CNN Architectures

Alexnet - 8 layers. - 60M
First time ReLU was used.
Used Local Response Normalisation - to crearte competition between neighbour nodes.
Used heavy data augmentation - best way to reduce overfitting. - random cropping, PCA

VGG - 19 layers. - 138M
Used small filter size - 3x3,stride=1,pad=1
3 3x3 has the same receptive field as that of 7x7, but it is deeper, so more complex boundaries.
Also number of parameters are less.

Googlenet - (Inception) - 5M - 22 layers - No FC
Stacking local networks(Inception-module) on top of each other.
To get rid of high computational power required for a module, we use a bottleneck approach, i.e using 1x1 conv - it will reduce the number of channels.
One inception module - 1x1,3x3,5x5,3x3 pool
So we add 1x1 before 3x3,5x5
And after 3x3 pool
We use auxillary classifiers in the middle, as you want the intermediate layers to keep on learning. Loss of these classifiers were weightd by 0.3 (This also provided additional regularisation)
Average pooling is better than FC for the last part

Resnet - 152 layers
No FC - we use a global averaging pooling layer.
Main point is - increasing no.of layers in plain networks gives bad performance , not due to overfitting, but bcoz deeper networks are harder to optimize.
Also the residual block helps in backprop.

Modern
The main objective was to increase the gradient flow.
And to not use FC layer - as this hugely increaes the number of parameters