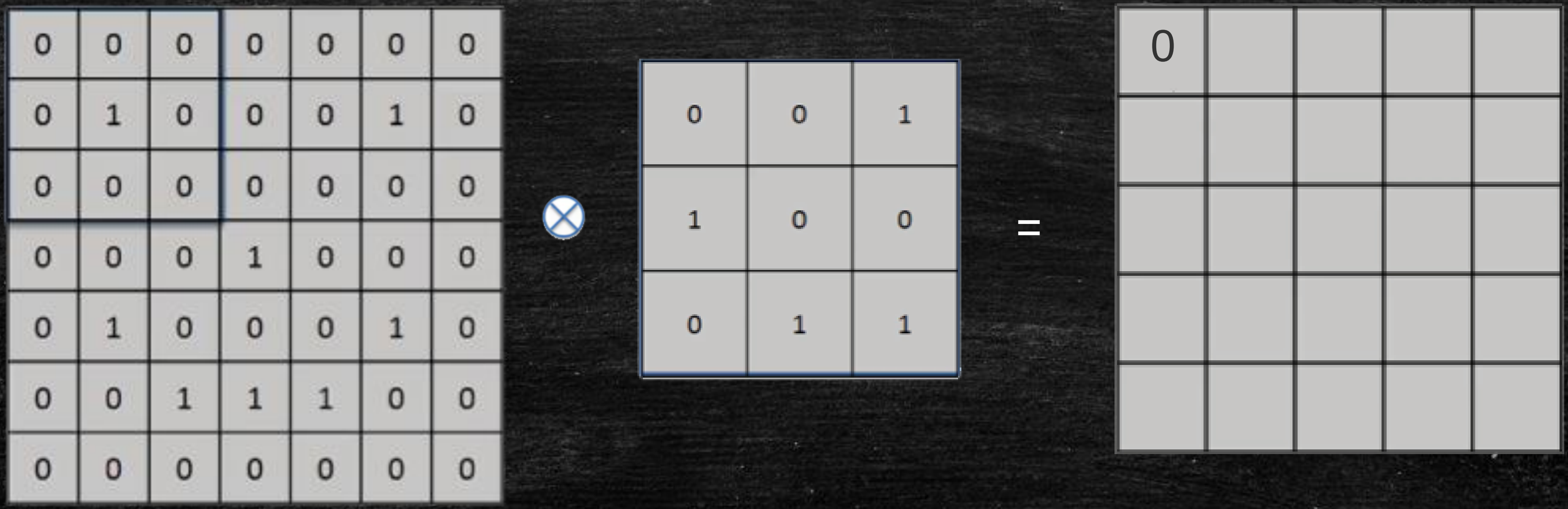
Filter

=

Feature map

Convolutional Neural Network

- Convolution



Input image

0	0	0	0	0	0	0
0	1	0	0	0	1	0
0	0	0	0	0	0	0
0	0	0	1	0	0	0
0	1	0	0	0	1	0
0	0	1	1	1	0	0
0	0	0	0	0	0	0

⊗

Filter

0	0	1
1	0	0
0	1	1

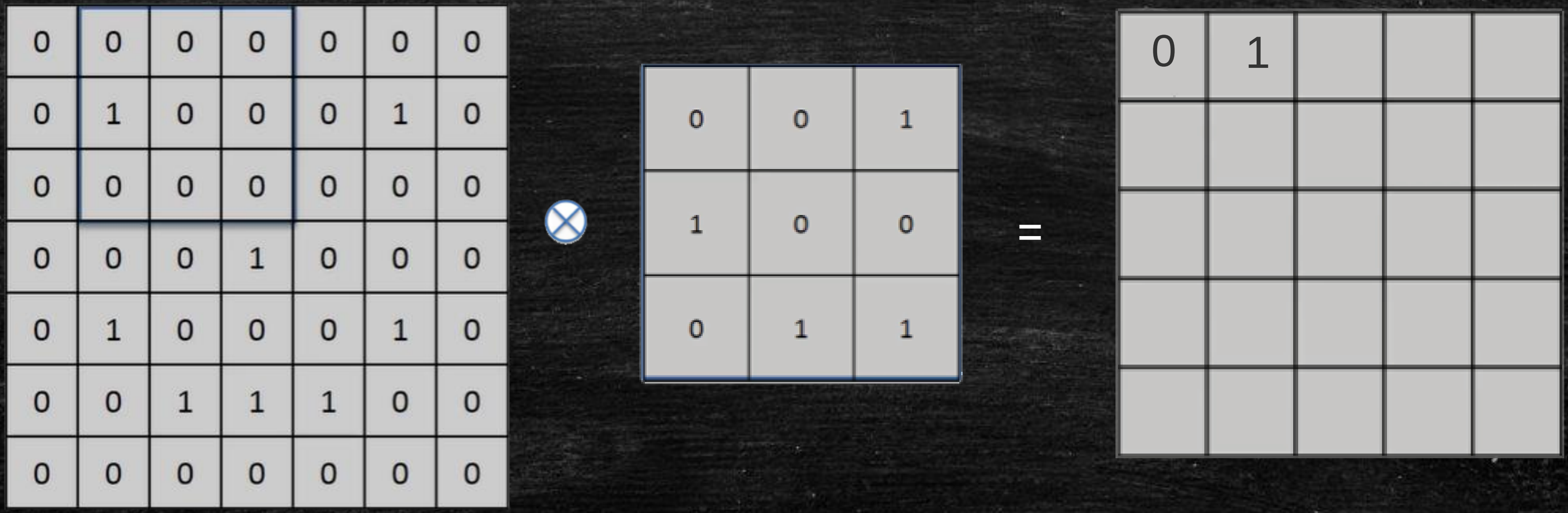
=

Feature map

0				

Convolutional Neural Network

- Convolution



0	0	0	0	0	0	0
0	1	0	0	0	1	0
0	0	0	0	0	0	0
0	0	0	1	0	0	0
0	1	0	0	0	1	0
0	0	1	1	1	0	0
0	0	0	0	0	0	0

Input image

⊗

0	0	1
1	0	0
0	1	1

Filter

=

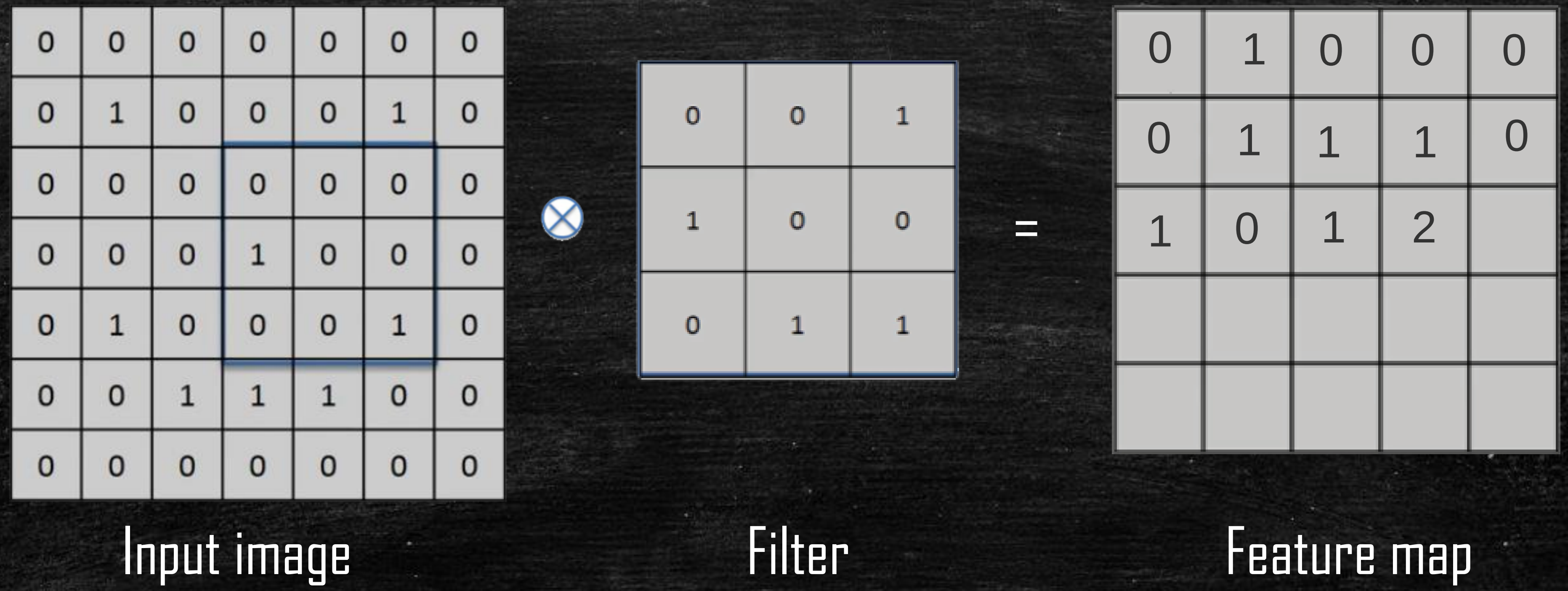
0	1			

Feature map

Convolutional Neural Network

- Convolution

$$(0\times 0)+(0\times 0)+(0\times 1)+(1\times 1)+(0\times 0)+(0\times 0)+(0\times 0)+(0\times 1)+(1\times 1)=2$$



0	0	0	0	0	0	0
0	1	0	0	0	1	0
0	0	0	0	0	0	0
0	0	0	1	0	0	0
0	1	0	0	0	1	0
0	0	1	1	1	0	0
0	0	0	0	0	0	0

Input image

⊗

0	0	1
1	0	0
0	1	1

Filter

=

0	1	0	0	0
0	1	1	1	0
1	0	1	2	

Feature map

Convolutional Neural Network

- Convolution

0	0	0	0	0	0	0
0	1	0	0	0	1	0
0	0	0	0	0	0	0
0	0	0	1	0	0	0
0	1	0	0	0	1	0
0	0	1	1	1	0	0
0	0	0	0	0	0	0

Input image

⊗

0	0	1
1	0	0
0	1	1

Filter

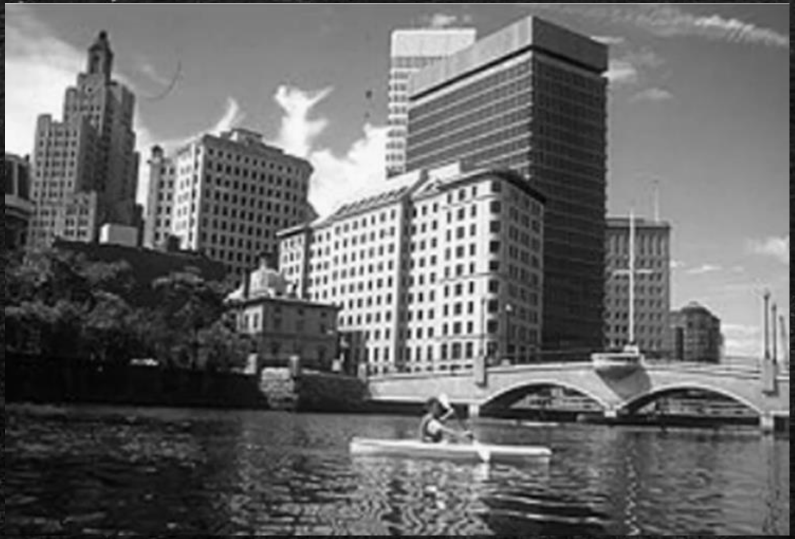
=

0	1	0	0	0
0	1	1	1	0
1	0	1	2	1
1	4	2	1	0
0	0	1	2	1

Feature map

Convolutional Neural Network

- Convolution



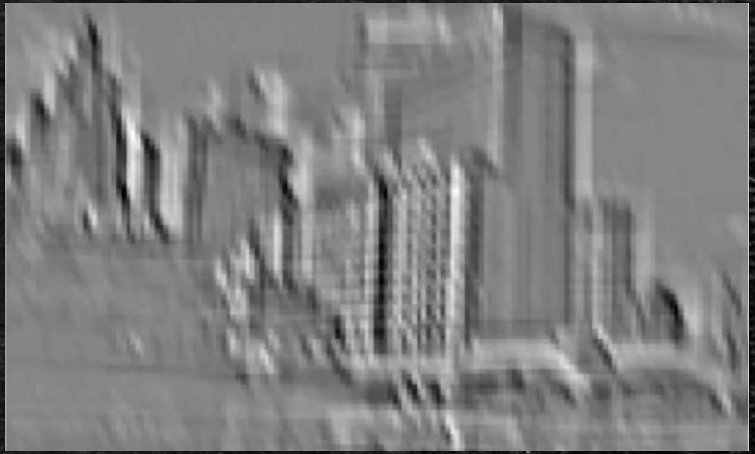
Input image

⊗

1	0	-1
2	0	-2
1	0	-1

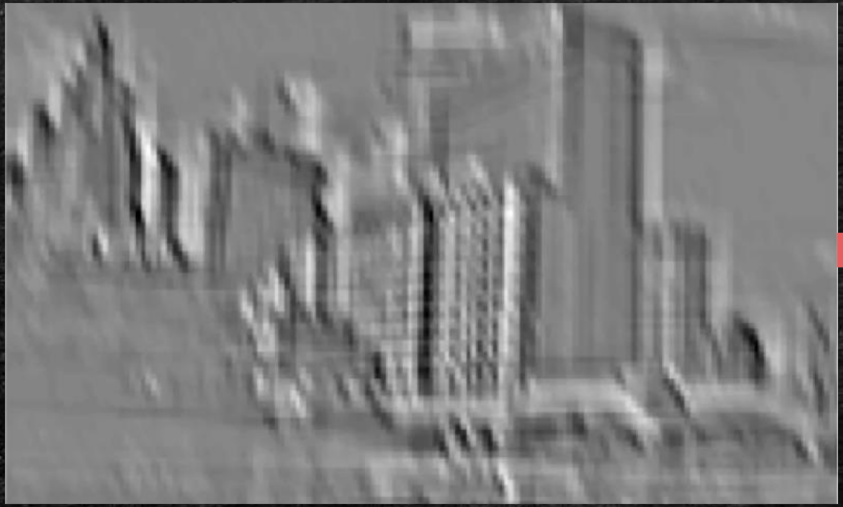
Filter

=



Convolutional Neural Network

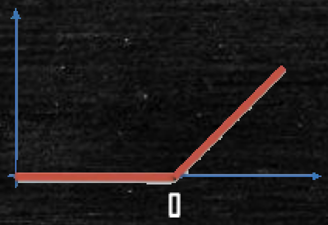
- Convolution

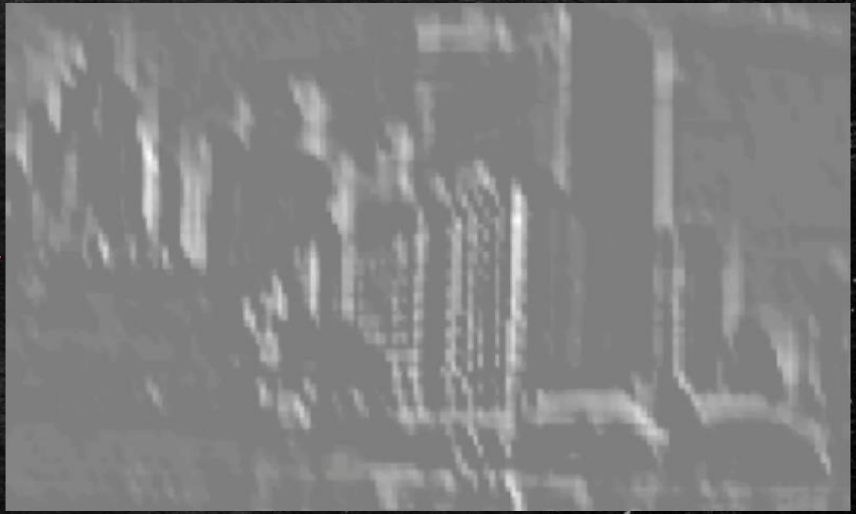


Black: Negative

White: Positive

Rectifier





Convolutional Neural Network

- Convolution

Understanding Convolutional Neural Network with A Mathematical Model (2016):

http://arxiv.org/pdf/1609.04112.pdf

Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification (2015):

https://arxiv.org/pdf/1502.01852.pdf

Convolutional Neural Network

- Pooling

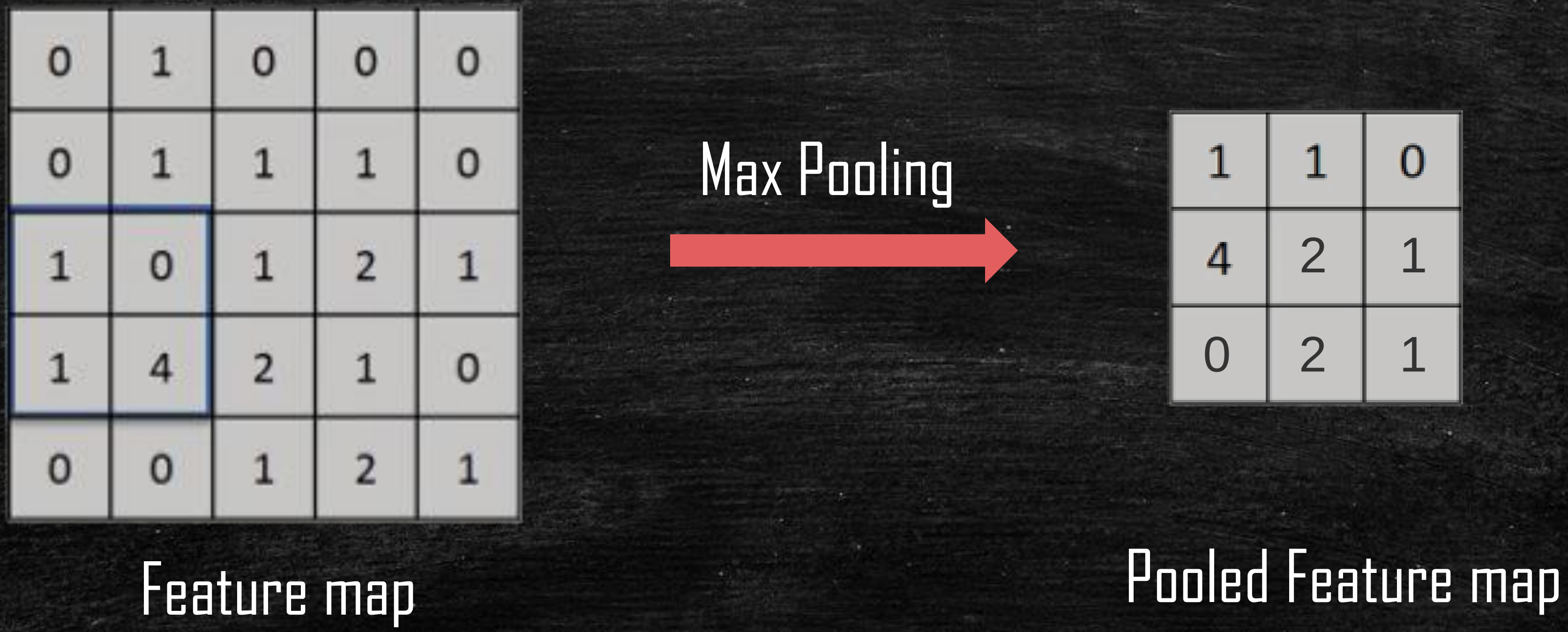
  Max Pooling

  Sum Pooling

  Mean Pooling

Convolutional Neural Network

- Pooling



0	1	0	0	0
0	1	1	1	0
1	0	1	2	1
1	4	2	1	0
0	0	1	2	1

Feature map

Max Pooling

1	1	0
4	2	1
0	2	1

Pooled Feature map

Convolutional Neural Network

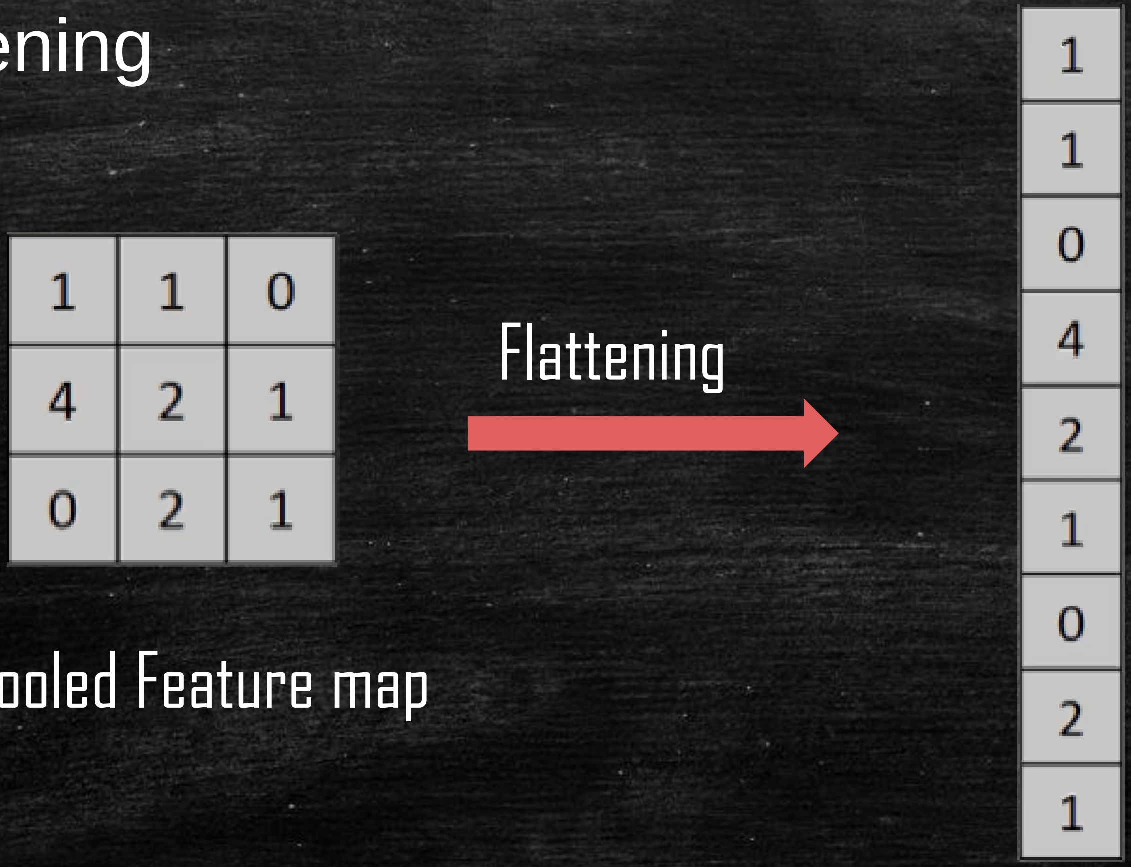
- Pooling

Evaluation of Pooling Operations in Convolutional Architectures for Object Recognition (2010):

http://ais.uni-bonn.de/papers/icann2010_maxpool.pdf

Convolutional Neural Network

- Flattening

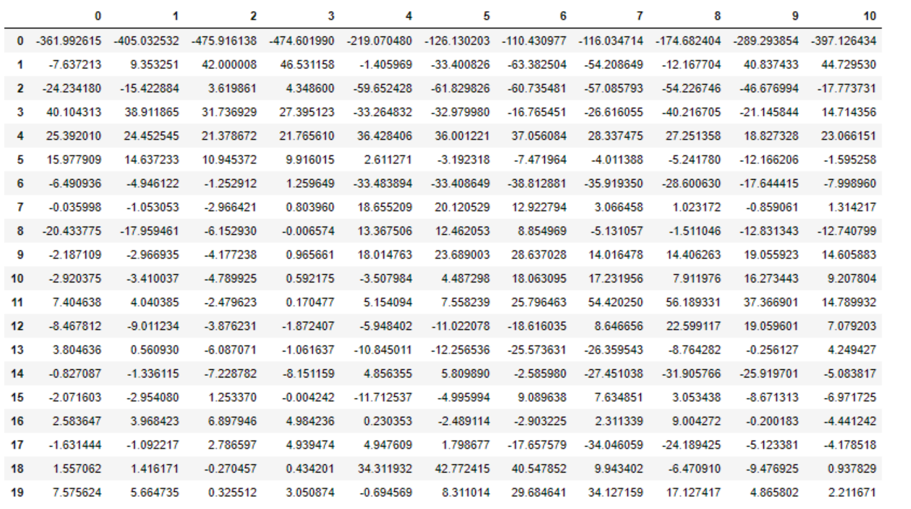


1	1	0
4	2	1
0	2	1

1
1
0
4
2
1
0
2
1

Pooled Feature map

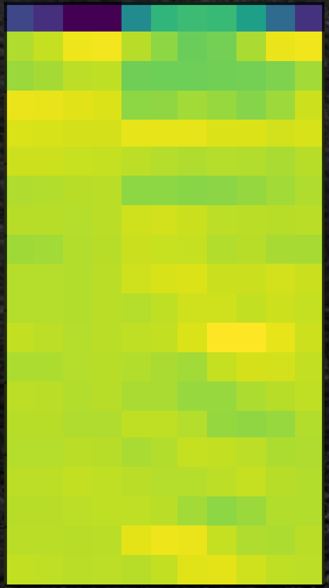
Convolutional Neural Network



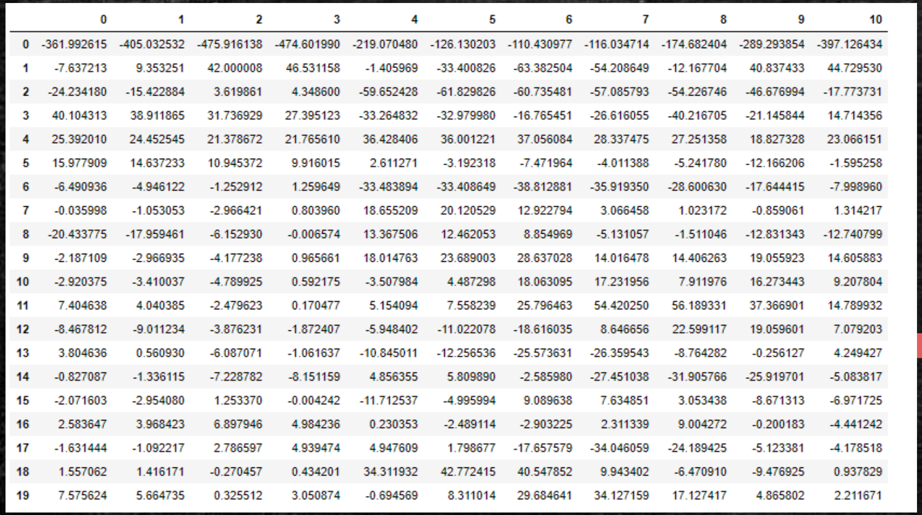
	0	1	2	3	4	5	6	7	8	9	10
0	-361.992615	-405.032532	-475.916138	-474.601990	-219.070480	-126.130203	-110.430977	-116.034714	-174.682404	-289.293854	-397.126434
1	-7.637213	9.353251	42.000008	46.531158	-1.405969	-33.400826	-63.382504	-54.208649	-12.167704	40.837433	44.729530
2	-24.234180	-15.422884	3.619861	4.348600	-59.652428	-61.829826	-60.735481	-57.085793	-54.226746	-46.676994	-17.773731
3	40.104313	38.911865	31.736929	27.395123	-33.264832	-32.979980	-16.765451	-26.616055	-40.216705	-21.145844	14.714356
4	25.392010	24.452545	21.378672	21.765610	36.428406	36.001221	37.056084	28.337475	27.251358	18.827328	23.066151
5	15.977909	14.637233	10.945372	9.916015	2.611271	-3.192318	-7.471964	-4.011388	-5.241780	-12.166206	-1.595258
6	-6.490936	-4.946122	-1.252912	1.259649	-33.483894	-33.408649	-38.812881	-35.919350	-28.600630	-17.644415	-7.998960
7	-0.035998	-1.053053	-2.966421	0.803960	18.655209	20.120529	12.922794	3.066458	1.023172	-0.859061	1.314217
8	-20.433775	-17.959461	-6.152930	-0.006574	13.367506	12.462053	8.854969	-5.131057	-1.511046	-12.831343	-12.740799
9	-2.187109	-2.966935	-4.177238	0.965661	18.014763	23.689003	28.637028	14.016478	14.406263	19.055923	14.605883
10	-2.920375	-3.410037	-4.789925	0.592175	-3.507984	4.487298	18.063095	17.231956	7.911976	16.273443	9.207804
11	7.404638	4.040385	-2.479623	0.170477	5.154094	7.558239	25.796463	54.420250	56.189331	37.366901	14.789932
12	-8.467812	-9.011234	-3.876231	-1.872407	-5.948402	-11.022078	-18.616035	8.646656	22.599117	19.059601	7.079203
13	3.804636	0.560930	-6.087071	-1.061637	-10.845011	-12.256536	-25.573631	-26.359543	-8.764282	-0.256127	4.249427
14	-0.827087	-1.336115	-7.228782	-8.151159	4.856355	5.809890	-2.585980	-27.451038	-31.905766	-25.919701	-5.083817
15	-2.071603	-2.954080	1.253370	-0.004242	-11.712537	-4.995994	9.089638	7.634851	3.053438	-8.671313	-6.971725
16	2.583647	3.968423	6.897946	4.984236	0.230353	-2.489114	-2.903225	2.311339	9.004272	-0.200183	-4.441242
17	-1.631444	-1.092217	2.786597	4.939474	4.947609	1.798677	-17.657579	-34.046059	-24.189425	-5.123381	-4.178518
18	1.557062	1.416171	-0.270457	0.434201	34.311932	42.772415	40.547852	9.943402	-6.470910	-9.476925	0.937829
19	7.575624	5.664735	0.325512	3.050874	-0.694569	8.311014	29.684641	34.127159	17.127417	4.865802	2.211671

MFCC

Spectrum

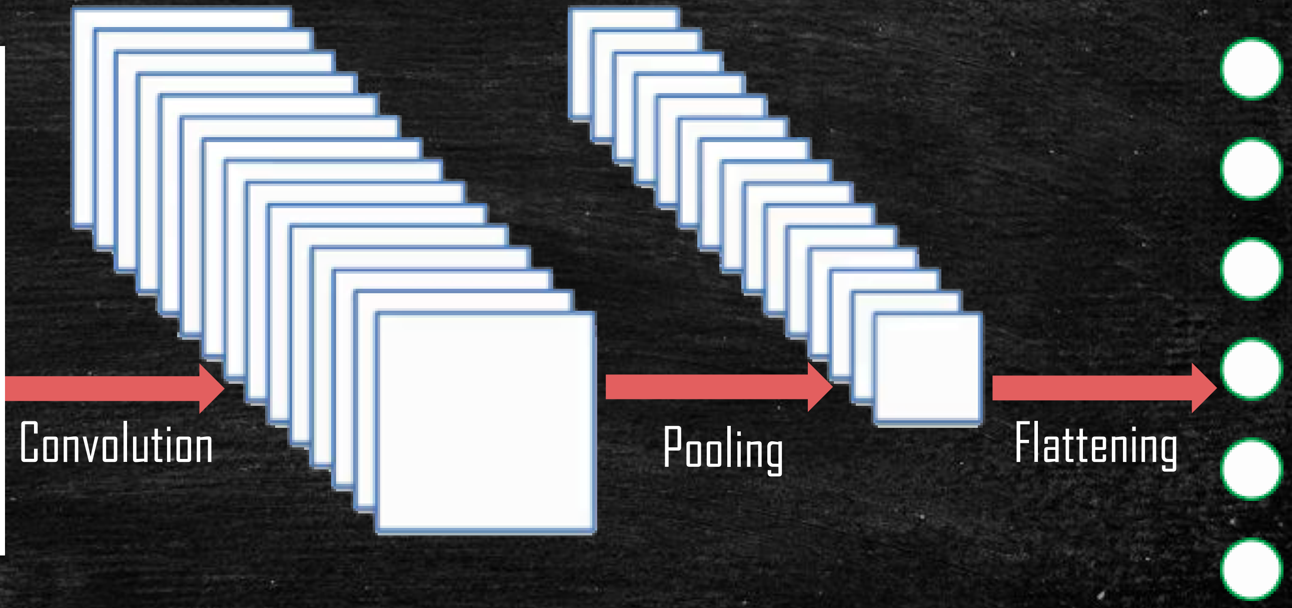


Convolutional Neural Network



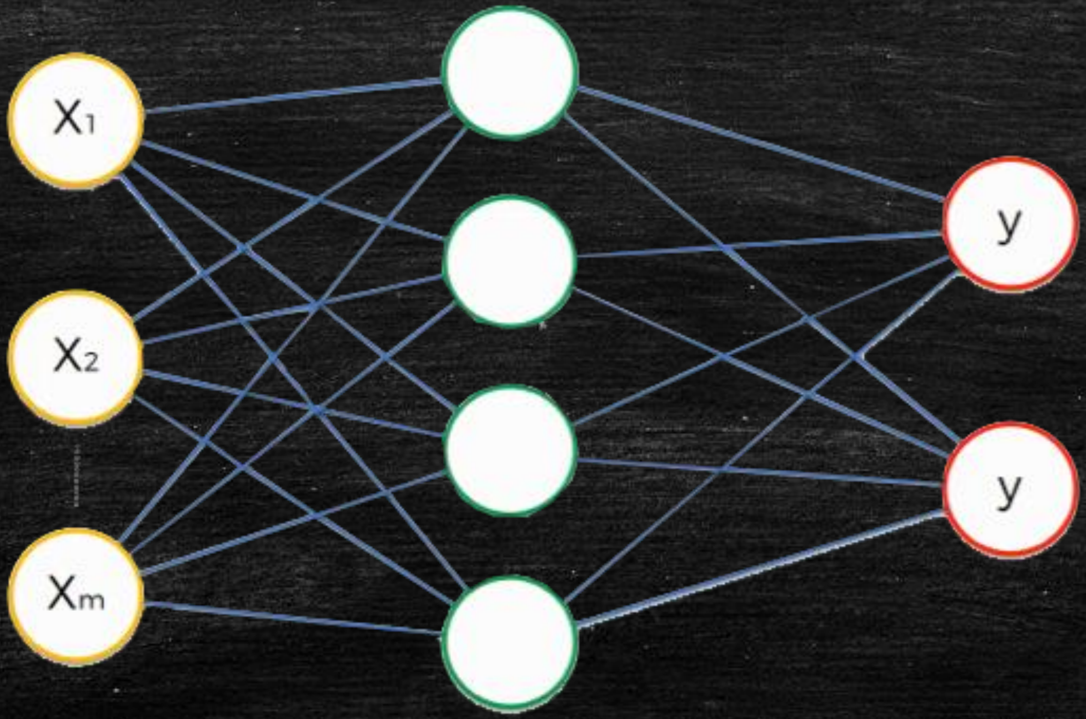
	0	1	2	3	4	5	6	7	8	9	10
0	-361.992615	-405.032532	-475.916138	-474.601990	-219.070480	-126.130203	-110.430977	-116.034714	-174.682404	-289.293854	-397.126434
1	-7.637213	9.353251	42.000008	46.531158	-1.405969	-33.400826	-63.382504	-54.208649	-12.167704	40.837433	44.729530
2	-24.234180	-15.422884	3.619861	4.348600	-59.652428	-61.829826	-60.735481	-57.085793	-54.226746	-46.676994	-17.773731
3	40.104313	38.911865	31.736929	27.395123	-33.264832	-32.979980	-16.765451	-26.616055	-40.216705	-21.145844	14.714356
4	25.392010	24.452545	21.378672	21.765610	36.428406	36.001221	37.056084	28.337475	27.251358	18.827328	23.066151
5	15.977909	14.637233	10.945372	9.916015	2.611271	-3.192318	-7.471964	-4.011388	-5.241780	-12.166206	-1.595258
6	-6.490936	-4.946122	-1.252912	1.259649	-33.483894	-33.408649	-38.812881	-35.919350	-28.600630	-17.644415	-7.998960
7	-0.035998	-1.053053	-2.966421	0.803960	18.655209	20.120529	12.922794	3.066458	1.023172	-0.859061	1.314217
8	-20.433775	-17.959461	-6.152930	-0.006574	13.367506	12.462053	8.854969	-5.131057	-1.511046	-12.831343	-12.740799
9	-2.187109	-2.966935	-4.177238	0.965661	18.014763	23.689003	28.637028	14.016478	14.406263	19.055923	14.605883
10	-2.920375	-3.410037	-4.789925	0.592175	-3.507984	4.487298	18.063095	17.231956	7.911976	16.273443	9.207804
11	7.404638	4.040385	-2.479623	0.170477	5.154094	7.558239	25.796463	54.420250	56.189331	37.366901	14.789932
12	-8.467812	-9.011234	-3.876231	-1.872407	-5.948402	-11.022078	-18.616035	8.646656	22.599117	19.059601	7.079203
13	3.804636	0.560930	-6.087071	-1.061637	-10.845011	-12.256536	-25.573631	-26.359543	-8.764282	-0.256127	4.249427
14	-0.827087	-1.336115	-7.228782	-8.151159	4.856355	5.809890	-2.585980	-27.451038	-31.905766	-25.919701	-5.083817
15	-2.071603	-2.954080	1.253370	-0.004242	-11.712537	-4.995994	9.089638	7.634851	3.053438	-8.671313	-6.971725
16	2.583647	3.968423	6.897946	4.984236	0.230353	-2.489114	-2.903225	2.311339	9.004272	-0.200183	-4.441242
17	-1.631444	-1.092217	2.786597	4.939474	4.947609	1.798677	-17.657579	-34.046059	-24.189425	-5.123381	-4.178518
18	1.557062	1.416171	-0.270457	0.434201	34.311932	42.772415	40.547852	9.943402	-6.470910	-9.476925	0.937829
19	7.575624	5.664735	0.325512	3.050874	-0.694569	8.311014	29.684641	34.127159	17.127417	4.865802	2.211671

MFCC



Convolutional Neural Network

- Full Connection



Implementation

Convolutional Neural Network

- Download Data

https://www.kaggle.com/c/tensorflow-speech-recognition-challenge